Universidade do Minho
Escola de Engenharia

Sara Maria Brito Araújo Cerqueira

A supportive, ergonomic, and Human-aware Human-Robot Collaboration Framework

August 2026

A supportive, ergonomic, and Human-aware Human-Robot Collaboration Framework
Sara Maria Brito Araújo Cerqueira
UMinho | 2026



Universidade do Minho
Escola de Engenharia

Sara Maria Brito Araújo Cerqueira

A supportive, ergonomic, and Human-aware Human-Robot Collaboration Framework

Doctoral thesis
Doctorate in Electronics and Computers Engineering
Automation, Control, and Robotics

Work carried out under the supervision of:

Professor Doctor Cristina Peixoto Santos
Professor Doctor Alexandre Silva
Professor Doctor Leonardo Gizzi

August 2026

Copyright and Terms of Use for Third Party Work

This dissertation reports on academic work that can be used by third parties as long as the internationally accepted standards and good practices are respected concerning copyright and related rights.
This work can thereafter be used under the terms established in the license below.
Readers needing authorization conditions not provided for in the indicated licensing should contact the author through the RepositóriUM of the University of Minho.

License granted to users of this work:



CC BY-NC-ND

https://creativecommons.org/licenses/by-nc/4.0/

Acknowledgements

"Eu sou quem tu és" Tenho a sorte de estar rodeada por pessoas que todos os dias me inspiram a ser melhor. A elas sou profundamente grata.

Aos meus pais, que me ensinaram os valores que tão orgulhosamente carrego e que sempre me apoiaram a ser quem eu quisesse ser.

Ao meu marido, Rodrigo, por ouvires quando precisava de desabafar. Por celebrares as minhas vitórias como se fossem tuas. Por me dares a mão mesmo quando as ideias são loucas. E por nunca encontrar dúvida no teu olhar.

Aos meus irmãos, cunhadas e sobrinhos, mas especialmente ao Zé que me incentivou a continuar o percurso académico.

À minha avó, que será sempre a minha maior fã. Às vezes a vida faz-me pensar que tenho um anjo da guarda a zelar por mim. Tenho a certeza de que lhe conheço o rosto.

À minha prima Susana e ao meu padrinho Zé, obrigada por todo o apoio e por me fazerem sempre sentir em casa.

À professora Cristina, que desde os tempos de mestrado arrancou-me da zona de conforto, com desafios e oportunidades. Acreditou em mim quando nem eu acreditava. Há pessoas que nos marcam. Levo-a no ADN da minha personalidade e identidade profissional. A Sara antes de a conhecer não é a mesma de agora. Um obrigado não chega!

Ao professor Alexandre e ao professor Leonardo, por aceitarem participar neste projeto e pela orientação nos momentos certos.

Aos meus colegas de laboratório, especialmente à Porta Aberta, vocês tornaram o Birdlab casa! Não consigo imaginar este percurso sem vocês. Um especial obrigado à Raquel por seres o meu contacto de emergência (literalmente).

Às pessoas que ao longo do tempo vestiram a camisola Ergoaware. Obrigada por acreditarem no projeto e por abraçarem o desafio. Aprendi tanto convosco! Levo um bocadinho de vós em mim. Um especial obrigado ao Diogo, Gilberto e Vítor, por não só vestirem a camisola como entregarem mais 4

anos da vossa vida a este projeto e visão.

À Dra. Mercedes, ao Dr. Elazer, ao Dr. Praneeth e a todos os colegas e amigos que fiz no MIT e em Boston. Obrigada por terem tornado esta experiência tão especial! À FCT, Fulbright e FLAD, por patrocinarem toda esta investigação e mobilidade internacional.

À Ana Pombeiro e à Bosch, por acreditarem no projeto e por todo o vosso apoio.

Às vezes questiono-me quem seria a Sara sem o Birdlab. Nunca saberei a resposta, mas tenho a certeza de que me tornei na melhor versão de mim depois de ter entrado por aquela Porta. Não só ganhei asas como aprendi a voar.

Statement of Integrity

I hereby declare having conducted this academic work with integrity.

I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

University of Minho, Braga, January 2026

Sara Maria Brito Araújo Cerqueira

Resumo

As lesões músculo-esqueléticas relacionadas com o trabalho constituem um desafio socioeconómico significativo na manufatura moderna, impulsionado pela fadiga física, posturas inadequadas e envelhecimento da força de trabalho. Apesar da adoção crescente de robôs colaborativos, os sistemas atuais carecem de consciência dos estados fisiológicos e ergonómicos humanos, limitando a prevenção de lesões e a personalização da assistência.

Esta tese propõe uma *framework* de colaboração humano–robô centrada no trabalhador, integrando sensorização vestível, perceção orientada por dados, avaliação ergonómica e controlo adaptativo. Foram desenvolvidas e validadas duas tecnologias de sensorização: uma peça de vestuário inteligente com unidades inerciais para monitorização do tronco e um sistema de eletromiografia de superfície em têxteis eletrónicos para análise da atividade muscular. Ambos asseguram qualidade de sinal fiável. Com base nestes dados, são explorados métodos de estimação de postura orientados por dados, reduzindo a dependência de modelos biomecânicos.

O reconhecimento de posturas é integrado com métricas ergonómicas articulares e modelos de exposição cumulativa, resultando num *pipeline* que quantifica o risco ergonómico por postura. Para além disso, são desenvolvidos modelos de *deep learning* para estimação da fadiga muscular a partir de sinais de eletromiografia de superficie.

Estas capacidades de perceção suportam estratégias de controlo centradas no ser humano, demonstradas num robô colaborativo. Uma estratégia de controlo agonista–antagonista redistribui a carga muscular de forma dinâmica, atrasando a progressão da fadiga, enquanto um controlador baseado em *reinforcement learning* se adapta às restrições ergonómicas e físicas do utilizador, assegurando assistência segura e personalizada. Adicionalmente, uma estratégia de controlo por admitância é implementada para otimizar a execução das tarefas e apoiar o treino de utilizadores iniciantes.

A validação experimental demonstra que a *framework* proposta reduz o risco ergonómico e viabiliza uma colaboração humano–robô mais segura, inclusiva e informada por fatores ergonómicos.

Palavras-chave Colaboração humano-robô, ergonomia, inteligência artificial, sensores vestíveis

Abstract

Work-related musculoskeletal disorders represent a major socio-economic challenge in modern manufacturing, driven by physical fatigue, awkward postures, and an aging workforce. Despite the increasing adoption of collaborative robots, current human–robot collaboration systems largely lack awareness of human physiological and ergonomic states, limiting their ability to prevent WRMSDs or provide inclusive and personalized assistance. This thesis addresses these limitations by proposing a human-aware HRC framework that integrates wearable sensing, data-driven perception, ergonomic assessment, and adaptive control.

The work begins with the development and validation of two wearable sensing technologies: a smart garment equipped with inertial measurement units for upper-body motion tracking, and an e-textile surface electromyography system for muscle activity monitoring. Both systems demonstrate reliable signal quality and robustness while avoiding the cost and rigidity of proprietary solutions. Building on these foundations, data-driven pose estimation methods are introduced to reduce reliance on explicit biomechanical models. To enable interpretable ergonomic assessment, posture recognition is combined with joint-level ergonomic scoring and cumulative exposure modeling, resulting in a reporting pipeline that allows ergonomic risk to be continuously quantified, attributed to specific postures, and visualized for practitioners. Deep learning models are then developed to estimate subject-independent muscle fatigue from sEMG data.

These perception capabilities are embedded into human-aware control strategies deployed on a collaborative robot. A fatigue-based agonist–antagonist control scheme dynamically redistributes muscular load to delay fatigue progression, while a reinforcement learning based controller learns user-specific ergonomic and physical constraints to provide safe and personalized assistance. An admittance-based control strategy further supports task quality and novice training through physically intuitive guidance.

Experimental validation demonstrates that the proposed framework reduces ergonomic risk, supports personalized assistance, and enhances task execution quality, contributing to healthier, more inclusive, and ergonomically informed human–robot collaboration.

Keywords Human–robot collaboration, ergonomics, artificial intelligence, wearable sensors

Contents

1 Introduction **1**

1.1 Motivation . . . 2

1.2 Problem Statement . . . 4

1.3 Goals and Research Questions . . . 6

1.4 Contributions to Knowledge . . . 9

1.5 Publications . . . 11

1.6 Thesis Outline . . . 13

2 Literature Review **15**

2.1 Wearables for ergonomic risk assessment . . . 15

2.1.1 Methods . . . 15

2.1.2 Postural Ergonomic Assessment . . . 16

2.1.3 Discussion . . . 27

2.2 Human Activity Recognition for Ergonomic Assessment . . . 31

2.2.1 AI-based activity recognition models . . . 31

2.2.2 Discussion . . . 37

2.3 Perception Models for Human–Robot Collaboration . . . 39

2.3.1 Muscle Fatigue Estimation . . . 39

2.3.2 Methods . . . 40

2.3.3 Models for Fatigue Estimation . . . 40

2.3.4 Sensors and sampling frequency . . . 46

2.3.5 Target muscles, tasks, and protocol . . . 46

2.3.6 Subjects and dataset size . . . 47

2.3.7 Fatigue Labeling . . . 47

2.3.8 Traditional sEMG-based indicators for fatigue estimation . . . 47

2.3.9 Model-based techniques for fatigue estimation . . . 48
2.3.10 Muscle fatigue detection based on machine/deep learning models . . . 49
2.3.11 Discussion . . . 52
2.4 Control Strategies for Human-aware Collaborative Robots . . . 54
2.5 General Insights . . . 61

I Tools for Ergonomic Risk Assessment 64

3 Wearable Technology for Human Awareness 65
3.1 Introduction Insight . . . 65
3.2 Ergowear . . . 66
3.2.1 Design Insights from Commercial Systems . . . 66
3.2.2 Solution Conceptualization . . . 68
3.2.3 Solution Development . . . 72
3.2.4 Experimental Methodology . . . 81
3.2.5 Results . . . 84
3.2.6 Discussion . . . 92
3.3 sEMG e-textile based electrodes . . . 96
3.3.1 Knowledge acquisition: Literature research . . . 96
3.3.2 Solution Description . . . 98
3.3.3 Experimental Methodology . . . 107
3.3.4 Results . . . 111
3.3.5 Discussion . . . 117
3.4 Conclusions . . . 120

4 Computational Methods for Human Motion and Ergonomics Analysis 122
4.1 Introduction Insight . . . 122
4.2 Deep-learning based Pose Estimation . . . 123
4.2.1 Knowledge acquisition: Literature research . . . 124
4.2.2 Solution Description . . . 125
4.2.3 Dataset acquisition . . . 125
4.2.4 Experimental Methodology . . . 128
4.2.5 Implementation Details . . . 132

4.2.6 Results . . . 138
4.2.7 Discussion . . . 147
4.3 Human Activity Recognition . . . 150
4.3.1 Solution Description . . . 150
4.3.2 Dataset Acquisition . . . 150
4.3.3 Experimental Methodology . . . 155
4.3.4 Implementation Details . . . 161
4.3.5 Results . . . 163
4.3.6 Discussion . . . 172
4.4 Ergo-report . . . 177
4.4.1 Solution Conceptualization . . . 177
4.4.2 Results . . . 186
4.4.3 Discussion . . . 190
4.5 Conclusion . . . 193

II Human-care Control Strategies 195

5 Ergoaware Framework Overview 196
5.1 Introductory Insight . . . 196
5.2 Conceptual Design and Functionalities . . . 196
5.2.1 Sensory systems . . . 198
5.2.2 Actuator . . . 200
5.2.3 Processing Station . . . 201
5.2.4 Position-based Control . . . 205
5.2.5 Velocity-based Control . . . 205
5.3 Conclusions . . . 205

6 Fatigue-Based Control strategy 207
6.1 Introductory Insight . . . 207
6.2 Solution Conceptualization . . . 207
6.2.1 Requirements . . . 207
6.3 Fatigue estimation Model . . . 208
6.3.1 Experimental Methodology . . . 208

6.3.2 Dataset Acquisition . . . 209
6.3.3 Dataset pre-processing and features selection . . . 214
6.3.4 Exploratory Data Analysis . . . 216
6.3.5 Data Input Preparation . . . 217
6.3.6 Dataset Split . . . 219
6.3.7 Implementation Details . . . 223
6.3.8 Results . . . 225
6.3.9 Discussion . . . 236
6.4 Fatigue-based Control Strategy . . . 241
6.4.1 Methods . . . 241
6.4.2 Implementation Details . . . 243
6.4.3 Results and Discussion . . . 246
6.5 Conclusions . . . 250

7 Learning Human Specificities based on Ergonomics and Physical restrictions 252
7.1 Introductory Insight . . . 252
7.2 Knowledge acquisition: Reinforcement Learning . . . 253
7.2.1 Application Domain Considerations . . . 253
7.2.2 Insights from MDP Design . . . 253
7.2.3 Reward Function Design and Human Considerations . . . 254
7.2.4 Algorithm Selection and Learning Strategies . . . 255
7.2.5 Learning Settings and Transferability . . . 255
7.3 Solution Conceptualization . . . 255
7.3.1 Solution requirements . . . 255
7.3.2 Elbow Contracture and Use Case Scenario . . . 256
7.3.3 Conceptional Overview of the RL-based Control . . . 256
7.4 Experimental Methodology . . . 258
7.4.1 Reinforcement Learning Design . . . 258
7.4.2 Simulation Environment and Human Posture Modeling . . . 264
7.5 Implementation Details . . . 266
7.5.1 Experimental Setup . . . 266
7.5.2 Action Space Shaping . . . 267
7.5.3 Pain and Ergonomic Assessment . . . 268

7.5.4 Reward Function . . . 268
7.5.5 Training Termination Criteria . . . 270
7.5.6 Hyperparameter Optimization . . . 270
7.5.7 ROS Implementation . . . 271
7.5.8 Experimental Validation . . . 273
7.6 Results . . . 275
7.6.1 Hyperparameter Optimization . . . 275
7.6.2 Real Environment Tests . . . 279
7.7 Discussion . . . 286
7.7.1 Discussion . . . 286
7.8 Conclusions . . . 289

8 Task Quality Based Control Strategy 291
8.1 Introductory Insight . . . 291
8.2 Knowledge acquisition . . . 292
8.3 Solution Conceptualization . . . 293
8.4 Methods . . . 293
8.4.1 Admittance Control Formulation . . . 294
8.4.2 Free-drive mode . . . 295
8.4.3 Guided-drive mode . . . 296
8.4.4 Implementation Details . . . 299
8.4.5 Experimental protocol . . . 302
8.4.6 Data Analysis . . . 304
8.5 Results . . . 305
8.5.1 Performance Metrics . . . 305
8.5.2 Usability and User Experience . . . 307
8.6 Discussion . . . 308
8.6.1 Performance metrics . . . 308
8.6.2 Usability and User Experience . . . 309
8.7 Conclusions . . . 310

9 Conclusions 311
9.1 Concluding Remarks . . . 311

9.2 Research Questions . . . 313
9.3 Prospect for future work . . . 319

III Appendices **350**

A Literature Review **351**
A.1 Wearables for ergonomic risk assessment . . . 351
A.2 AI-based activity recognition models . . . 352
A.3 Muscle Fatigue Estimation . . . 353
A.4 Control Strategies for Human-aware Collaborative Robots . . . 353

B Wearable Technology for Human Awareness **355**
B.1 Ergowear . . . 355
B.1.1 Hardware Implementation Details . . . 355
B.1.2 Sensor Specifications . . . 355
B.1.3 Communication Setup . . . 355
B.1.4 Wiring and Connectors . . . 356
B.1.5 Cabling . . . 356
B.1.6 Biofeedback Circuit . . . 356
B.1.7 Power Analysis . . . 357
B.1.8 Mechanical Integration . . . 357
B.1.9 Embedded Software Development . . . 358
B.2 sEMG e-textile based electrodes . . . 362
B.2.1 sEMG Acquisition Module: Android Application . . . 362

C Deep-learning based Pose Estimation **363**
C.1 Dataset acquisition . . . 363
C.2 Pre-processing . . . 364
C.3 Results . . . 364

D Human Activity Recognition **366**
D.1 Feature Selection . . . 366
D.2 Results . . . 368
D.2.1 Hyperparameter Optimization Results . . . 368

E Ergoreport **370**

E.1 Implementations Details . . . 371

E.2 Association between Ergonomic Risk and Postures . . . 372

F Fatigue-based Control Strategy **374**

F.1 Methods . . . 374

F.1.1 Participants . . . 374

F.1.2 Data Preprocessing and Feature Selection . . . 375

F.2 Results . . . 376

F.2.1 EDA . . . 376

F.2.2 Hyperaparameter Optimization Results . . . 378

F.3 Usability Questionnaire . . . 379

G Learning Human Specificities based on Ergonomics and Physical restrictions **380**

G.1 Results . . . 380

G.1.1 Real-World Deployment of the Pre-Trained Model . . . 380

H Task Quality Based Control Strategy **383**

H.1 Results . . . 383

H.1.1 Performance Metrics . . . 383

List of Figures

1 Expected impact of this work. 9

2 Overview of the thesis structure. 14

3 Body locations of the sensors and number of included studies that chose them. Circles represent sensors in the anterior (^{a}) part of the body, squares the ones in the posterior (^{p}) part of the body, filled symbols the ones placed bilaterally (^{b}), and empty ones unilaterally (either left, ^{l}, centre, ^{c}, or right, ^{r}). 25

4 Three 2D input window strategies: (a) multi-timestamp rows with concatenated channels; (b) single timestamp with one sensor per row; (c) mixed sensors and timestamps across rows 36

5 Movella Xsens MVN Link wearable system. (a) Arm-mounted MTx sensor attached via Velcro patch, with the zipper that covers the electronics open. (b) Back view of the suit showing the upper-body sensor cable branches routed along the back. (c) Body pack mounted at the lower back, housing the central electronics and cable terminations. . . . 67

6 Rokoko Smartsuit Pro II wearable system. (a) Back view of the suit as worn. (b) Garment laid flat, showing the overall layout. (c) Velcro patch used to secure sensors inside the garment. (d) Elastic strap used for sensor fixation. (e) Opened sensor housing, exposing internal components and strain relief. 68

7 Ergowear 2.0: Misalignment between the a) hand sensor and the forearm sensor, and b) on the back; c) Adhesive tape used to fix the sensor modules onto the respective anatomical 69

8 Smart garment layout showing (a) the sensing system and (b) the biofeedback system. . 72

9 Design features of the garment: (a) Light mesh placement on the elbow region; (b) Light mesh placement on the shoulder region; (c) Elastic pockets for securing the sensors. . . 73

10 System's hardware overview. 74

11 Sensors' communication architecture and wiring configuration. 75

12 Ergowear modules: (a) master module, (b) intermediary sensor module, and (c) extremity sensor module. 77
13 Ergowear smart garment: (a) front view, (b) lateral view, and (c) rear view. 78
14 Software architecture overview. 78
15 Illustration of the calibration procedures: (a) Six-position method used for accelerometer calibration. (b) Magnetometer calibration setup, with the sensors, processing station, and power bank mounted on a flat cardboard surface. 79
16 Comparison of high-bandwidth gyroscope data sampled at high and low frequencies, highlighting the loss of detail at lower sampling rates. Image from [1] 81
17 Examples of the movements performed during the usability tests. (a) Trunk flexion; (b) wrist flexion; (c) crossing arms; (d) placing a box on the top of a cabinet; 84
18 Oscilloscope capture showing five consecutive interrupt cycles (blue) and the execution window of the total sending routine (yellow), used to verify timing compatibility at a 240 Hz sampling rate. 86
19 Real sampling frequency and packet loss as a function of the distance between the main station and the processing unit. 87
20 Real sampling frequency (Hz), packet loss (%), and data corruption (%) of the wireless communication setup over time (in hours). 89
21 System's Biofeedback questionnaire answers. 90
22 System's Usability questionnaire answers. 90
23 Back view of the knitted structures (horizontal tracks): (a) Floating loops - Shieldex. (b) Base textile courses replaced with conductive fiber - Elitex SC. 100
24 Conductive track test samples using Elitex SC fiber: (a) Horizontal tracks, front view; (b) Horizontal tracks, back view; (c) Vertical tracks, front view; (d) Vertical tracks, back view. 100
25 Back view of the knitted structures (horizontal tracks): (a) Floating loops. (b) Base textile courses replaced with conductive fiber. 101
26 Block diagram of the sEMG acquisition system, highlighting a three-stage analog front-end with cutoff frequencies of 22.58 Hz (high-pass) and 512.1 Hz (low-pass), followed by adjustable gain amplification and digital processing via the Arduino Nano 33 IoT. . . . 102
27 AD8232 acquisition and filtering circuit schematic 103
28 Schematic of the LiPo battery charging circuit 104
29 The sEMG acquisition PCB: (a) Front view. (b) Back view. 105

30 Overview of the embedded software implemented on the Arduino platform for sEMG signal acquisition and processing. 106
31 Android application main screens. 107
32 Tensile testing setup: (a) Dynamometer and digital multimeter setup. (b) Four-wire measurement configuration. 108
33 Abrasion test setup: (a) Martindale abrasion tester. (b) Sample placed inside the rubbing head. 108
34 Electrode placement: (a) using Ag/AgCl electrodes. (b) using textile electrodes. 110
35 Test setup. 111
36 Variation of the resistance of the samples during the tensile strength tests, using a floating loop structure: (a) Shieldex 117/17 HCB, (b) Shieldex 235/36 HCB. 112
37 Variation in the resistance of the samples during the tensile strength tests, using Elitex yarns and a knitted structure. (a) Elitex SC with a thickness of 6 rows (horizontal). (b) Elitex with a thickness of 8 rows (horizontal). (c) Elitex SC with a thickness of 6 rows (vertical). (d) Elitex SC with a thickness of 8 rows (vertical). show the typical piezoresistive behavior of horizontal track samples. The red dashed circles show the more fluctuating behavior of the vertical track samples. 113
38 Comparison of sEMG signals acquired with different electrode types and conditions. From left to right: Ag/AgCl electrodes under (a) dynamic and (b) static conditions; 4 $\times$ 1 cm textile electrodes under (c) dynamic and (d) static conditions; and 2 $\times$ 1 cm textile electrodes under (e) dynamic and (f) static conditions. 115
39 End-to-end steps to build the human pose estimation models 126
40 (a) Participant instrumented with Xsens MTw Awinda and Ergowear. (b) Participant instrumented with two Xsens MTw Awinda systems on top of each other. 127
41 Frames, from video records captured during data acquisition, illustrative of datasets' movements for each of the 6 types of sequences: Calibration (a), Task (b), Sequence (c), Random (d), Circuit (e) and Validation (f). 129
42 Detailed breakdown of the data preparation and sensor fusion steps 130
43 Multiple sensor-fusion architectures: 1) Classical Filter-fusion; 2) Neural-Networks (global); 3a) Neural-Networks (complementary approach, abbreviated as Hybrid-Complementary-NN); 3b and 3c) Hybrid Filter-Neural-Networks, abbreviated as Hybrid-CFF-NN. 131

44 Boxplots of the measured acceleration values per segment: (a) Ergoaware dataset, (b) MtwAwinda dataset. 133
45 Boxplots showing the QAD error per segment: (a) Model-free LSTM on the Ergowear test set, (b) Model-free LSTM on the MTwAwinda test set, (c) LSTM-MG-detach on the Ergowear test set, and (d) LSTM-MG-detach on the MTwAwinda test set. 139
46 Cliff's δ effect sizes comparing classical filters against Neural Network (NN)-based models (negligible: $|\delta| < 0.15$, small: $0.15 \leq |\delta| < 0.33$, medium: $0.33 \leq |\delta| < 0.47$, large: $|\delta| \geq 0.47$). Positive δ indicates classical filters performed worst in relation the NN-based methods. 141
47 Cliff's delta effect-size analysis for the ablation studies. Positive values indicate higher Quaternion Angle Distance (QAD) errors for the ablation compared with the corresponding baseline. 144
48 Temporal results on 5min trial with and without magnetometer, for the COMP-LSTM model. 145
49 Pipeline of the development of the DL-based posture recognition models, where RS denotes random sampling. 151
50 Identification of typical postures. 151
51 Participant instrumented with the two Xsens MTw Awinda systems. Blue circles highlight the sensors used in the DL models, after the feature selection. 153
52 Participant in the experimental setup of the agriculture task circuit. 154
53 Participant in the experimental setup of the construction task circuit 155
54 Number of samples and percentage of posture classes in the dataset. 156
55 Proposed architectures. 159
56 Box plot of validation F1-scores for different combinations of (a) number of filters and layers, and (b) kernel size and number of filters. 164
57 Box plot of validation F1-scores for different combinations of (a) embedding size and number of layers, and (b) number of layers and heads, for transformer architecture. . . 165
58 Box plot of validation F1-scores for the different combinations of the number of convolutional layers and LSTM layers for CNN-LSTM architecture. 166
59 Box plot of validation F1-scores for the different combinations of the number of convolutional layers and encoder layers for CNN-transformer architecture. 167
60 Box plot of validation F1-scores for different combinations of LR and batch size, considering all the models of the five architectures. 167

61 Box plot of validation F1-scores for different combinations of WS and overlap, considering all the models of the four architectures. 168

62 Comparison of the classification output with the actual window labels and the original sample labels in a construction circuit. Green and red stripes highlight, respectively, correct and incorrect classifications by the CNN-TR 1. 170

63 Confusion matrix of the model CNN-TR 1. 171

64 Grad-CAMs for the class *stooping*. (a) Correct prediction. (b) Misclassified as *stooping* instead of *lifting/lowering*. The redder the pixels, the more important the signal regions are for the model's decision. Only acceleration signals are shown. 172

65 Importance of each Inertial Measurement Unit (IMU) considering the Grad-CAMs of all the True Positives (TP) windows in the test set, in (a) CNN 1 and (b) CNN 2. The posture classes are identified as follows: OT - *others*; N - *standing*; RE - *reaching*; ST - *stooping*; SQ - *squatting*; KN - *kneeling*; CA - *carrying* 173

66 Conceptual design of the system. 178

67 Behaviour of the joint kinematic wear index, V_{ij}, calculated using Equation 4.8 during wear phase and using Equation 4.9 during recovery phase. 183

68 GUI of the developed framework, showing the page with the options to upload the acquired data and perform the posture assessment. 186

69 Back flexion/extension LUBA scores and kinematic wear during the circuit based on the construction task, adopting (a) *stooping*, (b) *squatting*, or (c) *kneeling* postures for bricklaying. Red and green stripes highlight, respectively, wear and recovery phases. The red dashed line indicates the kinematic wear threshold of 0.7, suggested by [2]. To illustrate the trials, some frames from the latter example's recording are presented and identified in the plot. 188

70 Global LUBA scores (on a scale from 0 to 62) and postures adopted during (a) an agriculture and (b) a construction trial. The background colors refer to the categories regarding the need for corrective actions, defined in Table 32. 189

71 Summary report with the average percentage of time spent at each posture, averages of the mean LUBA scores (on a scale from 0 to 62) for each posture and 3 joint motions that breached the kinematic wear threshold of 0.7 for the longest time, considering all (a) agriculture and (b) construction trials. Note that the time percentages do not have to sum to 100%, as these are the averages of all trials. The colours refer to the categories regarding the need for corrective actions, defined in Table 32. 190

72 Ergoaware HRC framework overview. 197

73 ROS processing delay distributions for Delsys Trigno at 1259.259 Hz and Xsens MVN at 60 Hz. 200

74 Simulation environment in Rviz. 202

75 Simulation environment and collision model used for motion planning. Collision volumes are defined in red, around the workstation, feeder, robot structure, and a vertical interaction boundary in front of the human. 203

76 ROS node and topic graph of the Human package, ROS node and topic graph of the Human package. 203

77 Participant instrumented with 8 wireless Surface Electromyography (sEMG) sensors (Trigno Avanti, Delsys), placed on: 1) Right Deltoid Anterior, 2) Right Deltoid Medius, 3) Right Biceps Branchii, 4) Right Deltoid Posterior, 5) Left Deltoid Anterior, 6) Left Deltoid Medius, 7) Left Biceps Branchii. Sensor 8 is placed on the Left Deltoid Posterior. sEMG placement is the same for D1 and D2. 210

78 Self-perceived fatigue-level description. 211

79 Flowchart illustrating the experimental protocol sequence. 212

80 Experimental protocol. All movements were executed, one at once, with both right and left arm. For D2, only movement a, b, c, and d were executed. 213

81 Preprocessing and feature extraction pipeline for the muscle fatigue dataset. 214

82 Global (a) and per-subject (b) class distribution of the fatigue dataset. 215

83 Statistical analysis and feature selection pipeline for the muscle fatigue dataset. 216

84 Proposed Architectures 221

85 Mean absolute Cliff's delta effect size computed across subjects and trials. (a) Feature ranking obtained by averaging across all trials and combinations, with error bars indicating the standard deviation; (b) average Cliff's delta across sEMG features for each fatigue-level comparison. 225
86 (a) Heatmap of statistical significance across window length and overlap combinations; (b) Significance rate heatmap based on a one-sided Wilcoxon test, confirming whether the observed differences follow the expected direction. 226
87 Distribution of self-perceived fatigue label durations across subjects and trials, shown overall and stratified by muscle group. 227
88 Distribution of performance metrics (accuracy, weighted F1-score, and macro F1-score) across all CNN experiments. 229
89 Distribution of performance metrics (accuracy, weighted F1-score, and macro F1-score) across all LSTM experiments. 229
90 Distribution of validation weighted F1-scores for LSTM experiments across batch sizes and loss functions (weighted cross-entropy and focal loss) 230
91 Distribution of validation weighted F1-scores for LSTM experiments across feature windows and overlap for batch size 1. 230
92 Distribution of performance metrics (accuracy, weighted F1-score, and macro F1-score) across all LSTM with time-attention experiments. 231
93 Distribution of validation weighted F1-scores for LSTM experiments across batch sizes and loss functions (weighted cross-entropy and focal loss) 231
94 Distribution of validation weighted F1-scores for the LSTM with time-attention (batch size = 1) across feature window sizes and overlaps. 232
95 Distribution of performance metrics (accuracy, weighted F1-score, and macro F1-score) across all Time-Frequency LSTM with Attention experiments. 232
96 Distribution of validation weighted F1-scores for Time-Frequency LSTM with Attention experiments across batch sizes and loss functions (weighted cross-entropy and focal loss) 232
97 Validation weighted F1-score distributions of the Frequency–Time LSTM with attention as a function of feature window size and overlap for all batch sizes 233
98 Confusion Matrix for the best performing model (TF LSTM with attention). 235
99 Confusion matrix of the #52 TF Long Short Term Memory (LSTM) with attention evaluated. 235
100 Conceptualization of the control strategy. 242

101 Visualization of ROS computation graph. Schematic of the communication between nodes through topics and services. 243

102 Collaborative task and the two workstation configurations used for fatigue prevention. (a) Anterior deltoid activation. (b) Posterior deltoid activation. The locations of the sEMG sensors are visible in the figures. 246

103 Mean normalized sEMG RMS (%MVC) of the anterior (A) and posterior (P) deltoid muscles across five participants (P1–P5), comparing HRC and No HRC conditions. Error bars represent the combined standard deviation of both muscles, computed as the root mean square (RMS). 247

104 Validation results for participant P2. (a) Normalized muscle activity and envelopes of the anterior and posterior deltoids with the control strategy enabled. (b) Estimated fatigue levels, where posterior deltoid estimation starts after fatigue is detected in the anterior deltoid and the cobot changes configuration. (c) Anterior deltoid activity during task execution without the control strategy. 248

105 Results of the questionnaire applied to evaluate the implemented work. The bars indicate mean subjective scores given by the subjects for each question. The black lines indicate the standard deviation of mean. 249

106 Results of the subjective evaluation related to (a) the performance of the implemented control strategy and (b) usability in industrial tasks. 250

107 Elbow contracture ROM. Red represents the painful elbow's ROM. 256

108 Conceptual diagram of the RL-based control system for co-manipulation transport task. 257

109 RL framework overview for learning human specificities, in ergonomics and physical constraints. 259

110 DQN diagram illustrating the main functional components of the algorithm. 260

111 Discrete state and action space representation used in the Q-Learning formulation. (a) Discretized workspace of size $w_{ss} \times h_{ss}$, where each cell represents a possible object position (x, z) with discretization step Δ_d. (b) Examples of valid action sets, where green markers indicate admissible transitions within workspace boundaries and orange circle markers denote the current object position. Σ_{sso} represents the state space frame. 263

112 Continuous state and action space representation used in the DQN formulation. (a) Continuous workspace of size $w_{ss} \times h_{ss}$, where object positions are defined in the state space frame Σ_{sso} using Cartesian coordinates (x, z). (b) Actions defined by movement direction α and displacement magnitude d. 264

113 Human kinematic model used to compute human joint angles in the pre-training phase. 265

114 Simulation (a) and real-world (b) environments used for control strategy validation. In the simulation, the yellow rectangle defines the state-space boundary, and the orange sphere represents the object midpoint, used as the object position. In the real-world setup, the participant performs the task while equipped with the Xsens motion capture system. 267

115 Schematic of communication between nodes through topics and services (Nodes and topics depicted at red are only used in real-world environment settings and the ones at green solely in simulation environment). 272

116 Participant instrumented with Movella's MVN Awinda in the "upper body and no hands" configuration. For illustration purposes, IMUs are shown on the outside of the straps. . . 274

117 Correlation matrix for a) 10 cm and for b) 6.25 discretization level, correlating hyperparameters and performance metrics. The hyperparameters include Learning Rate, Discount Factor and Epsilon Decay Duration. The performance metrics include reward per episode, average ergonomic risk level, steps taken per episode and the episode in which the reward started to converge. Matrix values correspond to Pearson correlation coefficients, denoted as r_{corr}. 278

118 Correlation matrix relating DQN hyperparameters and performance metrics for successful experiments. Matrix values correspond to Pearson correlation coefficients, denoted as r_{corr}. 279

119 Metrics obtained from the Q-Learning (both discretization levels) pre-trained models' execution in the real-world environment for P1. Red line represents metrics achieved in simulation. 280

120 Q-Learning trajectories for Participant 1 under the 10 cm discretization during real-world fine-tuning. (a) Executed trajectories across trials. (b) Reward map from simulation, highlighting the final fine-tuned trajectory with green squares and the simulation pre-trained trajectory with light-blue circles. 281

121 Q-Learning trajectories for Participant 1 under the 6.25 cm discretization during real-world fine-tuning. (a) Executed trajectories across trials. (b) Reward map (from simulation) highlighting the final fine-tuned trajectory (green) and the simulation pre-trained trajectory (light blue). 282

122 Metrics obtained from the DQN pre-trained model's execution in the real-world environment for P1. Red line represents metrics achieved in simulation. 283

123 Human posture at the end of (a) pre-trained policy execution and (b) episode 2 during fine-tuning of the DQN model. In (a), the elbow is highly extended (≈13°), indicating pain risk, whereas in (b) the elbow remains flexed (≈56°), within a safe range. 283

124 Executed trajectories by the Deep Q Network (DQN) model: (a) pre-trained model. black crosses represent state transitions that resulted in pain; (b) fine-tuning of the model. (c) fine-tuned model. 284

125 Usability questionnaire responses 287

126 Conceptualization of the task quality control strategy. 293

127 Block diagram of the admittance controller. (a) shows the general control architecture, common to both modes, where human-applied forces are measured through the F/T sensor and processed by the admittance control to generate desired end-effector velocities. In free-drive mode (b), compliance is achieved through variable damping c while inertia m is predefined and kept constant. In guided-drive mode (c), damping c and inertia m are varied according to the alignment between human-applied force and the reference trajectory, while stiffness k is predefined. 294

128 Guided-drive modulation strategy. The applied force $\vec{F}_{ext}$ is evaluated using its angular deviation α from the local trajectory tangent $\vec{T}$ and the distance d from the reference trajectory. Forces aligned with or directed toward the reference path reduce resistance, whereas forces driving the end-effector away from the reference increase resistance. . . 297

129 Experimental setup and trajectories. (a) UR10e collaborative robot with tool, (b–d) cardboard trajectories used during experiments: square wave, sine wave, and spiral. 301

130 ROS node and topic diagram for the admittance controller architecture. Services used to initialize the controller (trigger-based) are not depicted. 302

131 sEMG electrode placement on the dominant hand. (1) Deltoid Anterior, (2) Deltoid Medius, (3) Biceps Brachii (long head), and (4) Triceps Brachii (lateral head). 304

132 Performance metrics and muscle activation across control modes and trajectories ($n = 10$). (a) Mean interaction force (N). (b) Trajectory Root Mean Square Error (RMSE, mm).(c) Total task duration (s). (d) Average muscle activation across all monitored muscles (% MVC). (e) Individual muscle activation (% MVC) for the Biceps Brachii, Triceps Brachii, Deltoid Anterior, and Deltoid Medius. Error bars denote standard deviation ($\pm$ SD); 306

133 2-D Cartesian position (in meters) of the sine wave reference trajectory (in blue) and the trajectory performed by a participant (in red). The corresponding mean exerted force, trajectory error, and execution time are reported for each control mode: (a) UR10e free-drive, (b) implemented free-drive, and (c) guided-drive. 307

134 Participant preferences between the Free-drive Drive Mode and Guided-drive Mode . . . 308

135 PRISMA flow diagram for the search strategy. 352

136 PRISMA flow diagram for the research on activity recognition models 353

137 Ergowear's main board PCB. 358

138 Interior view of ergowear modules: (a) master module; (b) intermediary module; (c) extremity module. 358

139 Low-level controller software routine. The circular arrow with a clock icon represents the periodic timer interrupt, while the circular arrow with a lightning bolt indicates an event-driven WiFi callback. *start cmd* is sent by the main station. *send flag* is set by the sensory acquisition routine. 359

140 Android application flowchart. 362

141 Spearman's correlation between features 366

142 Kruskal-Wallis scores of the kinematic signals, with the red dashed line delimiting the top 25% 367

143 Flowchart of the proposed ergonomic assessment framework. 371

144 AWBA risk levels (on a scale from 1 to 4) and postures adopted during (**a**) an agriculture and (**b**) a construction trial. 372

145 Summary report with the average percentage of time spent at each posture and averages of the mean AWBA risk levels (on a scale from 1 to 4) for each posture, considering all (**a**) agriculture and (**b**) construction trials. Note that the time percentages do not have to sum to 100%, as these are the averages of all trials. The colours refer to the risk levels, as defined in Table 29. 373

146 Filtered sEMG signals for Subject 2 across eight uni-articular movements: (a) time-domain with fatigue labels, and (b) frequency-domain spectra. 375

147 Trial ranking based on the significance rate computed across all window lengths and overlap combinations. 377

148 Trial durations per trial. 377

149 Distribution of validation weighted F1-scores for the BiLSTM with time attention (batch size = 16) across feature window sizes and overlaps. 378

150 Validation weighted F1-score distributions of the Frequency–Time BiLSTM with attention as a function of feature window size and overlap for (a) batch size 1, (b) batch size 16, and (c) batch size 32. 378

151 Questionnaire applied to participants after the validation protocol. 379

152 Evolution of key metrics during the fine-tuning of the DQN pre-trained model in real-world environment. The metrics include a) reward obtained per episode, b) average pain and e) ergonomic risk levels, c) average Hubber loss, d) episode in which the reward metric started to converge and f) steps taken per episode. All metrics were monitored at the end of each episode. 382

List of Tables

1 Summary of the included studies on postural ergonomics monitoring, regarding their goals, ergonomic methods, sensing systems, and biofeedback systems 17

2 Summary of the studies included on postural ergonomics monitoring, regarding their algorithms, validation protocols, ground truths, and metrics 20

3 Summary of studies on activity recognition, highlighting study goals and sensing systems. 32

4 Summary of studies on activity recognition, including classes, preprocessing, AI algorithms, input, dataset split, protocols, ground truth (GT), and metrics. 33

5 Summary of studies on muscle fatigue detection, including model-based and ML/DL approaches. Reference, sensor, sampling frequency, filters, muscles, task, and protocol are shown. 41

6 Summary of studies using machine learning and deep learning methods for fatigue estimation and associated evaluation metrics. 44

7 Summary of HRC control strategies: reference, optimization method, objective, behaviour, low-level controller, modelled variable, interaction scenario, task, and validation subjects. 55

8 State-of-the-art limitations and derived research requirements. 63

9 Summary of System Requirements 71

10 Execution times of system processes at a 250 Hz sampling rate 85

11 Influence of the distance between the main station and the low-level processing unit on the wireless communication performance. 87

12 Influence of wall obstruction and wireless interference on the wireless communication performance. 88

13 Results of the autonomy tests under different operating modes. 88

14 Hardware comparison between the previous (prev.) iteration of Ergowear, the current version (curr.), and the XSens MVN Link. 91
15 Overview of commercially available silver-coated conductive yarns 98
16 Shieldex 117/17 HCB, Shieldex 235/36 HCB, and Imbut Elitex Skin Contact characteristics. 99
17 Average electrical resistance for Martindale's abrasion Test. Shieldex samples were fabricated using a floating loop structure, while Elitex samples were produced with an integrated knitted structure. 114
18 sEMG acquisition RMSE. The values highlighted in green showcase the electrodes with better acquisition performance. The values highlighted in red show the electrodes with the worst performance in terms of sEMG acquisition. 116
19 Summary of the train, validation and test data splits for Ergowear and MTwAwinda datasets. 134
20 Sensor Fusion average results, in QAD°, and inference time (ms) on Ergowear(D1) and MTwAwinda (D2) data. 138
21 Average QAD° and inference time (ms) results, on Ergowear and MTwAwinda data, of Classical Sensor Fusion filters and best NN-based sensor fusion method. 140
22 Average QAD° results of the ablation studies on Ergowear data. To ease the results analysis, performance improvements are in bold and underlined, slight performance decreases ($< 1°$) are underlined, and performance decreases are in normal text, all in relation to the baseline results. 142
23 Average QAD° results of the ablation studies on MTwAwinda data performance improvements are in bold and underlined, slight performance decreases ($< 1°$) are underlined, and performance decreases are in normal text, all in relation to the baseline results. . . 143
24 Hyperparameters explored in the grid search 162
25 Weights learned by the top 5 hybrid CNN-LSTM models (sorted by F1-score) to fuse both subnetworks' outputs, in the validation 166
26 Weights learned by the top hybrid CNN-Transformer models (sorted by F1-score) to fuse both subnetworks' outputs, in the validation 167
27 Hyperparameters, number of training epochs, training and inference times, and test metrics of the three best models of each architecture 169
28 Test metrics per class for the hybrid CNN-TR 1 171
29 Finite state machine rules for ergonomic assessment using AULA and ALLA 179
30 AWBA rules 179

31 Finite-state machine for the ergonomic assessment based on the LUBA method 180
32 Posture categories regarding the need for corrective actions defined by the LUBA method 181
33 Typical ergonomists' difficulties in postural risk assessment and the corresponding ErgoReport's solution to address them 185
34 Results of the 7-point Likert scale questionnaire for ErgoReport's usability assessment . 191

35 Hyperparameters explored in the grid search 224
36 Top three models per architecture ranked by validation weighted F1. $F1_m$ and $F1_w$ denote the macro-averaged and weighted F1-scores, respectively. 228
37 Test performance and inference time for the evaluated architectures. 234
38 Description of each topic present in ROS graph. 244
39 Mean normalized sEMG RMS (%MVC) of the anterior and posterior deltoid muscles for each participant under No HRC and HRC conditions. The reduction percentage is computed relative to the No HRC condition, where positive values indicate a decrease in muscle activation with HRC. 248
40 Mean normalized sEMG RMS (%MVC) of the anterior and posterior deltoid muscles for each participant under No HRC and HRC conditions. The reduction percentage is computed relative to the No HRC condition, where positive values indicate a decrease in muscle activation with HRC. 248

41 Initial States and End Episode Conditions 267
42 Q-Learning Hyperparameter Search Space 271
43 DQN Hyperparameters Search Space 271
44 Best-performing models from Q-Learning's HPO (10 cm discretization). Selected model is highlighted in bold 276
45 Best-performing models from Q-Learning's HPO (6.25 cm discretization). Selected model is highlighted in bold 277
46 Best-performing models resulting from DQN hyperparameter optimisation. The selected model is highlighted. 277
47 Performance Comparison of Q-Learning and DQN Algorithms for Different Human Anthropometries 286

48 Admittance controller parameters for free-drive and guided-drive modes. 303

49 Participant preferences between the manufacturer's (UR10e) free-drive mode and the implemented free-drive mode. 307

50 Estimated current consumption of the wearable system components. 357

51 Ergowear data acquisition (D1), participants' main anthropometric data. 363

52 MTwAwinda data acquisition (D2), participants' main anthropometric data. Some of the participants were also present in the first data collection. 364

53 Descriptive analysis of MTwAwinda and Ergoaware datasets 364

54 Average QAD (°) and inference time (ms) obtained from the ablation studies on the Ergowear dataset. Performance improvements with respect to the baseline are shown in bold and underlined, slight performance decreases (less than $1°$) are underlined, and larger performance decreases are reported in normal text. 365

55 Average QAD° and inference time (ms) results of the ablation studies on MTwAwinda data performance improvements are in bold and underlined, slight performance decreases ($< 1°$) are underlined, and performance decreases are in normal text, all in relation to the baseline results. 365

56 Top 10 CNN models considering F1-score (their hyperparameters and validation metrics in the epoch with the lowest loss) 368

57 Top 10 LSTM models considering F1-score (their hyperparameters and validation metrics in the epoch with the lowest loss) 368

58 Top 10 transformer models considering F1-score (their hyperparameters and validation metrics in the epoch with the lowest loss) 369

59 Top 10 hybrid CNN-LSTM models considering F1-score (their hyperparameters and validation metrics in the epoch with the lowest loss) 369

60 Top 10 CNN–Transformer models considering F1-score (their hyperparameters and validation metrics in the epoch with the lowest loss) 369

61 Participants' anthropometrics, exercise habits, and caffeine intake for Dataset 1. 374

62 Anthropometric characteristics of the participants included in Dataset 2. 374

63 sEMG feature extraction methods. 376

64 Metrics obtained from the Q-Learning (both discretization levels) pre-trained models' execution in the real-world environment for participant 1. 380

65 Metrics obtained from the DQN pre-trained model's execution in the real-world environment for participant 1. 381

66 Average of the performance metrics for the 10 participants during square wave, sine wave and spiral line trajectories 383

Acronyms

ADC Analog to Digital Conversion.

AI Artificial Intelligence.

BiLSTM Bidirectional Long Short Term Memory.

biRNN Bidirectional Recurrent Neural Network.

BLE Bluetooth Low Energy.

CMC Coefficient of Multiple Correlation.

CNN Convolutional Neural Network.

cobot Collaborative Robot.

CWT Continuous Wavelet Transform.

DHM Digital Human Model.

DL Deep Learning.

DMP Dynamic Movement Primitive.

DoF Degrees of Freedom.

DQN Deep Q Network.

DTW Dynamic Time Warping.

EDA Exploratory Data Analysis.

EE End Effector.

EKF Extended Kalman Filter.

ELM Extreme Learning Machine.

EMG Electromyography.

FC Fully Connected.

FFT Fast Fourier Transform.

FIR Finite Impulse Response.

FoF Factory of the Future.

FSR Force Sensing Resistor.

GAP Global Average Pooling.

GDP Gross Domestic Product.

GPR Gaussian Process Regression.

Grad-CAM Gradient Weighted Class Activation Mapping.

GRNN General Regression Neural Network.

GRU Gated Recurrent Unit.

GSR Galvanic Skin Response.

GT Ground Truth.

GUI Graphical User Interface.

HAR Human Activity Recognition.

HPE Human Pose Estimation.

HPO Hyperparameter Optimization.

HRC Human Robot Collaboration.

ICC Intraclass Correlation Coefficient.

iEMG Integrated Electromyography.

IK Inverse Kinematics.

IMU Inertial Measurement Unit.

IQR Interquartile Range.

IRR Infinite Impulse Response.

KF Kalman Filter.

KPI Key Performance Indicator.

LR Learning Rate.

LSTM Long Short Term Memory.

LUBA Postural Loading Upper Body Assessment.

MAV Mean Absolute Value.

MDF Median Frequency.

MDP Markov Decision Process.

MFFNet Multidimensional Feature Fusion Network.

MHT Maximum Holding Time.

ML Machine Learning.

MLP Multilayer Perceptron.

MNF Mean Frequency.

MoCap Motion Capture.

MSD Musculoskeletal Disorder.

MSE Mean Squared Error.

MVC Maximal Voluntary Contraction.

NARX Nonlinear Autoregressive Network with Exogenous Inputs.

NN Neural Network.

NWU North West Up.

OWAS Ovako Working Posture Analysing System.

pHRI physical Human-Robot Interaction.

POMCP Partially Observable Monte Carlo Planning.

POMDP Partially Observable Markov Decision Process.

PRISMA Preferred Reporting Items for Systematic Reviews and Meta Analyses.

PWM Pulse Width Modulation.

QAD Quaternion Angle Distance.

REBA Rapid Entire Body Assessment.

ReLU Rectified Linear Unit.

RL Reinforcement Learning.

RMS Root Mean Square.

RMSE Root Mean Square Error.

RNN Recurrent Neural Network.

ROM Range of Motion.

ROS Robotic Operating System.

RPi4 Raspberry Pi 4.

RTC Real Time Clock.

RTT Round-Trip Time.

RULA Rapid Upper Limb Assessment.

SDI Strap Down Integration.

sEMG Surface Electromyography.

SNR Signal to Noise Ratio.

SOA State of the Art.

SSC Slope Sign Change.

STFT Short Time Fourier Transform.

STS Sensor to Segment.

SUS System Usability Scale.

SVM Support Vector Machine.

TGCN Temporal Graph Convolutional Network.

TP True Positives.

UDP User Datagram Protocol.

WRMSD Work Related Musculoskeletal Disorder.

WS Window Size.

XAI Explainable Artificial Intelligence.

Chapter 1

Introduction

This research proposal presents the main considerations, challenges, and contributions related to the research and development activities undertaken over the past 4 years within the scope of the doctoral program in Electronics and Computers Engineering, on the field of Control, Automation, and Robotics. This PhD workplan was developed in the Biomedical Robotic Devices Laboratory (BiRDLab)-Center for MicroElectroMechanical Systems (CMEMS) of University of Minho (UMinho). Part of the research was also developed in the Massachusetts Institute of Technology (MIT), in Boston, USA. This research was supervised by Professor Doctor Cristina P. Santos, assistant Professor in the Department of Industrial Eletronics, researcher at CMEMS research unit and head of BiRDLab, and Professor Doctor Alexandre Ferreira da Silva, Professor in the Department of Industrial Electronics and researcher at CMEMS research unit, and Leonardo Gizzi, Research Associate and Head of Neuromechanics Laboratory at Universität Stuttgart, and Senior Research Scientist at Fraunhofer IPA - Institut für Produktionstechnik und Automatisierung IPA. Additionally, this Ph.D. included the support of Bosch Car Multimédia Portugal,S.A.(Bosch BrgP), through Engineer and Robocist Eurico Damásio and Ergonomist Ana Pombeiro, Principal Research Scientis at MIT's Institute of Medical Engineering and Science, Doctor Mercedes Balcells-Camps, and the director of MIT's Institute of Medical Engineering and Science and Harvard-MIT Biomedical Engineering Center, Professor Doctor Elazer Eldeman. This PhD was supported by a doctoral fellowship granted by MPP2030-FCT, with the refence SFRH/BD/151382/2021, and was inserted in the Research projects FAIR (2022.05844.PTDC) and INTEGRATOR (2022.15668.MIT). The research stay at MIT was supported by a Fulbright and a FLAD grants.

Large languages Models were used to make minor improvements to the writing. They were not used to generate any ideas presented in this work.

1.1 Motivation

Work Related Musculoskeletal Disorders (WRMSDs) represent a critical challenge in modern manufacturing, with a direct impact on productivity, workforce well-being, and economic sustainability. Manufacturing industries play a central role in the economic growth of Europe, but they also place significant physical demands on workers. WRMSDs account for 53% of all occupational diseases in the European Union [3, 4], leading to substantial economic losses estimated at EUR 240 billion annually, equivalent to 2% of the EU's Gross Domestic Product (GDP) [5]. In the UK alone, over 30.7 million working days were lost in 2018 due to workplace injuries, including 0.5 million WRMSDs cases, with costs reaching GBP 5.2 billion [6].

The most frequently affected areas include the back, shoulders, and neck, with 30.7% of the Portuguese working population reporting back pain [7]. Among the main contributors to WRMSDs are physical fatigue and awkward postures, affecting 35% and 50.5% of workers, respectively [4, 8]. These conditions not only impair workers' physical capabilities but also lead to absenteeism, reduced productivity (up to 36%), sick leave, and early retirement [9].

The problem is further exacerbated by a rapidly aging workforce. According to the European Commission's 2024 Ageing Report [10], although the participation rate of individuals aged 20 to 64 is projected to increase from 79.4 percent in 2022 to 82.7 percent by 2070, it will not be sufficient to counterbalance the decline in the working-age population. Between 2022 and 2070, the European Union is expected to lose approximately 25 million workers, corresponding to a 12 percent reduction in the labour force. In parallel, the number of older adults supported by the working population will increase substantially. By 2070, there will be only 14 employed individuals for every 10 persons aged over 65, compared to 22 in 2022. At the same time, the total number of hours worked is projected to decline by 9 percent. As a direct consequence of these combined trends, labour productivity will become the dominant determinant of economic growth.

These trends highlight the urgency to implement preventive, productivity-sustaining solutions in the workplace, particularly those that address physical strain and promote long-term worker well-being.

After an injury, workers experience a substantial reduction in productivity, estimated at 53% [11], together with increased absenteeism and a higher likelihood of early retirement [3]. Most workplaces are not designed to accommodate workers affected by Musculoskeletal Disorder (MSD). Many individuals continue working while experiencing pain [12]. However, they struggle to perform their tasks effectively [13]. This combination of reduced work capacity and inadequate workplace adaptation leads to significant productivity losses, with documented economic impacts such as an estimated cost of $32 billion in Australia, and

is associated with negative effects on workers' mental health [3]. Persistent pain and reduced work ability contribute to frustration and increase the risk of depression [13].

Together, these factors highlight the need for preventive and inclusive workstations for workers affected by WRMSDs, designed to accommodate the physical limitations associated with MSD.

Collaborative Robots (cobots) are part of many manufacturing industries. Still, most are used in co-existence with humans, mostly to avoid the physical barriers required by industrial robots to ensure workers safety. Thus, these cobots are not being explored to their full potential. Cobots, in general, still miss abilities to understand Humans and deliver the best assistance. Whilst the construction of cobots is mature, perception and cognition algorithms to provide safer and more sensitive cobots are still an unattended need [14]. Research in Human Robot Collaboration (HRC) is needed to advance robotics towards Industry 5.0, while seizing the beginning of the digital transformation era in manufacturing, and attaining an industry-readiness level.

Collaborative robots can enhance task performance and ensure product quality [15] by learning the proper execution of tasks and supporting workers when errors occur. Evidence indicates that production defects decrease when operators work with a cobot capable of intervening if the task is not performed correctly [15].

To attain a truly collaborative environment, that provides customized robot motion to both the users' and production demands [16], it is mandatory to address HRC. To successfully minimize WRMSDs risk, first cobots should become Human-aware, i.e., have an holistically perception of Human physiological and ergonomic factors. This demands unobtrusive and agile sensors as well as time-efficient algorithms to process and interpret the collected data. Sensory systems such as OptiTrack™ [17]), depth cameras (e.g., Microsoft Kinect [18], and wearable devices (e.g., Xsens MVN Awinda [19], EMG sensors [2]) have been used to enhance the robot's awareness of human intentions and states. However, with the exception of Kinect, these commercial systems are generally expensive, complex, and require time consuming setup procedures, which can negatively affect worker comfort and acceptance. Moreover, they are often difficult to integrate into deployed industrial solutions, limiting their applicability in real world scenarios. The Kinect, in turn, has the drawbacks inherent to the camera-based solutions such as the restricted detection area region and light conditions, and the susceptibility to occlusion. In contrast, IMUs and sEMG sensors integrated into smart garments offer a lightweight, cost effective, and unobtrusive alternative. These technologies enable continuous, real time monitoring of posture and fatigue, while providing information on physical limitations. As such, they support holistic and personalized ergonomic assessment, laying the foundation for intelligent cobotic assistance.

1.2 Problem Statement

Despite the growing adoption of collaborative robots and advances in wearable sensing technologies, several critical challenges remain in achieving truly intelligent and adaptive HRC for the prevention of WRMSDs.

First, cobots still lack the ability to perceive and respond to the human operator's physiological and ergonomic states, such as postural ergonomic risk and muscle fatigue.

As one of the main risk factors for WRMSDs, postural assessment has been widely addressed through wearable systems [20, 21] and integrated into HRC applications [22, 23]. These approaches demonstrate the feasibility of objective, sensor based ergonomic evaluation using IMU-based wearable sensing systems. However, as noted before adoption in industrial settings remains limited due to high overall costs, with proprietary software licenses [24] required for advanced biomechanical analysis often exceeding the cost of the sensing hardware itself. Although access to raw sensor data is possible without these licenses, its interpretation requires substantial technical and mathematical expertise, which further restricts widespread use. As a result, there is a need for improved wearable solutions that emphasize simplified and robust designs aligned with industrial constraints and worker acceptance, as well as data-driven methods that bypass the reliance on explicit biomechanical modeling. Such approaches would reduce the dependency on specialized expertise, thereby facilitating broader adoption and development of IMU-based wearable solutions. This thesis hypothesizes that deep learning based pose estimation can offer a data-driven alternative by directly learning the relationship between inertial sensor data and joint angles, without requiring explicit biomechanical models or manually defined kinematic constraints. By shifting the burden from model design to data-driven inference, these methods have the potential to simplify deployment while maintaining sufficient accuracy.

In contrast, muscle fatigue, which affects approximately 35% of European workers, particularly in the upper body [5], remains considerably less addressed in current HRC systems, despite its established role in the development of WRMSDs [25, 26]. While sEMG stands out as the most commonly used modality for monitoring muscle activity and fatigue [27–29], conventional Ag/Cl electrodes may cause skin irritation [30]. Moreover, disposable Ag/AgCl electrodes would require daily placement at specific anatomical locations, which limits practicality and user compliance in industrial settings. Their single use nature also introduces recurring consumable waste, raising sustainability concerns for long term deployment. Alternatives such as Delsys Trigno Avanti exist, providing reusable dry electrode solutions; however, they rely on adhesive interfaces that are sensitive to sweat and movement, and require expertise to accurate place the

sensor over specific muscle bellies, with suboptimal positioning directly affecting signal quality. This thesis investigates e-textile based sEMG sensors as an alternative, as they are reusable and can be seamlessly embedded into garments that hold the electrode in place. By integrating sensing directly into clothing, these systems can help reduce setup complexity, allowing workers to don the garment without additional preparation.

Regarding fatigue-based HRC control strategies, most approaches focus on monitoring a single muscle [17, 31], which leads to cyclic activation rest patterns and incomplete fatigue management. In addition, these strategies are either subject dependent or rely on self defined global fatigue thresholds to trigger assistance [2,31,32], assumptions that lack physiological grounding and presume uniform fatigue behavior across users . Fatigue progression is also insufficiently modeled, particularly the transition to fatigue [33,34], which is critical for early intervention. This thesis explores deep learning based fatigue estimation models trained on data annotated with three self perceived fatigue levels, including a transition to fatigue, with the aim of supporting subject independent modeling across users and muscle groups.

Wearable-based ergonomic assessment systems typically generate large volumes of joint level kinematic data [35–37], which can be difficult for ergonomists to interpret in practice [35–37]. Risk is often summarized in terms of the time a worker spends within predefined ergonomic risk levels, while the underlying information is limited to joint angles and their temporal evolution [35–37]. As a result, identifying which specific postures occur at particular phases of a work cycle requires manual interpretation, often involving visual reconstruction of joint configurations to infer the performed posture. This thesis proposes to reduce this interpretability gap by integrating human activity recognition, powered by AI into the ergonomic assessment pipeline. By automatically recognizing postures, ergonomic risk levels can be directly associated with discrete postural categories rather than isolated joint angles. This enables ergonomic reports to be context-aware and continuously quantify how much time is spent in each posture and when these postures occur within the work cycle, providing more actionable information for task analysis and redesign.

Prevention alone is insufficient, and inclusive measures are required to support workers once musculoskeletal disorders occur. MSDs account for 30% of all years lived with disability [38], leading to substantial productivity losses, absenteeism, and early retirement, with annual costs in Australia alone reaching $32 billion [38]. Despite this impact, human sensitive control strategies that account for physical limitations, such as reduced range of motion or task specific constraints, remain largely overlooked in current HRC systems. This thesis explores Reinforcement Learning (RL) as a learning based approach for capturing user specific physical restrictions. Its inherent decision making capabilities enable the development of adaptive controllers that evolve through interaction with both the human and the environment. RL has

demonstrated potential in human robot interaction, particularly through inverse reinforcement learning, which has been used to capture ergonomic preferences in tasks such as object handovers [39]. However, while RL has shown effectiveness in virtual environments, its application to real world scenarios remains challenging, particularly with respect to sample efficiency and the safety of exploratory actions [40].

Lastly, Industry 5.0 environments demand manufacturing workstations that are flexible, agile, and capable of adapting rapidly to mass product customization [41]. In this context, HRC has demonstrated a positive impact on productivity and task performance in manufacturing settings [42], particularly when cobots assist in detecting and correcting suboptimal task execution [15]. However, despite these benefits, interaction often remains limited to passive coexistence, lacking the dynamic and intention aware coordination required for robust and adaptive collaboration [43]. In this context, this thesis presents a control strategy that leverages admittance control to guide the user toward the desired path. This combination enables corrective interaction during execution, supporting users in learning the intended trajectory while preserving flexibility and responsiveness.

1.3 Goals and Research Questions

The main goal of this proposal is to develop an innovative HRC framework that enables continuous human monitoring and adaptive control. The framework builds on wearable motion capture systems and learning based control strategies to shape cobot assistance according to human state and task demands, through a digital representation of the operator's work. It aims to demonstrate how adaptive cobot responses can prevent or delay muscle fatigue, improve ergonomic support while accommodating user specific physical limitations, and enhance task quality.

To achieve this main goal, this proposal will attain the following intertwined objectives:

- **Objective 1**: To design, implement, and technically validate custom wearable sensing systems based on IMU and e-textile based sEMG for human motion and muscle activity monitoring. The objective focuses on establishing reliable hardware and firmware foundations for wearable data acquisition, emphasizing signal quality, power efficiency, and suitability for integration into garment-based systems. These sensing platforms are intended to serve as enabling technologies for future ergonomic and fatigue-related applications. Chapter 3 addresses this objective. This objective is evaluated through the Key Performance Indicator (KPI) 1, considering: For the IMU-based wearable system: (i) reliable wireless communication, quantified by a packet loss rate below 1%; (ii) mechanical robustness and integration stability, verified by uninterrupted data acquisition and the absence

of sensor detachment or cable failures during extended recording sessions. For the sEMG system: (iii) signal quality comparable to a commercial gold-standard system, quantified by a root mean square error (RMSE) < 20% when benchmarked against Delsys sEMG electrodes.

- **Objective 2**: To design, implement, and systematically evaluate learning based inertial sensor fusion approaches for human pose estimation, aiming to abstract explicit biomechanical modeling and analytical filtering while preserving accuracy, robustness, and real time feasibility. The proposed framework should estimate full body or upper body segment orientations directly from raw MARG data and be applicable across heterogeneous sensing conditions, including low cost wearable sensors and magnetically disturbed environments. To achieve this, multiple neural network based, hybrid, and classical fusion strategies will be developed and benchmarked using a synchronized inertial motion capture dataset acquired with both low end and high end hardware. Chapter 4.1 addresses this objective. It will be measured by KPI 2, namely: (i) orientation estimation accuracy quantified by an average Quaternion Angle Distance comparable to state of the art classical fusion filters; (ii) inference time below 5 ms;

- **Objective 3**: To design, implement, and technically validate an intuitive and holistic ergonomic assessment tool for postural risk evaluation, grounded in inertial data and ergonomic risk assessment standards. The tool should associate detected postures with ergonomic risk levels, accounting for cumulative posture-induced joint stress, and provide the foundation for future biofeedback strategies. To achieve this, a Human Activity Recognition (HAR) model will be developed and trained on a custom dataset specifically acquired for this purpose. Chapter 4.2 and 4.3 address this objective and will be measured by KPI 3: (i) HAR model performance > 95% accuracy and inference time < 0.2 s, (iii) development of an expedited and intuitive ergonomic assessment Graphical User Interface (GUI) for postural analysis.

- **Objective 4**: To design, implement, and technically validate time-efficient Artificial Intelligence (AI)-based methods for real-time muscle fatigue estimation that are subject and muscle independent, and to integrate these estimates into a fatigue-aware control strategy for human–robot collaboration. The objective is to enable timely detection and management of fatigue during task execution, supporting personalized assistance aimed at delaying the onset of muscle fatigue. Chapter 6 adresses this objective. KPIs: (i) fatigue level estimation accuracy above 70%; and (ii) reduction of fatigue exposure (muscle activity and self-perceived fatigue) > 30%.

- **Objective 5**: To conceptualize, develop, and evaluate a proof-of-concept reinforcement learning

based controller that personalizes cobotic assistance according to individual ergonomic characteristics and, when present, physical limitations. The objective is to enable the controller to learn ergonomically optimal collaboration strategies for each user while accounting for pain-related constraints, thereby benefiting both healthy workers through improved ergonomics and workers with physical limitations through adapted assistance. Chapter 7 addresses this objective. KPIs: (i) ergonomic risk maintained at low levels (average RULA score < 2.5); (ii) absence of pain risk according to the adopted machine-state-based pain proxy.

Objective 6: To design, implement, and validate a task quality management module for human–robot collaboration that integrates variable admittance control to support novice training while ensuring consistent task execution quality. The module aims to balance accuracy, physical effort, and usability by modulating assistance strategies in response to user performance and task demands. Chapter 8 addresses this objective. KPIs: (i) trajectory tracking error < 5 mm and required interaction force < 10 N; (ii) evaluation of user experience and system usability through questionnaires and standardized metrics, targeting a System Usability Scale (SUS) score > 70%.

The outlined objectives will allow to answer the following research questions:

- **Research Question 1** (Objective 1): How effective are commercial knitting threads for sEMG signal acquisition and electrical transmission, and to what extent can a textile-based sEMG electrode reliably acquire electromyographic signals? This RQ is answered in chapter 3.

- **Research Question 2** (objective 2): To what extent can data driven inertial sensing pipelines replace explicit biomechanical modeling for pose estimation, when compared across sensor quality and modeling strategies? This RQ is answered in chapter 4.

- **Research Question 3** (Objective 3): How can postural ergonomic risk be continuously and objectively quantified and attributed over time in a manner that is interpretable and usable for end users, and how do architectural design choices and spatio-temporal information from inertial sensors influence the performance and interpretability of the posture recognition models that enable this assessment? This RQ is answered in chapter 4.

- **Research Question 4** (Objective 4): To what extent can a deep learning model accurately estimate muscle fatigue in a subject-independent manner and inform a control strategy to effectively mitigate fatigue progression and enhance task endurance? This RQ is answered in chapter 6.

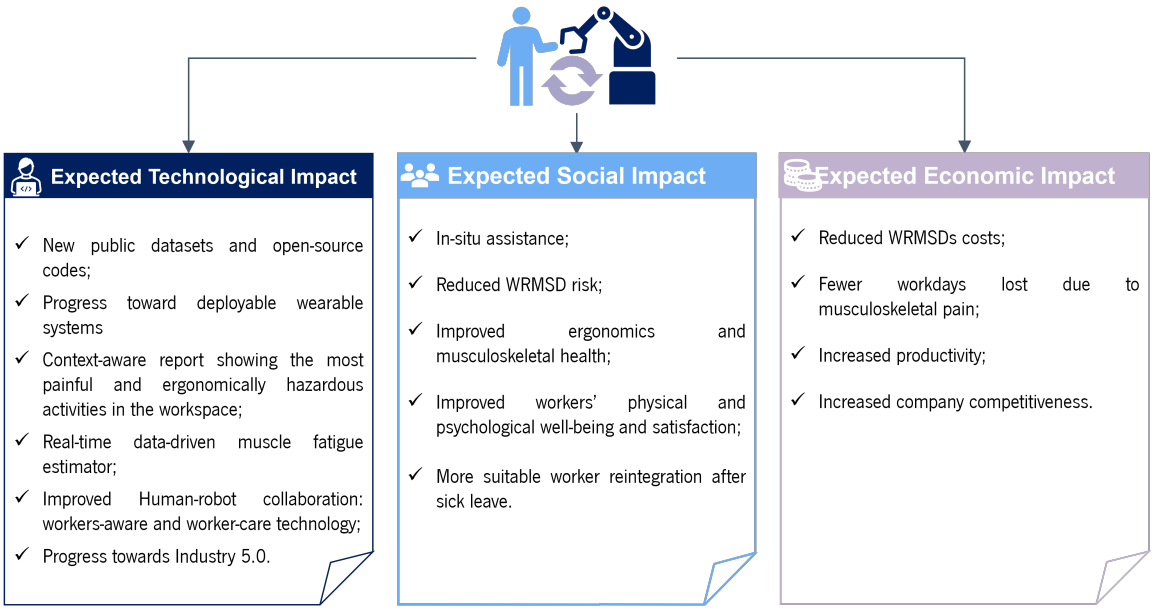


Figure 1: Expected impact of this work.

- **Research Question 5** (Objective 5): Can collaborative robots learn individual ergonomics and physical constraints to provide personalized, effective assistance during shared tasks? This RQ is answered in chapter 7.

- **Research Question 6** (Objective 6): How can an admittance-based control strategy be designed to support novice training and task quality management, and how do users interact with and experience this control approach? This RQ is answered in chapter 8.

Potential impacts resulting from the accomplishment of this PhD proposal's main goal and proposed objectives include significant technological advances able to upgrade industries and increase productivity. This step-forward towards innovation and industrial modernization improves companies' competitiveness, creating value to our society and economy. Socio-economic impact comes from contributions to reduce costs and number of WRMSDs. Moreover, in an aging society, with older workforce expected, these technologies can enhance companies by allowing them to motivate and give sense of purpose to operators who are aging or suffer from previous work-limiting WRMSDs. Figure 1 summarizes the expected impact of this work.

1.4 Contributions to Knowledge

- An extensive and critical literature review covering ergonomic assessment methods, biofeedback strategies

- The acquisition and public release of three novel datasets addressing IMU-based wearable sensing, posture recognition, and muscle fatigue assessment, contributing to open science by enabling reproducible research, fair comparison, and the training and evaluation of machine learning and deep learning models by the broader research community (**(Chapters 4 and 6)**).

- The design, development, and experimental characterization of e-textile-based sEMG electrodes for wearable muscle activity monitoring (**Chapter 3**).

- A systematic investigation of data-driven inertial pose estimation, assessing how far learning-based pipelines can replace classical estimation strategies without explicit biomechanical models through benchmarking across low-cost and high-end sensors and across learning-based, classical, and hybrid approaches (**Chapter 4**) .

- A comprehensive benchmarking analysis of deep learning models for human activity and posture recognition, evaluating convolutional, recurrent, transformer-based, and hybrid architectures under a unified experimental protocol for the recognition of eight posture classes, while highlighting the impact of architectural and training choices on model performance (**Chapter 4**).

- An intuitive and user-friendly ergonomic assessment computer application that integrates posture recognition with standardized ergonomic risk metrics, enabling real-time, expedited, and holistic postural risk evaluation, and whose technical validity was assessed through expert evaluation by professional ergonomists (**Chapter 4**).

- The development and validation of a subject- and muscle-independent deep learning approach for muscle fatigue estimation from sEMG signals, and its integration into a fatigue-aware control strategy for adapting collaborative robot assistance during task execution (**Chapter 6**).

- The development of human-sensitive controllers for human–robot collaboration that provide personalized assistance based on ergonomic risk and physical constraints, together with a benchmark of reinforcement learning approaches and a training methodology combining simulation and real-world adaptation (**Chapter 7**).

- The development of an admittance-based control strategy for task quality monitoring, capable of adapting robot motion in response to interaction forces and correcting user trajectories relative to a task-specific reference (**Chapter 8**).

1.5 Publications

The work here described allowed the publication of the following journal articles, conference papers, and book chapters. Contributions as second author or co-author, whether in journal articles, in conference papers or book chapters, relies on the conceptual design of the proposed solution, innovation or strategy, performing the experimental validation, and supporting the paper elaboration and review. Scopus profile recognizes a total of 17 publications, with only 5 papers not resulting from this thesis, 107 citations by 96 documents, and an h-index of 6. Google Scholar profile indicates a total of 203 citations and an h-index of 7. The following sources can be visited for further information: i) ResearcherID ; ii) ORCID 0000-0002-8097-5507; and iii) SciProfiles: 57208669452.

Journal Articles

- Martins, D. R., **Cerqueira, S. M.**, Pombeiro, A., da Silva, A. F., Rocha, A. M. A. C., Santos, C. P.
  ErgoReport: A Holistic Posture Assessment Framework Based on Inertial Data and Deep Learning.
  Sensors, 25(7), 2282, 2025.
  doi:10.3390/s25072282

- Martins, D. R., **Cerqueira, S. M.**, Santos, C. P.
  Seeking optimal and explainable deep learning models for inertial-based posture recognition.
  Knowledge-Based Systems, 306, 112700, 2024.
  doi:10.1016/j.knosys.2024.112700

- Martins, D. R., **Cerqueira, S. M.**, Santos, C. P.
  Combining inertial-based ergonomic assessment with biofeedback for posture correction: A narrative review.
  Computers and Industrial Engineering, 190, 110037, 2024.
  doi:10.1016/j.cie.2024.110037

- **Cerqueira, S. M.**, Vilas Boas, R., Figueiredo, J., Santos, C. P.
  A Comprehensive Dataset of Surface Electromyography and Self-Perceived Fatigue Levels for Muscle Fatigue Analysis.
  Sensors, 24(24), 8081, 2024.
  doi:10.3390/s24248081

- Palermo, M., **Cerqueira, S. M.**, André, J., Pereira, A., Santos, C. P.
  From raw measurements to human pose – a dataset with low-cost and high-end inertial-magnetic sensor data.
  Scientific Data, 9(1), 591, 2022.
  doi:10.1038/s41597-022-01690-y

- Martins, V., **Cerqueira, S. M.**, Balcells, M., Edelman, E. R., Santos, C. P.
  A Human-Sensitive Controller: Adapting to Human Ergonomics and Physical Constraints via Reinforcement Learning.
  Submitted to IEEE Transactions on Systems, Man and Cybernetics: Systems , 2025.
  Preprint available on ArXiv

- **Cerqueira, S. M.**, Palermo, M., Santos, C. P.
  Deep Inertial Pose: A deep learning approach for human pose estimation.
  Submitted to Engineering Applications of Artificial Intelligence
  Preprint available on ArXiv

Conference Articles

- Martins, V., **Cerqueira, S. M.**, Balcells, M., Edelman, E. R., Santos, C. P.
  Towards a Human-Sensitive Controller: Learning Human Specificities in Ergonomics and Physical Constraints.
  ICARSC 2025, pp. 158–163.
  doi:10.1109/ICARSC65809.2025.10970165

- Coelho, L., **Cerqueira, S. M.**, Martins, V., André, J., Santos, C. P.
  A DMPs-based Approach for Human-like Robotic Movements.
  ICARSC 2024, pp. 201–206.
  doi:10.1109/ICARSC61747.2024.10535923

- Martins, V., **Cerqueira, S. M.**, Coelho, L., Santos, C. P.
  An Admittance-based Control Strategy for Human-Robot Collaboration Task Quality Management.
  ICARSC 2024, pp. 207–212.
  doi:10.1109/ICARSC61747.2024.10535950

- Martins, D., **Cerqueira, S. M.**, Vieira, T. P., Pombeiro, A., Santos, C. P.

Towards a User-specific Ergonomics-based Approach for an Activity Assessment Tool.
ICARSC 2023, pp. 220–225.
doi:10.1109/ICARSC58346.2023.10129578

- Boas, A. V., Andre, J., **Cerqueira, S. M.**, Santos, C. P.
A DMPs-based Approach for Human-Robot Collaboration Task Quality Management.
ICARSC 2023, pp. 226–231.
doi:10.1109/ICARSC58346.2023.10129609

- Martins, J., **Cerqueira, S. M.**, Silva, A. F., Catarino, A., Rocha, A., Santos, C. P.
Towards a Smart-Vest for Forward Posture Monitoring: Improving Usability with E-Textiles.
ENBENG 2023, pp. 215–218.
doi:10.1109/ENBENG58165.2023.10175331

1.6 Thesis Outline

This Ph.D. thesis is organized into nine chapters. Chapter 1 introduces the research context, motivation, and objectives. Chapter 2 reviews the state of the art in wearable sensing, human activity recognition, and perception and control strategies for HRC, identifying key limitations and open challenges.

Chapter 3 presents the design, development, and technical validation of a smart garment instrumented with inertial measurement units and haptic actuators, together with the design, development, and characterization of e-textile-based sEMG electrodes and their acquisition system. Chapter 4 addresses inertial-based human motion analysis and ergonomic assessment of task execution, leveraging posture recognition to analyze exposure and posture distribution during representative work activities. In particular, section 4.2 investigates data-driven approaches for human pose and joint angle estimation, comparing deep learning, hybrid, and classical sensor fusion methods across low-cost and high-end inertial sensors. Chapters 4.3 and 4.4 focus on data-driven ergonomic reporting, covering human activity recognition and ergonomic risk assessment, respectively.

Chapter 5 introduces the ErgoAware HRC framework and details the computation of ergonomic metrics. Chapter 6 focuses on muscle fatigue estimation using biosignals, exploring learning-based approaches to detect fatigue progression during repetitive tasks and the associated control strategy.

Chapter 7 presents a reinforcement learning–based control strategy that personalizes cobotic assistance according to individual ergonomics and physical limitations, demonstrating how simulation-based training combined with real-world fine-tuning enables safe collaborative transport. Chapter 8 presents the

design and experimental evaluation of a task quality management control strategy based on admittance control, analyzing its effects on execution accuracy, physical effort, and user experience in the context of novice training.

Finally, Chapter 9 summarizes the main contributions, discusses limitations, and outlines directions for future work. Figure 2 provides an overview of the thesis topics.

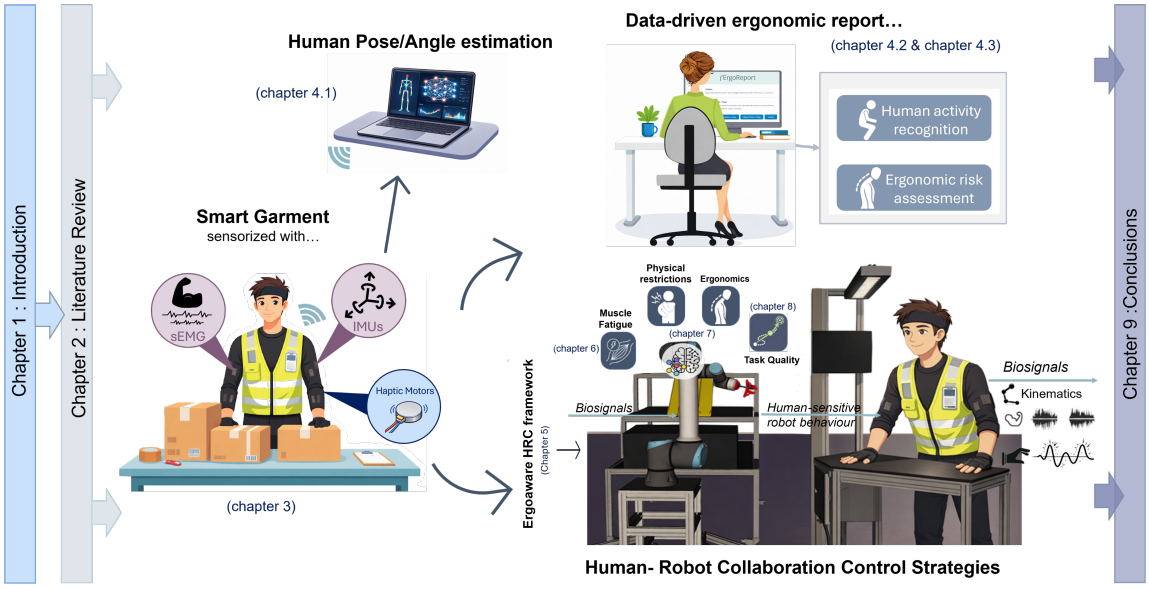


Figure 2: Overview of the thesis structure.

Chapter 2

Literature Review

This literature review surveys the state-of-the-art across four interconnected domains: **(i)** wearables for human awareness, focusing on sensing technologies that capture kinematics and posture for ergonomic risk assessment; **(ii)** HAR, which transforms wearable data into information on posture and movement, making ergonomic assessment more informative; **(iii)** perception algorithms, which exploit wearable data to estimate fatigue risk, enabling robots to interpret human fatigue state; and **(iv)** control strategies for supportive and personalized collaboration, addressing methods that allow robots to adapt their behavior in real-time to human needs. Together, these domains provide the basis for both ergonomist-oriented and robotics-oriented solutions aimed at reducing work-related musculoskeletal risks and improving worker well-being.

2.1 Wearables for ergonomic risk assessment

This section is based on the narrative review published in Computers & Industrial Engineering Journal, entitled ***Combining Inertial-based Ergonomic Assessment with Biofeedback for Posture Correction: a Narrative Review***. The review examines inertial sensor–based systems that both assess workers' postural ergonomics risk. With this work, we aim to answer the following questions: (i) "What types, number, locations and settings of wearable devices were adopted in the literature studies?"; (ii) "Which ergonomic criteria govern these postural assessments?". Although the journal manuscript also presents a biofeedback analysis, this component was not included here in order to limit the scope of the thesis.

2.1.1 Methods

An electronic search was conducted in the IEEE Xplore and Scopus databases, following PRISMA guidelines, covering studies from 2013 to 2023. Full details of the search strategy, keywords, inclusion/exclusion

criteria, and the PRISMA flow diagram are provided in Annex A.1.

2.1.2 Postural Ergonomic Assessment

This subsection reviews current ergonomic methods for posture assessment and how workers' postures are evaluated, whether by analyzing individual body parts separately or the body as a whole.

A summary of the relevant manuscripts found in the search on ergonomics is presented in Tables 1 and 2. The first divides the manuscripts according to the study goal, the ergonomic method/scale applied, highlighting the sensors used (their type, location and sampling frequency) and the biofeedback system (concurrently, i.e., in real time; or terminal, that is, presented after the trial). Table 2 summarises the algorithms used and the validation methodology of the identified studies, presenting the experimental protocols, ground truths, and metrics used for the validation.

Table 1: Summary of the included studies on postural ergonomics monitoring, regarding their goals, ergonomic methods, sensing systems, and biofeedback systems

Reference	Study Goal	Ergonomic Method	Sensors			Biofeedback System
			Number and Type	Location	Sampling Frequency	
[37]	Real-time local ergonomic risk assessment	Standard ISO 11226	2 Bluetooth YEI 3-Space IMUs (9-DoF)	Back of head and thoracic level	10 Hz	Visual and auditory (both concurrent)
[35]	Real-time local ergonomic risk assessment	Rapid Upper Limb Assessment (RULA) complemented with Postural Loading Upper Body Assessment (LUBA)	4 MPU-9250 IMUs (6-DoF)	Back of head, T4 and upper arms	100 Hz	Haptic (concurrent) and visual (terminal)
[23]	Real-time local ergonomic risk assessment	RULA	17 Xsens MVN Awinda IMUs (9-DoF)	Head, sternum, pelvis, shoulders, upper arms, forearms, hands, upper legs, lower legs and feet	20 Hz	N/A

(continued on next page)

(continued from previous page)

Reference	Study Goal	Ergonomic Method	Sensors			Biofeedback System
			Number and Type	Location	Sampling Frequency	
[44]	Local and global ergonomic risk assessment	RULA	17 Xsens MVN Awinda IMUs (9-DoF)	Head, sternum, pelvis, shoulders, upper arms, forearms, hands, upper legs, lower legs and feet	60 Hz	N/A
[20]	Real-time local and global ergonomic risk assessment	RULA	7 wireless Colibri IMUs (9-DoF) and 2 bi-axial SG65 goniometers	- IMUs: head, chest, sacrum, upper arms and forearms; - Goniometers: hands	100 Hz	Visual and auditory
[21]	Local and global ergonomic risk assessment	RULA	3 wireless Xsens MTw IMUs (9-DoF)	Back of head, thoracic level and sacral level	75 Hz	N/A
[45]	Local and global ergonomic risk assessment	RULA and Rapid Entire Body Assessment (REBA)	17 Xsens MVN Link IMUs (9-DoF)	Head, sternum, pelvis, shoulders, upper arms, forearms, hands, upper legs, lower legs and feet	N/M	Visual (terminal)
[46]	Local and global ergonomic risk assessment	REBA and K-score	9 wireless Xsens MTw IMUs (3-axis acc, 3-axis gyr and 3-axis mag)	Head, sternum, sacrum, right upper arm, right forearm, right hand, right thigh, right shank and right foot	100 Hz	N/A
[36]	Local and global ergonomic risk assessment + qualitative postures classification	Standard ISO 11226	8 wireless proprietary IMUs (9-DoF)	Upper back, lower back, upper arms, upper legs and lower legs	50 Hz	Visual (terminal)

(continued on next page)

(continued from previous page)

Reference	Study Goal	Ergonomic Method	Sensors			Biofeedback System
			Number and Type	Location	Sampling Frequency	
[47]	Qualitative postures classification + global ergonomic risk assessment	Ovako Working Posture Analysing System (OWAS) and Maximum Holding Time (MHT)	5 MetaMotionC IMUs (6-DoF)	Head, chest centre, right upper arm, right thigh and right calf	N/M	Visual (every 30 min and terminal) and auditory (concurrent)
[48]	Qualitative postures classification + global ergonomic risk assessment	OWAS and MHT	5 MetaMotionC IMUs (6-DoF)	Forehead, chest centre, right upper arm, right thigh and right crus	25 Hz (subject 1) 50 Hz (others)	N/A

N/A - Not Applicable; N/M - Not Mentioned; 6-DoF IMU: 3-axis acc + 3-axis gyr; 9-DoF IMU: 3-axis acc + 3-axis gyr + 3-axis mag

Table 2: Summary of the studies included on postural ergonomics monitoring, regarding their algorithms, validation protocols, ground truths, and metrics

Reference	Algorithms	Experimental Protocol	Ground Truth	Metrics
[37]	- Joint angle estimation using tilt-twist method; - Calculation of MHT for real-time warning threshold	- N = N/M; - Typical operations of construction workers on a construction site	**Technical validation:** N/A. **Ergonomic tool validation:** N/A	N/A
[35]	- Joint angle estimation using a Kalman filter; - Computation of the risk level through the discretization of joint angles based on thresholds (state machine)	- N = 5; - 5 general tasks, containing different working postures; - Each trial performed 4 times: 2 without biofeedback and 2 with it	**Technical validation:** - Collaborative robot arm. **Ergonomic tool validation:** N/A	**Technical validation:** - Root Mean Square Error (RMSE). **User performance:** - Trial execution time; - Time spent at each risk level. **System usability:** - Questionnaires following System Usability Scale (SUS)
[23]	- Joint angle estimation: N/M; - Computation of the risk level through the discretization of joint angles based on thresholds; - Calculation of an index that integrates the present and past risk scores	- N = 1; - Assembly of a corner joint with three aluminium profiles	**Technical validation:** N/A. **Ergonomic tool validation:** N/A	**User performance:** - Kinematic wear index
[44]	- Computation of the risk level through the discretization of joint angles based on thresholds; - Calculation of an index that integrates the present and past risk scores	- N = 3; - Performance of isolated postures, construction task circuit, and agriculture task circuit	**Technical validation:** N/A. **Ergonomic tool validation:** N/A	**User performance:** - LUBA scores; - Time spent at each risk level; - Kinematic wear index

(continued on next page)

(continued from previous page)

Reference	Algorithms	Experimental Protocol	Ground Truth	Metrics
[20]	- Biomechanical model of the upper body; - Joint angle estimation by means of loosely coupled extended Kalman filters; - Computation of the risk level through the discretization of joint angles based on thresholds	- N = 12; - Industrial manual task, composed of 4 subtasks; - 6 subjects without real-time biofeedback and 6 with it	**Technical validation:** N/A. **Ergonomic tool validation:** N/A	**User performance:** - Trial execution time; - Time spent at each risk level; - Frequency of threshold breaching. **System usability:** - 5-point Likert scale questionnaire; - Subjective observations
[21]	- Joint angle estimation: N/M; - Time window segmentation; - Discretization of joint angles based on thresholds; - Computation of the risk level considering a time threshold	- N = 1 (surgeon); - Real laparoscopic surgery operation	**Technical validation:** N/A. **Ergonomic tool validation:** - Camera recording	**User performance:** - RULA score; - Time spent at each risk level
[45]	- Joint angle estimation: N/M; - Computation of the risk level through the discretization of joint angles based on thresholds; - Estimation of joint contact forces and moments; - Computation of lower-back compression force and strength percent capable of joints	- N = 20 (healthy); - 15 experimental tasks derived from three common jobs in the shipbuilding process (manual handling, painting and welding)	**Technical validation:** N/A. **Ergonomic tool validation:** - For posture: expertise assessment through video camera recordings; - For biomechanical analysis: 3DSSPP	**Ergonomic tool validation:** - For posture: Intraclass Correlation Coefficient (ICC); absolute difference between scores; accuracy; - For biomechanical analysis: Coefficient of Multiple Correlation (CMC); relative error. **User performance:** - RULA and REBA scores; - Time spent at each risk level

(continued on next page)

(continued from previous page)

Reference	Algorithms	Experimental Protocol	Ground Truth	Metrics
[46]	- Joint angle estimation: Xsens proprietary sensor fusion; - Computation of the risk level through joint angles' discretization based on thresholds	- N = 10; - Material handling task	**Technical validation:** - Camera-based MoCap (Vicon). **Ergonomic tool validation:** - EMG (Trigno, Delsys)	**Technical validation:** - RMSE; - Cohen's Kappa coefficient. **Ergonomic tool validation:** - Spearman correlation coefficient. **User performance:** - REBA and K-score scores
[36]	- Joint angle estimation: N/M; - Computation of the risk level through the discretization of joint angles based on thresholds (state machine)	- N = 6 (not seriously injured in the last year); - Bricklaying routine tasks, replicating real working environments	**Technical validation:** N/A. **Ergonomic tool validation:** - Expertise assessment through video camera recordings	**User performance:** - Productivity Score; - Posture Score
[47]	- Deep neural network model for posture recognition; - Time threshold-based application of the ergonomic rules	- N = 30 (18 construction workers + 12 managers); - Routine construction tasks (6 trades)	**Technical validation:** N/A. **Ergonomic tool validation:** - Postures manually labelled through video camera recordings	**User performance**: - Count of MHT breach; - Total duration of MHT breach; - Detected MHT time; - Posture frequency; - Posture proportion. **System usability:** - 5-point Likert scale questionnaire

(continued on next page)

(continued from previous page)

Reference	Algorithms	Experimental Protocol	Ground Truth	Metrics
[48]	- Incremental learning model for posture recognition; - Time threshold-based application of the ergonomic rules	- N = 9 (construction workers); - Routine construction tasks (5 trades)	**Technical validation:** N/A. **Ergonomic tool validation:** - Postures manually labelled through video camera recordings	**User performance**: - Count of MHT breach; - Total duration of MHT breach; - Detected MHT time; - Posture frequency; - Posture proportion; - OWAS ergonomic risk level

N/A - Not Applicable; N/M – Not Mentioned; N - number of participants

Ergonomic Methods Adopted

Two main levels of postural ergonomic assessment were identified. A local approach evaluates multiple body segments independently, assigning a risk level to each, which supports WRMSD prevention by identifying segments exposed to higher ergonomic risk. In contrast, a global assessment assigns a single risk score to the whole body by combining information from multiple segments. Three of the reviewed studies applied a global assessment. In two cases, the evaluation relied on subjective or qualitative posture descriptions [47, 48], while the third used angular ranges [36]. The first two studies also included posture classification, which in [47] was required to perform the ergonomic assessment. Most studies based their assessments on angular measurements, particularly those adopting a local approach. Four studies focused exclusively on segment-specific assessment [23, 35–37], while three combined local assessment with a global score to provide a broader view of WRMSD risk [20, 21, 44–46]. Real time assessment was implemented in four studies [20, 23, 35, 37].

Several ergonomic criteria were found to be the basis of the posture assessments: Standard ISO 11226 [36, 37], RULA [20, 21, 23, 35, 45], LUBA [35, 44], REBA [45, 46], OWAS [47], and Maximum Holding Time (MHT) [47], and a novel K-score proposed by martinez2022k. With respect to the ergonomic methods that assess individual segments, the majority focus only on the sagittal plane. In connection with this, [21] pointed out some gaps in RULA tool, namely, the lack of angular thresholds to evaluate torsion and lateral bending. In order to mitigate that, they chose a threshold of three times the precision of the IMUs, for robust recognition, and [35] combined two ergonomic methods, with RULA assessing the sagittal plane and LUBA the coronal plane.

Hardware and System Characteristics

With regard to the sensors used to record body movement, all reported studies used IMUs, containing 3-axis accelerometers and 3-axis gyroscopes [35, 47, 48], or combining those with a 3-axis magnetometer [20, 21, 23, 36, 37, 44–46]. There is no agreement about the number and location of sensors. The number of sensing units varied between 2 and 8, except for 3 studies that used commercial systems which contain 17 IMUs: Xsens MVN Link, which embeds the sensors in a suit [45], and Xsens MVN Awinda [23, 44]. The IMUs were incorporated into a personal protective equipment by [37]. [35] developed a smart garment to embed the sensors, actuators and remaining hardware. Figure 3 displays the sensor placement in the studies, where it stands out that all but one placed a sensing unit on the head. Regarding limbs, [20, 23, 35, 36, 44, 45] placed inertial sensors on both left and right limbs, whereas [47, 48] collected data from the right limbs only. Two goniometers on the hands were added in a study to record wrist

angles [20], in order to complement the inertial system, since distal segments are the most challenging segments to achieve a lower joint angle estimation error. Sampling frequencies ranged from 10 Hz [37] to 100 Hz [20, 35], with two studies not reporting this value [45, 47].

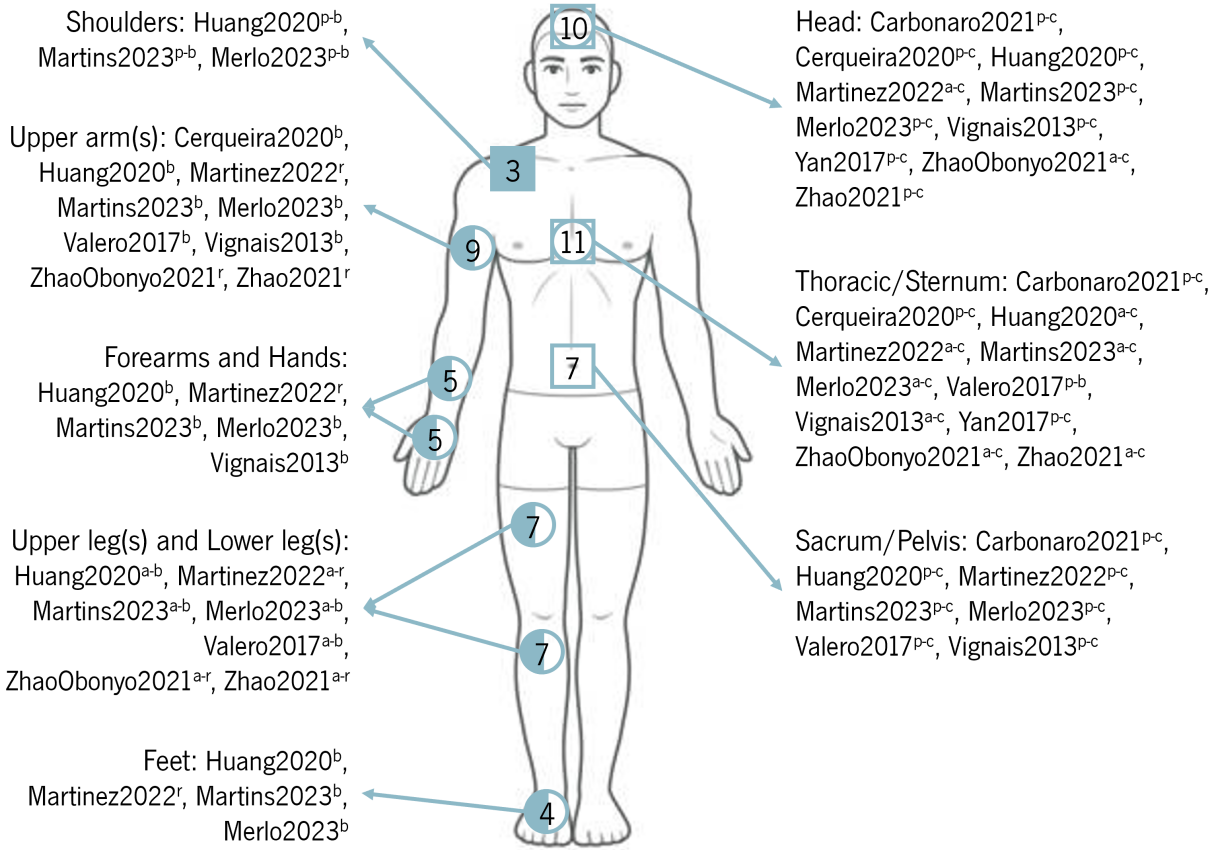


Figure 3: Body locations of the sensors and number of included studies that chose them. Circles represent sensors in the anterior (^{a}) part of the body, squares the ones in the posterior (^{p}) part of the body, filled symbols the ones placed bilaterally (^{b}), and empty ones unilaterally (either left, ^{l}, centre, ^{c}, or right, ^{r}).

Of these articles, six reported a biofeedback system. All of them included visual type: [35, 36, 45] showed relevant information about the ergonomic risk after the trial; Real-time auditory signals were delivered in three studies [20, 37, 47]. Local haptic cues were also given in real time by [35].

Algorithms for Ergonomic Assessment

Regarding the algorithms used to estimate ergonomic risk from inertial data, most studies followed a common pipeline: joint angles were estimated through sensor fusion using Kalman filtering [20, 35], tilt-twist decomposition [37], sensor-to-segment transformation from Xsens proprietary orientation estimates [46], or a commercial Motion Capture (MoCap) system [44]. Then joint angles are discretized into risk levels based on the thresholds defined by the adopted ergonomic method [20, 21, 23, 35–37, 44–46].

From this common baseline, studies diverged in how they quantified and accumulated risk over time. Some introduced cumulative exposure metrics: [23] and [44] proposed a kinematic wear index inspired by RC circuit analogies, where risk accumulates during work and dissipates during recovery, and [37] accumulated MHT in an angle-dependent manner per joint plane. Others focused on condensing multi-segment information into global scores: [36] computed a weighted posture score across body parts alongside a productivity score, while [21] assigned risk based on the highest level sustained beyond a minimum time threshold. [46] introduced the K-score, combining upper- and lower-limb posture scores to provide higher sensitivity to joint-angle changes than conventional categorical ergonomic scales. [45] took a different direction entirely, extending postural analysis with a static biomechanical model to compute lower-back compression force, addressing the known oversimplification of standard ergonomic tools for critical segments.

Finally, [47] and [48] departed from angle-based methods altogether, using deep learning to classify postures into OWAS categories, with overall risk driven by the most restrictive segment threshold and posture proportion limits over working time.

System's Validation

The systems were validated with experimental protocols involving, in several cases, construction or construction-related manual tasks [36, 37, 44, 45, 47, 48], given construction's leading role in WRMSD reports [49]. Some studies recruited actual construction workers [37, 47, 48], while others used participants without sector-specific experience. Beyond construction, [44] also designed an agriculture circuit and isolated posture trials, [21] evaluated a real surgery operation, [20, 23, 46] used industrial tasks, and [35] opted for general multi-profession postures. No study recruited participants with musculoskeletal disorders. Participant counts ranged from 1 [21, 23] to 30 [47], with [37] not reporting this number.

Five studies recorded trials on video [21, 36, 45, 47, 48], with all but [21] using expert video-based posture labelling as ground truth for ergonomic tool validation. [45] additionally used 3DSSPP as reference for lower-back compression force. [35] skipped ergonomic tool validation but performed technical validation of joint angle estimation against a collaborative robot arm. [46] also performed technical validation during the experimental tasks, using a camera-based MoCap system with reflective markers as reference. Four studies [20, 23, 37, 44] provided no ground truth of any kind.

Metrics were organised into four categories. *Technical validation* metrics assess sensor and angle estimation accuracy; two studies performed this type of validation. [35] reported an RMSE between 2.57° and 4.95° (1.43–2.5% of the full range), while [46] reported mean RMSE values below 4° for all joint

angles except the neck, and substantial agreement between IMU-based and Vicon-based ergonomic scores, with mean Cohen's Kappa coefficients of 0.67 or higher. Notably, [45] cautioned that their results relied on a commercial MoCap system and that low-cost inertial sensors with sensor fusion could yield worse outcomes.

Ergonomic tool validation metrics assess the trustworthiness of the computed ergonomic scores. [45] reported an ICC of $\sim$0.83, absolute score differences below 1.0, and accuracy above 88%, showing high consistency with expert ratings; lower-back compression force agreed well with 3DSSPP (CMC $>$ 0.89, relative RMSE $<$ 9.5%). [46] evaluated the relationship between Electromyography (EMG) amplitude and ergonomic scores, reporting that the K-score, unlike REBA, showed significant correlation with fatigue-induced changes in EMG amplitude. [36] and [21] confirmed correct classification of harmful postures qualitatively, without providing specific metrics.

User performance metrics quantified ergonomic behaviour during tasks. Standard scores such as RULA, LUBA, and REBA, alongside time spent at each risk level, were reported by several studies [20, 21, 35, 44–46]. [46] also reported the K-score as a more sensitive posture-based risk indicator. [23, 44] used a kinematic wear index capturing joint ergonomic history, and [20, 35] also compared trial execution times with and without feedback. [36] introduced a productivity score, which improved with experience, unlike the posture score. [47, 48] assessed MHT breach counts, durations, and posture proportions, finding that deep learning misclassification of awkward postures led to underestimation of MHT breaches and OWAS risk, though overall posture proportions and risk levels were correctly identified.

Finally, *system usability* metrics captured user satisfaction through questionnaires: [35] applied SUS guidelines, and [20, 47] used 5-point Likert scales. Across all cases, participants showed high acceptance of the wearable sensors, rating them as comfortable and non-intrusive.

2.1.3 Discussion

Overall, this literature review demonstrated the potential of wearable technology to not only monitor and/or assess posture. The key ideas presented regarding sensing and actuation technology, and ergonomic methods are discussed in the present section to answer the research questions. Accordingly, current gaps and future directions are pointed out.

What Types, Number, Locations and Settings of Wearable Devices Were Adopted in the Literature Studies for ergonomic assessment?

Currently, the most trendy wearable technology in data collection are IMUs. The number of degrees of freedom (DoFs) of the sensing devices has a considerable impact on angle estimation. It is frequent to find studies that only combine accelerometer and gyroscope. However, most studies also consider magnetometer data (9 DoF). With 6-DoF IMUs, orientation data from the gyroscope is corrected only with information from the accelerometer (roll and pitch angles), whereas 9 DoF join magnetometer data, which provides a global heading direction based on a magnetic field (yaw angle), allowing a higher accuracy and robustness against drift over time [20, 50]. Nonetheless, magnetometers can be affected by magnetic disturbances [20, 51], and they increase the sensing system's cost as well. Considering all studies, the number of IMUs is not consensual; it depends on the complexity of the postural assessment. Apart from that, the level of intrusiveness and inconvenience was rated as low by real workers [47], reporting that the sensors did not reduce their productivity.

Regarding ergonomic assessment, excepting [45], who employed the REBA method, the authors focused only on the upper body since most ergonomic methods do not give much relevance to the lower limbs. This is reflected in the sensors' body location: the majority of these studies did not collect information about lower-limb motion, but the ones that do, place one sensor on the upper leg and another on the lower leg, being able to measure knee angle. To measure head inclination, a sensor is placed on the head in most systems. Since the trunk is one of the most critical segments to musculoskeletal disorders, all studies from Section 2.1.2 measured this angle, using one or two IMUs placed in the back or chest. Almost all of the selected studies placed IMUs on either the shoulder(s) or the upper arm(s), in order to track shoulder joint movements. Few studies addressed forearms/hand movements since inertial sensors have higher angle estimation errors for extremity joint movements [52]. The accuracy of the collected data is dependent on the choice of the sensors' locations [37]. Moreover, the sensors must not interfere with other safety equipment, as [47] reported to happen.

From the included studies, three integrated sensors and all hardware into a smart garment ready to be worn autonomously [35, 50, 53], whereas one used a commercial suit with inertial sensors [45].

The choice of the sampling frequency depends on the temporal dynamics of the activities, and this frequency must always be superior to the human movement maximum frequency, 10 Hz, although the major frequencies range between 0.3 and 3.5 Hz [54]. However, the more samples per second, the greater the demands in terms of a fast response from the orientation filter, in order to achieve real-time ergonomic assessment or, at least, online.

Which Ergonomic Criteria Govern These Postural Assessments?

Ergonomic tools can be divided into local risk assessment, global, or both, where local scores assigned to individual body segments are combined into an overall risk value, as already proposed by methods like RULA or REBA. The combined approach was the most common in the literature, as it provides richer information on both the overall WRMSD risk and the most exposed body parts. All included studies expressed postural risk through quantitative scores, with most relying on angular values per articulation DoF as inputs; only [47, 48] used nominal qualitative postures instead. It is worth noting that angular threshold-based assessments depend on a user-defined neutral posture reference [35, 36, 51, 55], meaning that improper initial alignment or bad postural habits may lead to misjudged risk levels. A related gap concerns dynamic activities: most ergonomic methods were originally developed for static postures, and no established standard exists for dynamic assessment [36].

Beyond input type, the ergonomic methods themselves carry structural limitations. Most pay insufficient attention to motions outside the sagittal plane, forcing authors to define their own thresholds for movements like arm abduction or trunk lateral bending [20]. Despite this, RULA remains the dominant tool, largely due to its good inter-observer reproducibility [21]. Another recurring simplification concerns the lumbar zone, the body part most affected by WRMSDs, which most tools handle inadequately [20, 45]. One study tackled this directly by complementing postural analysis with a biomechanical model to estimate lower-back compression force [45].

A further limitation shared by most systems is the disregard for cumulative exposure, i.e., the influence of time previously spent at each risk level. Different studies approached this problem in different ways: [21] assigned the risk score as the highest level sustained beyond a minimum time threshold; [48] considered posture proportions over working time and tracked MHT breaches; [37] accumulated MHT per joint plane following ISO 11226; and [23, 44] introduced a cumulative ergonomic index inspired by the RC circuit time response, which increases as the user sustains risky postures and recovers otherwise.

Concerning validation, none of the studies validated angle estimation against another MoCap system online, and only [35] conducted an offline technical validation, albeit on a robot arm rather than on the human body. As noted by [56], this is a critical gap: without a known estimation error, the reliability of the ergonomic assessment itself cannot be fully trusted. Most experiments were also conducted in controlled lab environments rather than real field conditions, which limits the ecological validity of the results. On the subjective side, questionnaire-based evaluations consistently pointed towards good technology acceptance [35, 50, 53, 55, 57, 58], intuitiveness [35, 58, 59], perceived usefulness [35, 47, 60], and wearability [35, 53, 55]. Some workers reported that the system helped them identify and correct risky postures [47],

though a few noted that feedback occasionally caused distraction or stress at first [50].

Key Limitations and Future Directions

This review highlights both the potential and limitations of wearable technologies for postural ergonomic assessment. Existing systems can identify hazardous tasks and workplace conditions, supporting interventions to reduce WRMSD risk, yet their methodological foundations remain constrained. Most approaches rely on ergonomic tools that adopt local, global, or combined assessment strategies, with the latter being the most prevalent due to its ability to capture both overall risk and segment-specific exposure, as in RULA and REBA. In all cases, ergonomic risk is quantified using posture scores based on joint angles measured relative to a neutral posture, which introduces user-dependent variability [35, 36, 51, 55].

Current methods exhibit several gaps: they focus on static postures [36], emphasize sagittal-plane movements while neglecting motions such as arm abduction or trunk rotation [20], and often oversimplify critical areas like the lumbar spine [20, 45]. Some studies complemented postural with biomechanical analyses [45], and others considered time spent in risky postures using cumulative or time-weighted scoring [21, 23, 36, 37, 47, 48], though such approaches remain uncommon.

RULA was the most frequently applied tool due to its reproducibility [21], while REBA was less frequently used [45, 46]. Most studies focused on the upper body, placing sensors on the back, shoulders, or arms, with fewer addressing the lower limbs or forearms [52]. Sensor placement critically affected accuracy [37], though intrusiveness was generally rated low by workers [47]. The number of DoF also influenced performance: 9-DoF IMUs improved robustness but increased cost and susceptibility to magnetic disturbances [20, 50, 51].

Validation remains a major gap. No study validated angle estimation online against MoCap, although one performed an offline test with sensors mounted on a robot [35]. Most experiments were lab-based, limiting ecological validity and raising concerns about the reliability of ergonomic assessments [56].

A remaining translational gap concerns the use of commercial inertial MoCap systems for routine occupational ergonomic assessment. Systems such as Xsens provide multi-segment kinematic data, but their cost, setup complexity, and dependence on proprietary software may limit accessibility and large-scale workplace deployment. This reinforces the need for wearable systems that preserve sufficient ergonomic coverage while reducing cost, calibration requirements, and user burden.

2.2 Human Activity Recognition for Ergonomic Assessment

While wearable technologies provide direct measurements of human motion, their full value emerges when these signals are contextualized. Human activity recognition (HAR) enables continuous and objective identification of postures and movements, linking ergonomic risk not only to the worker's condition but also to the specific activities during which risk is generated. By situating ergonomic assessment within the context of tasks and postures across a work shift, HAR makes evaluations more informative and actionable, highlighting when and how hazardous situations occur. This subsection reviews recent trends in AI-based models for automatic HAR, with particular emphasis on approaches that leverage wearable sensing modalities such as IMUs, as introduced in Section 2.1. It aims to answer *(i) What sensors, sensor placements, sampling frequencies, algorithms, and training pipelines have been employed in recent HAR models based on wearable technologies?*

2.2.1 AI-based activity recognition models

A systematic literature search on AI-based activity recognition models was conducted in the Scopus database in January 2023. Full details of the search strategy, keywords, inclusion/exclusion criteria, and the PRISMA flow diagram are provided in Annex A.2.

Table 3 presents the goals and sensors used in these studies, and Table 4 summarizes the recognized classes, data preprocessing steps, AI algorithms, the dimensions of model input, train/validation/test split, data acquisition protocol, the Ground Truth (GT) used to validate and evaluate the results, and the metrics utilized. The recognized classes were divided into three kinds or levels: **static postures**, like standing, squatting or kneeling; **dynamic postures**, such as walking, pushing/pulling or transitional movements between postures; and **activities**, i.e., more complex movements which integrate different postures performed sequentially, like vacuuming or playing basketball.

Table 3: Summary of studies on activity recognition, highlighting study goals and sensing systems.

Reference	Study Goals	Sensors (Type)	Location	Sampling Frequency
Chen et al. [61]	Daily life activity recognition model	4 Shimmer IMUs (6-DoF)	Right wrist, hip, chest, left ankle	204.8 Hz
Kuncan et al. [62]	Daily life activity recognition model	5 Xsens MTx IMUs (9-DoF)	Thighs, chest, wrists	25 Hz
Luptáková et al. [63]	Daily life activity recognition model	Smartphone (6-DoF)	Waist (bag)	100 Hz
Kim et al. [64]	Daily life activity recognition model	WISDM: smartphone (3-axis acc); UCI-HAR: smartphone (6-DoF)	WISDM: upper leg (pocket); UCI-HAR: waist	20 Hz (WISDM); 50 Hz (UCI-HAR)
Kim et al. [65]	Working posture recognition (construction)	2 Wi-Fi proprietary IMUs (9-DoF)	Neck, lower back	30 Hz
Zhao et al. [66] and [48]	Working posture recognition (construction)	5 MetaMotionC IMUs (6-DoF)	Forehead, chest, right upper arm, right thigh, right calf	25 Hz (subj. 1); 50 Hz (others)
Yan et al. [67]	Working posture recognition (construction)	8 proprietary IMUs (9-DoF)	T3–T4, L5–S1, upper arms, forearms, thighs	10 Hz
Jang et al. [68]	Bad head and shoulder posture recognition	4 EBIMU24GV3 wireless IMUs (acc + mag)	Right tragus, C7, acromions	10 Hz

Notes: N/M = Not Mentioned; 6-DoF: 3-axis acc + 3-axis gyr; 9-DoF: 3-axis acc + 3-axis gyr + 3-axis mag.

Table 4: Summary of studies on activity recognition, including classes, preprocessing, AI algorithms, input, dataset split, protocols, ground truth (GT), and metrics.

Reference	**Classes**	**Preprocessing**	**AI Algorithms**	**Model Input**	**Dataset Split**	**Experimental Protocol**	**GT**	**Metrics**
Chen et al. [61]	13 SP, DP, AC	Sliding windows (2.5s, 50% overlap)	EEMD feature extraction; sparse logistic regression (Bayesian reg.); FLS-SVM	N/M	Leave-one-subject-out CV	N=23, protocol N/M	Labels via Android app	Accuracy
Kuncan et al. [62]	19 SP, DP, AC	5s non-overlapping segments	Motif pattern frequencies + LSTM	9120 windows (WS$\times$45)	10-fold CV	N=8, each activity 5min	One label per trial	Accuracy, Recall, Precision, F1
Luptáková et al. [63]	18 SP, DP, AC	Paired double-length windows; norm. (mean=0, var=1)	Transformer	83,129 windows (3s$\times$6)	70/15/15% split	N=90, per-activity records	One label per trial	Accuracy, Precision, Recall, Macro F1
Kim et al. [64]	6 SP+DP (WISDM/UCI-HAR)	Sliding windows (50% overlap); SMOTE balancing	Transformer	WISDM: 1,098,207 (80$\times$3); UCI-HAR: 1,318,272 (128$\times$6)	70/30% split	WISDM N=36; UCI-HAR N=30	WISDM: Android app; UCI-HAR: video labels	Accuracy, Macro F1, Epoch time, Test time
Kim et al. [65]	14 SP+DP	Normalization; sliding windows (1/6 overlap)	Two-stacked LSTM	32,954 + 9,572 + 10,738 windows (6$\times$18)	49/21/30% split	N=4 (pile), N=3–4 (asphalt); real sites	Video labels	Accuracy, PR-AUC

Reference	Classes	Preprocessing	AI Algorithms	Model Input	Dataset Split	Experimental Protocol	GT	Metrics
Zhao et al. [66]	7 SP+DP	Down-sampling; channel norm.; sliding windows (50%)	ConvLSTM (CNN+2 LSTM layers)	N/M, windows 1s$\times$30	72/18/10% split	N=4, 20–30min routine tasks, construction sites	Video labels	Macro F1
Zhao et al. [48]	9 SP+DP	Same as [66]	Incremental learning (ConvLSTM-based)	Same as [66]	72/18/10% split	N=9, routine tasks, construction sites	Video labels	Macro F1
Yan et al. [67]	15 SP+DP	Sliding windows; drop multi-label windows	CNN (ResNet-18)	16,326 windows (60$\times$19)	60/20/20% split	N=15, 15 isolated motions + 3 simulated tasks	Video labels	Accuracy
Jang et al. [68]	12 SP	N/A	MLP, CNN	MLP: 93,291 vectors (24 feats.); CNN: 93,291 (4$\times$6)	70/15/15% split	N=28, forced neck-/shoulder postures, lab env.	Webcam markers	Accuracy

Notes: SP = static postures, DP = dynamic postures, AC = activities. EEMD = Ensemble Empirical Mode Decomposition.

FLS-SVM = Fuzzy Least Squares-SVM. N/M = Not Mentioned, N/A = Not Applicable, N = participants.

Hardware and System Characteristics

All the analysed studies used IMU sensors with 3-axis accelerometers, either combined with a 3-axis gyroscope [48, 61, 63, 66], or a 3-axis magnetometer [68], or both, performing 9 Degrees of Freedom (DoF) [62,65,67]; still, [64] used an accelerometer-only dataset. Some studies used only 1 sensor [63,64], others 2 [65], 4 [61, 68], 5 [48, 62, 66], and one used 8 [67]. These sensing units were placed on head [48, 66, 68], neck/cervical spine [65, 68], thoracic spine [67], lower back [65, 67], chest/sternum [48, 61, 62, 66], hip/waist [61, 63, 64], both shoulders [68], upper arm(s) [48, 66, 67], forearms/wrist(s) [61, 62, 67], upper leg(s) [48, 62, 64, 66, 67], lower leg(s) [48, 66] and left ankle [61]. Some studies placed IMUs on both left and right sides of the body [62, 67], while others collected data from only one side [48, 61, 66, 68]. The sampling frequencies were 10 Hz [67, 68], 20 Hz [64], 25 Hz [62], 30 Hz [65], 50 Hz [48, 64, 66], 100 Hz [63], and 204.8 Hz [61].

Types of Recognised Classes

Following the notation defined above, different kinds of classes were recognised: one study only distinguished static postures [68], while others identified static and dynamic postures too [48, 64–67], and others combined static and dynamic postures with activities [61–63]. In these cases, the activities were associated with daily life or sports. The number of distinct classes varied between 6, in [64], and 19, in [62].

Algorithms for Activity Recognition

Regarding the models, most studies employed neural networks, with the exception of Chen et al. [61], who adopted a Machine Learning (ML) based approach. Their method combined statistical and time frequency features extracted using Ensemble Empirical Mode Decomposition, followed by normalization. Dimensionality reduction was performed via sparse multinomial logistic regression with Bayesian regularization, and classification was achieved using a Fuzzy Least Squares Support Vector Machine (SVM).

Deep learning approaches were dominated by Convolutional Neural Network (CNN)s [48, 66–68] and LSTMs [48, 62, 65, 66], sometimes combined into ConvLSTM models [48, 66]. Simpler DNNs [68] and transformers [63, 64] were also reported. Unlike ML pipelines, DL models typically required minimal preprocessing, using raw inertial data segmented by sliding windows with 0–50% overlap [48, 61, 64–67]. Some studies added downsampling [48, 66], input normalization [48, 63, 65, 66], or data balancing via SMOTE [64]. As a data augmentation strategy, [63] combined pairs of records into double-length windows, downsampling them too by omitting every second step. [62] extracted motif pattern histograms from multi-

channel IMU signals as input features for a LSTM network, while [65] fed raw data directly into a two-stacked LSTM.

Inspired by the transformer architecture originally proposed by Vaswani et al. [69], Lupták ová et al. [63] and Kim et al. [64] adapted the model for activity recognition with targeted modifications. Lupták ová et al. applied normalization before the multi-head attention module, while Kim et al. replaced sinusoidal positional encoding with a convolutional backbone for higher dimensional embedding. Kim et al. further introduced a convolutional module after multi head attention, inserted between two feed forward layers within conformer blocks. A CNN, specifically a ResNet-18, was applied by Yan et al. [67]. Jang et al. [68] classified static postures into variations of forward head, rounded shoulder, and elevated shoulder, distinguishing standard from nonstandard alignments and symmetric from asymmetric elevations. Their method compared the performance of a Multilayer Perceptron (MLP) and a CNN using raw inertial data as input.

Zhao et al. [66] proposed a hybrid CNN–LSTM model and compared two training strategies: subject specific training using inertial data from individual workers and a generalized approach combining data from multiple workers . The former targeted subject-specific feature learning for improved accuracy, while the latter aimed at subject-invariant representations. Unlike many deep learning approaches relying on raw inertial signals, channel-wise normalization was applied to standardize the data. In a follow up study, Zhao et al. [48] introduced an incremental learning strategy enabling adaptation to new subjects without full retraining, evaluating one to one and many to one adaptation schemes.

Input representations varied across studies, ranging from 1D vectors of statistical features or raw data [61, 68] to 2D windows arranging sensor channels and timestamps as motion images [48, 63–68], as illustrated in Figure 4.

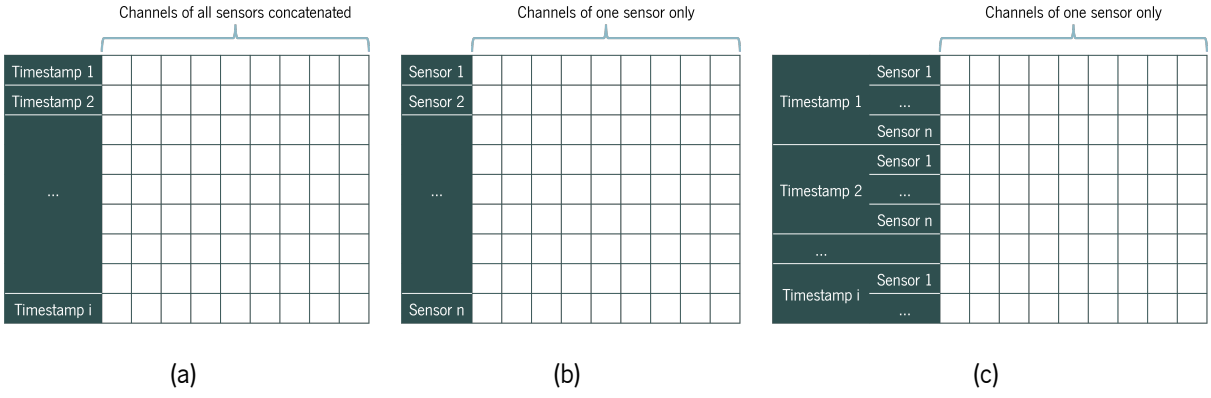


Figure 4: Three 2D input window strategies: (a) multi-timestamp rows with concatenated channels; (b) single timestamp with one sensor per row; (c) mixed sensors and timestamps across rows

System's Validation

The reviewed studies varied widely in dataset size and recording duration. Reported datasets ranged from fewer than 10,000 windows [62,65] to around 90,000 samples [68], with several works not specifying dataset size at all [48, 61, 64, 66]. In terms of effective recording time, datasets spanned from less than 30 minutes [65] to over 17 hours [63], highlighting the heterogeneity in data availability.

Dataset split strategies also differed substantially. Only Chen et al. [61] applied subject-independent validation (leave-one-subject-out cross-validation), while Kuncan et al. [62] used 10-fold cross-validation. Most studies relied on simple random hold-out splits, typically assigning 49–72% to training, 15–21% to validation, and 10–30% to testing. This likely places samples from the same subject in both training and testing sets, introducing data leakage and inflating performance. Variations included larger test sets [65], equal validation and test proportions [63, 67, 68], or no explicit test set [64].

The number of subjects per dataset ranged from as few as 3 [65] to as many as 90 [63]. Protocols reflected either sector-specific tasks, such as construction site activities [48,65,66], simulated real working conditions [67], or controlled laboratory settings [68]. Four studies drew from public datasets, including KU-HAR [63], WISDM and UCI-HAR [64], a daily/sports dataset [62], and the hierarchical dataset by Leutheuser et al. [61].

GT was most often obtained through video-based manual labeling [48, 64–67]. Alternative approaches included mobile annotation apps [61, 64], optical motion capture [68], and simplified protocols with isolated class recordings [63], while some studies did not specify the GT method [62].

Evaluation metrics were also inconsistent. Accuracy was the most commonly reported metric, with occasional use of precision and recall [62, 63]. To address class imbalance, several studies adopted macro F1-score [48, 63, 64, 66] or precision–recall curves [65]. Only one study considered computational efficiency, reporting training and testing times [64].

Overall, substantial variability in labeling strategies, validation protocols, and evaluation metrics limits comparability across studies and highlights the need for more consistent validation practices in wearable-based HAR research.

2.2.2 Discussion

In line with previous findings, IMUs were the dominant sensing modality. While some studies relied on accelerometers and gyroscopes (6 DoF), most incorporated magnetometers (9 DoF) to improve orientation estimation. Although magnetometers can enhance activity discrimination, they introduce additional com-

putational cost and sensitivity to disturbances. Moreover, some studies reported that acceleration alone contributed most to posture recognition [62, 66], suggesting that increased sensing complexity does not always translate into improved performance.

Sensor configuration remains an open challenge. The number and placement of sensors varied considerably across studies, reflecting task-specific requirements rather than a standardized design. Sensor placement was shown to significantly affect recognition performance [70], yet no consensus exists on optimal configurations. Most systems relied on trunk and limb sensors, but their selection was often heuristic. In this context, XAI techniques could provide additional insight into the contribution of different sensors or body segments, potentially supporting more informed sensor placement strategies.

Sampling frequency introduces a further trade-off between performance and computational cost. Although human motion frequencies are typically below 10 Hz [54], and recognition performance has been shown to plateau around 20 Hz [71], higher sampling rates are still commonly used, increasing processing demands without clear benefits for classification accuracy.

The definition of activity classes also lacks consistency. This work distinguishes between complex activities, static postures, and dynamic postures, yet most studies focused on posture recognition rather than full activity understanding. Transitional movements were inconsistently handled, either split into multiple classes or grouped into a single category, which directly affects classification difficulty and reported performance. Additionally, limited dataset sizes and restricted subject diversity were frequently reported [65, 67, 70], constraining generalization. While some works addressed this through personalized models [48, 66], these approaches improve accuracy at the expense of scalability.

From an algorithmic perspective, most approaches followed a similar pipeline based on sliding window segmentation and supervised learning. Classical methods relied on handcrafted features and models such as support vector machines [61], whereas deep learning approaches, particularly CNNs and LSTMs, dominated more recent work [62, 65, 67, 68]. Hybrid architectures and transformer-based models have been introduced to better capture temporal dependencies [48, 63, 66], but their adoption remains limited. Despite architectural diversity, performance gains are often reported on relatively simple datasets, making it difficult to assess their robustness in realistic conditions.

Model optimization further adds variability. Hyperparameter selection, including window size, network depth, and learning settings, was typically tuned empirically [48, 63, 64, 66], with limited standardization across studies.

Validation practices represent a major limitation. Most studies relied on random hold-out splits at the sample level, which likely include data from the same subjects in both training and testing sets, introduc-

ing data leakage and inflating performance. Only a small number of works employed subject-independent validation [61]. Evaluation metrics were also inconsistent, with accuracy being the most commonly reported measure, despite known sensitivity to class imbalance. Alternative metrics, such as F1-score or precision–recall curves, were used inconsistently.

Overall, while reported recognition accuracies are often high, these results are achieved under heterogeneous and frequently weak validation protocols. The lack of standardization in sensing configurations, dataset design, class definitions, and evaluation procedures limits comparability across studies and raises concerns about the generalizability of current HAR approaches to real-world applications.

2.3 Perception Models for Human–Robot Collaboration

The growing awareness regarding the high economic costs and life impact on the worker's quality of life, due to WRMSDs, led to a research thrust in ergonomics-sensible HRC. However, there are still some open challenges in HRC research, namely improving cobot's decision-making capabilities, to make this collaboration efficient and closer to the industrial and user needs.

2.3.1 Muscle Fatigue Estimation

Muscle fatigue is typically defined as the decline in the ability of a muscle to generate or maintain the required force after prolonged or repeated activity [72]. A common protocol to measure fatigue is by assessing the loss of Maximal Voluntary Contraction (MVC) force through short maximal efforts before, during, or after a fatiguing task [73]. While this method is precise, it is not suitable for everyday applications because it requires controlled conditions, maximum effort, specialized equipment, and interruption of the ongoing activity [73].

A more practical and widely used approach relies on analysing changes in the sEMG signal, which reflects the underlying physiological alterations induced by fatigue at both central and peripheral levels [74]. At the central level, as the fatigue sets in, motor unit recruitment shifts from an asynchronous to a more synchronised pattern, fatigued motor neurons require increased neural drive to maintain firing, and the overall motor unit firing rate decreases. At the peripheral level, imbalances in sodium and potassium concentrations, disrupted calcium dynamics in the sarcoplasmic reticulum, and reduced actin-myosin cross-bridge formation collectively decrease muscle fibre conduction velocity [74].

These physiological changes manifest directly in the sEMG signal: the decreased muscle fibre conduction velocity and altered motor unit firing patterns shift the spectral content towards lower frequencies,

while the increased neural drive and motor unit synchronisation lead to a concurrent rise in overall amplitude [75]. Together, these spectral and amplitude changes provide measurable and non-invasive indicators of fatigue progression during ongoing activity.

This literature review is guided by two key questions: (i) *What methods are currently employed to estimate muscle fatigue, and what muscles, sensors, sampling frequencies, and algorithms are most commonly used?* (ii) *How are sEMG-based fatigue estimation methods validated, and what limitations are reported in their application?*

2.3.2 Methods

A structured literature search on fatigue estimation methods and cobot control strategies was conducted in Scopus in January 2022 and updated in August 2025 through targeted screening of recent publications and reference checking. The search strategy, keywords, selection criteria, and screening process are reported in Annex A.3.

2.3.3 Models for Fatigue Estimation

Table 5 presents the main aspects of the studies included in this literature analysis and regarding model-based methods for muscle fatigue detection. The studies consider: (i) the sensor used, (ii) sampling frequency, (iii) filters used, (iv) analyzed muscle(s), (v) task, and (vi) protocol. The studies based on machine and deep learning can be found in the table 6. Detailed information on these neural network models, such as i) architectures used, ii) number of classes, iii) ground truth, iv) dataset, v) metrics are present in this table.

Table 5: Summary of studies on muscle fatigue detection, including model-based and ML/DL approaches. Reference, sensor, sampling frequency, filters, muscles, task, and protocol are shown.

Reference	Sensor	Freq. (Hz)	Filters	Muscle	Task	Protocol
Model-based approaches						
Peternel *et al.* (2016) [31]	Delsys sEMG	N/M	N/M	Posterior / anterior deltoid	Sawing	N/M
Peternel *et al.* (2018) [17]	Delsys sEMG	N/M	N/M	Posterior / anterior deltoid	Sawing, Polishing	N/M
Peternel *et al.* (2018) [32]	Force sensor; Optitrack™	N/A	N/A	Triceps / Biceps Brachii	Polishing	N/M
Peternel *et al.* (2019) [2]	Force sensor; Optitrack™	N/A	N/A	Triceps / Biceps; Deltoid; Pectoralis Major; Latissimus Dorsi	Polishing, Drilling	N/M
Yaacoub *et al.* (2023) [76]	N/A	N/A	N/A	N/A	Co-manipulation	N/M
Boyer *et al.* (2021) [77]	Delsys sEMG	1926, 926	Butterworth band-pass (20–500 Hz, 2nd); notch (60 Hz)	Medial deltoid	Arm elevations	8 adults; 7 elevation tasks; 25 min rest between trials
Ebied *et al.* (2020) [27]	Myo armband (sEMG)	200	Pre-filtered	Pronator teres	Elbow flexion 90° with 6 kg load	15 males; 120s until fatigue

Reference	Sensor	Freq. (Hz)	Filters	Muscle	Task	Protocol
Kanal *et al.* (2019) [78]	Delsys sEMG	2000	N/M	Deltoids, Triceps	Shoulder flexion/abduction; elbow extension	20 subjects; isometric contractions; fatigue self-reported vs. model
Machine and deep learning approaches						
Papakostas *et al.* (2019) [79]	sEMG	1926	Median filter	Deltoid, Triceps	Shoulder flex/abduction; elbow extension (WAM arm)	3 exercises × 3 reps; held until fatigue; onset manually marked
Feng *et al.* (2022) [33]	sEMG	1000	Butterworth high-pass (5 Hz); band-stop (50 Hz)	8 muscles (Trapezius L/R, Psoas major L/R)	Flight simulation	Up to 8h or fatigue; questionnaire + sEMG validation
Zhang *et al.* (2021) [80]	sEMG	4000 (D1); 1962 (D2)	High-pass 15 Hz; low-pass 140 Hz	D1: N/M; D2: Deltoid, Triceps	Shoulder flexion/abduction; elbow extension	3 exercises, 3 reps, 10s each
Krishnamani *et al.* (2021) [28]	sEMG	1000	Notch (50 Hz); Butterworth band-pass (10–400 Hz)	Biceps brachii	Elbow flexion/extension with load	Dynamic curls with 6 kg dumbbell; until exhaustion
Wang *et al.* (2020) [81]	sEMG	2000	Wavelet packet threshold denoise	Lower-body muscles	Cycle ergometer	60–70 rpm; stop if <60 rpm; respiratory gases measured
Wang *et al.* (2021) [82]	sEMG	2000	Wavelet packet threshold denoise	Lower-body muscles	Cycle ergometer	70–75 rpm; stop if <70 rpm; respiratory gases measured
Chen *et al.* (2021) [83]	sEMG	1000	Slice cleaning (noise removal)	Biceps, Brachioradialis	Dumbbell lifting	Repetitions until difficulty; pre/post fatigue sEMG

Reference	Sensor	Freq. (Hz)	Filters	Muscle	Task	Protocol
Terracina *et al.* (2019) [34]	sEMG	1000	Butterworth band-pass (20–450 Hz)	14 trunk muscles	30-min walk	MVC + Biering–Sorensen; continuous RPE rating
Jiang *et al.* (2022) [84]	IMUs, MoCap, Force plates	100	Butterworth low-pass (20 Hz); zero-phase (6 Hz)	Lower-body muscles	Squats	Repeated to exhaustion; Simulated OpenSim dataset

Table 5: Summary of model-based and ML/DL studies on muscle fatigue detection.

Table 6: Summary of studies using machine learning and deep learning methods for fatigue estimation and associated evaluation metrics.

Reference	Architecture	Input	Output	Ground Truth	Dataset	Split	Acc. (%)	F1 (%)
Papakostas *et al.* (2019) [79]	Linear SVM; SVM–RBF; GB; ET; RF	Time/spectral features	2 classes: Non-fatigue, Fatigue	User-reported	10 subjects	LOO variants (E1–E4)	N/M	E1: GB 70.4; E2: SVM 71.7; E3: ET 77.8; E4: GB 76.6
Feng *et al.* (2022) [33]	GRU	5 time + 4 freq. features	3 classes: Non-fatigue, Transition, Fatigue	Critical-point rule (post hoc labeling)	8 male	per sample, 6/2/2 (train/val/test)	98.42 (test)	99.04 (test)
Zhang *et al.* (2021) [80]	MFFNet	Filtered raw sEMG	D1: 4-class; D2: (0/1)	D1: N/M; D2: User-reported	D1: 2 subj.; D2 [79]: 10 subj.	mixed (per sample): N/M, cross (per subj.): N/M	Mixed: D1 96.45; D2 78.25; Cross: D1 71.68%; D2: D1 71.24%	Mixed: D1 96.49; D2 77.03; Cross: N/M
karthick *et al.* (2021) [28]	MLP; ELM	Geometric features	2 classes (0/1)	First curl = Non-fatigue; last curl = Fatigue	58 subj. (3F/55M)	per sample, 10-fold CV	ELM 94.09; MLP 91.44	ELM 93.75; MLP 91.05
Wang *et al.* (2020) [81]	CNN–SVM	Raw sEMG (wavelet denoised)	2 classes (Normal/Fatigued)	V-slope anaerobic threshold; split at VT time	20 subj. (14M/6F)	16 train; 4 test (by subj.)	83.51	N/M

Reference	Architecture	Input	Output	Ground Truth	Dataset	Split	Acc. (%)	F1 (%)
Wang *et al.* (2021) [82]	LSTM	Time-freq. features (wavelet denoised)	2 classes (0/1)	V-slope AT; split at AT time	20 male	per sample, (70/10/20), (60/10/30), (50/10/40)%	95.18; 92.24; 89.36	N/M
Chen *et al.* (2021) [83]	LSTM + Attention	Raw (10,000 slices, 300 samples)	2 classes (0/1)	Labels set at collection	10 subj.	per sample, Train 3500; Test 1500 (slices)	90.40 (test)	N/M
Terracina *et al.* (2019) [34]	SVM	Time-freq. features	3 classes (Non-fatigue, Transition, Fatigue)	RPE: 6–11 Non-fatigue; 12–16 Transition; 17–20 Fatigue	13 male	10-fold CV	Single-subj 98.50; Across 99.10	N/M
Jiang *et al.* (2022) [84]	Spatio-temporal attention Transformer	True + forecast motion signals	10 classes (levels)	Self-report scale 1–10 (10=max)	Real: 12 adults; Sim: 50 virtual (OpenSim)	Sim+exp for train; held-out exp for val	83.50 (test)	82.00 (test)

2.3.4 Sensors and sampling frequency

Regarding type of sensor to estimate fatigue, most studies relied on surface electromyography (sEMG) as the primary modality [17, 27, 28, 31, 33, 34, 77–83], with only one study integrating IMUs, force plates, and musculoskeletal simulation [84]. This predominance reflects the sensitivity of sEMG to fatigue-induced physiological changes, although it demands high acquisition frequencies and careful preprocessing. However, these studies predominantly rely on conventional standalone sEMG acquisition systems and Ag/AgCl electrodes. While such electrodes provide reliable signal quality under controlled experimental conditions, they require skin preparation, may cause discomfort or irritation during prolonged use [30], and are poorly suited to unobtrusive long-term monitoring in industrial environments.

sEMG signals were typically acquired between 926 Hz and 4000 Hz [77, 80], with most works reporting rates around 1000–2000 Hz [27, 28, 33, 34, 78, 79, 81–83]. Filtering commonly included band-pass filters in the range of 10–450 Hz and notch filters at 50/60 Hz to suppress powerline interference [28, 34, 77]. By contrast, IMU-based recordings were acquired at 100 Hz, sufficient to capture the slower kinematic dynamics of human movement [84].

2.3.5 Target muscles, tasks, and protocol

The choice of muscles was strongly task-dependent. Upper-limb muscles such as deltoideus, triceps brachii, biceps brachii, and pronator teres were studied during lifting, flexion–extension, and arm elevation tasks [17, 27, 28, 31, 77–80]. Trunk muscles were monitored during prolonged walking and endurance protocols [33, 34], while lower-limb muscles were emphasized in cycling and squat-based protocols [81, 82, 84]. Some studies combined multiple muscle groups, reflecting differences in fatigue manifestation across anatomical and functional contexts [2, 33, 34].

Experimental tasks ranged from static isometric contractions such as holding a load until exhaustion [27, 78] to dynamic activities like dumbbell lifting [28, 83], cycling [81, 82], squats [84], and walking [34]. Other studies addressed occupationally inspired tasks such as polishing, drilling, or sawing [2, 17, 31, 32]. A few extended protocols to long-duration simulations, for example flight operation lasting up to 8 hours [33]. In most cases, participants were pushed to exhaustion, either by sustaining contractions until failure or repeating cycles until voluntary cessation.

2.3.6 Subjects and dataset size

Sample sizes were generally modest, often between 8 and 20 participants [27, 34, 77, 78, 83, 84], with some studies involving larger cohorts (e.g., 58 subjects in [28]). To mitigate recruitment constraints, Jiang et al. [84] incorporated simulated data from 50 virtual bodies modeled in OpenSim, although these simulated approaches may limit the extent to which findings can be generalized to real-world conditions.

2.3.7 Fatigue Labeling

Definitions of fatigue varied substantially across studies. Common strategies included self-reported onset [78, 79], performance-based thresholds such as cadence drop during cycling [81, 82], subjective scales like the rate of perceived exertion (RPE) [34], and rule-based labeling [33], in which pre-task segments were assumed non-fatigued, post-task fatigued, and intermediate windows classified as transition states . Physiological thresholds based on ventilation or gas exchange were also used as ground truth in cycling protocols [81, 82].

2.3.8 Traditional sEMG-based indicators for fatigue estimation

Early work on muscle fatigue estimation relied primarily on offline analysis of sEMG signals, with amplitude and spectral features forming the foundation of most approaches. Ebied *et al.* [27] examined Root Mean Square (RMS) and Median Frequency (MDF) during a static task in which participants sustained a load with the elbow flexed at 90° for 120 s. Using the Myo armband with eight sensors, continuous sEMG was segmented into one second epochs for feature extraction. The results confirmed expected fatigue trends, increasing RMS and decreasing MDF, while revealing substantial inter-individual variability. Notably, some participants exhibited pronounced amplitude changes with minimal frequency shifts, indicating that MDF alone is an unreliable fatigue marker and motivating multimodal feature combinations.

Building on these offline findings, subsequent studies have pursued real-time detection, a prerequisite for preventive and assistive applications. Kanal *et al.* [78] proposed an online algorithm that estimated MDF slopes every 0.125 s and applied temporal smoothing over five samples, defining fatigue onset when no slope exceeded the initial reference within a 40 sample window. Validation against self reported fatigue onset showed promising performance for shoulder flexion, with detection occurring within approximately 9.44 s of the subjective report, but weaker results for shoulder abduction and elbow extension, where delays exceeded 12 s. Together, these findings highlight both the potential and the task-dependence of real time MDF based methods.

To overcome limitations of Short Time Fourier Transform (STFT) methods, particularly the quasi-stationarity assumption, Boyer *et al.* [77] proposed a Continuous Wavelet Transform (CWT)-based framework incorporating contraction detection, time–frequency decomposition, and smoothing of MDF trajectories. Experiments on static and dynamic shoulder abductions under varying loads showed steeper MDF slopes for heavier and static tasks, consistent with physiological expectations. Compared to STFT, CWT yielded smoother trajectories and higher time–frequency resolution, while contraction detection reduced spurious fatigue detections in dynamic conditions.

Taken together, these studies mark a clear progression from offline RMS/MDF feature tracking towards online and higher-resolution methods, with CWT approaches offering particular promise. Yet, variability across tasks and subjects remains a central challenge, and systematic validation against reliable ground truth is still lacking.

Beyond sEMG, other physiological modalities have been explored as fatigue proxies. Galvanic Skin Response (GSR), for instance, exploits the rise in skin conductivity associated with increased sweat gland activity under fatigue [85–88]. While it consistently shows sensitivity to fatigue, its utility as a standalone marker is limited by susceptibility to confounders such as temperature, humidity, stress, and emotional state, as well as sensor placement and skin preparation. It is therefore best regarded as a complementary modality, whose integration with sEMG or other physiological signals may provide more robust fatigue estimation, particularly in ergonomics and human–robot collaboration contexts.

2.3.9 Model-based techniques for fatigue estimation

As seen so far, analysing the changes of the sEMG signal in the frequency-domain is an efficient tool to evaluate muscle fatigue. It is well established that fatigue is typically accompanied by an increase in signal amplitude and a downward shift in spectral frequency content, most notably a decrease in the MDF [2]. While this method is well suited for constant or quasi-static muscle efforts, it becomes less reliable in dynamic tasks where varying contractions affect MDF trajectories. Although Boyer *et al.* [77] proposed an algorithmic refinement to address this limitation, the challenge of defining an appropriate window length for tracking MDF during dynamic movements remains. To overcome these shortcomings, researchers have proposed model-based approaches to fatigue estimation [2].

One of the most influential contributions came from Peternel *et al.* [31], who introduced a fatigue model inspired by resistor–capacitor (RC) circuit dynamics. In their formulation, capacitor voltage represented fatigue, resistor conductance corresponded to muscle effort, and capacitance reflected endurance capacity. sEMG signals were filtered, rectified, and normalized to the subject's MVC, yielding a dimensionless

activation level $A_i(t)$ for each muscle. The fatigue index $V_i(t)$ was then modeled as a function of current effort $G_i(t)$ and endurance capacity C_i, with capacity calibrated through reference exertions sustained until task failure. Muscles were empirically considered fatigued when V_i reached 70%.

This framework was later extended to account for recovery. Peternel *et al.* [17] added a discharging component to represent recuperation when activation fell below a threshold, enabling the model to switch between fatigue and recovery modes. In this version, fatigue onset was defined at $V_i = 30\%$, and recovery was governed by a rate parameter R, empirically set to 0.5.

Subsequent work advanced this approach by replacing muscle activity with muscle force as the input variable, exploiting the approximately linear relationship between activity and force within a limited range. In [2, 32], muscle forces were estimated through a two stage process. First, an offline musculoskeletal simulation in OpenSim, solved via Static Optimization, generated training data linking joint torques and endpoint forces to muscle force estimates. A Gaussian Process Regression (GPR) model was then trained to approximate this mapping. In the online stage, interaction forces measured by the robot's force torque sensors and arm kinematics obtained from OptiTrack™ MoCap system were used to estimate real time muscle forces, which were subsequently incorporated into the fatigue model to produce a force based fatigue index.

In a related direction, Yaacoub *et al.* [76] proposed a higher-level fatigue model based on the progressive reduction of joint torque-generating capacity rather than individual muscles. Their approach relied on a Digital Human Model (DHM) to compute joint loads during a co-manipulation task, and fatigue was represented as a decline in the maximum voluntary torque capacity of each joint over time. This formulation enabled the robot to reason about overall human fatigue during repetitive tasks and informed a decision-making framework for object placement policies. While this approach avoids the sensor calibration of EMG- or force-based models and naturally accounts for multi-joint effects, it remains an abstraction: fatigue was estimated at the simulation level, without direct physiological measurements, and validated only *in silico*.

Nevertheless, these methods are highly subject-dependent and require individual calibration for each worker. Moreover, the threshold values used in the models lack clear physiological justification, being arbitrary. This limits their direct use as realtime cost functions in adaptative HRC control without personalization layers.

2.3.10 Muscle fatigue detection based on machine/deep learning models

Based on the reviewed literature, traditional methods for estimating muscle fatigue are limited by their complexity, time-consuming procedures, and frequent confusion between fatigue and muscle activity

monitoring. These approaches also show poor real-time performance and relatively low accuracy in fatigue detection [33]. Furthermore, muscle fatigue progression varies considerably between individuals and across repeated trials in the same subject, making it difficult to define universal fatigue thresholds [34].To address these challenges and abstract inter- and intra-subject variability in sEMG features, researchers have increasingly turned to ML algorithms and Deep Learning (DL) models.

Among ML approaches, Papakostas *et al.* [79] evaluated five ML classifiers (linear SVM, SVM with an Radial Basis Function (RBF) kernel, Gradient Boosting, Extra-Trees, and Random Forest), achieving modest F1-scores below 70% initially. While post-processing improved results above 70%, these methods remained limited by their dependence on manual feature extraction and prior knowledge of sEMG signals and struggles to generalize to complex, non-stationary data [80].

Wang *et al.* [81] proposed a hybrid fatigue estimation approach combining deep learning and classical machine learning, in which an improved wavelet thresholding method is applied for sEMG denoising, a CNN learns higher level feature representations, and final classification is performed by a Gaussian kernel SVM replacing the fully connected layer. Fatigue was labeled into "normal" or "fatigued" states based on the anaerobic threshold, which was derived from ventilation thresholds computed using the V slope method applied to oxygen uptake (VO_2) and carbon dioxide (CO_2) curves.

In a subsequent study, Wang *et al.* [82] proposed a muscle fatigue classification framework in which sEMG signals are first denoised using an improved wavelet packet thresholding method and then transformed into a feature set comprising representative time- and frequency-domain descriptors. These features were used as inputs to a multi layer LSTM network trained using a holdout strategy with balanced fatigue and non fatigue samples, and were also provided to benchmark CNN and Gaussian kernel SVM models under the same evaluation protocol. Using this common feature representation, the LSTM-based model achieved the highest accuracy (95.18%) compared with the CNN (92.72%) and SVM (90.30%), along with superior sensitivity, specificity, and precision relative to these baseline models [82].

Feng *et al.* [33] proposed a DL-based muscle fatigue assessment method introducing a transition-to-fatigue label for early detection. The pipeline relied on 3 minute windows with 5 second steps, signal cleaning and denoising, and fatigue labeling based on a critical fatigue time derived from post-experiment questionnaires combined with sEMG analysis. Nine time- and frequency-domain features were extracted per segment and classified using a Gated Recurrent Unit (GRU)-based network, achieving an F1-score of 99.02%, outperforming ResNet-18 (98.03%) and LSTM (98.39%). However, the fatigue labeling strategy relies on subjective and post hoc criteria, raising questions about its robustness and reproducibility.

Similarly, Chen *et al.* [83] investigated the integration of multi source sEMG signals by comparing mod-

els trained on biceps brachii activity alone with models that also incorporated brachioradialis signals. Their proposed LSTM with attention outperformed both a standard LSTM and a GRU with attention, achieving a test accuracy of 90.4%, while the multi source configuration consistently outperformed the single source one. These results suggest the benefit of multi muscle inputs and the use of attention mechanisms to emphasize informative patterns in non stationary sEMG signals.

Alternative neural network structures have also been explored. Krishnamani *et al.* [28] investigated MLP and Extreme Learning Machine (ELM) models using cyclostationarity based geometric features, including area, perimeter, inertia, and bending energy, extracted from surface sEMG signals. The ELM, a single hidden layer feedforward network with fixed randomly initialized input weights and analytically learned output weights, achieved higher accuracy than the MLP (94.09%), indicating the potential of efficient feedforward architectures for fatigue detection. However, the evaluation relied only on the first curl, labeled as non fatigue, and the last curl, labeled as fatigue, discarding intermediate segments that capture fatigue progression.

However, although the reported metrics appear highly promising, a critical methodological limitation undermines many of these results [28, 33, 82, 83]. In most studies, datasets are split at the sample level rather than at the subject level, allowing data from the same individuals to appear in both training and test sets. This practice introduces data leakage, as the model is effectively tested on information it has already seen during training, which likely inflates performance metrics and masks the true challenge of achieving subject-independent generalisation.

This effect is explicitly demonstrated by Zhang *et al.* [80], who evaluated their Multidimensional Feature Fusion Network (Multidimensional Feature Fusion Network (MFFNet)) under both mixed (sample level) and cross subject (subject independent) protocols. The first dataset included only two subjects and addressed a four class fatigue problem, while the second involved ten subjects and a binary classification task. Under mixed evaluation, MFFNet achieved high performance, with F1 scores of 96.49% and 77.03% and accuracies of 96.45% and 78.25%, respectively. When a cross subject protocol was enforced, ensuring that each subject appeared exclusively in either training or test sets, performance dropped substantially across all models, with deep learning approaches reaching accuracies of 61.34–71.68% on the two subject dataset and 57.94–71.24% on the ten subject dataset. Notably, despite the higher class complexity, the dataset with only two subjects still yielded higher cross subject accuracy, suggesting strong subject specific effects and highlighting the difficulty of achieving generalizable fatigue estimation.

Terracina *et al.* [34] proposed a short term fatigue forecasting approach that classified human state as non fatigued, transition to fatigue, or fatigued up to 25 s ahead. Ground truth was defined using the Borg

Rating of Perceived Exertion (6–11, 12–16, and 17–20). The study involved 13 subjects, each performing trials lasting approximately 70 minutes, resulting in a large amount of continuous data per subject. The pipeline comprised feature extraction, feature forecasting, and classification. Forecasted features from 14 muscles, yielding 70 features, were obtained using a Normalized Least Mean Square filter with a 100 sample (25 s) prediction horizon and a 200 sample (50 s) training window, reduced to two Linear Discriminant Analysis components, and classified using an SVM. The classifier was tuned and validated in a subject-specific manner, which, together with the large within subject dataset, likely contributes to the very high reported accuracies of 98% in single subject and 99% in cross subject evaluations under 10-fold cross validation.

User independent fatigue forecasting was later explored by Jiang *et al.* [84], who proposed a time series forecasting framework to predict future fatigue levels. The approach was validated on experimental data from 12 subjects performing repeated squats to exhaustion and on simulated squatting data generated in OpenSim using 50 musculoskeletal models. Fatigue labels were obtained from self reported ratings on a 1–10 scale. A Transformer based encoder decoder model with spatiotemporal self attention was used to forecast future IMU windows up to 80 timesteps ahead, which were then fed to a separate fatigue classifier. Using leave one out cross validation, the method achieved approximately 83% classification accuracy on forecasted data and a Pearson correlation of 0.92 between predicted and reported fatigue.

2.3.11 Discussion

The present literature review analyzed recent studies on muscle fatigue estimation, focusing on the sensing modalities, labeling strategies, validation protocols, and real-time feasibility that determine their suitability for preventive human–robot collaboration.

In terms of fatigue estimation, the main limitations identified with this literature review were: (i) The reported performance of many deep learning based studies is potentially misleading, as models are often trained and evaluated using sample level splits, leading to data leakage, overly optimistic results, and limited insight into subject independent generalizability [28, 33, 80, 82, 83]; **ii)** the several existing models fail in real-time performance [27, 33]; **iii)** several works focus primarily on monitoring muscle activity rather than explicitly estimating muscle fatigue [27, 77, 78]; **iv)** there is still a lack of subject-independent models, since muscle fatigue development varies between subjects and even in the same subject for different trials [80, 84]; **v)** several approaches rely on self-defined global fatigue thresholds, which lack physiological grounding [2, 17, 31, 32]; **vii)** only few works incorporate an intermediate transition-to-fatigue label [33, 34], despite its importance for early detection. As a result, most methods only recognize fatigue

once it has already occurred, limiting their utility for prevention.

From the sixteen studies analyzed, 81% employed sEMG to monitor muscle activity and feed fatigue estimation models [17, 27, 28, 31, 33, 34, 77–83]. In an attempt to avoid the use of sEMG sensors, 19% (3) explored the use of machine learning algorithms with kinematic-based sensors, namely, 13% used an optitrack™ system [2, 32] and the rest, 6%, used IMUs [89]. Among sEMG-based studies, 69% monitored upper-body muscles, particularly the triceps and biceps brachii, posterior and anterior deltoid, pectoralis major, and latissimus dorsi. These findings reinforce that sEMG is the most widely used and effective modality for fatigue assessment, and that upper-limb muscles dominate research focus. This emphasis is aligned with epidemiological evidence showing that more than 30% of WRMSDs affect the upper body [5], underscoring the importance of studying fatigue in these muscle groups.

Although sEMG is effective for detecting fatigue-related physiological changes, conventional electrodes present practical limitations for continuous use, including skin preparation, possible discomfort during prolonged recordings, and limited suitability for unobtrusive garment integration. These constraints have increased interest in textile-integrated dry electrodes, which appear to be a promising alternative for wearable and long-term muscle activity monitoring [90–92].

With respect to ground truth, most studies rely on self-reported fatigue indicators. Participants typically rated their fatigue using the modified Borg Rating of Perceived Exertion scale (6–20) [34], or through binary labels (non-fatigued vs. fatigued) [79, 80, 83]. Only two studies explicitly included a transition-to-fatigue state: Feng *et al.* classified samples before, after, and around the fatigue breakpoint into non-fatigue, fatigue, and transition-to-fatigue classes, respectively [33], while Terracina *et al.* defined Borg scores between 12–16 as transition-to-fatigue [34]. A distinct approach was proposed by Wang *et al.* [81, 82], who determined fatigue onset by measuring oxygen uptake and carbon dioxide exchange with a gas analyzer, identifying the anaerobic threshold as the fatigue breakpoint.

Overall, deep learning models generally outperformed traditional machine learning approaches in terms of reported accuracy. However, these gains should be interpreted with caution, as many evaluations rely on sample level data splits, which limits conclusions about subject independent performance. As a result, the high accuracies often reported, frequently above 90%, do not necessarily reflect robustness or generalizability in real world fatigue estimation scenarios.

2.4 Control Strategies for Human-aware Collaborative Robots

This subsection presents the main aspects of the studies included in this literature analysis, regarding the robot's control. The studies are organized by categories and consider: (i) relevant aspects of the high-level controller, namely optimization method, optimization objective, and the behaviour (i.e., if is reactive or not); (ii) low-level controller; (iii) modelled variable in the robot control; (iv) interaction scenario; and (v) task.

The literature review in this chapter follows a targeted screening approach supported by reference checking, rather than a formal Preferred Reporting Items for Systematic Reviews and Meta Analyses (PRISMA)-based systematic review. Some of the studies discussed in this section were identified through the structured literature search reported in Section 2.3. The search strategy, keywords, selection criteria, and screening process are reported in Annex A.4. This section addresses the following question: *How has human state-informed control been integrated into HRC, and what tasks, experimental settings, modelled variables, and control strategies have been adopted?* Table 7 presents an overview of the included studies.

Table 7: Summary of HRC control strategies: reference, optimization method, objective, behaviour, low-level controller, modelled variable, interaction scenario, task, and validation subjects.

Reference	High-level controller			Low-level controller	Modelled variable	Interaction scenario	Task	Validation subjects
	Objective	Optimization / learning method	Behaviour					
Shafti *et al.* (2020) [93]	Minimize postural risk ergonomic	Non-optimization	Adaptative	N/M	Position	Assistive holding	Soldering	5
Van der Spaa *et al.* (2020) [22]	Minimize postural risk ergonomic	Gradient-free optimization	Adaptative	N/M	Position	Co-manipulation	Manual load handling	4
Xie *et al.* (2023) [94]	Minimize postural ergonomic risk	Learning-based (gradient-based RL)	Fixed	N/M	Position and orientation	Assistive holding	Fine wire insertion	20
Amani *et al.* (2024) [95]	Minimize postural ergonomic risk	Learning-based (Q-Learning)	Fixed	N/M	Position	Handover	Grabbing and receiving boxes in virtual reality	4
Zolotas *et al.* (2024) [96]	Minimize postural ergonomic risk and improve balance stability	Heuristic variability policy	Adaptative	N/M	Position	Handover	Pick-up task	16
Papadopoulos *et al.* (2024) [97]	Minimize postural ergonomic risk	Hybrid: Learning (LfD DMPs) + Optimization (HQP, admittance)	Adaptative	N/M	Pose, joint velocity	Co-manipulation	Pick-up task	6

Table 7 (continued)

Reference	High-level controller			Low-level controller	Modelled variable	Interaction scenario	Task	Validation subjects
	Objective	Optimization / learning method	Behaviour					
Busch *et al.* (2017) [18]	Minimize ergonomic & proxemics cost; maximize object visibility	Gradient-based optimization	Fixed	N/M	Position	Handover	Handover	39
Chen *et al.* (2018) [98]	Maximize comfort (postural & muscular)	Gradient-based optimization	Fixed	N/M	Position	Assistive holding	Cutting and drilling	5
Marin *et al.* (2018) [99]	Find optimal location in latent space	Gradient-based optimization	Fixed	N/M	Position	Assistive holding	Drilling	1
Figueredo *et al.* (2020) [16]	Maximize comfortability (REBA + muscle activity)	Gradient-based optimization	Fixed	N/M	Position	Assistive holding	Simulated peg-in-hole assembly	1
Falerni *et al.* (2024) [100]	Learn user ergonomic preferences	Learning-based: Preference learning (pairwise) + RULA penalty	Fixed	N/M	Pose	Handover	Pick-up task	20
Bestick *et al.* (2018) [39]	Learn user ergonomic preferences	Learning-based (Bayesian Inverse RL)	Adaptative	N/M	Position	Handover	Handover	5

Table 7 (continued)

Reference	High-level controller			Low-level controller	Modelled variable	Interaction scenario	Task	Validation subjects
	Objective	Optimization / learning method	Behaviour					
Peternel *et al.* (2019) [2]	Manage and minimize muscle fatigue	Gradient-based optimization	Adaptative	Cartesian impedance	pose	Co-manipulation	Drilling, polishing	6
Yaacoub *et al.* (2023) [76]	Minimize long-term fatigue accumulation	Planner: Partially Observable Markov Decision Process (POMDP) solved with Partially Observable Monte Carlo Planning (POMCP)	Adaptative	N/M	Pose (position and orientation)	Co-manipulation	equivalent to drilling	0 (simulation only)
Merlo *et al.* (2024) [23]	Minimize fatigue and ergonomic risk	Planner: AO* search on AND/OR Graph with ergonomic costs	Adaptative	N/M	Role allocation / position?	Collaborative assembly	Pick-up task	12
Peternel *et al.* (2016) [31]	Minimize fatigue	DMP (Locally Weighted Regression)	Adaptative	Hybrid force/impedance controller	Trajectory	Co-manipulation	Sawing	1
Kim *et al.* (2018) [19]	Minimize joint-torque effort	Gradient-based optimization	Adaptative	Cartesian impedance	Position	Co-manipulation	Manual load handling	1

Table 7 (continued)

Reference	High-level controller			Low-level controller	Modelled variable	Interaction scenario	Task	Validation subjects
	Objective	Optimization / learning method	Behaviour					
Rahal *et al.* (2020) [101]	Minimize workload (RULA-inspired)	Gradient-based optimization	Adaptative	N/M	Velocity	Teleoperation	Pick and place	15

Current solutions in HRC control have predominantly focused on optimization-based frameworks [2, 16, 18, 19, 22, 98, 99, 101]. These approaches are generally adaptative, meaning the robot continuously adapts its control commands at the control rate in response to changes in the human's state (e.g., posture, fatigue). In contrast, fixed controllers compute an action or configuration that remains fixed over a task cycle or longer horizon, without step-wise adjustment to user variability. While adaptative optimization has proven effective in specific scenarios, it relies heavily on manually designed objective functions, such as maximizing comfort [16] or minimizing REBA scores [22], which limits its ability to adapt to varying task constraints. Moreover, shifting from one ergonomic objective (e.g., risk reduction) to another (e.g., task efficiency) often requires a complete reformulation of objectives and algorithms [95].

Beyond traditional optimization, planning-based approaches have also been explored. Yaacoub *et al.* [76] formulated fatigue-aware HRC as a continuous-state POMDP and solved it using POMCP, an online Monte Carlo tree-search planner. The method selects the object pose to be executed at each task cycle based on the current belief about the human fatigue state, with the objective of minimizing cumulative fatigue over repeated cycles. Therefore, the controller is adaptative, as the robot action is adapted at the cycle level. Similarly, Merlo *et al.* [23] integrated ergonomic costs into AO*-based planner operating on AND/OR graphs, using the KWear (kinematic wear) index to dynamically allocate roles between human and robot in collaborative assembly. AO* is a best-first graph search algorithm designed for problems with multiple possible solution paths, while AND/OR graphs generalize classical search trees by representing tasks that can be achieved either by alternative sub-tasks (OR nodes) or by combinations of sub-tasks (AND nodes). Their planner is adaptative, reassigning tasks at each step, highlighting how planning can also support real-time ergonomic adaptation.

Heuristic controllers offer a lighter alternative. Zolotas *et al.* [96] contrasted an "assistive" handover policy, which minimizes center-of-mass deviation, with a "stimulating" strategy that deliberately introduces variability by placing handover points near the boundaries of the user's base of support. This adaptative heuristic reduced REBA scores and improved balance margins, suggesting that controlled variability can mitigate over-assistance, although at the cost of increased motion jerkiness.

Learning-based approaches extend adaptability beyond predefined heuristics. Peternel *et al.* [17, 31] proposed controllers that gradually assume control as fatigue increases, combining online learning with impedance adaptation. Their key advantage is progressive assistance aligned with human effort; however, the approach is limited to tasks characterized by repetitive, periodic motions, rather than industrial workflows composed of sequential task phases, which restricts its applicability to more general industrial settings. Papadopoulos *et al.* [97] combined Learning from Demonstration with optimization by adapt-

ing Riemannian Dynamic Movement Primitive (DMP)s online through hierarchical quadratic programming. A nominal co-carrying trajectory is learned from a single demonstration and continuously deformed via posture-dependent virtual forces, computed by an ergonomics module and applied through an admittance layer, to promote safer joint configurations. This hybrid approach enables continuous trajectory adaptation across task configurations while remaining responsive to human state and environmental constraints. However, it increases computational load due to online optimization, requires careful tuning of multiple control gains and thresholds, and remains sensitive to demonstration quality and DMP design choices.

Model-free RL has been explored primarily for posture optimization. Xie *et al.* [94] proposed GOLA-HRC, a gradient-based online, model-free method framed as RL, which iteratively updates the robot end-effector pose using finite-difference gradients of a continuous awkward posture score. By repeating the task and adjusting the pose between repetitions, the method converges to a personalized end-effector configuration that minimizes each worker's ergonomic score, accounting for individual anthropometrics and movement patterns, and reducing awkward postures compared to fixed baselines. However, the learned assistance corresponds to a single optimized static pose per worker–task pair, with no continuous adaptation to posture variations within a task cycle. Falerni *et al.* [100] extended this idea with a human-in-the-loop, preference-based optimization framework in which the robot proposes pairs of handover poses and the user provides explicit pairwise preferences with confidence levels. An active learning scheme is used to construct a surrogate cost function that jointly encodes learned user preferences and an ergonomic penalty based on RULA, such that the optimizer searches for handover poses that simultaneously match the user's preferred configurations (low surrogate cost) and yield low ergonomic risk. While this enables strong personalization, the approach requires repeated task interruptions for user queries and converges to a single optimized static pose rather than a controller that continuously adapts during execution. Notwithstanding, RL approaches in human–robot collaboration raise several practical concerns [102–104]. A primary limitation is the large amount of training data required to learn stable policies, which is difficult to obtain in close physical interaction scenarios where human behavior is highly variable and unpredictable, such as handovers [103]. To mitigate this, Ghadirzadeh *et al.* [104] collected extensive real-world interaction data with the robot operating in a passive mode and used it to populate a simulator that evaluated a large set of candidate robot actions, enabling offline policy learning. Nevertheless, accurately modeling human responses and interaction dynamics in simulation remains challenging, and policy performance is sensitive to modeling assumptions [102, 104]. In addition, ergonomic objectives and discomfort are often encoded only implicitly through reward functions rather than being explicitly modeled, making learned policies highly dependent on reward design and limiting interpretability [102]. Together, these factors

constrain scalability, robustness, and deployment of RL-based methods in HRC scenarios [103, 104].

All the reviewed papers present relevant results with different levels of complexity. Nonetheless, none presented a method capable of including the user's intrinsic limitations (e.g., a shoulder lesion). Bestick *et al.* presented a preliminary work on learning Human ergonomic preferences for handovers, using Bayesian inverse reinforcement learning (IRL) [39]. Their driving motivation was the lack of confirmation that the standard ergonomic methods typically used are in fact the best solution to model ergonomic cost. Thus, they developed an estimation, where the person's ergonomic cost is learnt online, formulating queries consisting of selecting a handover configuration for an object, and using as label the person's joint configuration when picking the object. The authors' conclusions suggest that IRL algorithms can predict the human behaviour and adapt to it. Therefore, it can be an interesting tool to model subject-specific behaviour.

2.5 General Insights

This literature review demonstrated that substantial progress has been made toward ergonomically informed HRC through advances in wearable sensing, perception algorithms, ergonomic assessment methods, and human-aware control strategies. Wearable technologies, particularly IMU- and sEMG-based systems, have proven effective for non-invasive monitoring of posture, movement, and muscle activity. In parallel, data-driven perception methods have shown strong performance in posture and activity recognition, while ergonomic frameworks enable the translation of kinematic data into quantitative risk indicators. Together, these developments confirm the feasibility of monitoring human physical state and incorporating ergonomic considerations into collaborative robotic systems.

Despite these advances, several limitations constrain real-world deployment and reduce preventive effectiveness. Most wearable posture assessment systems rely on static or quasi-static ergonomic criteria derived from observational tools, which limits their applicability to dynamic tasks. Consequently, cumulative exposure, recovery effects, and the temporal evolution of risk are insufficiently modeled. Although some studies introduced time-weighted or cumulative indices, these approaches remain fragmented and are rarely integrated into decision-making or control strategies.

At the perception level, inertial sensing has emerged as the most practical solution for continuous monitoring, yet its accuracy and robustness remain strongly dependent on sensor quality, placement, and algorithmic design. Moreover, ergonomic reports generated from wearable data are often poorly contextualized, providing isolated risk scores without explicit attribution to specific postures or task phases.

Human activity and posture recognition can address this limitation by structuring ergonomic information temporally and semantically, enabling more holistic and interpretable assessments of ergonomic exposure.

Fatigue estimation is another active research area, with deep learning models showing promising performance, particularly when using sEMG signals. However, most approaches focus on fatigue detection rather than early identification, rely on binary labels or self-defined thresholds with limited physiological grounding, and frequently use sample-level data splits that inflate reported performance and limit conclusions about subject-independent behavior. Subject-independent performance, real-time feasibility, and explicit modeling of transition-to-fatigue states remain open challenges, limiting current methods' preventive potential. Further, most studies rely on standalone dry sEMG electrodes, which constrain wearability and long-term monitoring; e-textile-based sEMG electrodes represent a relevant direction for garment-integrated, continuous fatigue monitoring.

In HRC, control strategies increasingly incorporate ergonomic objectives, yet most frameworks optimize posture or fatigue in isolation. Optimization-based approaches depend on manually defined ergonomic cost functions, which must be reformulated when objectives change, while learning-based methods have primarily focused on optimizing static task configurations or require large amounts of interaction data to achieve stable policies. Across these strategies, personalization remains limited: few methods explicitly account for user preferences, and none directly adapt assistance based on individual physical restrictions or pain-related constraints.

The work presented in this thesis addresses the fragmentation observed in the literature by integrating wearable sensing, perception algorithms, continuous ergonomic assessment, and adaptive control within a unified, human-aware collaborative robotic framework.

Table 8: State-of-the-art limitations and derived research requirements.

Research axis	State-of-the-art limitation	Requirement derived	Chapter
Wearable sensing	Existing IMU prototypes present hardware limitations, and most research relies on proprietary solutions, constraining reproducibility and future extensibility.	Establish an open, low-cost upper-body IMU platform as a foundation for future integration with complementary modalities such as sEMG.	Chapter 3
	Conventional sEMG electrodes require skin preparation and are unsuitable for long-term garment-integrated monitoring.	Validate textile-integrated dry sEMG electrodes as a viable modality for continuous, unobtrusive muscle activity monitoring.	Chapter 3
Computational methods	Ergonomic assessment relies on static, snapshot-based scoring; therefore, dynamic tasks and the temporal evolution of risk are not captured.	Develop methods enabling continuous, dynamic ergonomic assessment from inertial data, moving beyond static scoring.	Chapter 4
	Cumulative postural exposure is rarely modelled and is not integrated into assessment pipelines.	Introduce cumulative ergonomic metrics that quantify accumulated postural load over time within interpretable assessment outputs.	Chapter 4
Muscle fatigue estimation	Fatigue models rely on sample-level validation, binary labels, and self-defined thresholds lacking physiological grounding, limiting generalizability and integration into adaptive assistance.	Develop and validate a data-driven fatigue estimator under subject-independent conditions, capture fatigue as a gradual process, and integrate its predictions into an adaptive collaborative control strategy.	Chapter 6
Human–robot collaboration	Optimization-based HRC frameworks rely on manually designed ergonomic cost functions, limiting adaptability across objectives and tasks.	Develop a learning-based control framework that embeds ergonomic objectives through experience rather than manual cost-function design.	Chapter 7
	No existing HRC approach accounts for individual physical limitations or pain-related constraints.	Enable personalized assistance by explicitly encoding physical restrictions and pain-related constraints into the control strategy.	Chapter 7

Part I

Tools for Ergonomic Risk Assessment

Chapter 3

Wearable Technology for Human Awareness

3.1 Introduction Insight

Wearable sensing technologies have gained substantial relevance in industrial and occupational health contexts, driven by the need for continuous, objective monitoring of human physical state. In particular, systems based on IMUs and sEMG enable real-time assessment of posture and muscle activity without disrupting normal work activities, supporting a shift from periodic, subjective evaluations toward holistic, continuous, data-driven ergonomic assessment [26]. This technological relevance is further reflected in the rapid expansion of the wearable sensor market, which is projected to grow by 16.65% between 2025 and 2030 [105]. Despite this growth, existing solutions remain poorly aligned with the requirements of industrial deployment, motivating the development of purpose-built wearable platforms for ergonomic monitoring.

While commercial IMU-based systems such as Movella's Xsens [24] and Rokoko Smart Suit Pro [106] exist, they fail to meet industrial requirements. further, these systems lack multi-modal sensing integration and require complex setup procedures involving multiple straps and time-consuming calibration routines. Similarly, conventional sEMG systems rely on wet Ag/AgCl electrodes that require skin preparation, precise anatomical placement, and can cause skin irritation during prolonged use [30]. Both types of systems present high costs and usability barriers that prevent widespread adoption in industrial settings.

This chapter addresses the development of two sensing systems as research platforms for ergonomic monitoring. The IMU-based motion capture system builds upon previous prototypes (ergowear v0 [35], ergowear v1 [107]), achieving reliable performance through systematic hardware optimization and professional garment manufacturing. The textile-integrated sEMG electrodes explore commercially available conductive fibers for biosignal acquisition, with the long-term goal of integrating both sensor types into unified smart garments for holistic monitoring. While proof-of-concept implementations, these systems establish technical foundations for future ergonomic assessment tools, emphasizing accessible, customiz-

able platforms free from proprietary constraints.

3.2 Ergowear

In recent years, inertial-based MoCap systems have been increasingly adopted for human motion tracking across a wide range of applications. Their main advantage is wearability, allowing natural and unconstrained movement without mobility restrictions. They can also be used both indoors and outdoors without requiring a controlled or restricted environment, unlike camera-based systems [108]. While academic research has produced numerous custom inertial systems tailored to specific applications, from single-limb monitoring [109, 110] to ergonomic evaluations [20], these proof-of-concept designs typically prioritize technical validation over the practical usability required for commercial viability. Commercial systems, by contrast, prioritize design elements critical for user adoption.

3.2.1 Design Insights from Commercial Systems

Building upon the literature review in Chapter 2.1 on wearables for human-awareness and ergonomic risk assessment, this work examined commercial MoCap systems, including Movella [24, 111], Rokoko [106, 112], NOITOM [113], Nansense [114], and AiQ Synertial [115], to identify industrial design solutions absent from academic prototypes. The analysis focused on practical implementation details: sensor housing design, attachment mechanisms, cable management, and component integration. These insights were originally written in 2021 and revised in 2025. The Xsens MVN Link version referred to in this manuscript is the legacy version, since a new version of this system, featuring updated sensor attachment and specifications, has been available since 2025.

Most commercial IMU-based motion capture systems adopt similar attachment strategies based on Velcro fixation and zippered fabric tunnels to secure and conceal sensors. Xsens MVN Link integrates 17 wired IMUs within Lycra pockets, routing cables through internal tunnels to a lower-back body pack [24, 111]. Rokoko Smartsuit Pro II follows the same approach with 19 removable IMUs, centralized in a hub pocket, and incorporates segment tightening and strain relief at sensor entry points to protect internal wiring, using PCBs with wire-to-board connectors (Figure 6e) [112]. Nansense and AIQ Synertial adopt comparable designs with thinner fabric tunnels and centralized modules, integrating up to 22 and 18 IMUs, respectively [114, 115].

In contrast, wireless systems rely on external attachment. Xsens Awinda and Noitom Perception Neuron mount IMUs directly onto body segments using elastic straps or holders, prioritizing modularity and ease

of deployment over integration [24, 113, 116].



(a) (b) (c)

Figure 5: Movella Xsens MVN Link wearable system. (a) Arm-mounted MTx sensor attached via Velcro patch, with the zipper that covers the electronics open. (b) Back view of the suit showing the upper-body sensor cable branches routed along the back. (c) Body pack mounted at the lower back, housing the central electronics and cable terminations.

In terms of textiles and materials, Xsens MVN Link uses a tight Lycra full-body suit [24, 111], while Xsens Awinda requires no dedicated suit, using straps over clothing [24, 116] and a lycra shirt. Rokoko Smartsuit Pro II employs nylon-elastane stretch fabrics available in sizes S-XL [112]. Nansense uses stretchy performance fabrics in S-L sizes without disclosed composition [114]. AIQ Synertial uses Lycra-based fabric (80% nylon, 20% spandex) [115]. Perception Neuron systems use elastic Velcro straps with unspecified composition [113].

Regarding communication, all systems support wireless transmission: Xsens MVN Link and Rokoko stream via Wi-Fi [**?**, 24, 112]; Xsens Awinda uses a proprietary 2.4 GHz protocol to a dedicated receiver [24, 116]; Nansense supports 2.4 and 5.0 GHz with onboard logging [114]; AiQ Synertial supports USB, Wi-Fi, and Bluetooth at 200 Hz [115]; and Noitom Perception Neuron 3 supports Wi-Fi, USB, and microSD for local recording [113].

In terms of component concealment, integrated garment-based solutions such as Xsens MVN Link, Rokoko Smartsuit Pro II, Nansense, and AIQ Synertial embed sensors and wiring within fabric layers or tunnels, resulting in a clean and protected design with minimal external hardware [24, 111, 112, 114, 115]. In contrast, modular systems such as Xsens Awinda and Perception Neuron expose sensors on external straps, prioritizing flexibility and ease of deployment over visual integration [24, 113, 116].

Despite these differences, all systems require removal of electronic components prior to washing.

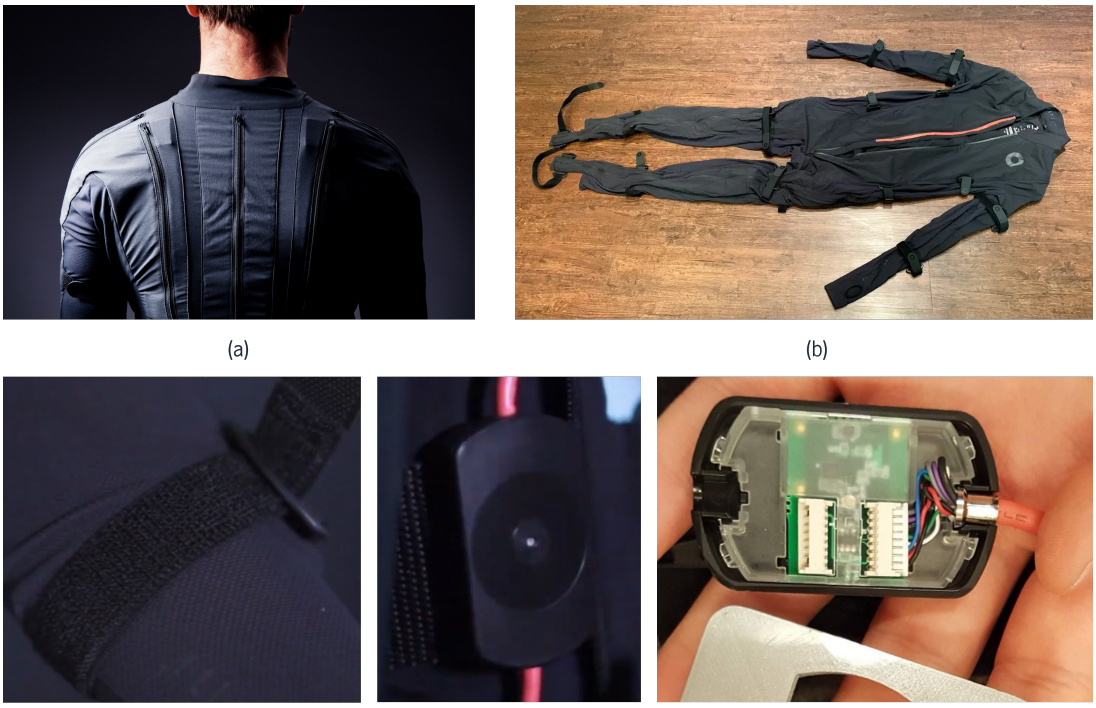


Figure 6: Rokoko Smartsuit Pro II wearable system. (a) Back view of the suit as worn. (b) Garment laid flat, showing the overall layout. (c) Velcro patch used to secure sensors inside the garment. (d) Elastic strap used for sensor fixation. (e) Opened sensor housing, exposing internal components and strain relief.

Garments can then be machine or hand washed depending on material specifications, with some systems such as Rokoko explicitly supporting machine washing at 30°C [24, 112–115].

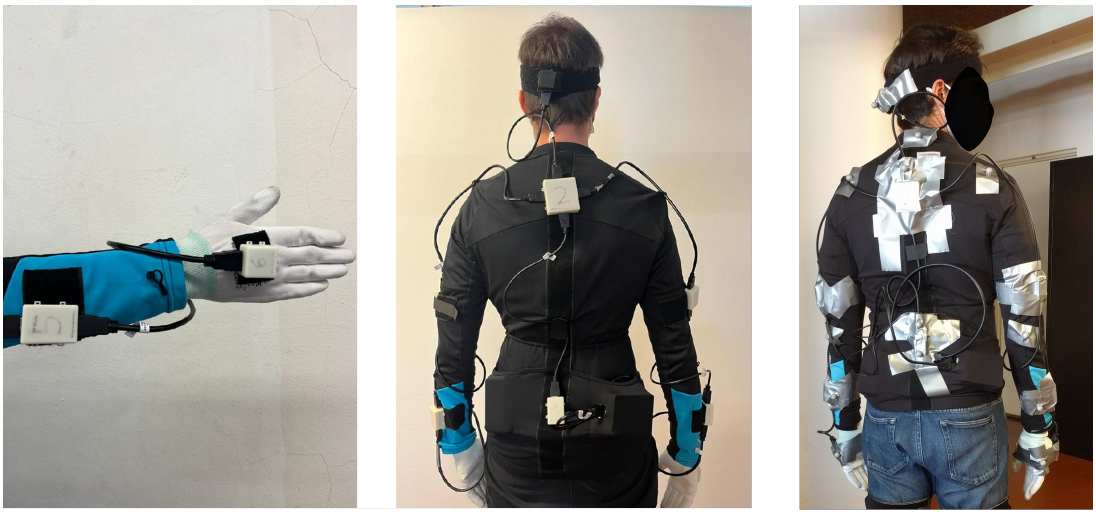
3.2.2 Solution Conceptualization

Ergowear is an inertial-based garment developed as the third iteration of two earlier prototypes. The design and implementation of the first two versions are detailed in [35, 107]. The development of this prototype focuses on improving usability, aesthetics, and overall functionality.

Insights from previous versions

This section identifies the main limitations of the previous Ergowear prototype (v1 [107]) and outlines the design changes implemented to improve performance, usability, and user acceptance. As described in [117], kinematic data were collected from 21 participants across six trials, including factory-like bimanual reaching, full upper-body motions, office tasks, and free movements. Further details are provided in Section 4.2.3.

The first issue concerned the HDMI cables and connectors linking sensor modules. Their weight and stiffness caused sensor tilting, while Velcro attachments provided insufficient support, leading to displacement. These effects compromised measurement accuracy, particularly joint angle estimation, due to incorrect sensor orientation (Fig. 7a-b). This required redesigning the connection interface to ensure secure mounting while preserving flexibility.



(a) (b) (c)

Figure 7: Ergowear 2.0: Misalignment between the a) hand sensor and the forearm sensor, and b) on the back; c) Adhesive tape used to fix the sensor modules onto the respective anatomical

A second limitation was poor garment fit. During dynamic movements, the suit shifted relative to the body, degrading sensor accuracy and vibrotactile feedback effectiveness. Although adhesive tape provided temporary improvement (Figure 7c), a redesign was required. Participants also reported reduced dexterity due to gloves. Consequently, fingerless gloves were integrated to prevent slipping while preserving precision, and shorts were connected to the jacket to form a bodysuit that prevents upward displacement. A tight fit was prioritized to ensure stable sensor positioning and reliable signal acquisition.

The system also caused thermal discomfort. The cyclist jacket retained heat, while the Raspberry Pi 4 added significant thermal load with 1200 mA consumption (out of 1827 mA total). It also limited sampling rates to approximately 100 Hz with nine sensors, which is insufficient for Strap Down Integration (SDI). These constraints motivated the use of breathable fabrics and a more energy-efficient processing unit with higher sampling capability.

Users reported discomfort from the headband due to heat and pressure, requiring redesign. Additionally, forearm sensors placed over muscle tissue shifted during contractions, introducing low-frequency noise. Repositioning them over bony regions near the wrist mitigated this issue. The sensor casing was

also redesigned to be more compact, improving comfort and usability.

Finally, users expressed discomfort wearing the suit in public due to its technical appearance, with visible wiring and electronics. Commercial systems typically conceal these components within the garment. The new design addresses this by hiding technical elements, improving aesthetics and making the system more suitable for workplace use.

Based on insights from commercial systems, academic systems reviewed in the literature, and previous prototype versions, a requirement table was defined (Table 9).

System Conceptualization

The system was conceptualized as a garment-integrated sensing and biofeedback platform whose architecture had to satisfy three concurrent objectives: reliable full-body kinematic acquisition, effective on-body vibrotactile feedback, and acceptable wearability for extended use. Figure 8 depicts the conceptualized garment.

A wired sensing architecture was selected instead of a wireless alternative. Although wireless systems offer higher modularity, they introduce challenges related to synchronization, packet loss, and battery management. In this case, wired communication ensured deterministic data synchronization across all nine IMUs and avoided the need for individual batteries on each sensor. This reduced mass at distal segments, which is important for preserving comfort and natural movement. This decision introduced the need to route cables within the garment. Rather than addressing this later, cable routing was considered from the beginning as part of the garment structure.

The garment architecture follows a principle observed in commercial systems, where electronic components are integrated within the textile to improve protection and user acceptance. In this design, sensing and fixation are handled by a compressive inner layer, while an outer layer houses and conceals the electronics. This separation supports both functional and aesthetic requirements, allowing sensor placement to be optimized for body contact while reducing visible wiring and exposed components, which are associated with a prototype-like appearance and lower user acceptance in workplace settings.

Sensor fixation was defined as a key requirement, as accurate inertial measurements depend on maintaining a stable orientation relative to the body. Elastic pockets were selected instead of Velcro patches, as they provide continuous contact with the body, adapt to local curvature, and can be integrated more easily into the garment. Velcro, in contrast, introduces local stiffness and requires additional assembly steps. This choice prioritizes stable sensor placement over reconfigurability.

The contribution of this system lies in how sensing, feedback, fixation, and wearability were addressed

Table 9: Summary of System Requirements

Type	**Requirement**	**Ideal**	**Reason**	**Priority**
Functional	Measure human movement (f = 10 Hz)	—	Accurate capture	High
Functional	Monitor trunk, neck, arms, wrists	—	Anatomical coverage	High
Functional	Wireless communication	—	Mobility	High
Functional	Real-time data streaming	—	Immediate feedback	High
Functional	Joint angle estimation	—	Motion analysis	High
Functional	Allow Ergonomic risk estimation	—	RULA feedback	High
Functional	Encourage posture improvement	—	Long-term benefit	Medium
Functional	Integration with cobot	—	HRC support	Medium
Functional	Interoperability	—	Compatibility	Medium
Usability	Comfortable to wear	—	Adoption	High
Usability	No motion restriction	—	Natural movement	High
Usability	Lightweight	0.5–1.0 kg	Comparable to jacket	Medium
Usability	Module height $<$12 cm	$<$12 cm	Respect intimate-space/proxemic constraints	Medium
Usability	Skin-safe materials	—	Prevent irritation	High
Usability	Easy to don/doff	$<$1 min	Like clothing	High
Usability	Compact, discrete form	—	User acceptance	Medium
Usability	Intuitive interface	—	Low training need	High
Usability	Rechargeable	—	Daily use	Medium
Usability	Aesthetic design	—	Visual appeal	Low
Technical	8h autonomy	8 hours	Full work cycle	High
Technical	Packet loss $<$1%	$<$1%	Reliable data	High
Technical	Data Corruption $<$2.5%	0% target	Data integrity	High
Technical	Latency $<$19 ms	$<$19 ms	Real-time feedback	High
Technical	10 m range	10 m	Workplace flexibility	Medium
Technical	0% command failure	0%	Robust control	High
Technical	No I2C errors	—	Sensing reliability	High

together within a single garment. Commercial systems typically focus on motion capture performance, while academic prototypes often demonstrate functionality without full integration. In contrast, this design treats the garment as the interface between the body and the electronics from the beginning, resulting in a system that is stable on the body, visually integrated, and suitable for ergonomic applications.

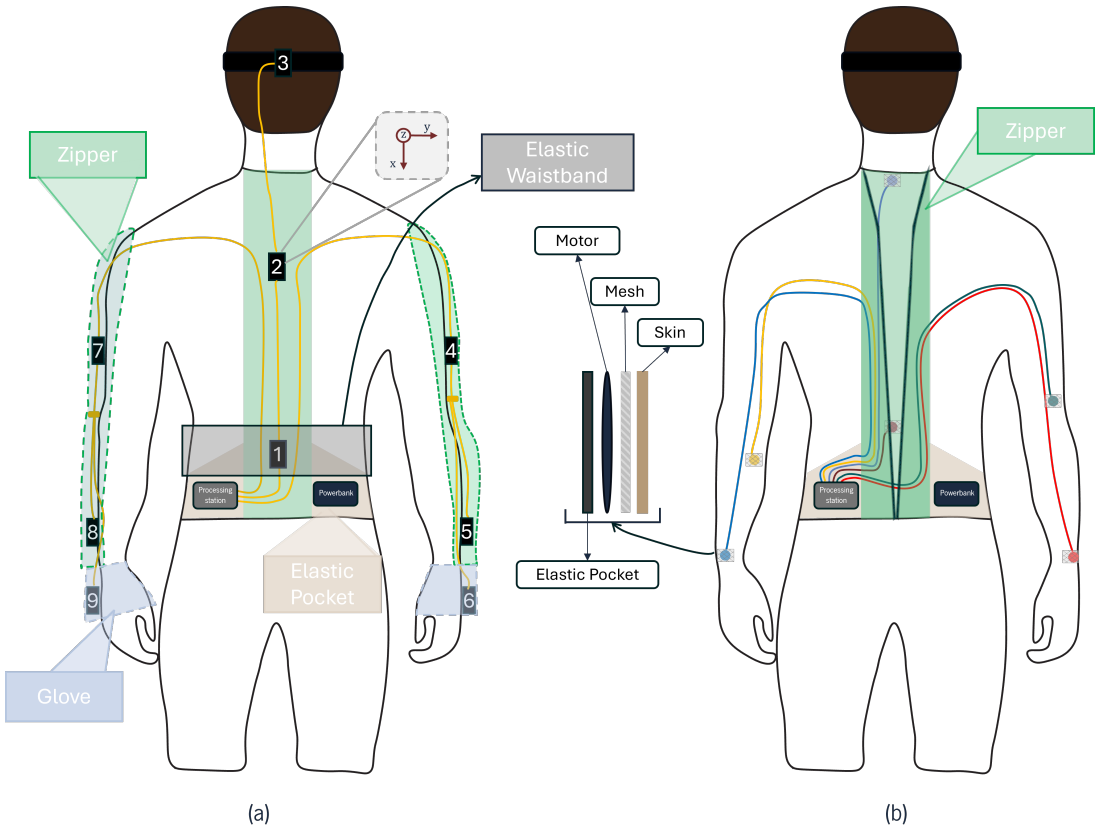


Figure 8: Smart garment layout showing (a) the sensing system and (b) the biofeedback system.

3.2.3 Solution Development

Textile Design

The textile design implements the requirements defined during system conceptualization through an iterative process, where multiple prototypes were evaluated to improve sensor stability, mobility, and comfort.

To improve stability in the lumbar region, the garment was redesigned as a bodysuit with an integrated elastic waistband. The waistband is inserted through slotted openings and provides adjustable compression, reducing relative movement between the garment and the body during dynamic tasks.

Fabric selection followed the need to balance sensor fixation and user mobility. The chosen textile composition (95% polyamide, 5% elastane) provided sufficient compression to maintain sensor–skin contact

while preserving range of motion. A higher elastane composition (92% polyamide, 8% elastane) was also tested, but resulted in excessive compressive stress, restricting joint movement and reducing comfort. This shows that increasing elastane content does not necessarily improve wearability, as higher compression can degrade usability.

To preserve mobility in highly articulated regions, such as the shoulders and elbows, lighter and less dense mesh structures were introduced (Figure 9). These reduce resistance during movement, while areas containing sensors maintain more compressive behavior to ensure stable fixation.



Figure 9: Design features of the garment: (a) Light mesh placement on the elbow region; (b) Light mesh placement on the shoulder region; (c) Elastic pockets for securing the sensors.

To support the biofeedback system, textile regions where vibration motors are placed were designed to maintain direct contact with the body. Low-density mesh and elastic pockets were used to reduce damping and ensure consistent perception of the feedback.

An elastic headband with an integrated pocket was developed to secure the head sensor while maintaining comfort. In addition, fingerless gloves were integrated to stabilize distal sensors without affecting manual dexterity.

An outer textile layer was added to house and conceal the electronic components. Zippers provide

access to the internal hardware while maintaining a clean external appearance. Textile tunnels were not used to route cables between the trunk and sleeves, as they would increase stiffness in the shoulder region and restrict movement. Instead, cable routing was integrated into the garment while preserving flexibility in high-mobility areas.

Hardware Development

The system comprises a data acquisition unit, a biofeedback module, a low-level processing unit, a wireless communication interface, a power supply unit, and the necessary interconnections. An overview of the hardware architecture is shown in Figure 10. More details on the hardware implementation can be found in appendix B.1.1.

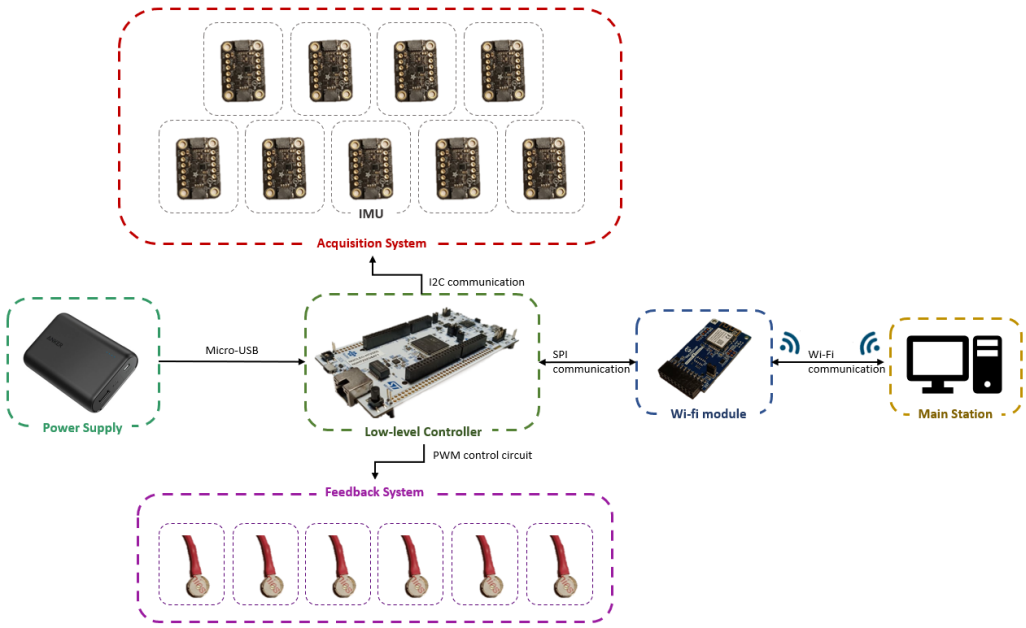


Figure 10: System's hardware overview.

Acquisition System The acquisition system consists of nine IMU sensors distributed across the upper body. The selected configuration, based on the LSM6DSOX and LIS3MDL, provides improved bias stability and reduced noise compared to the MPU-9250 used in previous prototypes [118–120]. This choice was motivated by the need to reduce drift in orientation estimation while maintaining a low-cost solution.

The use of the integrated connectors of the sensor enabled compact module design and simplified integration into the garment, avoiding the need for custom PCBs at the sensing level (Figure 11b).

Communication Architecture The communication architecture is constrained by the sensor boards, which support a limited number of I^2C addresses. To interface all nine sensors without introducing multi-

plexers, the system was distributed across multiple I^2C buses. These were organized into three branches corresponding to the trunk/head and each arm, aligning the communication structure with the physical layout of the garment (Figure 11a). This approach reduces communication bottlenecks and simplifies system organization.

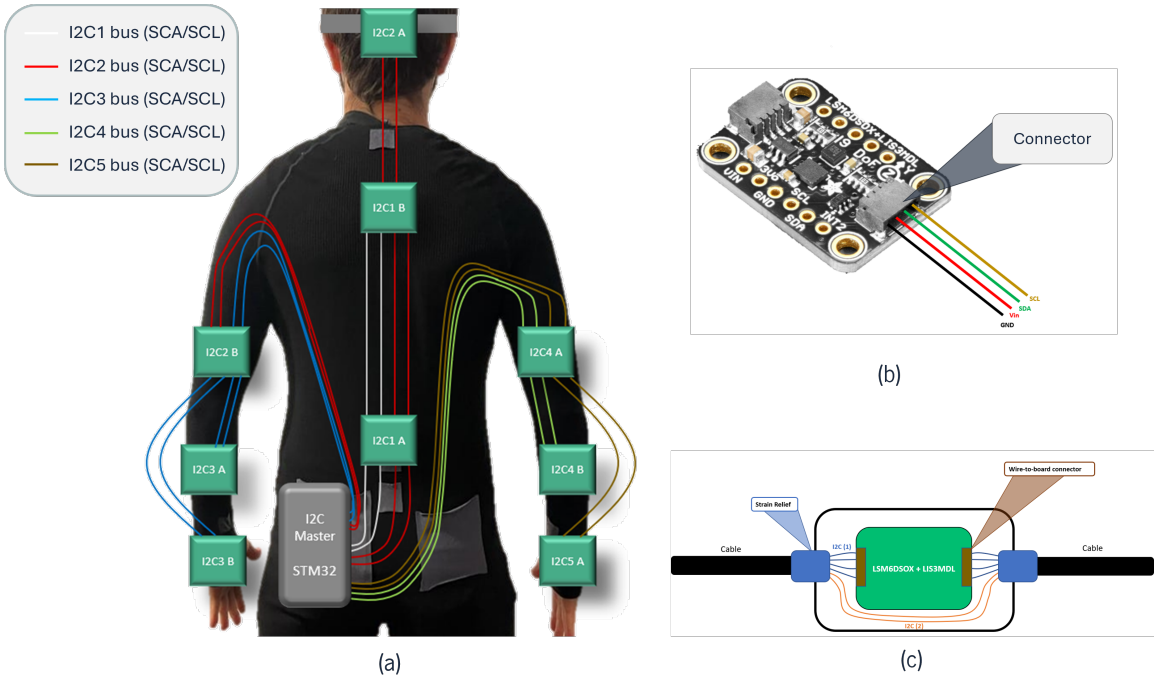


Figure 11: Sensors' communication architecture and wiring configuration.

To maintain continuous communication across sensors within each branch, a pass-through wiring approach was implemented, allowing signals to propagate along the chain without requiring independent connections to each module (Figure 11c).

Mechanical Integration and Cabling Mechanical integration of the sensing system was driven by the need to preserve user mobility while maintaining reliable connections. Sensor interconnections were positioned along the midpoint of anatomical segments, avoiding joints to reduce cable stress during movement.

A branching solution was introduced at the upper arm to separate connections to the forearm and hand. This avoided placing connectors near the wrist or introducing excess cable length, both of which would interfere with movement. Connector selection prioritized mechanical robustness and ease of use to ensure reliable operation under repeated motion.

Biofeedback System The biofeedback system includes six vibrotactile motors placed on key body regions to indicate poor ergonomic posture. Coin-style ERM motors were retained from previous prototypes,

as they provided reliable performance and adequate perceptual feedback [121].

Due to the current requirements of the motors, a dedicated control circuit was implemented using Pulse Width Modulation (PWM) signals. This enables both safe operation and modulation of vibration intensity.

Processing System The processing architecture is divided into a low-level embedded controller and a main processing station implemented on a personal computer. The embedded controller handles sensor acquisition, communication, and actuation, while the main station performs higher-level processing.

The selection of the embedded platform was driven by requirements for multiple I^2C interfaces, sufficient PWM outputs, and low power consumption. The Raspberry Pi 4 (RPi4) used in previous prototypes was excluded due to its high power consumption and heat generation. The NUCLEO-H723ZG was selected as it satisfies all functional requirements, particularly the availability of multiple I^2C buses, while maintaining lower power consumption.

Wireless Communication Wireless communication between the wearable and the main station is achieved using a WiFi module interfaced via SPI. WiFi was selected over Bluetooth due to its higher data throughput and more suitable support for continuous data streaming.

Power Supply The system is powered by an external power bank (Anker PowerCore 13000 mAh power bank (Anker, Changsha, China)) connected to the NUCLEO-H723ZG. The power design was guided by the requirement of continuous operation over approximately 8 hours.

In addition to capacity considerations, component placement was used to improve wearability. The power bank and processing unit were positioned symmetrically near the lower back to balance mass distribution and avoid asymmetric loading during use.

System Integration The hardware components were integrated into the garment through a combination of custom enclosures and distributed wiring, following the architectural constraints defined during the design phase. Sensor modules were designed to support both intermediary connections and terminal positions, enabling continuous signal routing along each branch while adapting to different anatomical locations.

Figure 12 depicts ergowear modules. The master module has dimensions of 146 mm $\times$ 92 mm $\times$ 37 mm and a mass of 189 g. Sensor modules measure approximately 52 mm $\times$ 22 mm $\times$ 13 mm and weigh 15 g (intermediary) or 12 g (extremity).

Mechanical integration was guided by two main requirements: protecting the electronics and preserving

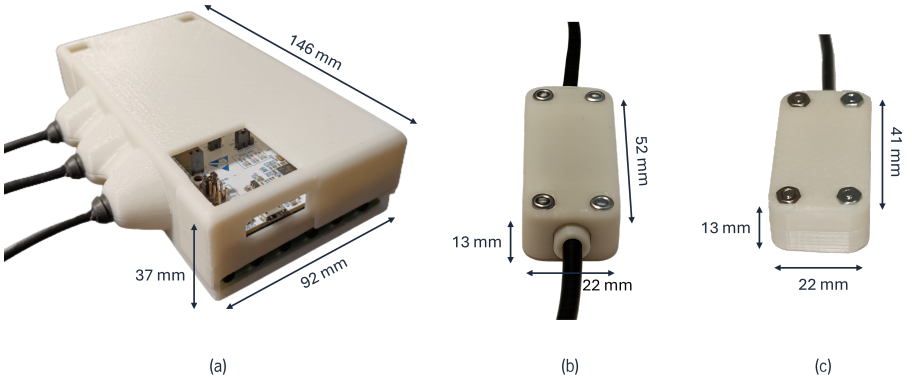


Figure 12: Ergowear modules: (a) master module, (b) intermediary sensor module, and (c) extremity sensor module.

user comfort. Custom enclosures were developed to house the processing unit and sensing modules, with geometries compatible with the textile pockets and rounded edges to avoid discomfort during wear. The modular design also allowed consistent placement of sensors across the body while maintaining flexibility in cable routing.

To ensure reliable operation during movement, costum strain relief mechanisms were incorporated at cable entry points, reducing mechanical stress on the connections. The overall wiring layout was distributed across the garment to avoid concentration of cables in high-mobility regions and to maintain a balanced integration.

The final system integrates sensing, processing, feedback, and power within a single wearable structure, while maintaining stability on the body and compatibility with the textile design. The complete wearable system, including electronics and garment, weighs approximately 1130 g. Figure 13 shows the full assembly, open back and sleeve of the wearable's outer layer.

Embedded Software Development

The system architecture is composed of two processing layers: a low-level processor embedded in the wearable unit and a main station implemented on a personal computer. The processing workload is distributed, with the low-level processor focusing on time-sensitive operations and the main station handling higher-level computations. An overview of the software architecture is presented in figure 14.

To ensure real-time performance under high sensor sampling rates, the tasks executed by the low-level processor were kept to a minimum. This processor is responsible for (1) handling sensor calibration; (2) acquiring raw sensor data; (3) applying SDI to compute orientation estimates; (4) transmitting the processed data wirelessly to a main processing unit; and (5) receiving and executing motor activation

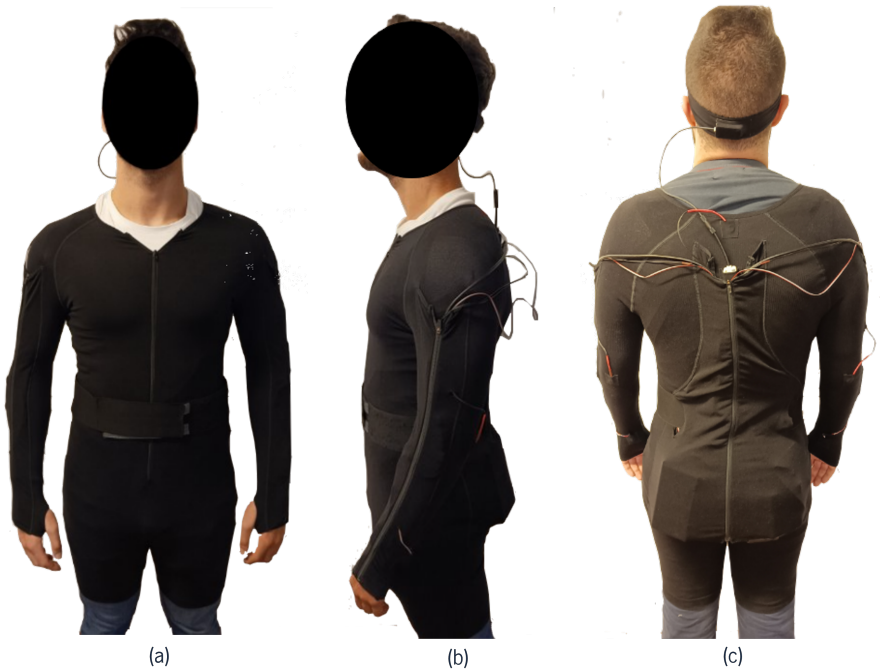


Figure 13: Ergowear smart garment: (a) front view, (b) lateral view, and (c) rear view.

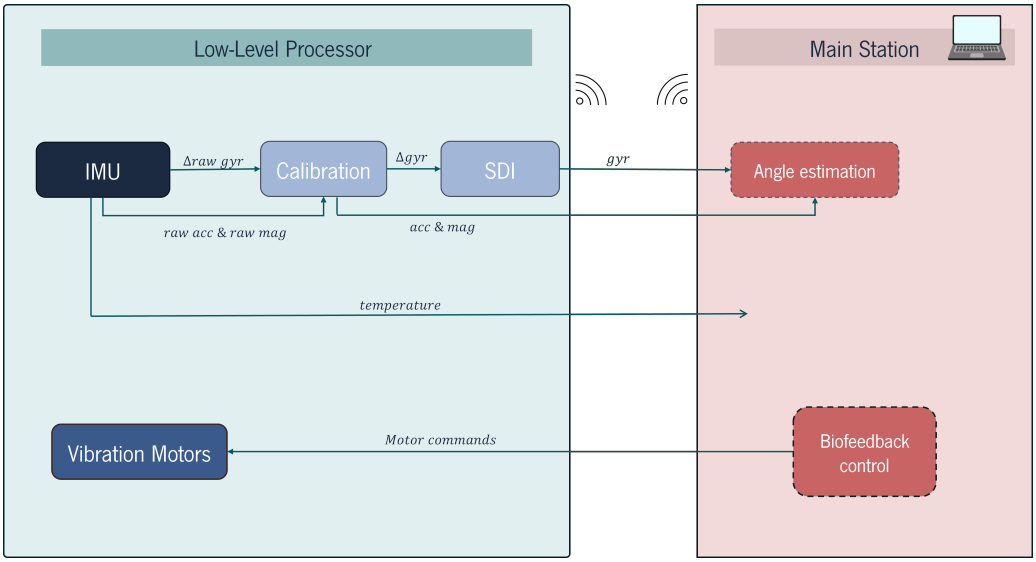


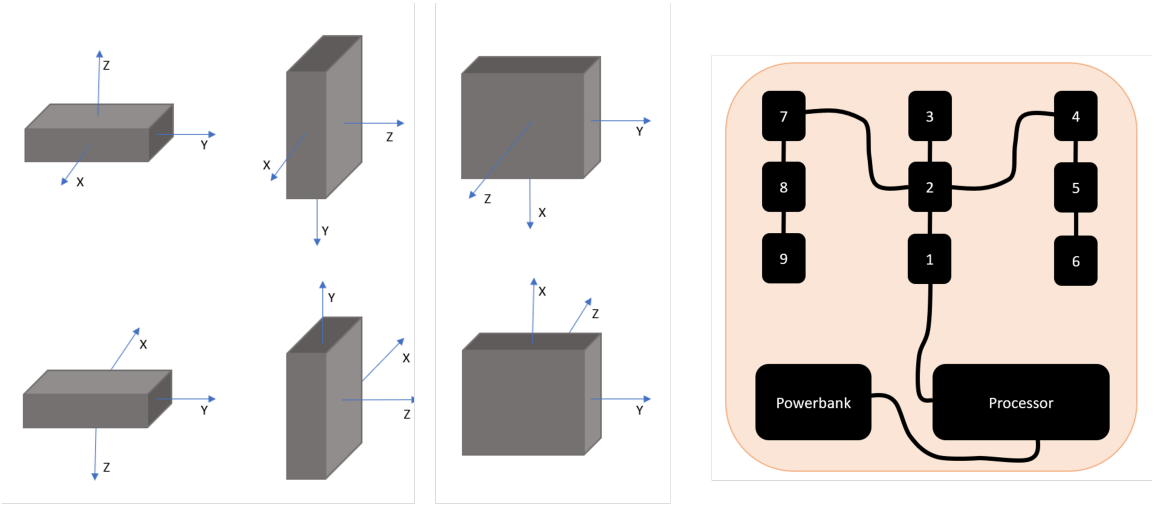
Figure 14: Software architecture overview.

commands from the main station. The main station receives the data and handles high-level processing and system coordination, while also providing the computational interface for extensions, such as joint angle estimation, ergonomic risk assessment, and postural biofeedback functionalities.

A detailed description of the embedded software architecture, including execution flow, interrupt handling, RTC setup, and UDP packet structure, is provided in Appendix B.1.9.

Sensor Configuration The LSM6DSOX + LIS3MDL sensors were configured with Output Data Rates (ODR) of 1.66 kHz and 80 Hz, and full-scale ranges of $\pm 16g$ and $\pm 16G$ for the accelerometer and magnetometer, respectively. For the gyroscope, an output data rate of 416 Hz was selected, as tests it provided greater stability in the measured values, along with a full-scale range of $\pm 2000^{\circ}/\mathrm{s}$. These values were based on observations from the experimental data presented in [117].

Sensor Calibration Before acquiring data from the IMUs, calibration is required to mitigate sensor-specific errors. The primary sources of error include bias and scale factor deviations. For magnetometers, additional distortions such as hard iron (constant magnetic offsets) and soft iron (environmental magnetic interference) effects must also be compensated [109]. Bias represents a constant offset from the true value and is corrected by subtracting this offset. In turn, scale factor error reflects the proportional discrepancy between measured and true values, and is corrected using predefined calibration routines.



(a) Accelerometer 6 position calibration method

(b) Magnetometer calibration setup

Figure 15: Illustration of the calibration procedures: (a) Six-position method used for accelerometer calibration. (b) Magnetometer calibration setup, with the sensors, processing station, and power bank mounted on a flat cardboard surface.

Accelerometer calibration was performed using a bench routine in which each sensor is placed statically in six orientations, each held for five seconds [122]. In each position, one of the sensor's axes (x, y, or z) is aligned with the gravity vector. For each axis, the maximum and minimum values are recorded and stored in the processor firmware, as illustrated in Figure 15a). Once the extreme values are obtained,

future measurements are calibrated using:

$$\text{calibrated_value} = \left(\frac{\text{raw_value} - \text{acc}_{\text{min}}}{\text{acc}_{\text{max}} - \text{acc}_{\text{min}}} \right) \times (\text{out}_{\text{max}} - \text{out}_{\text{min}}) + \text{out}_{\text{min}} \tag{3.1}$$

where `raw_value` is the uncalibrated measurement, $\texttt{acc}_{\text{min}}$ and $\texttt{acc}_{\text{max}}$ are the minimum and maximum measured values during calibration, and $\texttt{out}_{\text{min}}$ and $\texttt{out}_{\text{max}}$ represent the full-scale sensor range (−16 g to 16 g). This equation is applied independently to each axis.

Gyroscope calibration is performed on-body during system initialization. During a five-second static period, in which the user adopts the n-pose, gyroscope data are acquired at the nominal sampling rate. The mean value of each axis is computed and stored as the sensor offset, which is subtracted from subsequent measurements to reduce bias.

Among the sensing modalities, the magnetometer requires the most extensive calibration. The system is mounted on a planar cardboard support with all IMUs horizontally aligned, and each IMU is recorded for 60 seconds, starting with the x-axis aligned to magnetic North. The board is then rotated about each axis to sample a broad range of orientations, as shown in Fig. 15b). For each axis, the midpoint between the maximum and minimum values is computed and used as the magnetometer offset.

Strap Down Integration is a mathematical procedure used to compute angular displacement by integrating gyroscope data over time.

The algorithm samples angular velocity and integrates it to compute incremental orientation updates (Δq). Unlike direct integration, which is prone to drift and noise,SDI compensates for dynamic effects such as coning and sculling errors, which arise from high-frequency rotations and combined rotational–translational motion. These corrections improve accuracy during rapid or complex movements, and the integrated outputs are then provided to the system's attitude estimation filters.

In Movella's XSens systems, the SDIalgorithm is implemented with high-frequency sampling (typically 1kHz) of gyroscope and accelerometer data. As noted in [123], high sampling rates are critical to maintaining orientation accuracy. Validation studies in human motion tracking have also shown that lower sampling frequencies significantly increase orientation drift and that for reliable estimation of human movement, SDIrates of several hundred hertz are required. Figure 16 presents an example provided by Movella [1] that illustrates the impact of low sampling frequencies on the accuracy of motion capture.

In this project, the SDI block was implemented using gyroscope data only. During dynamic movements, accelerometers measure both gravity and motion-induced accelerations, making them unreliable for drift correction except in quasi-static conditions [124]. Unlike standard implementations that output orienta-

tion [123], the SDI block here estimates angular velocity (in degrees per second) over five consecutive frames using trapezoidal integration.

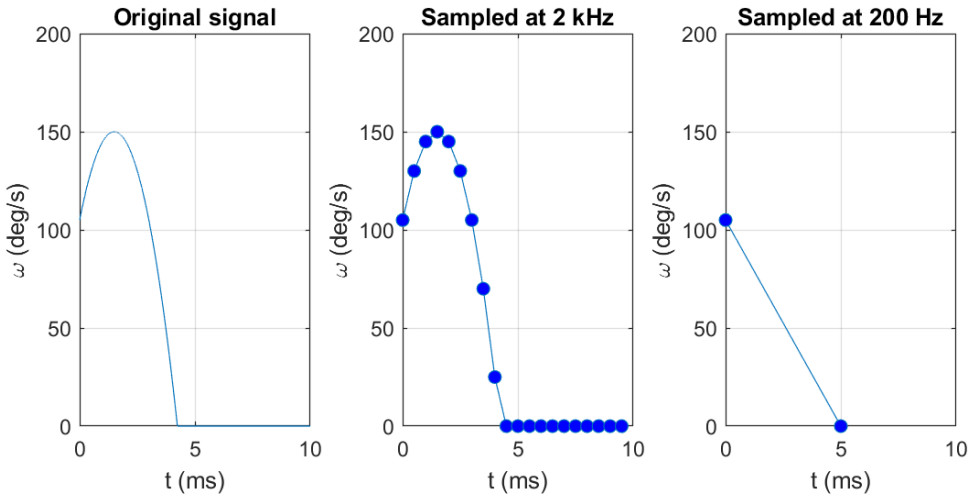


Figure 16: Comparison of high-bandwidth gyroscope data sampled at high and low frequencies, highlighting the loss of detail at lower sampling rates. Image from [1]

Trapezoidal Integration is a numerical method used to approximate the area under a curve by dividing it into small segments, each represented as a trapezoid [1]. When applied to gyroscope signals, this method estimates angular displacement by integrating angular velocity over time. For each pair of adjacent samples, the area is computed as the average of the two values multiplied by the time interval between them. This allows accurate estimation of rotation even from discrete data. Given a sampled gyroscope signal $\omega(t)$, the total rotation Δq over a time window can be approximated as:

$$\Delta q \approx \sum_{i=1}^{N-1} \frac{\omega_i + \omega_{i+1}}{2} \cdot \Delta t$$

where ω_i and ω_{i+1} are consecutive angular velocity samples, Δt is the sampling interval, and N is the total number of samples.

3.2.4 Experimental Methodology

To ensure proper system operation and define its specifications, a validation protocol was developed to characterize system timing, autonomy, storage capacity, communication reliability, and mechanical robustness. The protocol targeted the technical requirements defined in Section 3.2.2 and followed a procedure similar to that reported in [107]. All tests were performed with a fully charged battery and

conducted in the same controlled environment, namely a closed room. The *Wi-Fi Analyzer* application (Farproc) was used to survey surrounding wireless activity and identify occupied IEEE 802.11 channels. The analysis indicated the presence of six active Wi-Fi networks and eleven available channels in the test environment.

System Timings

Timing measurements for each core system process were obtained using oscilloscope recordings. To verify that real-time constraints were met, execution times of key tasks related to IMU acquisition, processing, and data transmission were measured by instrumenting each process with a GPIO toggle. The pin was set `HIGH` immediately before execution and `LOW` afterward, enabling precise duration measurement on the oscilloscope.

The analysis focused on the two algorithmic blocks that constrain sampling and output rates: the interrupt block and the sending block. The sending block includes the SDI algorithm, data packaging, and the Wi-Fi event handler, which were evaluated both individually and jointly to quantify their processing load. The interrupt block, responsible for IMU acquisition, was characterized by measuring the duration of individual sensor reads for both single-IMU and full-system acquisition. The total interrupt time, including gyroscope, accelerometer, magnetometer, and temperature readings, was used to estimate the maximum achievable sampling frequency.

To assess temporal stability, sequences of five consecutive interrupt cycles were analyzed and compared against the duration of the sending block to verify that both could reliably execute within the same sampling window. These measurements guided the selection of the system sampling rate, initially set to 250 Hz as the maximum frequency supporting stable acquisition and transmission.

Wireless Range Protocol

To assess the effect of distance on communication quality, the wearable system was placed on a planar table and positioned at predefined distances from the main station: 0.5, 1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, and 8.0 meters. The main station was a personal laptop running Ubuntu on VMware (VMware, Palo Alto, USA). The maximum test distance was limited by the physical constraints of the room. Each distance was tested in three trials of five minutes, totaling 27 trials and 135 minutes of data.

Wireless Interference Protocol

To determine the effect of obstructions, such as walls and other wireless interferences a test where the main station and the smart garment were separated by a 35 cm thick wall, with a total separation of approximately 0.5 meter. The environment also featured a high level of wireless interference, with 31 active networks detected using the WiFi Analyzer application. The test was repeated three times, with each trial lasting five minutes.

Autonomy Protocol

Battery autonomy was evaluated under two operating modes: (i) monitoring mode, with continuous data acquisition and transmission, and (ii) feedback mode, which additionally activated vibrotactile motors at 1-second intervals. In both cases, the system was placed 0.5 m from the main station and operated continuously until battery depletion. During each test, wireless communication metrics including sampling frequency, packet loss, and data corruption were recorded hourly to assess the effect of decreasing battery voltage on communication stability. All experiments were conducted in the same controlled environment used for the wireless range tests, with minimal electromagnetic interference.

Usability Assessment Protocol

Ergonomic and usability testing of the wearable system was conducted with 9 healthy volunteers (5 male, 4 female, 65,5 ± 10,0 kg, 169,4 ± 7,8 cm). Participants performed a structured sequence of movements designed to evaluate the system's wearability across a full range of upper body motions. These included: (i) crossing both arms; (ii) trunk flexion/extension; (iii) trunk rotation; (iv) trunk side-bending; (v) neck flexion/extension; (vi) neck rotation; (vii) neck side-bending; (viii) arms stretching at the front, up, and back; (ix) shoulder rotation; (x) wrist rotation; (xi) wrist flexion/extension; (xii) wrist ulnar/radial deviation; (xiii) jumping; (xiv) a lifting task involving picking up a box from the floor and placing it at maximum reachable height. Each motion was performed three times by every participant. Figure 17 displays examples of the tested movements.

To assess the effectiveness of the vibrotactile biofeedback, an additional test was performed in which each feedback motor was activated individually and randomly. Participants were asked to identify the location of the perceived vibration. After completing the physical tasks, each volunteer completed an anonymous usability questionnaire, based on the System Usability Scale (SUS) template, customized for this wearable. Responses were recorded on a 5-point Likert scale (1: Strongly Agree, 5: Strongly Disagree). Finally, to assess communication reliability during movement, wireless performance was monitored in a 5-minute trial for five of the participants. The aim was to detect any I2C communication failures during



(a) (b)

(c) (d)

Figure 17: Examples of the movements performed during the usability tests. (a) Trunk flexion; (b) wrist flexion; (c) crossing arms; (d) placing a box on the top of a cabinet;

operation.

3.2.5 Results

System Timings

The timing measurements for each core process in the system were acquired using oscilloscope recordings. A summary of the measured durations is presented in Table 10.

Table 10: Execution times of system processes at a 250 Hz sampling rate

Process	Duration	Description
Interrupt Block		
Single gyroscope read	250 µs	
All gyroscopes (9 IMUs)	2.25 ms	
All sensors (1 IMU)	600 µs	Acc., Mag., Temp. - No Gyro
All sensors (9 IMUs)	1.80 ms	Acc., Mag., Temp. - No Gyro
Combined (Gyro + 3 IMUs full)	4.10 ms	Interrupt cycle duration
Sending Block		
SDI computation	35 µs	
Data packaging	20 µs	Message assembly
Sending function only	1.55 ms	Transmission logic
Total sending routine	1.58 ms	Includes all above components

The sending block executes in approximately 1.58 ms, with the UDP transmission itself accounting for 1.55 ms, making it the dominant contributor to processing time. When the UDP sending event is disabled, the Wi-Fi handler introduces no additional delay. In contrast, the SDI computation and data packaging require only 35 µs and 20 µs, respectively, and therefore have a negligible impact on overall timing.

Gyroscope acquisition from a single IMU takes approximately 250 µs, while reading all nine gyroscopes requires about 2.25 ms. Acquiring accelerometer, magnetometer, and temperature data from all IMUs adds roughly 1.8 ms. When combined with gyroscope acquisition from three IMUs, which defines the interrupt workload, the total interrupt duration reaches approximately 4.1 ms.

A target sampling rate of 250 Hz would require the interrupt routine to complete in under 4 ms. Since the measured duration is 4.17 ms, the sampling rate was reduced to 240 Hz, yielding an effective output frequency of 48 Hz (240/5). After reconfiguration, five consecutive interrupt cycles were analyzed to verify temporal stability. Figure 18 shows the oscilloscope trace of the interrupt events (blue) and the sending block (yellow), confirming that the sending routine executes fully within a single interrupt cycle and that no overlap or timing overrun occurs across the five-sample window.

Given the sending function duration of 1.55 ms per cycle, the Atwinc1500 WiFi module operates in sending mode for approximately 0.0744 s per second, remaining in standby mode for roughly 92.56% of the time. Based on the datasheet values for current consumption in active and standby states, the

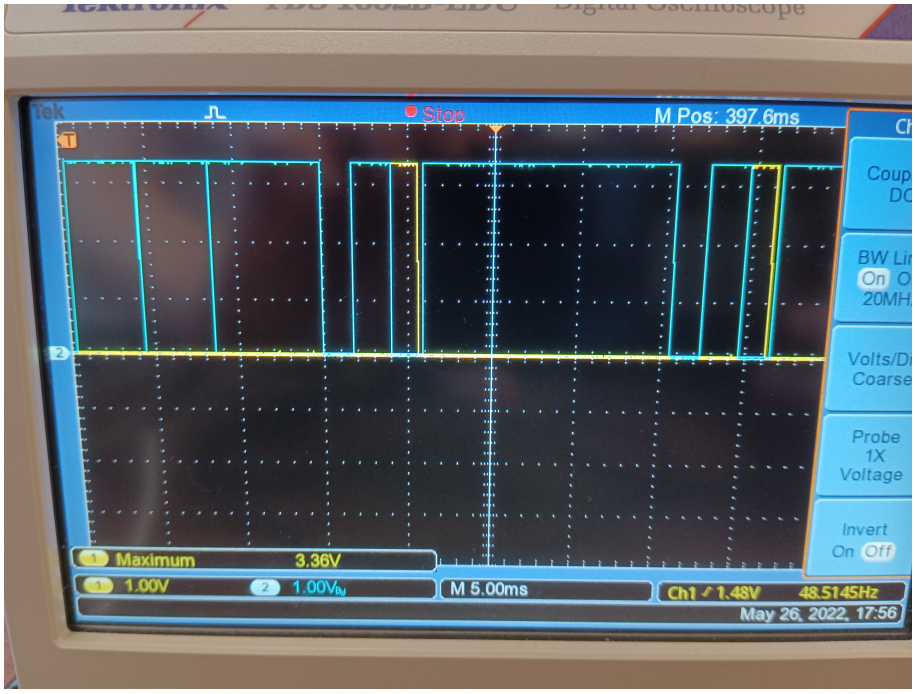


Figure 18: Oscilloscope capture showing five consecutive interrupt cycles (blue) and the execution window of the total sending routine (yellow), used to verify timing compatibility at a 240 Hz sampling rate.

WiFi module's estimated power usage is approximately 162.33 mAh during an 8-hour session [125]. This estimate was computed for a 250Hz sampling rate (50 transmissions per second) and is thus slightly conservative for the final configuration at 240Hz, where the expected power consumption is marginally lower. All subsequent tests were conducted with the system operating at a 240 Hz acquisition frequency.

Wireless Range tests

Table 11 and Figure 19 summarize the communication performance across all tested distances. No data corruption was detected at any distance. The sending frequency remained stable, ranging from a maximum of 48.50 Hz at 0.5 m to a minimum of 48.11 Hz, demonstrating consistent performance across the full range.

Packet loss was negligible up to a distance of 5 m. A maximum packet loss of 0.65% was observed at 6 m, still well below the 1% threshold defined by the system requirements. Beyond this distance, packet loss increased slightly but remained within acceptable limits for real-time operation.

The Round-Trip Time (RTT) results showed that latency remained within specifications. The highest and lowest average RTT values yielded estimated one-way latencies of approximately 17.32 ms and 16.3 ms, respectively. These values conform to the requirement of remaining below 19 ms. However, standard deviation in RTT was non-negligible, reaching up to 8.05 ms at 0.5 m, indicating occasional variance in

Table 11: Influence of the distance between the main station and the low-level processing unit on the wireless communication performance.

Distance (m)	Frequency (Hz)	Packet loss (%)	Data Corruption (%)	RTT (ms)
0.5	48.50 (± 0.91)	0.00	0.00	35.37 (± 8.05)
1	48.45 (± 0.96)	0.00	0.00	33.38 (± 6.85)
2	48.49 (± 0.96)	0.00	0.00	33.50 (± 7.10)
3	48.48 (± 0.94)	0.00	0.00	33.44 (± 6.56)
4	48.47 (± 0.94)	0.00	0.00	34.33 (± 7.98)
5	48.20 (± 2.67)	0.46	0.00	33.32 (± 6.49)
6	48.11 (± 4.11)	0.65	0.00	34.21 (± 7.22)
7	48.49 (± 1.14)	0.02	0.00	33.41 (± 6.58)
8	48.43 (± 1.79)	0.09	0.00	34.38 (± 6.96)

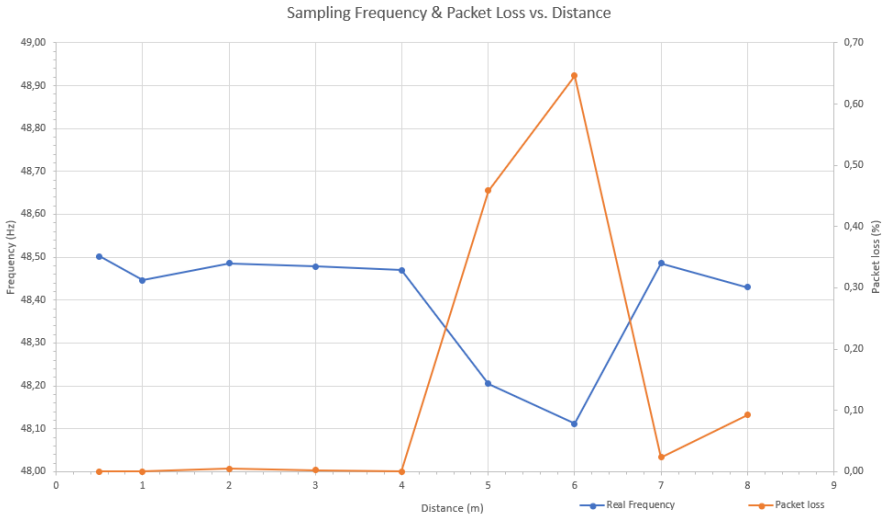


Figure 19: Real sampling frequency and packet loss as a function of the distance between the main station and the processing unit.

response time even at short ranges.

Wireless Interference tests

As shown in 12, the interference test resulted in an increase in packet loss of approximately 0.06% compared to the previous test, conducted at the same distance between the main station and the pro-

Table 12: Influence of wall obstruction and wireless interference on the wireless communication performance.

Interference	Frequency (Hz)	Packet loss (%)	Data Corruption (%)	RTT (ms)
W/ Wall	48.38 (± 1.51)	0.06	0.00	33.67 (± 8.03)
W/o Wall	48.50 (± 0.91)	0.00	0.00	35.37 (± 8.05)

cessing unit. Despite this slight increase, the remaining performance metrics remained stable, indicating that the presence of external interference—specifically the obstruction introduced by a concrete wall—did not significantly impact the wireless communication performance. These results suggest that, under the conditions in which the tests were conducted and at a separation distance of 0.5m, the system meets the predefined technical requirements. Furthermore, the findings reinforce the conclusion that the wireless communication performance of the new system represents a significant improvement over the previous iteration of the Ergowear.

Autonomy tests

Table 13: Results of the autonomy tests under different operating modes.

Operating Mode	Autonomy (h)
Monitoring mode	23:30
Biofeedback mode	15:30

As shown in Table 13, the system achieved an autonomy of 23 hours and 30 minutes in monitoring mode and 15 hours and 30 minutes in feedback mode. These durations significantly exceed the 8-hour operational requirement of the system.

Figure 20 presents the evolution of the wireless communication performance over a 23.5-hour monitoring period, including the real sampling frequency, packet loss, and data corruption. No packet loss or data corruption was observed throughout the experiment. The sampling frequency remained stable, with an average value of approximately 48.55 Hz and only minor fluctuations, ranging from 48.51 Hz to 48.60 Hz over time.

The estimated current consumption in monitoring mode was approximately 4561.68 mAh for 8 hours

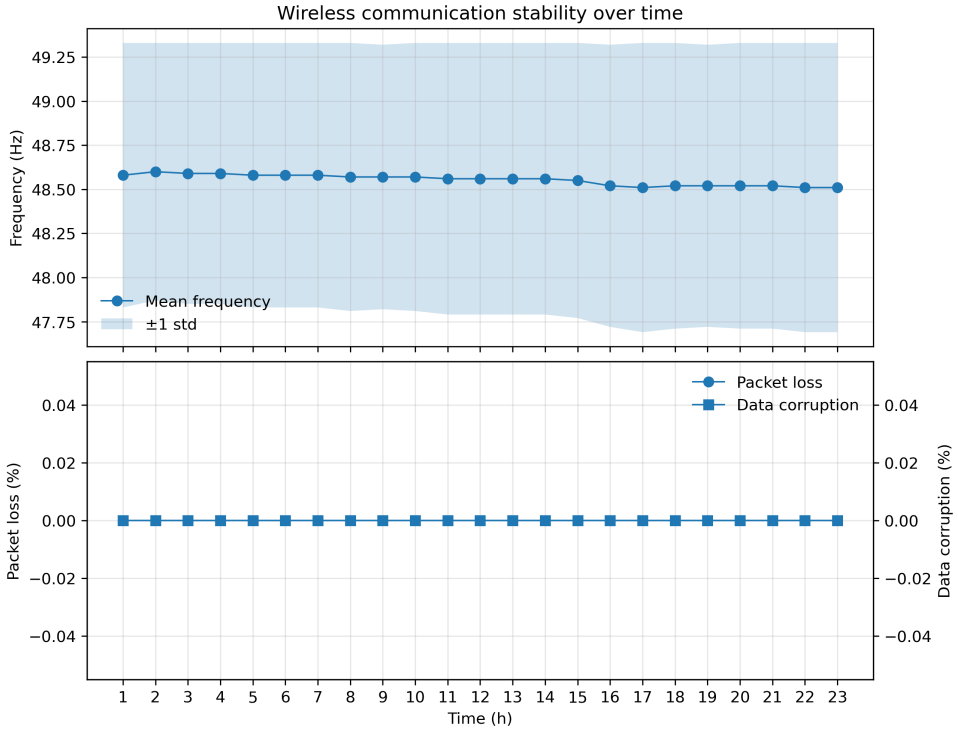


Figure 20: Real sampling frequency (Hz), packet loss (%), and data corruption (%) of the wireless communication setup over time (in hours).

of operation, as shown in Equation 3.2.

$$\textit{Current} = \frac{13400}{23.5} = 570.21\,\text{mA} \tag{3.2}$$

For feedback mode, the estimated consumption was higher, as described in Equation 3.3. These findings indicate that the current power supply is overdimensioned for standard use.

$$\textit{Current} = \frac{13400}{15.5} \times 8 = 864.52 \times 8 = 6916.13\,\text{mAh} \tag{3.3}$$

Usability Assessment tests

Figure 21 shows that all participants successfully identified the vibrotactile stimulus for all motors except the one positioned in the lumbar region. For this motor, only five participants reported clear perception; two were neutral, and two reported no or unclear sensation. This limitation was attributed to the lumbar curvature, particularly in female participants, and to suboptimal compression by the waistband, which failed to press the motor evenly due to the larger sensor module.

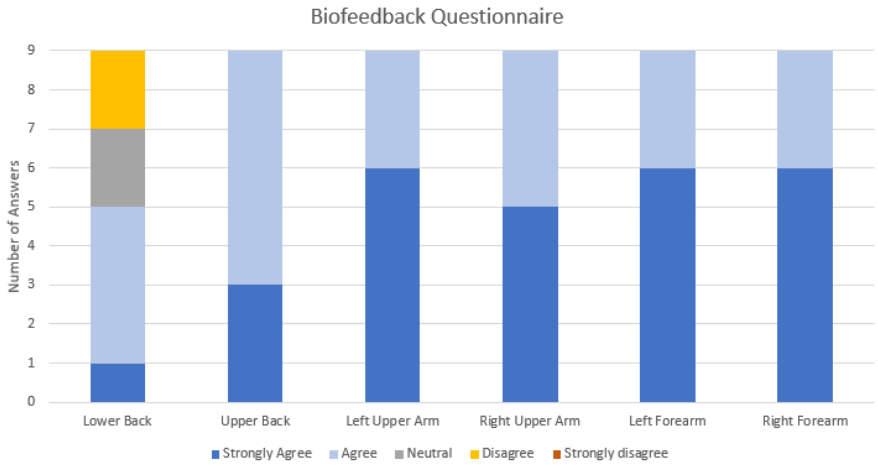


Figure 21: System's Biofeedback questionnaire answers.

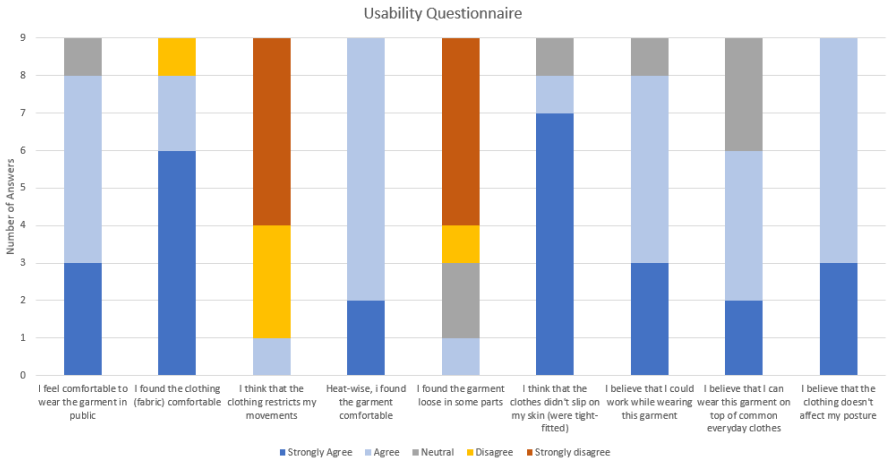


Figure 22: System's Usability questionnaire answers.

In terms of communication stability, one sensor failure occurred during the first subject's trial due to a short-circuit caused by a detached cable shield inside a connector. This issue was resolved by reinforcing the weld. No further communication failures were observed in subsequent trials.

The SUS-based questionnaire presented in figure 22 indicated strong overall acceptance of the wearable system. Participants reported no significant heat accumulation, no restriction of range of motion, and high comfort. Only one participant noted some limitation in upper body movement. Most volunteers expressed willingness to wear the system in public, suggesting high user acceptance.

Some limitations were reported. The most significant was discomfort caused by the thumb-support textile, particularly for male users with broader shoulders. This was linked to the lack of stretch in the sleeve design and the pressure exerted by the textile in that region. Issues related to the biofeedback motor housing were also identified. Motors occasionally slipped out of their pockets during motion. Additional

observations included: (i) Discomfort when sitting, due to fear of damaging the electronics, especially the powerbank and processing unit placed near the lower back; (ii) Minor skin irritation from the front zipper, particularly for participants not wearing collared clothing; (iii) Sensation of the lower back cable, attributed to the rigidity of the currently used intermediary cables.

On a positive note, sensor fixation was stable across trials. The sensor pockets and the central cable-connector design contributed to maintaining correct orientation and position relative to anatomical landmarks. The newly added waistband further improved fixation of the lower back sensor, overcoming a known limitation of the previous design.

Comparison with Previous Iteration and Commercial Systems

Table 14: Hardware comparison between the previous (prev.) iteration of Ergowear, the current version (curr.), and the XSens MVN Link.

Category		**Prev.**	**Curr.**	**LINK**
Weight (g)	Intermediary Modules	19	15	10
	Extremity Modules	11	12	10
	Master Module	210	189	150
	Power Supply	260	137	N/A
	Suit	356	399	485
	Full System	**1254**	**1130**	**1130**
Dimensions (mm^3)	Intermediary Modules	45×35×19 (29925)	52×22×13 (14872)	36×24.5×10 (8820)
	Extremity Modules	31×35×19 (20615)	41×22×13 (11726)	36×24.5×10 (8820)
	Master Module	95×73×45 (312075)	146×92×37 (496984)	160×72.5×25 (290000)
	Power Supply	100×80×20 (160000)	100×80×20 (160000)	N/A
Current Consumption (mA)		1827.27	570.21	305.26

The current Ergowear system was benchmarked against its previous iteration and against the XSens Link, a wired motion capture solution from the XSens MVN line [24]. A summary of the comparison is reported in Table 14.

The updated Ergowear is 124 g lighter than the previous version, corresponding to a 10% reduction in total system weight. Although the overall mass remains slightly above the 1 kg wearable target, the excess weight is primarily located near the user's center of mass at the hip, where it is evenly distributed. The intermediary sensor modules exhibit a 21% reduction in mass, corresponding to a decrease of 4 g per module. In terms of volume, the new modules are approximately 50% smaller, with a substantial

reduction in height, which is a critical factor for wearable comfort [126]. The XSens Link sensor remains more compact, with a total volume of 8 820 mm^3, compared to 14 872 mm^3 and 11 726 mm^3 for the Ergowear intermediary and extremity modules, respectively.

Power consumption shows a substantial improvement over the previous iteration. Excluding the vibrotactile motors, the current system draws nearly three times less current. Nevertheless, the XSens system achieves a 46% lower current draw, indicating a remaining performance gap. The total weight of the wearable suit increased by 43 g relative to the previous version, but remains approximately 18% lighter than the XSens solution.

In terms of wireless communication, the updated Ergowear consistently outperforms the previous iteration. Both systems exhibit zero packet corruption and comparable latency, while the new version demonstrates lower packet loss and a more stable sampling frequency throughout operation.

The total assembly cost of the updated Ergowear, excluding the custom-made suit and the powerbank, is approximately 400 € (2021 values). This cost is well below the target suggested by [127] and substantially lower than commercial alternatives. For reference (2021 values), the Perception Neuron system starts at 2 399 €, while high-end solutions such as AiQ Synertial reach costs of up to 10 650 €.

3.2.6 Discussion

System Timings

The timing analysis validates that the system successfully meets real-time constraints at the adjusted 240 Hz sampling rate. This validates the feasibility of meeting real-time constraints at the system's target frequency, supporting the feasibility of real-time motion capture applications. The sending function's dominance in the timing budget, accounting for 1.55 ms of the total 1.58 ms sending block duration, identifies it as the primary bottleneck for future optimization efforts. The necessity to reduce the sampling rate from 250 Hz to 240 Hz represents a minor compromise that maintains system functionality while ensuring reliable operation without timing conflicts. This adjustment demonstrates the importance of comprehensive timing validation in real-time embedded systems. The high standby time of the WiFi module (92.56%) suggests good power efficiency in the communication subsystem, contributing to the system's extended autonomy.

Wireless Range tests

Comparative analysis with a previous system iteration [107] reveals substantial improvement in communication stability. The earlier version exhibited packet loss rates consistently above 1.5%, reaching 3.4% at just 4 m. In contrast, the current system showed no packet loss below 4 m and only 0.65% at 6 m. Both systems reported zero data corruption. These improvements demonstrate a clear enhancement in wireless performance, with the new version addressing a key shortcoming of the older system, which previously failed to meet the required packet loss threshold [107]. Although latency cannot be directly compared between the two systems due to differing measurement methodologies, the current implementation showed consistent latency estimates across all distances. Ideally, latency should be measured with a millisecond-resolution real-time clock synchronized to global time. While this was feasible on the main station, integration into the STM processing unit was not viable due to limitations in the STM32 real-time clock libraries. In conclusion, the system maintained stable communication performance across the tested distance range, complying with all defined technical requirements, including sampling frequency, packet loss, and latency.

Wireless Interference tests

These results suggest that, under the conditions in which the tests were conducted and at a separation distance of 0.5m, the system meets the predefined technical requirements. Furthermore, the findings reinforce the conclusion that the wireless communication performance of the new system represents a significant improvement over the previous iteration of the Ergowear.

Autonomy tests

Compared to the previous version, which sustained monitoring mode for only 7 hours and 20 minutes using the same 13,400 mAh power supply, the current iteration shows a substantial improvement in energy efficiency. This reduction in power consumption allows the use of a smaller and lighter power supply, improving wearability and thermal performance.

These results indicate that the current power supply is overdimensioned for typical use. To reduce size and weight, a smaller alternative is proposed: the Bluehive 5300 mAh power bank (88×41×21 mm, 75,768 mm^3, 120 g). Although it does not support full-day operation in feedback mode, it is sufficient for monitoring mode, which reflects typical usage since vibrotactile feedback is only activated when corrective cues are required. If extended operation is needed, a second identical unit can be used. Overall, the results confirm that the system maintains stable wireless communication during prolonged operation, meeting

the defined performance requirements.

Usability Assessment tests

Although industrial validation for an 8-hour work cycle is still required, the initial results support the system's high usability and wearability. Proposed solutions for identified limitations include redesigning the thumb area using a more elastic mesh, similar to that used around the shoulder and elbow joints. While increasing compression or allowing direct contact between motor and skin were considered for motor housing issues, motor heating posed a comfort risk. Suggested solutions include the use of tighter entry channels or Velcro fastenings to secure the motors in place. Suggested improvements include repositioning the electronics higher along the spine to reduce pressure while maintaining balance, and replacing existing cables with more flexible shielded variants (6-wire, 28–32 AWG, ≤4 mm diameter, silicone jacket).

Comparison with Previous Iteration and Commercial Systems

The comparative evaluation highlights that the current Ergowear iteration substantially improves upon the previous prototype while remaining competitive with commercial motion capture systems in terms of cost and system-level integration. The deviation from the 1 kg weight target is primarily attributable to the power supply and processing unit rather than the sensing modules themselves. The selected battery is overdimensioned relative to the system's energy requirements, and its replacement with a smaller unit would be sufficient to meet the defined weight constraint. Similarly, the processing unit is currently implemented using a development board stacked with a custom PCB, an approach that prioritizes flexibility over compactness. Integrating these components into a single dedicated PCB would further reduce both mass and volume.

At the module level, additional reductions in size and weight are feasible. The current enclosures rely on external fasteners, which introduce unnecessary mass and assembly complexity. Snap-fit enclosure designs could eliminate this overhead, while custom-designed sensor PCBs would allow further reductions in module height. As the extremity modules share a standardized form factor, such optimizations would apply uniformly across the system.

However, the centralized processing strategy limits both system scalability and achievable sampling frequency, as all sensor data are transmitted to and processed by a single main controller. To approach sampling frequencies closer to those of commercial systems such as the XSens MVN Link, a viable architectural evolution is the integration of a lightweight microcontroller directly on a custom IMU sensor board, enabling local signal processing while also allowing the use of smaller and more energy-efficient proces-

sors. Another alternative is to migrate from the current multi-bus I^2C configuration to a high-speed serial interface such as SPI.Compared to I^2C, SPI provides higher bandwidth and more deterministic timing by avoiding shared-bus addressing and arbitration mechanisms [128]. At the system level, this can reduce communication overhead and interrupt-handling complexity associated with managing multiple parallel I^2C buses, thereby improving timing determinism and bandwidth availability for high-rate inertial data acquisition. However, this improvement comes at the cost of increased wiring complexity, as SPI typically requires additional chip-select lines per sensor module, which may impact cable routing, connector design, and overall system integration in wearable applications.

Wireless communication reliability is a critical requirement for wearable sensing in industrial environments. In this respect, the current Ergowear version resolves a key limitation of the previous prototype, which failed to meet packet loss requirements [107]. The improved stability in packet delivery and sampling frequency strengthens the system's suitability for continuous, real-time ergonomic monitoring.

Finally, the achieved cost reduction confirms that Ergowear meets its low-cost design objective. The combination of affordability, modularity, and non-proprietary design positions the system as a viable research and deployment platform. Rather than a finalized product, the current implementation represents a robust and extensible foundation for future developments in wearable ergonomic assessment.

3.3 sEMG e-textile based electrodes

Surface electromyography (sEMG) provides critical insights into muscle activity and fatigue [25, 27], affecting 35% of European workers [129] with 22.8% specifically reporting upper body issues [5]. While conventional wet Ag/AgCl electrodes offer high signal quality, their requirement for skin preparation and potential for skin irritation limit their practicality for continuous industrial monitoring [30].

This section explores textile-integrated sEMG electrodes as an alternative solution, investigating commercially available conductive threads and their performance in electromyographic signal acquisition. The content of this subchapter is primarily based on the research article *Integrating sEMG and IMU Sensors in an e-Textile Smart Vest for Forward Posture Monitoring: First Steps*, published in *Sensors* (2024) [130]. It aims to address research question 1 of this thesis: **How do commercially available knitting threads perform in sEMG signal acquisition and electrical signal transmission, and to what extent can a textile-based sEMG electrode effectively capture electromyographic activity?**

3.3.1 Knowledge acquisition: Literature research

sEMG textile-based electrodes

Recent advances in textile-based dry electrodes have highlighted their potential as viable alternatives to conventional gel electrodes for sEMG. Existing studies explore a wide range of material strategies and fabrication techniques aimed at balancing signal quality, comfort, durability, and garment integration [90–92, 131–133]. These approaches can be broadly categorized according to the underlying conductive material: silver-based textiles, conductive polymer coatings, and carbon-based materials.

Among the different material strategies, silver-based textiles are the most widely adopted, primarily due to their high electrical conductivity and mechanical robustness. Within this category, several studies have explored embroidery and knitting techniques using silver-coated yarns [90–92]. Embroidery-based designs enable precise electrode placement and geometrically stable structures, whereas knitted electrodes, particularly three-dimensional architectures [91], improve skin contact through enhanced mechanical compliance and compression. Beyond yarn-based approaches, sheet-type silver–carbon composites further reduce electrical resistance and achieve excellent signal quality [90], although at the cost of breathability. Laminated silver-plated stretch fabrics, as proposed by Ohiri et al. [132], represent a compromise between low impedance, washability, and ease of integration into garments.

Conductive polymer coatings, particularly PEDOT:PSS, represent a softer, biocompatible alternative.

Niijima et al. [133] integrated commercial PEDOT:PSS textiles (hitoe®) into a wearable cap, achieving signal quality comparable to Ag/AgCl electrodes while maintaining comfort over extended use. However, in-house coated textiles using PEDOT:PSS, as tested by Ohiri et al. [132], exhibited degradation with washing, limiting their long-term stability.

Graphene-based textile electrodes have also been explored as an alternative conductive solution. Ozturk et al. [131] investigated dip-coating nylon substrates with reduced graphene oxide to obtain flexible and comfortable electrodes. Although this approach preserves garment softness and conformability, the resulting electrodes exhibit higher sheet resistance and pronounced sensitivity to washing cycles, which limits their long-term electrical stability. As a result, graphene-based textiles are currently better suited for short-term or semi-disposable sEMG applications.

Overall, silver-based materials remain the most robust for repeatable sEMG acquisition. Coated textiles offer promising alternatives when user comfort and integration into soft garments are prioritized, though further work is needed to ensure their durability under real-world conditions.

Regarding electrode design, most electrodes fall within a size range of 1.5 to 2 cm for localized recordings, with circular or rectangular shapes. Sheet-based silver electrodes [90] have the lowest resistance (0.003–0.004 Ω/sq in [90]), granting them high Signal to Noise Ratio (SNR) and low impedance. However, they are more prone to delamination, are less breathable, and flexible.

Knitted and embroidered electrodes offer wearability (breathability and flexibility) but exhibit higher variability in resistance, depending on structural density, stitch type, and yarn coating. For example, stainless steel embroidered electrodes with a thread resistivity of 91.8 Ω/m achieved an optimized electrode resistance of approximately 0.64 Ω with optimized fill density [92]. Kim et al. [90] reported that silver-plated yarns (530Ω/m) used in knitted structures reached electrode resistances of 0.472 Ω/sq, while embroidered versions using the same yarn achieved even lower values, around 0.197 Ω/sq, particularly when using loop-based (moss stitch) configurations that enhanced contact stability. Catarino et al. [91] reported signal quality comparable to Ag/AgCl electrodes using 3D weft-knitted structures made with Elitex® yarn, which exhibited a resistivity of 60Ω/m. By contrast, graphene-coated and PEDOT:PSS-based textiles presented higher material resistances (13.96kΩ/sq and 3Ω/sq, respectively) and exhibited greater sensitivity to wear, deformation, or humidity [131, 132]. Overall, compact electrode dimensions combined with silver-based materials yielded the most electrically stable and mechanically robust designs for sEMG acquisition.

Commercially available silver-based conductive yarns

To support the design of textile-based electrodes, it is important to understand the current commercial landscape of silver-coated conductive yarns. Several suppliers, including Imbut [134], Swicofil [135], Shieldex [136], and AMANN [137], offer yarns specifically developed for integration into wearable systems. Since 28-gauge knitting machines are commonly used to produce tight-fitting, body-conforming garments for wearable applications, a critical factor in yarn selection is thickness (yarn count). Only yarns below 300dtex are compatible with this gauge. Several manufacturers, including Imbut [134], Swicofil [135], Shieldex [136], and AMANN [137], offer silver-coated conductive yarns that meet these requirements.

Table 15 summarizes the main characteristics of the selected commercial yarns, including thickness, electrical resistance, tensile strength, and cost.

Table 15: Overview of commercially available silver-coated conductive yarns

Brand	Model	Yarn count	Electrical resistance	Tensile strength	Price
Imbut	ELITEX SC	220 dtex	70 Ω/m	400 cN	–
Swicofil	Silver	78 dtex	5 Ω/cm, equivalent to 500 Ω/m	–	–
	117/17 HC+B	117 dtex	500 Ω/m	48 cN/tex	–
Shieldex	235/36 HC+B	235 dtex	600 Ω/m	60 cN/tex	375 € per 0.45 kg cone
	117/17 2-ply HC+B	295 dtex	300 Ω/m	44 cN/tex	515 € per 0.45 kg cone
AMANN	Silver-tech+	220 dtex	300 Ω/m	–	–

3.3.2 Solution Description

This study focused on developing a proof of concept for textile-based sEMG electrodes and conductive tracks, with the goal of enhancing the comfort and usability of wearable systems. Various fiber types, track widths, and electrode dimensions were evaluated using textile samples to characterize their electrical properties under mechanical stress and abrasion. The design and validation of the sEMG signal acquisition modules are also presented, together with the testing protocols applied to each component. Additionally, an Android application was developed to wirelessly control the acquisition modules using the BLE 4.2 protocol.

Test Sample Design

To evaluate how structural parameters affect the performance of conductive tracks and surface electrodes, 22 textile samples were produced with varying thicknesses (six and eight courses), orientations

(horizontal and vertical), and knitted structures (weft-knitted using conductive yarns with combinations of normal and floating loops). For surface electrodes, two different surface areas were tested (4 × 2 cm and 2 × 1 cm). Conductive fibers were selected based on electrical resistance, essential for signal performance, and yarn linear density, which determines compatibility with industrial knitting. High yarn counts increase thickness and compromise compatibility with 28-gauge knitting machines commonly used for wearable garments. Therefore, the selection prioritized fibers with the lowest possible electrical resistance and a yarn linear density below 300 dtex. From the ones available, three yarns met these criteria: 117/17 HCB [138] and 235/36 HCB [139] from Shieldex, and Elitex Skin Contact [140] from Imbut. Their main characteristics are summarized in Table 16.

The 117/17 HCB fiber has the lowest yarn count at 142 dtex, making it thinner, more flexible, and easier to knit with industrial looms compared to the 235/36 HCB (295 dtex) and Elitex SC (220 dtex) fibers. In terms of electrical resistance, Elitex SC performs best at 20 Ω/m, while the 117/17 HCB and 235/36 HCB fibers show higher values of 500 Ω/m and 600 Ω/m, respectively.

Table 16: Shieldex 117/17 HCB, Shieldex 235/36 HCB, and Imbut Elitex Skin Contact characteristics.

Parameter	**Shieldex 117/17 HCB**	**Shieldex 235/36 HCB**	**Imbut Elitex Skin Contact**
Electrical Resistivity	500 Ω/m	600 Ω/m	20 Ω/m
Yarn Count Silverized	142 dtex	295 dtex	235 dtex
Elongation	23%	24%	15–25%
Yield	70,000 m/kg	33,500 m/kg	22,000 m/kg
Tenacity	48 cN/tex	60 cN/tex	31.9 cN/tex

The direction in which the conductive track is placed affects how easy it is to knit and how the fabric behaves. Knitted fabrics are made row by row, forming horizontal lines (called courses) and vertical lines (called wales) [141]. A horizontal track follows the natural direction of knitting, so it is easier to produce. A vertical track goes against this direction and requires switching between the conductive yarn and the base yarn more frequently, which makes the process more complex.

To evaluate the influence of design parameters on electrical performance, conductive tracks were produced with two different thicknesses, six and eight courses, corresponding to approximate widths of 1.9mm and 2.5mm, respectively. In addition, two knitted structures were tested, as shown in Figure 23. In



(a)



(b)

Figure 23: Back view of the knitted structures (horizontal tracks): (a) Floating loops - Shieldex. (b) Base textile courses replaced with conductive fiber - Elitex SC.

the first structure, the base textile courses were entirely replaced by conductive fibers, fully integrating the track into the fabric (Figure 23a). The second structure used floating loops, where the conductive fiber was knitted only at specific points, forming internal loops that connected these points (Figure 23b). However, due to time constraints at the production facility and limited availability of Shieldex fiber, Shieldex samples were produced only with the floating loop structure, while Elitex samples were exclusively manufactured using the structure in which the base textile courses were fully replaced by conductive fibers.



(a) (b) (c) (d)

Figure 24: Conductive track test samples using Elitex SC fiber: (a) Horizontal tracks, front view; (b) Horizontal tracks, back view; (c) Vertical tracks, front view; (d) Vertical tracks, back view.

Track orientation was also considered. Due to the limited availability of Shieldex fiber and the higher fiber consumption required for vertical tracks, vertical tracks were made only with Elitex SC, which was available in greater quantity. All tracks were 10cm long. The designed samples are shown in Figure 24. In the vertical tracks (Figure 24 d), excess fiber is visible on the inner side, resulting from frequent switching between the conductive and base fibers. This surplus corresponds to the trimmed ends of the conductive fiber at each switch.

For the textile surface electrodes, two designs were tested to assess the influence of electrode surface area on electrical properties and sEMG signal acquisition. Both designs featured a rectangular shape but differed in size (2 $\times$1cm and 4 $\times$2cm). The floating loop structure was used in both cases, as it enhances contact between the conductive fibers and the skin. The fabricated electrode samples are shown in Figure 25.

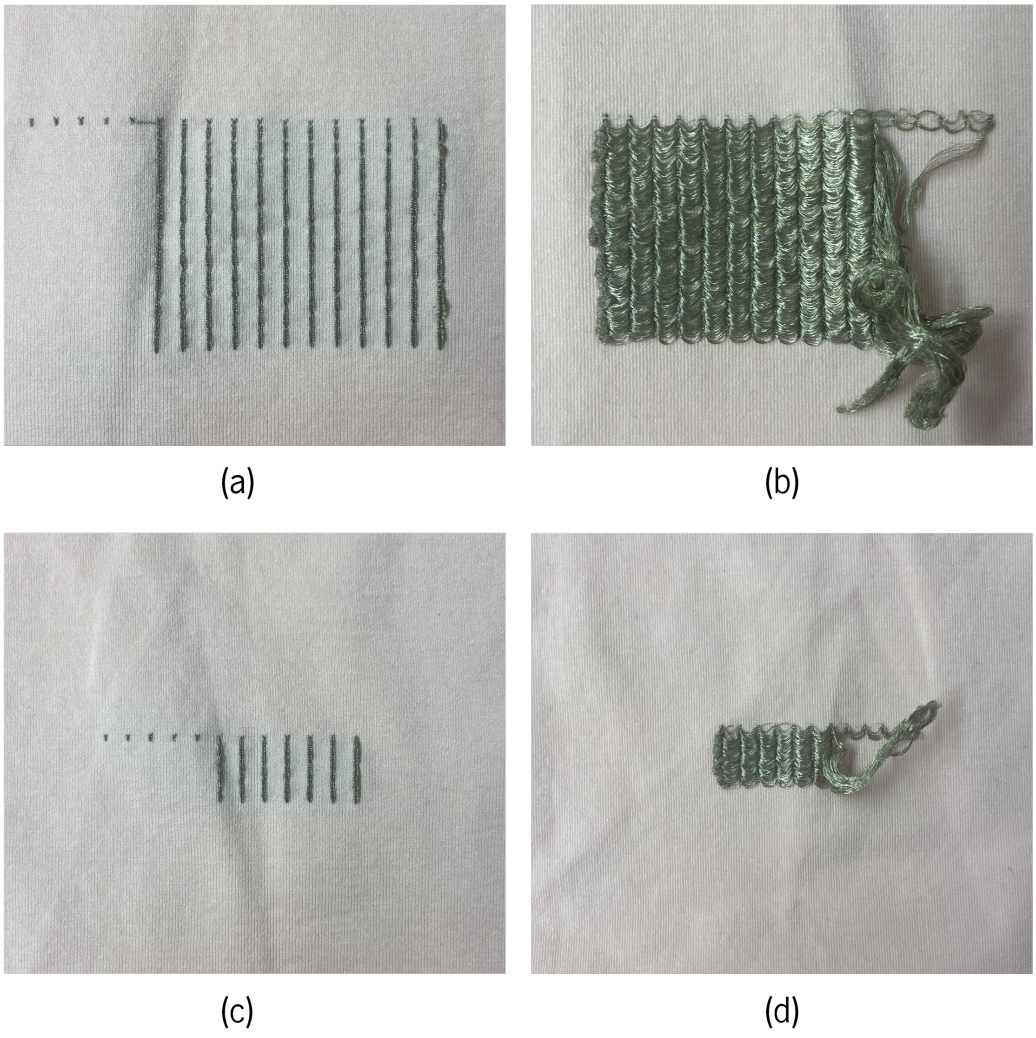


Figure 25: Back view of the knitted structures (horizontal tracks): (a) Floating loops. (b) Base textile courses replaced with conductive fiber.

sEMG Acquisition Module: Hardware Development

The sEMG signal acquisition circuit was developed using the AD8232 chip from Analog Devices [142], a compact and integrated signal conditioning block designed for biopotential measurements, including ECG and sEMG, that allows both amplification and filtering of the signal. AD8232's circuit comprises three main stages, as illustrated in Figure 26. Each of the stages can be configured according to the application requirements.

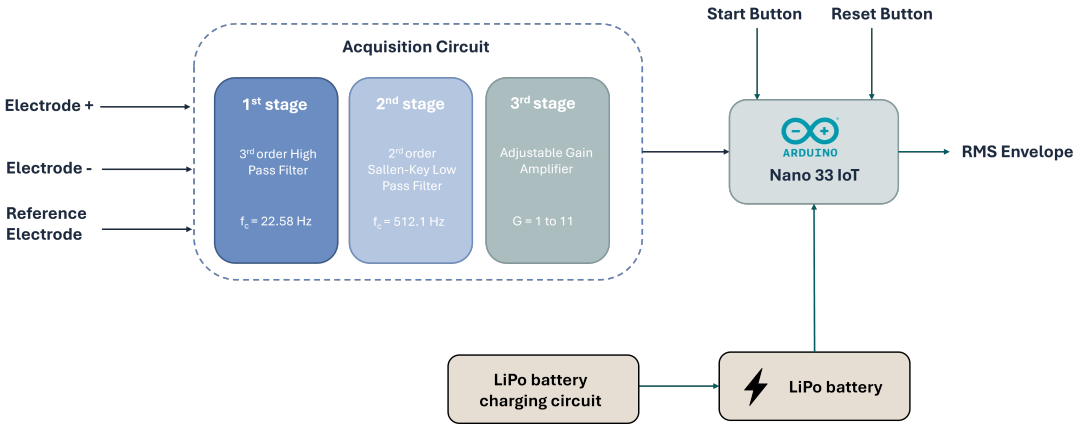


Figure 26: Block diagram of the sEMG acquisition system, highlighting a three-stage analog front-end with cutoff frequencies of 22.58 Hz (high-pass) and 512.1 Hz (low-pass), followed by adjustable gain amplification and digital processing via the Arduino Nano 33 IoT.

The first stage of the acquisition circuit consists of a third-order high-pass filter, configured with a cut-off frequency of 22.58Hz. The second stage implements a Sallen–Key low-pass filter that also provides amplification. This stage was designed with a cut-off frequency of 512.1 Hz and an output gain of 100. The resistor and capacitor values used in both stages were chosen not only to define the cut-off points but also to maintain an appropriate quality factor (Q factor), which determines the filter's selectivity and sharpness. According to the datasheet [142], a Q value of 0.7 is recommended to achieve maximum flatness and sharp cut-off. The selected components ratios yielded a Q factor of 0.696. The gain at the cut-off frequencies of the high-pass and low-pass filters is 36 and 37 dB, respectively, which corresponds to an attenuation of 4 and 3 dB. The filter characteristics and gain were defined using the AD8232 Filter Design Tool [142], provided by Analog Devices [142].

The selected frequency band, around 20–500Hz, is consistent with recommendations from SENIAM, which advise using minimal hardware filtering during EMG acquisition, limited to an amplifier bandpass between 10 and 500Hz to ensure anti-aliasing without compromising the integrity of the raw signal [143]. This range encompasses the majority of the EMG signal power spectrum while effectively attenuating

low-frequency motion artifacts and high-frequency noise. Additionally, Delsys [144] suggests that filtering between 20 and 500Hz is effective in improving the signal-to-noise ratio, while still preserving physiologically meaningful data. Although some systems may opt for a stricter upper limit of 400Hz, extending the range to 500Hz provides a design safety margin and accommodates signal variability across subjects and conditions [144].

Figure 27 shows each of the blocks used for signal's filtering and amplification. To ensure that the output signal remains entirely positive and within the input range of the Arduino Nano 33 IoT's Analog to Digital Conversion (ADC), a DC offset of 1.65V is applied to the REFIN pin using a voltage divider with two 1MΩ resistors.

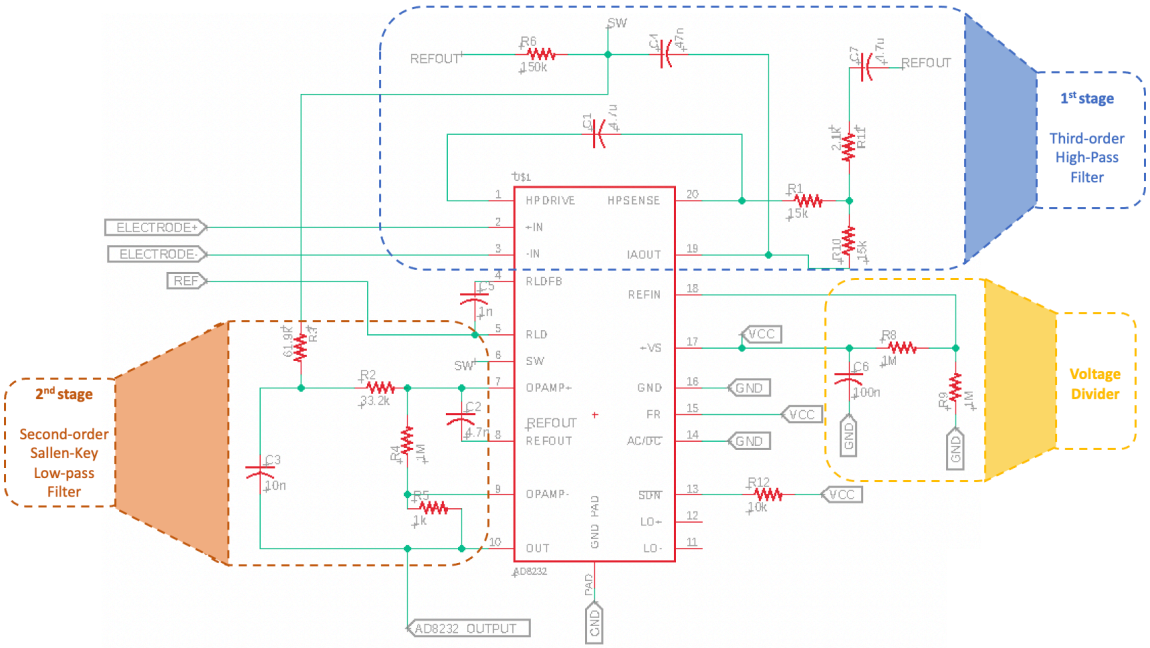


Figure 27: AD8232 acquisition and filtering circuit schematic

To further amplify the signal, a third stage was added, consisting of an non-inverting operational amplifier with adjustable gain connected to the output of the AD8232 chip. The gain of this stage is controlled by an AD5241BRZ1M digital potentiometer [145], integrated into the feedback loop. This potentiometer allows resistance values between 10kΩ and 1MΩ, enabling the amplifier gain to vary between 1 and 11.

To digitize the sEMG signal and manage system control, the circuit was interfaced with an Arduino Nano 33 IoT. This microcontroller performs ADC of the acquired signal and communicates with the digital potentiometer through the I2C protocol. The system also includes two user-accessible buttons: one for resetting the Arduino and another for initiating the calibration sequence and starting the acquisition process.

The Arduino Nano 33 IoT used in the acquisition modules enables on-site data processing and digital transmission to a central processing unit. This architecture was intentionally designed to reflect a future, more comprehensive system that will also integrate IMUs. For this reason, the STM32F405 Feather Express was included as a central processor to simulate the role of a unified CPU managing multiple sensing modalities. By digitizing the sEMG signal locally, this design reduces the computational burden on the central unit and minimizes the risk of signal degradation associated with transmitting analog signals over e-textile conductive paths. While a wireless body area network (WBAN) could alternatively be used for sensor communication, it introduces challenges such as packet loss and synchronization across nodes. These issues, although relevant, are beyond the scope of this study, which focuses on system reliability and robust signal acquisition.

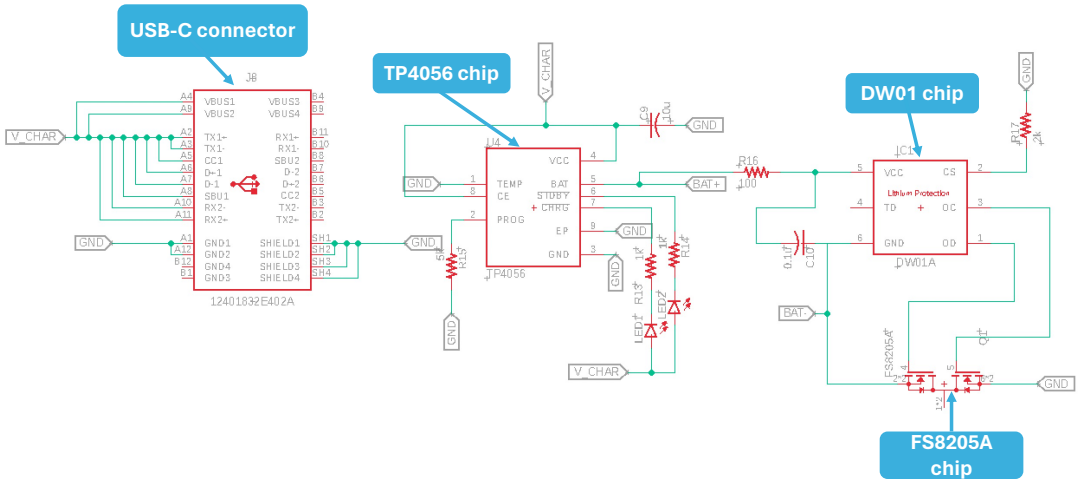


Figure 28: Schematic of the LiPo battery charging circuit

The module is powered by a LiPo battery. To support safe and efficient recharging, a dedicated charging circuit was implemented, which is depicted in figure 28. This circuit uses the TP4056 chip, which provides constant-current/constant-voltage (CC/CV) control. Battery protection is ensured by the DW01 chip, which guards against overcharge, over-discharge, and overcurrent conditions. The DW01 requires dual MOSFETs for charge and discharge switching, which are integrated in the FS8205A package. The charging interface connects via a USB-C port. To electrically isolate the acquisition system during charging, a manual switch was added between the power rail and the acquisition circuit.

Synchronization between the Arduino Nano 33 IoT in the sEMG acquisition module and the STM32F405 Feather Express is achieved by sending a flag via UART from the STM32 to the Arduino, which triggers the initiation of sEMG signal acquisition.

To allow the integration and connection of the sEMG electrodes to the developed PCB, snap fasteners

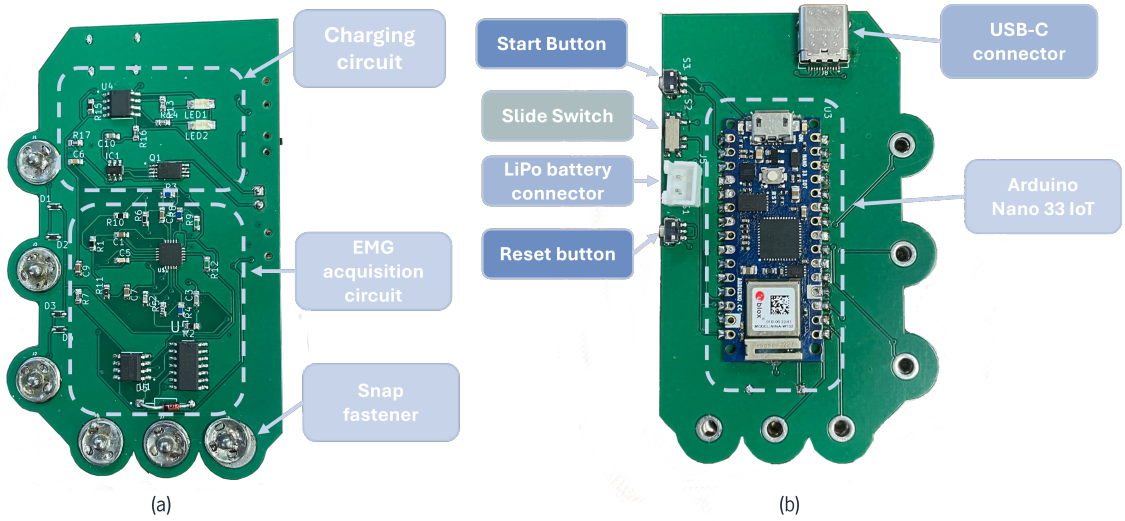


Figure 29: The sEMG acquisition PCB: (a) Front view. (b) Back view.

were used. Figure 29 illustrates the resulting module, which incorporates all the aforementioned circuits.

sEMG Acquisition Module: Software Development

The embedded software running on the Arduino platform is outlined in Figure 30, which illustrates its two main functional blocks.

The embedded software implemented on the Arduino platform is illustrated in Figure 30. The process begins with the Arduino initialization, where the I^2C and UART peripherals are configured to initialize the digital potentiometer and establish communication with the STM board. Bluetooth Low Energy (BLE)is also activated to support wireless communication with the Android application.

The sEMG signals captured by the circuit are influenced by muscle-specific properties, electrode placement, among other, which can introduce variability in the signal's amplitude. Therefore, to allow intra- and inter-subject comparisons [146] of the acquired signals, following initialization, the system enters a calibration phase, triggered either by a hardware button or an initialization flag sent over BLE. During this phase, sEMG signals are acquired at 1200 Hz to record a MVC lasting approximately 3 seconds. The RMS value of this signal's segment is computed, and stored as a normalization reference to enable intra- and inter-subject comparisons [146].

Signal acquisition and filtering are handled within a timer interrupt subroutine. The analog sEMG signal is sampled at 1200 Hz and digitized using the Arduino's 3.3 V reference voltage and 12-bit ADC resolution. The resulting digital signal is then processed using an Infinite Impulse Response (IRR) filter. This type of filter is computationally efficient, requiring fewer coefficients and less memory than a Finite

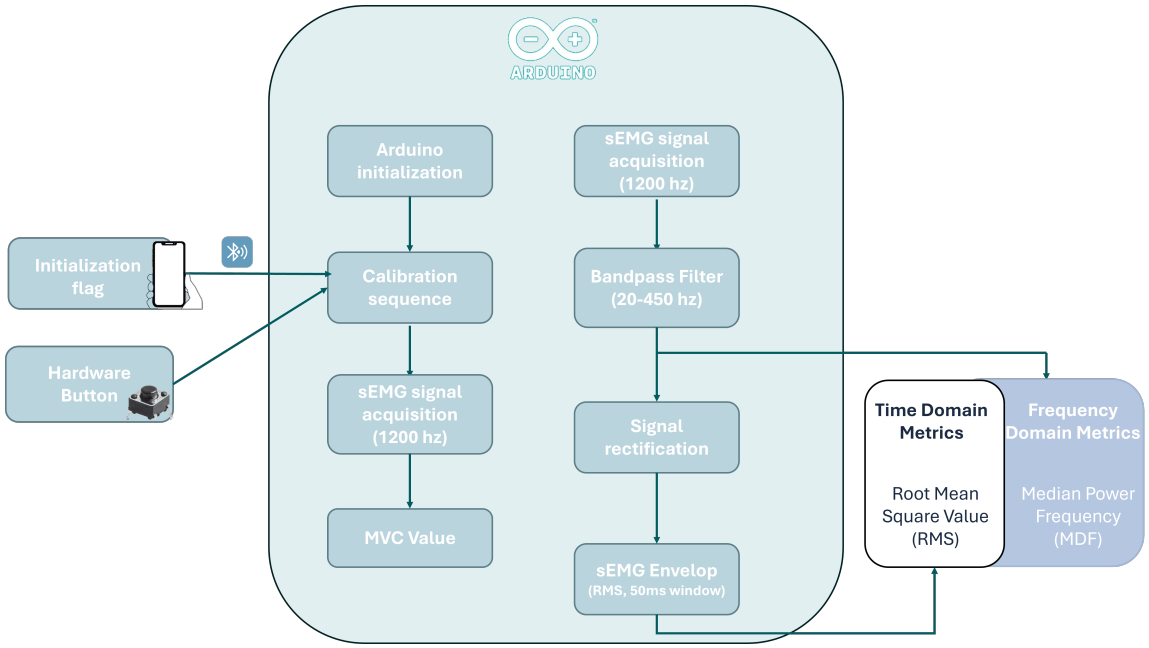


Figure 30: Overview of the embedded software implemented on the Arduino platform for sEMG signal acquisition and processing.

Impulse Response (FIR) filter. The general form of the implemented filter is shown in Equation (3.4).

$$y[n] = \left(\sum_{n=1}^{N-1} b_n x_{n-N} - \sum_{n=1}^{n=N} a_n y_{n-N} \right) \frac{1}{a_0}, \tag{3.4}$$

where a and b represent the filter coefficients, and N is the filter's order. The order chosen for the filter was 12, resulting in 12 coefficients a and 12 coefficients b, which were determined using Scipy's *signal* library. The lower and upper cut-off frequencies were set to 20 Hz and 450 Hz, respectively, in alignment with the specifications of the Delsys Trigno system [147].

The filtered samples are stored in a buffer. After acquiring and filtering 128 samples, the RMS value is determined, resulting in a new value every 107 ms, or a frequency of 9.34 Hz. After calculating three RMS values, these are sent via UART to the STM board, which translates into a sending frequency of 3.13 Hz.

sEMG Acquisition Module: Android Application

To facilitate control and communication with the signal acquisition modules, an Android application was developed that communicates with the Arduino using the BLE communication protocol. Figure 31 presents the developed application, while Figure 140 in appendix B.2 depicts the flowchart of the application. The application allows the user to connect to the acquisition modules, start their calibration sequence, and

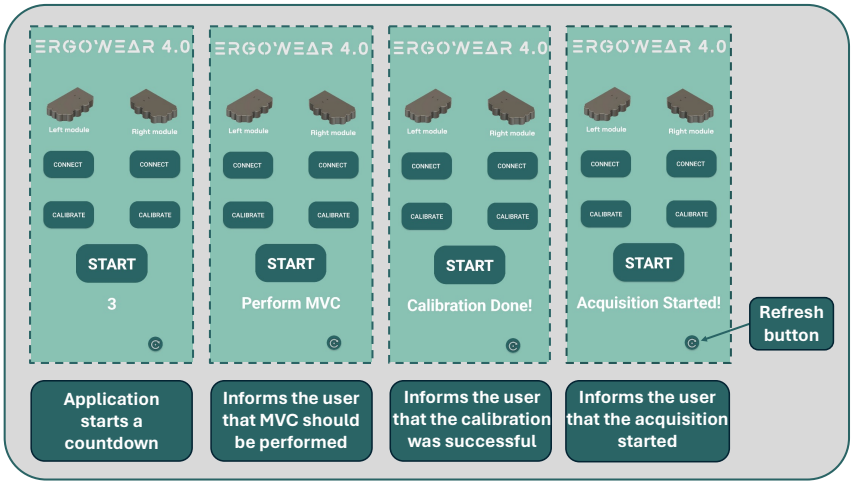


Figure 31: Android application main screens.

begin signal acquisition. The application starts by searching for available BLE devices in the surrounding area. If the acquisition modules are found, the user can start the connection process with the left module. When the connection is established, the left module can be calibrated by pressing the "Calibrate" button. A countdown starts, and at the end of it, a flag is sent to the Arduino of the acquisition circuit to start the calibration sequence. Once the left module has been calibrated, the user can repeat the process for the right module. After calibration of both modules, signal acquisition can be started by pressing the "Start" button. When this button is pressed, a flag is sent to the Arduino to start acquiring the signal.

3.3.3 Experimental Methodology

Sample Characterization: Tensile Strength Tests

Tensile strength tests were performed on the conductive track samples using a dynamometer configured to control the maximum extension, number of cycles, and extension speed. During the extension tests, the sample resistance was measured using an Agilent 34410A digital multimeter [148] with a four-wire configuration. This configuration ensures accurate resistance measurement by connecting the current and voltage sources to the same point on the sample, mitigating contact resistance between the probes and the sample. The testing setup is illustrated in Figure 32.

To determine the maximum length to which the samples should be stretched, a pre-test was conducted. For the samples with floating loop structures, the maximum value corresponded to the point where the textile loops were fully stretched and aligned, at 110mm. This corresponds to a 110% elongation relative to the original sample length (100mm). In samples where the knitted structure completely replaced the base textile courses, the samples were pulled until they increased their length by 45%. To simulate conditions

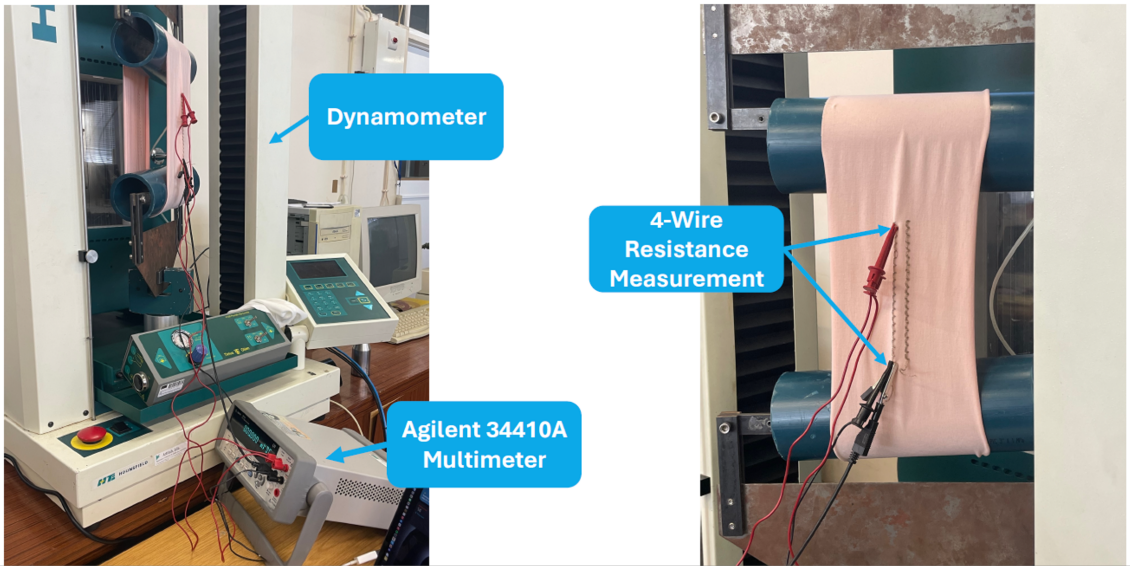


Figure 32: Tensile testing setup: (a) Dynamometer and digital multimeter setup. (b) Four-wire measurement configuration.

similar to dressing the garment, a speed of 200 mm/min was selected. The tests were repeated for 5 cycles to determine the average behavior of each sample.

Sample Characterization: Abrasion Tests

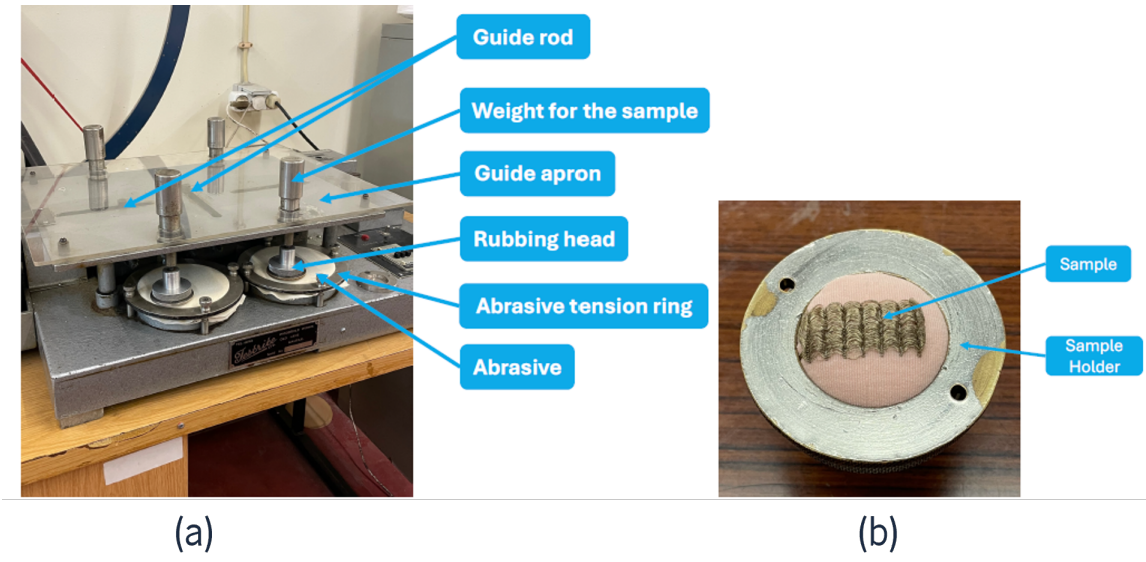


Figure 33: Abrasion test setup: (a) Martindale abrasion tester. (b) Sample placed inside the rubbing head.

Abrasion tests were conducted to evaluate the durability of the metallic coating on the conductive fibers

when exposed to repeated abrasion cycles. The Martindale abrasion tester (Figure 33a) was used for this purpose. The test sample was placed inside the rubbing head (Figure 33b), in contact with the surface of the abrasive, with a weight placed on top of it, allowing the sample to be pressed lightly onto the surface of the abrasive during a cyclic and complex pattern of Lissajous figures [149]. The weight placed on the samples had a mass of 595 g, and the abrasive used was worsted wool.

To measure the electrical resistance of the samples, an Agilent 34410A digital multimeter [148] was used with a four-wire configuration. The resistance was measured for up to 4000 abrasion cycles.

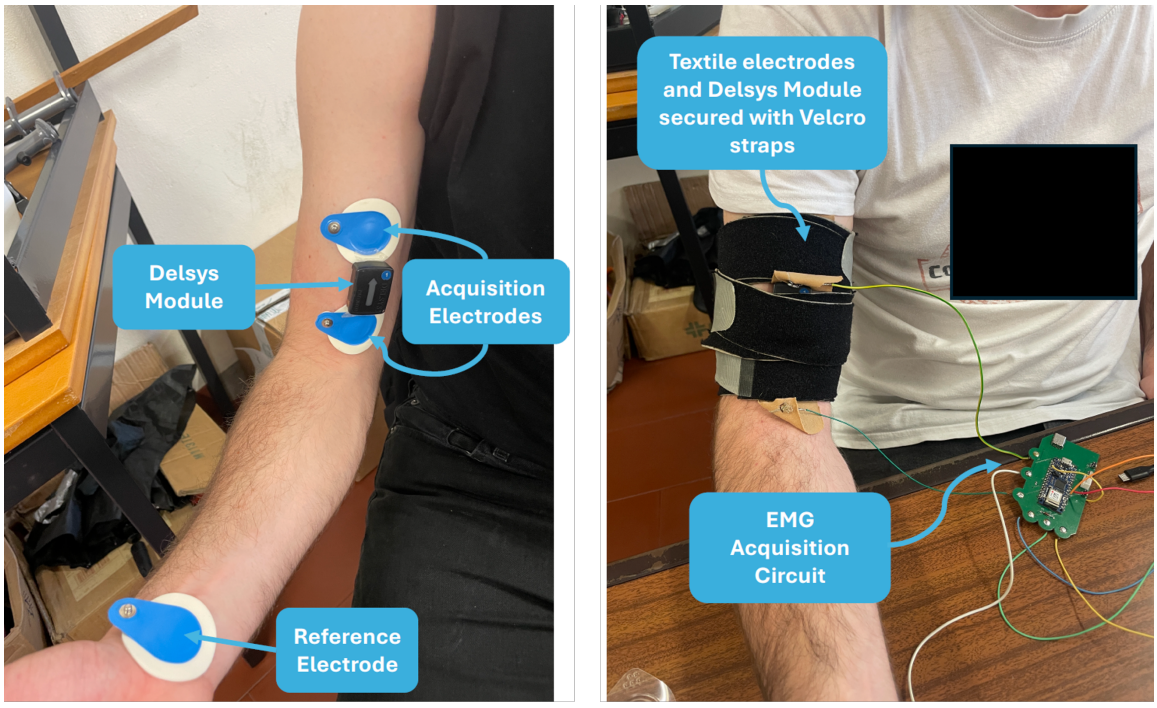
sEMG Signal Quality Validation

The quality of the signal acquired by the acquisition circuit was validated against a commercial, gold standard system: the Delsys Trigno Avanti sensor. The sEMG signal was acquired using two types of electrodes with different purposes: (1) commercial Ag/AgCl electrodes to evaluate the acquisition error of the developed sEMG acquisition circuit and (2) the developed textile electrodes (Elitex SC) to determine the error and assess the quality of the acquired sEMG signal. Data acquisition between systems was synchronized using a hardware trigger system.

To validate the electrodes, 11 healthy participants (4 females and 7 males, 24.8 ± 1.9 years, 66.0 ± 12.9 kg) voluntarily participated in the study. Inclusion criteria required the absence of clinical history or signs of motor injuries, particularly in the shoulder or arm. All participants provided written informed consent.

Data acquisition was conducted inside a laboratory in the School of Engineering of the University of Minho under the ethical procedures of the Ethics Committee in Life and Health Sciences (CEICVS 147/2021) and following the standard set by the Declaration of Helsinki and the Oviedo Convention.

The validation protocol was the same for both types of electrodes, as depicted in Figure 34. Validation experiments were on the Biceps Brachii, a superficial muscle of the arm. This choice was primarily motivated due to its anatomy, which allowed for easy fixation of the textile electrode samples using a Velcro strap. Electrodes were positioned following SENIAM guidelines [150], at one-third of the distance between the acromion and the cubitus fossa. The reference electrode was placed on the wrist. Figure 34 depicts the electrodes' positioning. No skin preparation was done in order to replicate the conditions conceptualized for the wearable, where users are expected to wear the prototype directly on the skin without shaving or cleaning. Both electrode sets were secured using Velcro straps, and data acquisition was synchronized via the Delsys hardware trigger. During the calibration process of the acquisition board, the subject performed the MVC for the biceps branchii [150] for a period of 3 s. Two contraction conditions



(a) (b)

Figure 34: Electrode placement: (a) using Ag/AgCl electrodes. (b) using textile electrodes.

were tested: static, isometric contraction and dynamic contraction consisting of five biceps contractions without any added weight. In static conditions, the subject contracted the biceps until fatigue was self-perceived by the participant.

To allow comparative analysis, sEMG data from the Trigno Avanti sensors were processed offline using EMGworks Analysis software v4.7.3.0 (Natick, MA, USA). Signals were first normalized to the recorded MVC. Subsequently, signal envelopes were extracted via amplitude analysis, corresponding to the RMS computed with a 0.25 s window and no overlap. The processed signals, expressed as muscle activation percentages, were then exported in `.csv` format.

The data from the EMG acquisition board were displayed on a serial monitor for real-time analysis. Although the circuit is capable of operating on a LiPo battery, during the validation experiments it was powered through the Arduino Nano 33 IoT, which received power via USB from the computer. The complete test setup is depicted in Figure 35.

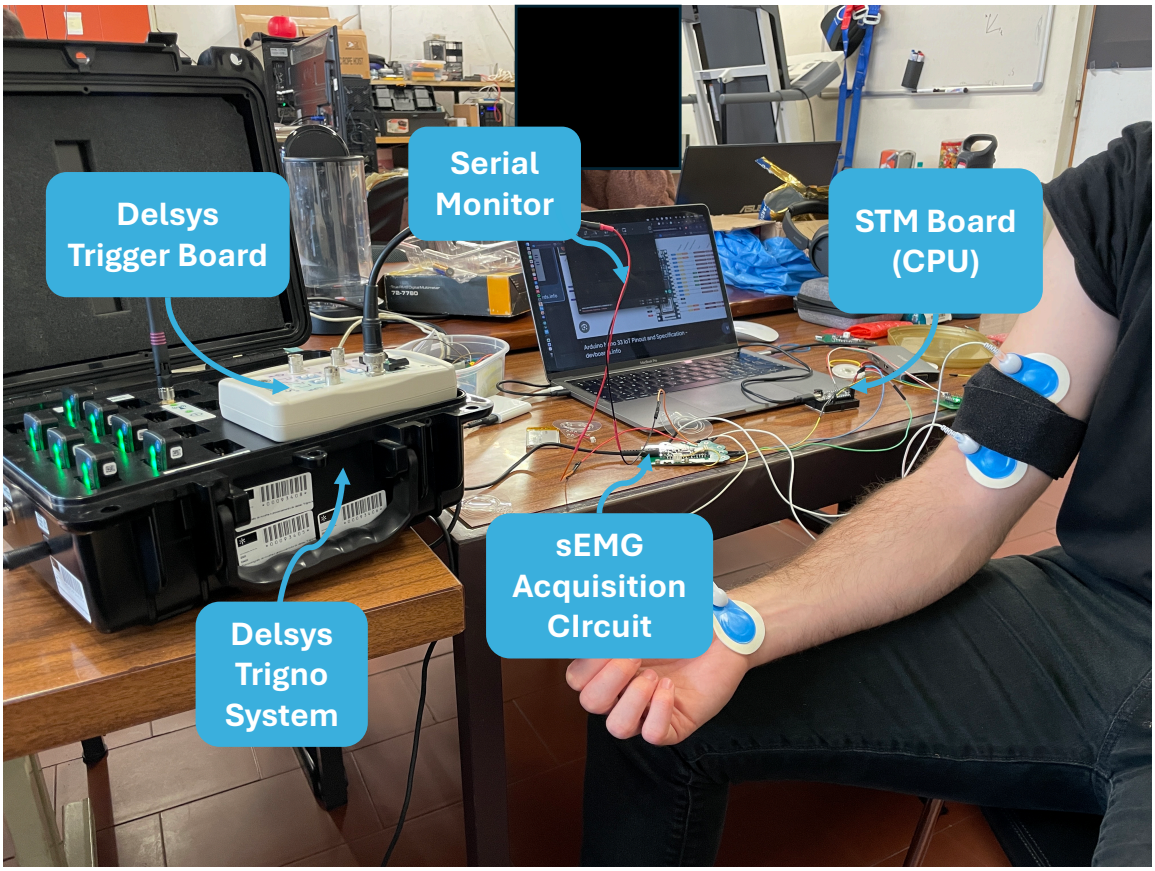


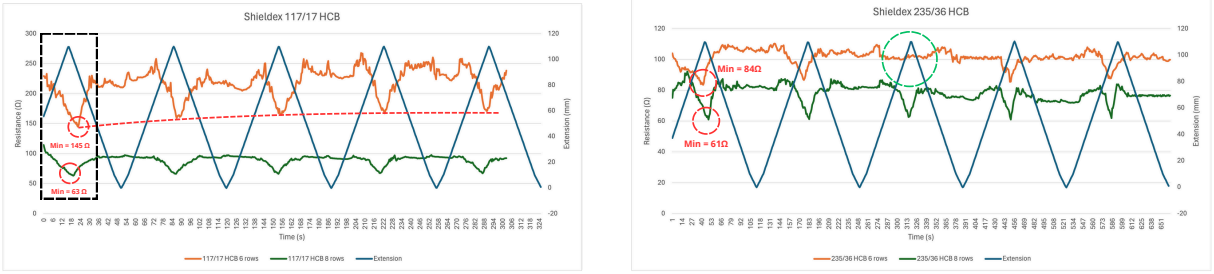
Figure 35: Test setup.

3.3.4 Results

Sample Characterization: Tensile Strength Tests

Figure 36 presents the results of the tensile strength tests for samples with Shieldex fibers and a floating loop structure. As can be observed, the resistance of the Shieldex 117/17 HCB sample with six rows of track thickness (Figure 36a) decreased with the extension of the textile (black dashed rectangle), reaching a minimum of 145 Ω at its maximum extension (100 mm). However, the minimum resistance increased during the test, likely due to the fact that the relaxation time between deformation cycles was not enough to allow the fibers to return to their initial shape. Comparably, the samples with 117/17 HCB and eight-row-thick tracks (Figure 36a) reached a minimum resistance of 63 Ω, with a slight increase observed in resistance during each traction cycle due to fabric deformation. The samples with Shieldex 235/36 HCB and eight-row-thick tracks (Figure 36b) had a minimum resistance of 61 Ω at maximum extension. In contrast to the Shieldex 117/17 HCB samples, there was a noticeable decrease in the fluctuation in minimum resistance between traction cycles. This might be explained due to the fiber's higher tenacity (60 cN/tex for the 235/36 HCB compared to 48 cN/tex for the 117/17 HCB).

Regarding the Shieldex 235/36 HCB samples with six-row-thick tracks, the lowest measured resistance was 84 Ω at maximum extension. In contrast to the eight-row sample, there was a more visible change in the minimum resistance between subsequent tensile cycles, suggesting a greater degree of fabric deformation, as seen in Figure 36a as represented by the red dashed line. During the tensile strength test on the six-row sample produced with Shieldex 235/36 HCB fiber, an error in the digital multimeter led to the absence of the variation in electrical resistance values for the third traction cycle, as marked with the green circle in Figure 36b.



(a) **Shieldex** 117/17 HCB

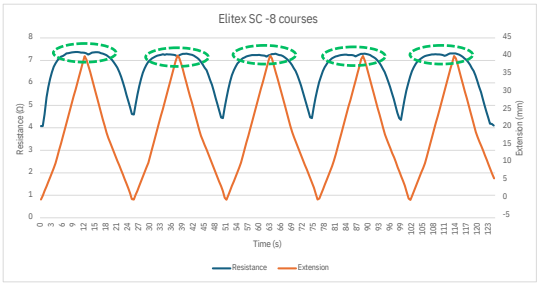
(b) **Shieldex** 235/36 HCB

Figure 36: Variation of the resistance of the samples during the tensile strength tests, using a floating loop structure: (a) Shieldex 117/17 HCB, (b) Shieldex 235/36 HCB.

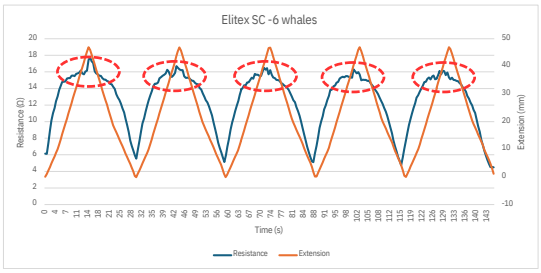
For the samples produced with the Elitex SC fiber and with a knitted structure that replaces the base textile courses with conductive fiber, the achieved results are presented in Figure 37. The tests were conducted until the sample reached 40% of its initial length, i.e., 45 mm. As traction was applied, the resistance increased, peaking at 9.5 Ω for six-row tracks (Figure 37a) and 7.4 Ω for eight-row tracks (Figure 37b). In the six-row vertical samples, for a stretch close to the maximum stretch tested (between 20% and 30%), it can be observed that the behavior is similar to the typical piezoresistive behavior of these structures, with a decrease in the resistance value as the length of the sample increases, as seen in the green highlighted sections in Figure 37a. For the eight-row sample, however, the variation in the resistance value for a stretch close to the maximum tested extension (between 20% and 30%) is practically null, as depicted in the green highlighted sections in Figure 37b. As for the samples with the vertical track, an increase in electrical resistance was observed as sample length increased. Resistance peaked at 45 Ω for the six-row track (Figure 37c) and 16 Ω for the eight-row track (Figure 37d). For extensions close to the maximum extension tested (between 20% and 30%), the variation in resistance with extension takes on more unstable behavior (highlighted in red in Figure 37c,d), which may be due to the excess fiber inside the sample resulting from the production process of the parallel wefts.



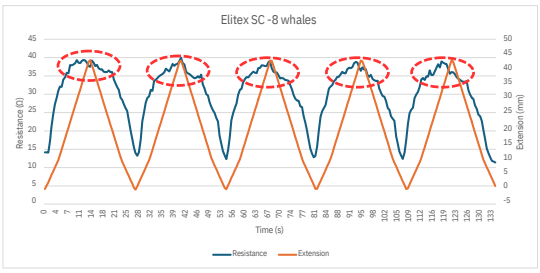
(a) **Elitex SC**, 6-row (horizontal)



(b) **Elitex SC**, 8-row (horizontal)



(c) **Elitex SC**, 6-row (vertical)



(d) **Elitex SC**, 8-row (vertical)

Figure 37: Variation in the resistance of the samples during the tensile strength tests, using Elitex yarns and a knitted structure. (a) Elitex SC with a thickness of 6 rows (horizontal). (b) Elitex with a thickness of 8 rows (horizontal). (c) Elitex SC with a thickness of 6 rows (vertical). (d) Elitex SC with a thickness of 8 rows (vertical). show the typical piezoresistive behavior of horizontal track samples. The red dashed circles show the more fluctuating behavior of the vertical track samples.

Sample Characterization: Martindale Abrasion Tests

Table 17 shows the resistance values measured on the different samples during the Martindale abrasion tests. There were significant changes in all samples with Shieldex fibers. After 1000 abrasion cycles, the resistances in the conductive tracks of both samples produced with Shieldex 117/17 HCB fiber exceeded the multimeter scale value (on the order of GΩ), suggesting substantial damage or possible loss of the metallic coating in some regions, which could have caused a large increase in resistance (highlighted in red in Table 17). For the conductive track samples produced with Shieldex 235/36 HCB fiber, resistance increased, but to a lower degree. For example, resistance in the six-row sample increased from 77 to 333 Ω after the first 1000 cycles and then to 720 Ω after 2000 cycles. However, the resistance exceeded the maximum value measurable by the multimeter after the final trial, showing deterioration of the metallic coating. For the Shieldex 235/36 HCB eight-row sample, the resistance also increased after each test, but the measured values were significantly lower, reaching a peak of 4200 Ω after 3000 cycles. This suggests far less wear of the metallic coating.

Table 17: Average electrical resistance for Martindale's abrasion Test. Shieldex samples were fabricated using a floating loop structure, while Elitex samples were produced with an integrated knitted structure.

Sample	Initial	1000 Cycles	2000 Cycles	3000 Cycles
Shieldex 235/36 HCB 6 rows	77 Ω	333 Ω	720 Ω	Break
Shieldex 235/36 HCB 8 rows	12 Ω	87 Ω	178 Ω	4200 Ω
Shieldex 117/17 HCB 6 rows	78 Ω	**Break**	-	-
Shieldex 117/17 HCB 8 rows	86 Ω	**Break**	-	-
Shieldex Electrode 2 × 1	1.3 Ω	11.7 Ω	29.4 Ω	47 Ω
Shieldex Electrode 4 × 2	**0.97** Ω	**30.4** Ω	**60** Ω	**170** Ω
Elitex SC 6 rows	1.3 Ω	2.1 Ω	3.1 Ω	4.0 Ω
Elitex SC 8 rows	0.9 Ω	1.4 Ω	1.8 Ω	2.2 Ω
Elitex SC Electrode 2 × 1	0.19 Ω	0.6 Ω	1.1 Ω	1.5 Ω
Elitex SC Electrode 4 × 2	0.13 Ω	0.5 Ω	0.8 Ω	1.1 Ω

Electrodes produced with Shieldex 117/17 HCB fiber exhibited substantial increases in electrical resistance after abrasion testing, reaching 3515% for the 2 × 1 electrode and 17425% for the 4 × 2 electrode.

In contrast, samples produced with Elitex SC fiber showed markedly smaller resistance increases under the same conditions. Although resistance rose after successive abrasion cycles, all Elitex SC samples remained below 5 Ω. Specifically, resistance increased by 207% in the six-row conductive track, 144% in the eight-row track (highlighted in green in Table 17), 689% in the 2 × 1 electrode, and 746% in the 4 × 2 electrode. While these increases are non-negligible, the abrasion tests represent extreme conditions. In practical use, textile electrodes are in contact with the user's skin, where abrasive forces are substantially lower than those applied during testing.

sEMG Signal Quality Validation

Since data from Trigno Avanti sensors and the team's sEMG system were acquired at different sampling frequencies (7.36 Hz vs. 9.34 Hz), signal alignment was performed using MATLAB's Dynamic Time Warping (DTW) function [151]. This method enabled peak-to-peak alignment of the RMS envelopes for accurate comparison.

Figure 38a shows the RMS envelopes recorded during dynamic tests for one participant, comparing signals acquired with Ag/AgCl electrodes using the custom EMG circuit against the Delsys reference. Fig-

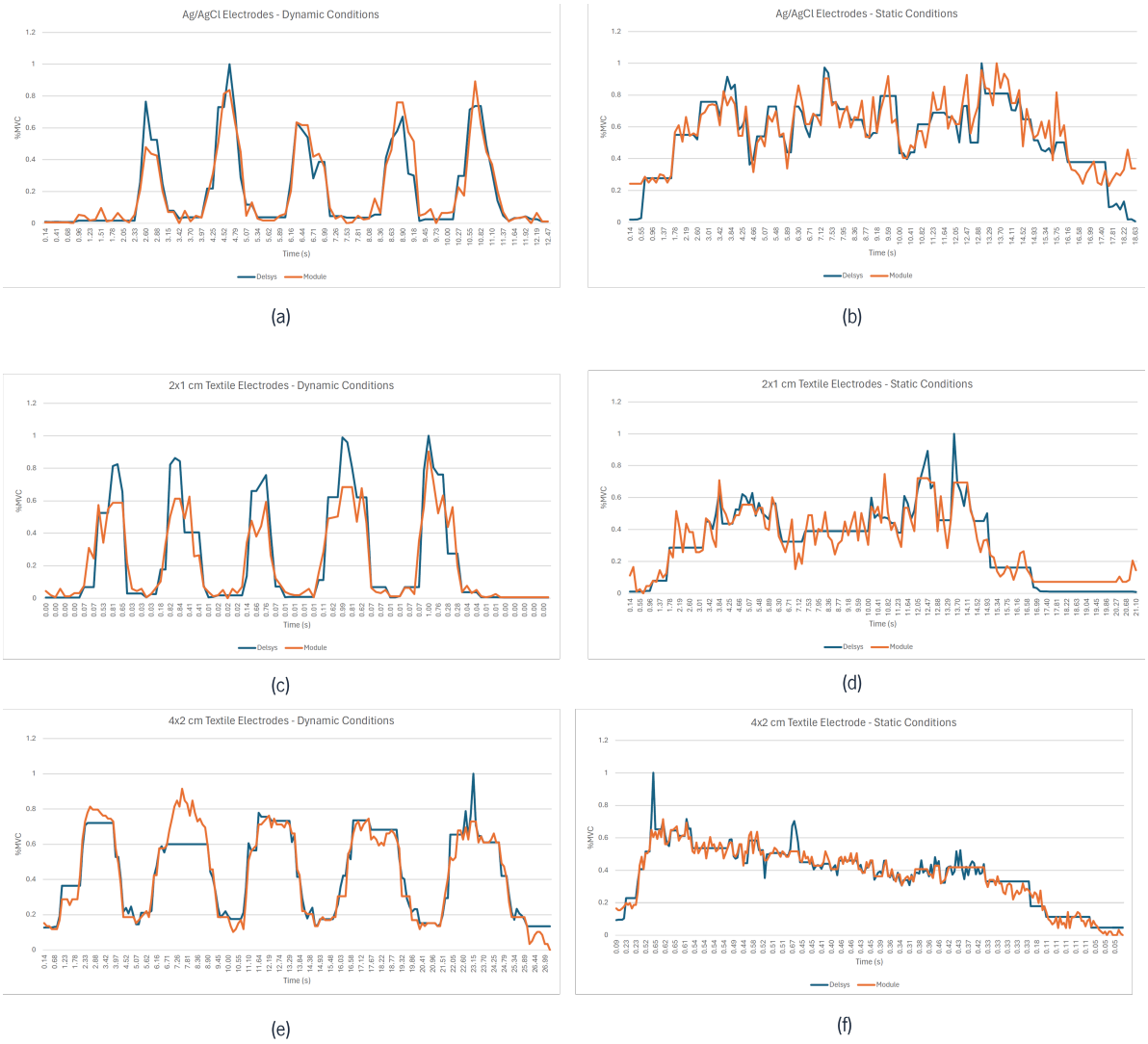


Figure 38: Comparison of sEMG signals acquired with different electrode types and conditions. From left to right: Ag/AgCl electrodes under (a) dynamic and (b) static conditions; 4 $\times$ 1 cm textile electrodes under (c) dynamic and (d) static conditions; and 2 $\times$ 1 cm textile electrodes under (e) dynamic and (f) static conditions.

ure 38b presents the corresponding RMS envelopes for static tests. During dynamic tests across the three electrode configurations (Figures 38a, 38c, and 38e), five distinct peaks are visible in the EMG signals, corresponding to the five voluntary contractions defined by the protocol. The signals acquired with the proposed system exhibit oscillations, likely due to fragile electrode-to-circuit connections and the presence of electromagnetic and electrostatic interference in the acquisition environment. A similar behavior is observed in static tests. As shown in Figures 38b, 38d, and 38f, the RMS envelopes from the custom acquisition circuit follow the behaviour observed in the Delsys reference during isometric contraction of the biceps brachii. However, residual oscillations remain, attributable to the same factors identified in the dynamic tests. Table 18 reports the RMSE values computed between the RMS envelopes of the Delsys reference signal (Trigno Avanti) and those obtained for each electrode configuration, for all participants.

Table 18 presents the RMSEs computed between the RMS envelopes of the Delsys reference signal (Trigno Avanti) and those obtained using each tested electrode configuration.

Table 18: sEMG acquisition RMSE. The values highlighted in green showcase the electrodes with better acquisition performance. The values highlighted in red show the electrodes with the worst performance in terms of sEMG acquisition.

Subject	Ag/AgCl (Dynamic)	Ag/AgCl (Static)	4 × 2 cm Electrode (Dynamic)	4 × 2 cm Electrode (Static)	2 × 1 cm Electrode (Dynamic)	2 × 1 cm Electrode (Static)
ID01	13.21%	9.19%	8.00%	7.13%	18.79%	22.12%
ID02	11.03%	17.88%	7.70%	9.98%	9.24%	10.14%
ID03	6.75%	9.19%	10.68%	9.38%	24.42%	7.83%
ID04	7.88%	10.33%	9.17%	11.53%	9.96%	–
ID05	14.06%	15.42%	11.22%	4.76%	14.68%	13.44%
ID06	13.74%	15.11%	13.64%	19.72%	16.45%	23.47%
ID07	10.85%	14.09%	9.74%	7.20%	27.23%	35.31%
ID08	11.85%	9.18%	7.46%	8.44%	14.33%	30.02%
ID09	7.76%	11.24%	9.19%	12.28%	15.90%	12.90%
ID10	6.60%	9.27%	11.71%	–	17.45%	17.93%
ID11	8.29%	11.39%	12.27%	19.25%	12.17%	8.45%
Avg ± SD	10.18% ± 2.83%	12.03% ± 3.07%	**10.07% ± 2.01%**	**9.97% ± 5.78%**	**16.42% ± 5.54%**	**16.51% ± 10.46%**

From the analysis of Table 18, and starting by the dynamic tests, it is possible to observe that the 4 ×2cm textile electrodes performed comparably to the commercial Ag/AgCl electrodes. Both maintained RMSE values below 14.1%, with average errors of 10.07% and 10.18%, respectively. Their consistency was also similar, reflected in standard deviations of 2.01% for the textile electrodes and 2.83% for the Ag/AgCl electrodes. In contrast, the 2 ×1cm textile electrodes showed higher variability and poorer performance, with an average RMSE of 16.42% (± 5.54%) and peak error reaching 27.23%.

In static conditions, the 4 ×2cm textile electrodes again outperformed the others, yielding the lowest average RMSE of 9.97%. However, their maximum error (19.72%) was higher than that of the Ag/AgCl

electrodes (17.88%), indicating lower consistency. This is also reflected in the higher standard deviation observed for the textile electrodes (5.78% vs. 3.17%).The 2 $\times$1cm electrodes, once again, underperformed on average, exhibiting the highest RMSE in static tests (16.51% $\pm$ 10.64%).

These results confirm that larger textile electrodes (4 $\times$2cm) provide signal quality comparable to standard gel-based electrodes, particularly during dynamic tasks, whereas smaller electrodes exhibit reduced reliability

3.3.5 Discussion

The tensile strength tests highlighted how both fiber composition and knit structure influence the electrical response of the samples under mechanical loading. In the case of Shieldex 117/17 HCB, resistance decreased notably when samples were stretched, dropping to 145 Ω in 6-row samples and to 63 Ω in 8-row samples (Figure 36aa). However, this minimum resistance tended to increase gradually across traction cycles, indicating a progressive decline in conductivity. This trend might be due to mechanical fatigue or structural changes, such as misalignment or distortion of the floating conductive loops, which interfere with current flow.

Samples made from Shieldex 235/36 HCB followed a similar pattern in terms of absolute resistance values, reaching 84 Ω and 61 Ω for 6- and 8-row samples, respectively (Figure 36bb). However, their performance was more consistent across cycles, showing less variation in minimum resistance. This improved stability can be attributed to the higher tenacity of the 235/36 HCB fiber (60 cN/tex), which gives it greater ability to return to its original configuration after each stretch. In comparison, the 117/17 HCB fiber (48 cN/tex) appears more prone to mechanical fatigue. Additionally, in both fiber types, the 8-row samples consistently exhibited lower resistance than the 6-row ones, confirming that increasing the number of conductive courses enhances conductivity.

Elitex SC samples also followed the expected trend of lower resistance in thicker tracks, measuring 9.5 Ω for 6-rows and 7.4 Ω for 8 (Figures 37a and 37b). However, they cannot be directly compared to Shieldex samples due to the differences in their knitted structures. While Shieldex tracks were produced using a floating loop design, the Elitex tracks were fabricated using a fully integrated structure, where the conductive fibers replaced the base textile courses entirely. These differences significantly affect mechanical behavior and current distribution.

Elitex samples, with integrated knitted structures, demonstrated stable and gradual resistance increases during stretching, without the fluctuations seen in Shieldex samples. This controlled response points to greater structural resilience under repeated loading. Furthermore, Elitex samples with horizon-

tally aligned conductive tracks showed low resistance variation under tension. Further, at 45% extension, the resistance is slightly lower, specially for the 6-row tracks, indicating that, in a future prototype, applying a pre-tension to these structures might be beneficial so that their electrical behavior remains in this range. In contrast, vertical track samples exhibited more oscillatory behavior at maximum extension. This instability may be attributed to internal reorganization of excess fiber left over from the knitting process, which affects how the conductive paths respond under strain.

Overall, these results confirm that floating-loop structures, such as those used in Shieldex samples, tend to decrease in resistance under tension due to fiber realignment. Meanwhile, fully integrated structures like those in Elitex exhibit increased resistance when stretched. This underscores the importance of fiber integration strategy, which must be tailored to meet the mechanical and electrical requirements of the target application. This suggests that the structural layout and integration method of the conductive fiber critically influence the piezoresistive behavior, which must be tailored to match the intended mechanical and electrical performance of the system.

The Martindale abrasion tests provided essential insights into the durability of the different fibers under prolonged mechanical stress. Among the tested materials, Shieldex 235/36 HCB demonstrated superior abrasion resistance compared to Shieldex 117/17 HCB, which is consistent with its higher yarn count and greater overall thickness. In both cases, 8-row configurations consistently outperformed 6-row ones, exhibiting lower electrical resistance after abrasion, in line with observations from the tensile strength tests. Although the conductive tracks differ structurally across fiber types, the electrodes were fabricated with a uniform design, enabling a direct comparison of material performance under identical abrasion conditions. The results clearly demonstrate Elitex SC's superior mechanical and electrical stability. For both electrode sizes, resistance remained below 1.5 Ω throughout the test cycles (Table 17), whereas Shieldex electrodes degraded significantly, reaching 42 Ω and 170 Ω depending on the size. This minimal resistance variation in Elitex SC electrodes points to greater durability under repeated mechanical stress. Furthermore, the combination of Elitex SC fibers with an integrated knitted structure resulted in a stable and robust configuration, maintaining electrical resistance below 4 Ω throughout the abrasion cycles.

The validation of the sEMG circuit and textile electrodes revealed key differences in performance across electrode sizes and testing conditions. During dynamic tasks, the 4 $\times$2cm electrodes delivered signal quality on par with standard Ag/AgCl electrodes, with nearly identical average RMSE values (10.07% vs. 10.18%). This performance consistency, confirmed by similar standard deviations, reinforces the suitability of larger textile electrodes for dynamic muscle activity monitoring.

Conversely, the 2 $\times$1cm electrodes showed not only higher RMSE values (16.42%) but also greater

variability between participants, indicating lower reliability. This pattern persisted in static tasks, where the 4 $\times$2cm electrodes again outperformed both Ag/AgCl and smaller textile electrodes. Although they achieved the lowest average RMSE (9.97%), their higher standard deviation (5.78%) suggests individual differences may affect performance consistency.

These results highlight a critical design consideration: surface area has a measurable impact on EMG signal acquisition. Larger electrodes likely benefit from improved skin contact and greater spatial averaging, which enhances signal amplitude and stability. This is consistent with findings from Barrera et al. [152] and Pani et al. [153], who also demonstrated that increasing electrode area improves signal fidelity.

Looking ahead, optimizing electrode size should be a design priority. Incorporating larger textile electrodes can improve both signal quality and robustness, directly reducing RMSE and improving system usability across users and conditions.

3.4 Conclusions

This chapter presented the development of two complementary sensory systems aimed at enabling human awareness in industrial settings through comprehensive physiological monitoring. The work focused on creating the hardware and firmware foundations necessary to capture both biomechanical and physiological data, though the integration of these systems and their processing algorithms remain for future implementation.

The first contribution of this work is the development of an improved IMU-based wearable system capable of monitoring upper body kinematics. Building upon a previous prototype, this iteration achieved significant improvements including a 69% reduction in current consumption (from 1827.27 mA to 570.21 mA), communication reliability (<0.65% packet loss), and professional manufacturing through partnership with a textile company. The system provides raw inertial data at 48 Hz, creating the foundation for joint angle estimation and ergonomic risk assessment. The second contribution involves the validation of e-textile based sEMG electrodes and their acquisition system. The dry electrodes achieved signal quality comparable to gold-standard Ag/AgCl sensors while eliminating skin preparation requirements. A complete acquisition circuit was developed and validated, demonstrating that textile-integrated muscle monitoring is feasible. Both systems were developed as standalone units providing raw sensor data. While not yet integrated, they represent the essential sensing modalities for comprehensive human awareness: motion capture for postural assessment and sEMG for muscle activity monitoring. Together, they form the hardware platform needed to eventually assess both biomechanical loading and muscle fatigue in workplace settings. The practical outcome consists of two validated systems that can be customized and integrated into various research environments without the constraints that come with proprietary commercial solutions.

Future work should focus on several practical improvements and integration challenges:

For the Ergowear system: (i) improving the routing and concealment of vibrotactile motor cables to reduce visibility and enhance aesthetics; (ii) conducting usability studies with factory workers to gather real-world feedback for iterative design improvements; (iii) developing custom IMU boards to reduce module size and cost; (iv) miniaturizing the processing board through reverse engineering of the NUCLEO-H723ZG, creating a custom PCB with only essential components for data acquisition and wireless communication; (v) experimenting with flexible PCBs to better integrate electronics within the garment fabric, potentially eliminating rigid enclosures.

For the sEMG system: (i) miniaturizing the acquisition hardware through reverse engineering of the

Arduino Nano platform, retaining only essential components for data acquisition; (ii) test the electrodes in smaller muscles to identify viable monitoring sites and assess whether electrode dimensions need to be adapted based on muscle size and anatomy; (iii) developing washability protocols to ensure long-term durability.

For system integration: (i) studying garment design requirements to seamlessly integrate both IMU and sEMG systems, including optimal sensor placement, cable routing, and pocket configurations; (ii) building a unified prototype that combines both sensing technologies with shared power and communication infrastructure; (iii) validating the integrated system in actual industrial environments over extended work shifts.

Chapter 4

Computational Methods for Human Motion and Ergonomics Analysis

4.1 Introduction Insight

While the previous chapter focused on hardware and firmware aspects, namely the design and validation of wearable sensing technologies, this chapter shifts the emphasis to algorithmic development. Its goal is to extract meaningful ergonomic insight from inertial data by transforming raw measurements into interpretable representations of human posture and movement.

The chapter follows a structured progression. First, it investigates data-driven approaches for joint angle estimation, aiming to infer human kinematics directly from inertial data while reducing reliance on complex biomechanical models and manual calibration procedures. Second, it proposes and evaluates learning based models for human activity recognition, with a particular focus on posture/activity recognition, which is essential for contextualizing joint kinematics within work related tasks. Finally, it combines joint angle information and recognized posture classes to derive higher level ergonomic indicators, enabling the generation of holistic reports that quantify postural exposure and ergonomic risk over time.

Together, these contributions demonstrate how inertial sensing, when coupled with appropriate computational methods, can support comprehensive analysis of human motion and ergonomics, bridging the gap between wearable sensing technologies and practical ergonomic assessment. Rather than relying on data acquired with the Ergowear prototype, a commercial inertial sensing system (Xsens MTw Awinda) was used to decouple algorithmic development from sensing quality, allowing the proposed methods to be developed and validated independently of hardware specific constraints.

4.2 Deep-learning based Pose Estimation

This subchapter is primarily based on two journal articles: “From raw measurements to human pose – a dataset with low-cost and high-end inertial–magnetic sensor data”, published in Scientific Data [154], and “Deep Inertial Pose: A deep learning approach for human pose estimation”, submitted to Engineering Applications of Artificial Intelligence and currently available as a preprint [155]. This work address the research question 2 of this thesis: “To what extent can data driven inertial sensing pipelines replace explicit biomechanical modeling for pose estimation, when compared across sensor quality and modeling strategies?”

As discussed in the previous chapter, several commercial inertial MoCap systems are available. Among them, Movella Inc. (Xsens Link and Awinda) is one of the few providers offering inertial systems validated for biomechanical analysis [156]. However, adoption remains limited due to high costs, driven by both hardware expenses and proprietary software licenses required for advanced biomechanical analysis. Although raw data can be accessed without these tools, its interpretation requires substantial technical and mathematical expertise.

This work addresses this limitation by investigating the integration of deep learning-based pose estimation into wearable inertial systems, aiming to enable low-cost and versatile solutions suitable for real-world deployment.

Human Pose Estimation (HPE) based on inertial sensing relies on measurements of acceleration, angular velocity, and magnetic field to estimate body segment orientations and joint angles. Conventional approaches employ sensor fusion algorithms to compensate for sensor limitations, such as gyroscope drift, accelerometer noise, and magnetometer sensitivity to magnetic disturbances [157]. Common solutions include Kalman Filter–based methods [109, 158–160] and complementary filters, such as the Madgwick [161] and Mahony [162] filters. These approaches require a sensor-to-segment (Sensor to Segment (STS)) calibration step to map sensor orientations to anatomical segments and compute joint kinematics.

Sensor fusion and STS calibration remain major sources of error, as calibration procedures depend on user-executed poses and are sensitive to sensor drift, noise, and misalignment caused by garment motion. These challenges are further amplified by wearability constraints and remain active research topics [109, 163].

Commercial systems such as Xsens MVN Awinda achieve high accuracy through proprietary hardware and software pipelines designed to mitigate these error sources. This includes gyroscope drift compensation via SDI [1], a dedicated Kalman Filter variant [164], and a dynamic STS calibration procedure

integrated within a biomechanical human body model [165, 166]. While effective, this approach increases system cost and setup complexity.

In contrast, this work explores a data-driven pose estimation strategy that avoids explicit biomechanical modeling and analytical fusion pipelines. Using IMUs sensors and deep learning methods, the objective is to learn motion dynamics directly from inertial measurements and assess whether such models can compensate for common error sources, including suboptimal sensor placement and dressing variability. Several neural network architectures are evaluated using both low-cost sensors (MPU9250) and high-end commercial units (MTw Awinda) to examine performance across sensing conditions. If successful, this approach could enable scalable, accessible real-time pose estimation and simplify integration into wearable platforms by reducing reliance on expert-driven motion modeling.

4.2.1 Knowledge acquisition: Literature research

Human Pose Estimation is a vast research field, with extensive literature on the use of DL algorithms to estimate 3D human pose [167, 168]. However, these approaches are predominantly based on images and videos, where the main challenges are related to visual factors such as occlusions, image quality, and noise. These challenges are fundamentally different from those associated with inertial-based pose estimation, which relies on sparse, noisy, and drift-prone sensor signals.

DL methods have also been explored using inertial data, primarily targeting full-body pose reconstruction from a reduced number of sensors. For example, Huang et al. [169] addressed real-time human pose reconstruction during a diverse set of daily motions, including locomotion and full-body activities, using only 6 high-cost Xsens MTw Awinda sensors instead of the full 17-sensor configuration, which was used as ground truth. Their approach relied on a Bidirectional Recurrent Neural Network (biRNN) and achieved reconstruction errors of $15.85° \pm 12.87°$ and $17.54° \pm 6.49°$, depending on the dataset used. Yi et al. [170] improved the method (achieving $12.93° \pm 6.15°$ and 8.85 ± 4.82, respectively, for the same datasets) with a multistage pose estimation and a fusion-based global translation estimation, using both a biRNN and a Recurrent Neural Network (RNN). Working on the same topic and with the same data, Liao et al. [171] proposed a temporal convolutional encoder and human kinematics regression decoder NN that attained an angle error of $13.37° \pm 5.35°$, and $13.76° \pm 5.21°$, respectively.

In the context of gait analysis, sensor reduction has also been investigated using NNs [172, 173]. [172] employed LSTMs, General Regression Neural Networks (GRNNs) and Nonlinear Autoregressive Network with Exogenous Inputss (NARXs) to estimate sagittal lower limbs' joint angles using only one IMU, placed on the foot. The LSTM outperformed the remaining, achieving errors between $1.91°$ and $2.57°$, in comparison

to the GT data. However, the GT of this study was the joint angles output of a Kalman Filter (KF) fed by data from 4 IMUs, placed on the right side at the foot, lower leg, upper leg, and pelvis. Consequently, the reference itself is subject to estimation errors, which may affect the reliability of the reported performance. [173], in turn, presented a LSTMs-based NN designated IMU2Pose, which estimated 17 joint angles from 3 IMUs positioned at both heels and left wrist, obtaining a RMSE lower than 12.28°. This approach was trained using synthetic IMU data, generated from the [174] dataset, whose data was captured using Vicon optical motion capture system.

Despite these advances, no study directly addresses the problem considered in this work. Existing approaches focus on reconstructing full-body pose from sparse inertial inputs or on estimating joint kinematics from reduced sensor configurations. In contrast, they do not investigate the use of NNs for sensor fusion itself, that is, the estimation of joint angles directly from raw inertial measurements by replacing or augmenting classical model-based filtering methods. As a result, their methodologies are not directly comparable with the approach proposed in this thesis. Hybrid strategies combining NNs with classical filtering methods have been explored in other domains. For example, Coskun et al. [175] combined LSTMs with a KF for image-based pose estimation, achieving a 13.8% reduction in joint error compared to standalone methods. This suggests that integrating learning-based and model-based approaches is a promising direction, although it remains largely unexplored for inertial sensor fusion.

4.2.2 Solution Description

Figure 39 illustrates the end-to-end pipeline followed in this work for inertial human pose estimation (HPE), encompassing three main stages: dataset acquisition, data preparation, and model building. Each stage was designed to ensure the development of a robust and generalizable HPE system. First, a custom dataset was acquired under two distinct scenarios to capture diverse magnetic conditions and hardware configurations. Next, the raw data underwent verification and calibration procedures to ensure signal reliability and to align inertial measurements with anatomical segments. Finally, multiple sensor-fusion architectures were explored and evaluated using 3D kinematic outputs, supported by systematic ablation testing to assess model component contributions.

4.2.3 Dataset acquisition

After an extensive review of publicly available datasets, none were found to meet the specific requirements of this study. As a result, a custom dataset was acquired [154] to enable the design, training, and validation of the proposed inertial human pose estimation (Human Pose Estimation (HPE)) methods.

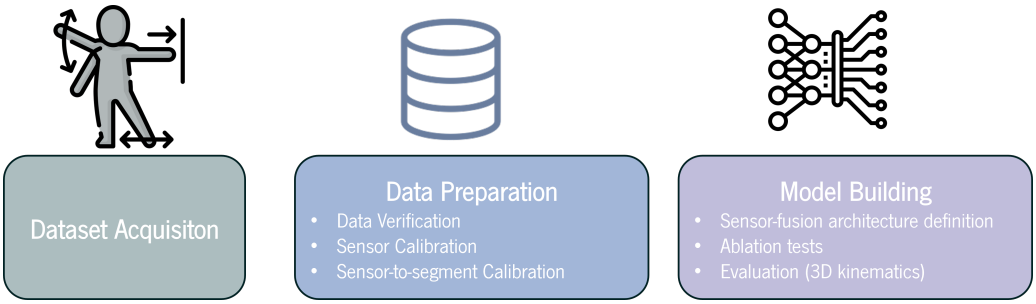


Figure 39: End-to-end steps to build the human pose estimation models

To rigorously evaluate performance and robustness, two distinct data acquisition scenarios were implemented.

The first scenario (D1) used low-cost hardware in an indoor environment characterized by strong magnetic interference—conditions commonly encountered in rehabilitation, office, and industrial settings. In contrast, the second scenario (D2) employed high-quality hardware MTw Awinda (Xsens Technologies B.V., Movella Inc., The Netherlands), operated with the MTManager software in an outdoor setting selected to minimize magnetic disturbances. This configuration was designed to represent a best-case scenario. Despite the differences in hardware and environmental conditions, both acquisition protocols maintained similar movement variability to ensure comparability.

Participants

21 healthy participants (body mass = 66.0 ± 7.8 kg, body height = 171 ± 8.6 cm) and 10 healthy participants (body mass = 64.4 ± 8.5 kg, body height = 171 ±8.1 cm) were recruited for the first and second data collection, respectively. All participants accepted to participate, voluntarily (Tables 51 and 52 in the Appendix) and provided their written and informed consent to participate in the study, according to the ethical conduct defined by the University of Minho Ethics Committee in Life and Health Sciences (CEICVS 006/2020), that follows the standard set by the declaration of Helsinki and the Oviedo Convention.

Instrumentation

Participants in both datasets were asked to wear sneakers and tight-fitting clothing (e.g., jeans, leggings, shirts, or strap tops). For D1, participants were instrumented as follows: i) MTw Awinda full-body motion tracking system (17 IMU sensors on bony landmarks including head, torso, limbs, and feet) following manufacturer guidelines [116]; ii) Ergowear smart garment prototype (v1 version) [107], embedding 9 low-cost IMU sensors (MPU9250) on the upper body. Sensors were secured with duct tape to minimize displacement (Figure 77a)). For D2, instrumentation was similar (Figure 77b)): i) MTw Awinda system as

ground truth, same sensor placements; ii) a second MTw Awinda system, with sensors placed exactly over the first set. Data was acquired using Xsens MtManager.

All sensors were placed by the same researcher to ensure repeatability and minimize misplacement errors.



(a) Ergowear (b) MTwAwinda

Figure 40: (a) Participant instrumented with Xsens MTw Awinda and Ergowear. (b) Participant instrumented with two Xsens MTw Awinda systems on top of each other.

Experimental Protocol

After sensorizing participants, anthropometric data was collected following Xsens guidelines [176] and entered into MVN Analyze to personalize the biomechanical model. Calibration was performed in *N-Pose* and repeated as needed to ensure high quality.

Participants performed six movement sequences (Figure 41), each targeting distinct motion patterns. In D1 and D2, each sequence was repeated 3 and 5 times, respectively. While sequence types remained consistent, individual execution varied, especially in the *Random* trials.

The six sequences were:

1. **Calibration** – Upper-body poses and movements for sensor-segment alignment. Only dynamic calibration was included in D1.

2. **Task** – Simulated factory task involving bimanual reaching with low variability.

3. **Circuit** – Office-like activity including sitting, walking, lifting, and writing, possibly affected by magnetic interference.

4. **Sequence** – Structured motions covering the full range of upper-body joint motion.

5. **Random** – Approximately 1.5 minutes of free, varied motion (e.g., dancing, imitating sports).

6. **Validation** – Isolated maximum ROM movements per upper-body joint, collected only in the Ergowear 1.0 acquisition (D1).

This amounts to 2.5M and 1M samples of data sampled at 60Hz, synchronized with a GT inertial MoCap system. Given the different number of sensors and, therefore, inputs / outputs contained in each dataset (9 vs 17), each one was treated separately to pre-process, train and evaluate solutions, following similar steps (Figure 42).

4.2.4 Experimental Methodology

Pipeline description

To transform raw inertial measurements, from the collected dataset, into full-body pose estimates, a structured multi-step pipeline was implemented, compliant with inertial HPE literature [163, 177, 178] guidelines, to obtain the skeleton pose and compare results with the commercial MoCap systems. This process is illustrated in Figure 42. The pipeline consists of five main steps: **step 1)** sensors' scale/bias calibration **step 2)** sensor-to-segment calibration [108, 178]; **step 3)** data pre-processing; **step 4)** whole body sensor-fusion; Traditionally, this sensor fusion is performed for each sensor individually with classical filters (e.g. Kalman, Madgwick, among others) [179]. Innovation comes from the use of DL methods to replace these traditional approaches for sensor fusion. **Step 5)** is related to the 3D kinematics tool developed to allow easy visualization of the estimated pose.

When using raw IMU data from the higher-end hardware from Xsens (D2), however, the sensor bias calibration step is not required, since the manufacture offers a sensor's self-calibration procedure.

Sensor Fusion

Calibrated IMU data, transformed into segment-aligned reference frames through a simple sensor-to-segment (STS) alignment procedure, was then processed using sensor fusion methods (Step 3 in Figure 42). These methods estimate the orientation of each body segment, represented as unit quaternions. The initial orientation at the first timestep is computed using the TRIAD algorithm [180], which produces a quaternion close to the identity since subjects always begin each trial in a predefined calibration pose used as a reference.



(a) Calibration (b) Task

(c) Sequence (d) Random

(e) Circuit (f) Validation

Figure 41: Frames, from video records captured during data acquisition, illustrative of datasets' movements for each of the 6 types of sequences: Calibration (a), Task (b), Sequence (c), Random (d), Circuit (e) and Validation (f).

Three types of approaches with similar inputs and outputs were explored, as depicted in figure 43.

Classical Fusion-Filter Follows the model-based classical literature [161, 162, 180, 181] and serves as a strong and relevant baseline on which most existing solutions are based [166, 179]. It relies on mathematical analysis of IMU complementary sensor data (i.e. gyroscope to capture high frequency dynamics

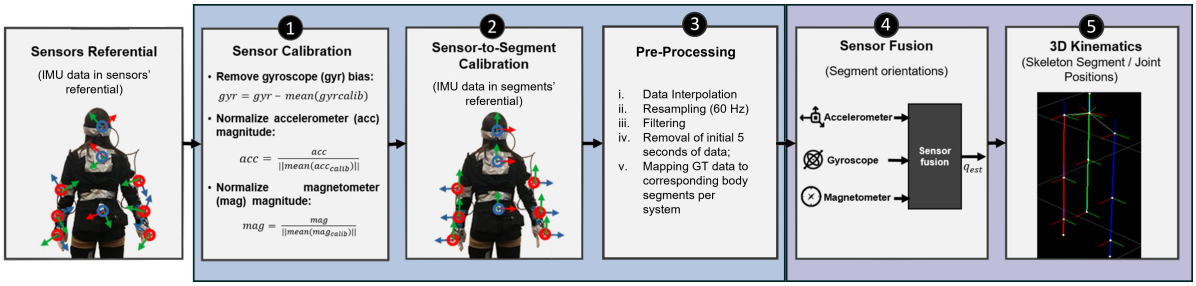


Figure 42: Detailed breakdown of the data preparation and sensor fusion steps

.

and accelerometer/magnetometer to provide global reference) [180]. In simpler complementary methods [161, 162, 180, 182], two different orientation solutions are averaged, by a gain factor: i) the gyroscope data integrated over time (dependent on previous step, making it recurrent), and ii) the accelerometer and magnetometer data using a TRIAD method (not dependent on previous steps).

Neural-Networks Uses model-free NNs to directly learn the mapping from IMU sensor readings, to orientations from the training data (figure 43-2).

This approach makes it straightforward to simultaneously combine data across all sensors, learning an internal latent biomechanical model representation, without explicit mathematical modeling. This combined with the use of information from multiple timesteps can reduce the effects of noise, magnetic disturbances and soft-body artifacts, which traditional systems have trouble dealing with, since these are hard to model explicitly [183].

Hybrid-Filter-Neural-Networks Combines model-based fusion-filters with model-free NNs, with the objective of obtaining advantages from both approaches, namely: relying on model-based priors to improve generalization over the complete input space and facilitate training, while using NNs to learn to address unmodeled system dynamics and perform whole body sensor fusion.

Two hybrid approaches for combining the NNs with filters were explored: **i)** the first (figure 43-3a), abbreviated as Hybrid-Complementary-NN, follows a complementary strategy inspired by the classical complementary filter. Gyroscope measurements are integrated over time to produce the next quaternion, while a global orientation quaternion is independently predicted by the NN using the full set of IMU measurements. The final orientation estimate is obtained by combining these two quaternions through a weighted fusion, where the weights control the relative contribution of the integrated gyroscope data and the network prediction. **ii)** the second approach, abbreviated as Hybrid-CFF-NN, relies on a classical fusion filter to produce an initial quaternion estimate. This estimate, together with the raw IMU measurements,

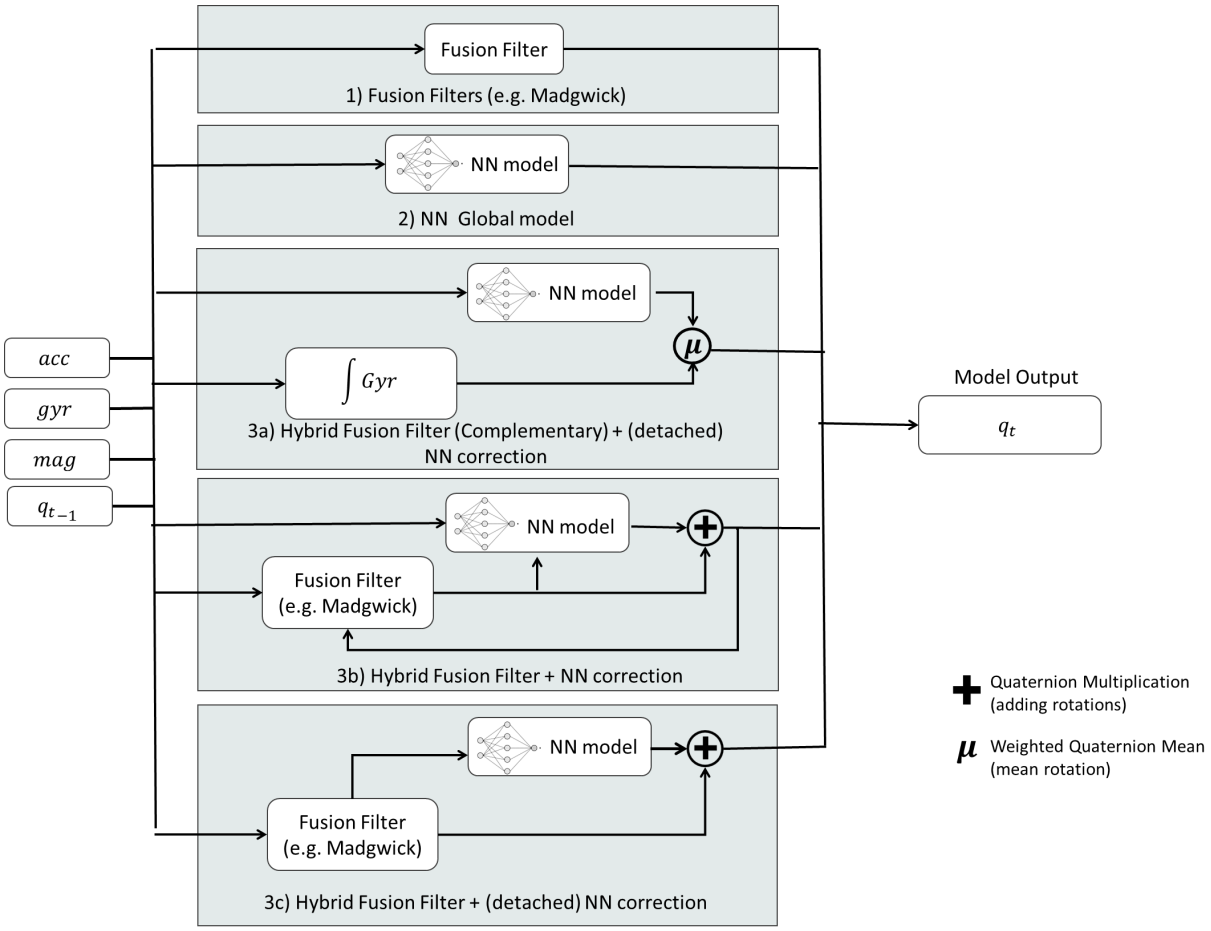


Figure 43: Multiple sensor-fusion architectures: 1) Classical Filter-fusion; 2) Neural-Networks (global); 3a) Neural-Networks (complementary approach, abbreviated as Hybrid-Complementary-NN); 3b and 3c) Hybrid Filter-Neural-Networks, abbreviated as Hybrid-CFF-NN.

is then provided as input to a NN, which outputs a residual quaternion correction. Two variants of this approach were investigated: 1) a feedback-based configuration (Figure 43-3b), in which the quaternion predicted by the NN is fed back into the fusion filter, directly influencing its internal state; 2) a detached configuration (Figure 43-3c), where the fusion filter operates independently of the NN, and the network applies corrections without altering the filter's internal state.

3D kinematics

The estimated segment orientations are used to compute the three-dimensional positions of the body joints through forward kinematics. Each segment orientation is first aligned to the kinematic model's reference frame, then used to rotate a base vector of the segment's length. Joint positions are computed by propagating these rotated vectors along the kinematic chain, starting from the root (pelvis). For Ergowear, the kinematic model comprises nine segments: pelvis, T12, neck, right shoulder, left shoulder, right elbow,

left elbow, right wrist, and left wrist. For the MTw Awinda system, a more detailed model with seventeen segments is used, including the pelvis, sternum, head, upper and lower limbs, and feet. Segment lengths within the kinematic chain are defined using default anthropometric values from [184], and the initial configuration is set according to an N-pose. This procedure yields the complete body pose, including both segment orientations and positions, which can then be used for visualization and evaluation purposes (Figure 42, step 4). It should be noted that the kinematic computations are performed in a global segment reference frame rather than using a conventional parent–child joint hierarchy. This choice simplifies the implementation, as the outputs of the sensor fusion and filtering stages are already expressed in global segment orientations.

4.2.5 Implementation Details

Calibration

The low-cost Ergowear sensors were calibrated (step 1 of figure 42) to remove bias and scale factors over all axis, following standard procedures on IMU sensor literature [185].

Sensor-to-segment calibration (step 2 of figure 42) was then performed, transforming the inertial data from the sensors' referential to each of the corresponding segments' referential. The segments' referential aligns with the world referential in the North West Up (NWU) frame of reference when the user holds the T-Pose. This is consistent with the frame of reference used by the GT's Xsens Analyze software [24], enabling direct comparison of the data.

For the MTw Awinda dataset, an offline STS calibration was performed for each subject using an optimization-based approach [186]. The calibration minimized the Mean Squared Error (MSE) between the inertial sensor measurements (accelerometer and gyroscope) and the corresponding GT trajectories, resulting in relatively small orientation offsets. The optimization was conducted using multiple long-duration trials characterized by slow dynamics and high variability, primarily the sequence and task trials, which yielded lower calibration error than standard calibration procedures. The resulting sensor-to-segment transformation was assumed to remain constant across trials for each subject, as the sensors were firmly secured using straps that provided stable attachment and minimized relative motion.

Unfortunately, this approach did not work reliably for the Ergowear data, due to unpredictable sensor displacement across trials. These displacements resulted from subject movement and from limitations of the smart-garment prototype, which was unable to firmly secure the sensors to the body segments. Consequently, a simpler static calibration method was adopted [108]. The transformation was estimated

using the static N-pose recorded at the beginning of each trial (first 5 seconds). This method assumes an approximately consistent initial sensor placement and corrects sensor orientation in the pitch and roll axes by aligning them with the gravity vector, using Earth's gravitational field as a reference.

Pre-Processing

The datasets were then preprocessed to make the data more suitable, for training and evaluating the models. This included: **i)** interpolation of missing or corrupt data samples; **ii)** resampling of the Ergowear data to a constant 60Hz in order to match the GT system (not required for the MTwAwinda since data was acquired at this frequency); **iii)** filtering out all trials where issues were detected (*e.g.* loose sensors, data desynchronization or corruption); **iv)** removing the initial seconds ($\sim$5s) of each trial, which contained static N-Pose calibration data; **v)** mapping all GT segments' sensor data to the corresponding in each dataset (*i.e* 9 upper-body segments for Ergowear and 17 full-body segments for the MTwAwinda, from the original 23 from the Xsens^{TM} Analyze software); **vi)** normalization of the data to more suitable ranges (accel in m/s^2 was converted to g, magnetometer was normalized by the average magnitude of the measured magnetic field. The gyroscope data (in rad/s) already had appropriate magnitude ranges and was therefore left unchanged, as were the unit quaternions used to represent the GT orientations.

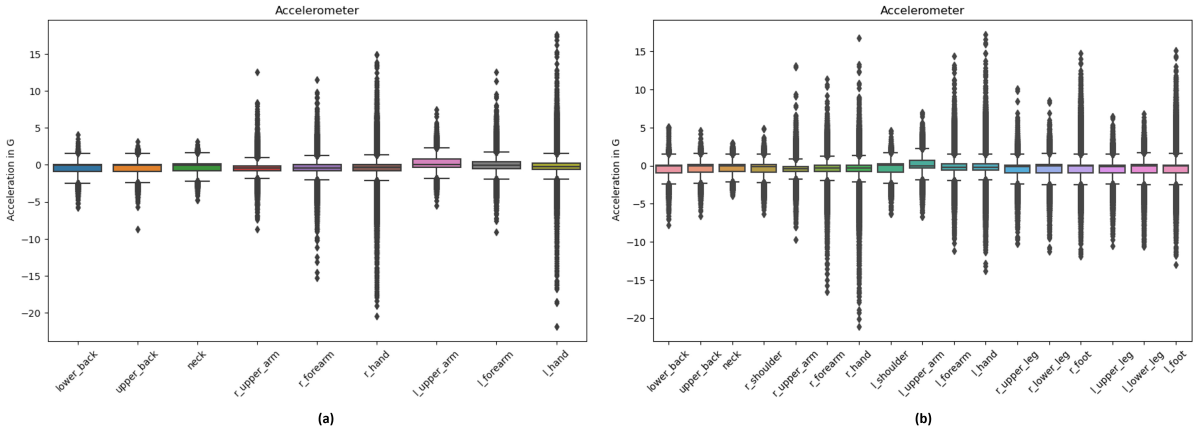


Figure 44: Boxplots of the measured acceleration values per segment: (a) Ergoaware dataset, (b) MtwAwinda dataset.

.

Table 53, in appendix C.2 summarizes the descriptive statistics of the Ergowear and MTwAwinda datasets. With the exception of the MTwAwinda magnetometer data, both datasets exhibit a wide range between minimum and maximum values and predominantly positive kurtosis, indicating the presence of outliers and deviations from a Gaussian distribution. Skewness values are generally close to zero across both datasets, indicating approximately symmetric distributions. A moderate positive skew is only observed

in the MTwAwinda magnetometer signals, suggesting a slight asymmetry toward higher values. This behavior is consistent across corresponding sensor modalities, with accelerometers and gyroscopes showing comparable statistical properties between datasets. The main difference concerns the magnetometer signals. Although their means and standard deviations are similar, Ergowear shows much larger minimum and maximum values, likely due to strong magnetic disturbances in the experimental environment, as described in Section 4.2.3. In contrast, the MTwAwinda magnetometer data show a more constrained range and lower kurtosis. This is relevant because magnetometer disturbances can affect heading information and therefore increase the complexity of the sensor-fusion problem. Figure 44 illustrates the distribution of accelerometer measurements across all sensors for both datasets. Across both datasets, sensor signals are approximately centered around zero. The accelerometer standard deviation is approximately 0.6 G (6 m/s^2), which is consistent with acceleration levels observed in daily human activities such as walking [187]. The remaining sensors are not shown for space and readability reasons. Limb extremities, such as the hands, feet, and forearms, present higher acceleration value ranges, which is consistent with their larger movement amplitudes during manual and whole-body movements.

Data Splits

Each dataset was split by subject into training (70%), validation (20%), and test (10%) sets. The split was kept fixed across all experiments to ensure consistent and fair benchmarking of different models and hyperparameter configurations. The training split only contained *task*, *circuit*, *sequence* and *random* trials, given that the *calibration* and *validation* trials contained little whole-body dynamic movements (these were still used in the validation and test splits). The resulting data is summarized on Table 19.

Table 19: Summary of the train, validation and test data splits for Ergowear and MTwAwinda datasets.

Dataset	Splits	Subjects	Trials	Samples
Ergowear	train	15	184	773.6k
	val	4	24	101.9k
	test	2	51	236.4k
MTwAwinda	train	7	117	480.1k
	val	1	20	73.2k
	test	2	41	188.5k

Sensor Fusion Models: Pipeline Details

All models were implemented and trained using the PyTorch [188] DL framework, with PyTorch Lightning [189] employed to streamline training and evaluation within a Python environment.

Baseline filter-fusion models followed the same processing pipeline as the DL models but did not require training, as they contain no learnable parameters. Their performance was therefore evaluated solely on the test set using the same metrics. By default, all IMU sensor modalities, namely acceleration, angular velocity, and magnetic field data, were used as inputs, and all models predicted segment orientations.

Full motion trajectories were used for both training and evaluation. This choice is necessary due to the recurrent nature of the fusion filters and the DL models, which are based on RNNs and are intended to operate continuously in real-world applications. For the DL models, full trajectories also enabled back-propagation through time during training.

NN models Following existing literature [169, 170, 190], recurrent-based NNs were adopted, with LSTM selected as the baseline architecture. These models were preferred over window-based approaches [69, 191] due to their conceptual alignment with the recursive nature of classical filtering methods [180]. All hidden layers were followed by batch normalization [192] and a ReLU activation function [193], while output layers remained linear. To assess the impact of the recurrent architecture, multiple variants were evaluated, including vanilla RNN, GRU [194], Temporal Graph Convolutional Network (TGCN) [195], and LSTM [196].;

Hyperparameters Reasonable hyper-parameters for training were found empirically and kept constant for all the combinations of models tried. The AdamW [197] optimizer was selected with an initial learning rate of 2e-3 which was decayed to 1e-5 over 25 epochs using a cosine-annealing schedule with warmup [198] with a batch size of 64. Gradient clipping [199] with a range of [-0.2, 0.2] was applied during training for all models. At the end of the training, the best model was selected according to the validation loss.

Data Augmentation and Regularization To increase data variability and improve model generalization, random data augmentation was applied at training time to both the input signals and the corresponding GT.

The applied augmentations included four strategies. Additive noise, either Gaussian or uniform, was applied to the sensor signals with a signal-to-noise ratio (SNR) uniformly sampled from the interval $(0.001, 0.3)$. Sensor dropout was implemented by randomly setting individual sensor signals to near-zero values (1×10^{-8}). Time-step dropout was performed by randomly zeroing out individual time-steps, where each sam-

ple in the sequence had an independent probability of being masked. Small rotational offsets were applied to the sensor data, with perturbations of up to $\pm 18^\circ$ around the X-axis, from -22.5° to $+45^\circ$ around the Y-axis, and up to $\pm 180^\circ$ around the Z-axis. The augmentation probabilities were progressively increased during training. By the final epoch, the probabilities reached 50%, 15%, 15%, and 1% for additive noise, sensor dropout, time-step dropout, and rotational offsets, respectively.

In addition to these data-level augmentations, model-level regularization was applied during training through dropout [200], with a rate of 20% across network layers, and weight decay [201] with a coefficient of 1×10^{-5}.

Loss All DL models were trained using the QAD loss (Eq. 4.1) [202], which directly minimizes the angle distance between two quaternions, while avoiding discontinuities associated with Euler representations and classical distance metrics (e.g. MSE), which, when tried, led to high training instability.

$$\text{QAD} = 2 \arccos\left(|q_{\text{target}} \cdot q_{\text{pred}}|\right), \quad \text{QAD} \in [0, \pi] \tag{4.1}$$

Performance Metrics

All models (classical filters and NNs) were evaluate based on the QAD [202] metric (Eq. 4.1), converted to degrees for easier interpretation. Evaluation was performed in online inference mode, processing one timestep at a time while propagating the hidden state causally, matching real-world deployment conditions.

QAD - The average QAD metric across time and across all sensors was used, in general, to compare results from different methods.

QAD Boxplots - The QAD distribution was plotted for each of the predicted segment orientations and visualized in boxplots, to better analyze the error for each segment individually.

QAD Timeplots - The QAD was plotted over time for each of the predicted segment orientations, giving a better notion of how the error evolves overtime (*e.g.* drift) or in specific complex dynamics.

Inference Time - Inference time was evaluated on an Intel™ i9-10940X (3.30 GHz) CPU. This metric is critical for assessing real-time feasibility and corresponds to the time required to load the inputs and perform a forward pass for a single sample. The reported values correspond to the average inference time per sample, computed over all samples in the test split.

Effect-size analysis

To quantify the practical magnitude of the performance differences between methods, Cliff's delta (δ) was computed over the QAD error distributions. Cliff's delta was used instead of conventional significance testing because the objective was to quantify the magnitude and direction of performance differences, rather than only determine whether two distributions differ. This is particularly relevant here because the comparison is based on millions of timestep- and segment-level QAD samples; under these conditions, conventional significance tests can become overly sensitive, making even negligible numerical differences statistically significant. Cliff's delta provides a more practical comparison by indicating how often one method produces higher errors than another. Given two QAD error distributions, X_1 and X_2, Cliff's delta was computed as

$$\delta = \frac{\#(x_1 > x_2) - \#(x_1 < x_2)}{n_1 n_2}, \tag{4.2}$$

where $x_1 \in X_1$, $x_2 \in X_2$, and n_1 and n_2 are the corresponding sample sizes. Values of δ range from -1 to 1. Positive values indicate that the first distribution tends to contain higher QAD errors, whereas negative values indicate lower QAD errors. Effect-size magnitude was interpreted using commonly used thresholds: $|\delta| < 0.147$ as negligible, $0.147 \leq |\delta| < 0.33$ as small, $0.33 \leq |\delta| < 0.474$ as medium, and $|\delta| \geq 0.474$ as large [203].

Model Variants

Several model variants were evaluated to isolate how specific design choices affect the proposed sensor-fusion pipeline. The corresponding results are presented in Section 4.2.6. The variants explored include:

i) the use of a magnetic disturbance rejection heuristic based on the measured magnetic field magnitude, following the approach proposed by Madgwick *et al.* [182];

ii) sensor fusion performed with and without magnetometer data, to assess robustness under magnetic perturbations;

iii) alternative network output representations, including Euler angles, quaternions, and the 6D rotation representation [204]. Regardless of the predicted representation, all outputs were converted to quaternions during post-processing for loss computation and metric evaluation, ensuring a common representation space for comparison while preserving representation-specific learning characteristics.

iv) training using fixed temporal windows of 1, 2, 5, and 50 timesteps [196], as opposed to full motion trajectories. Although recurrent models trained on full trajectories can be deployed online through stepwise inference with hidden-state propagation, short-window variants were evaluated to assess whether the mod-

els could maintain accurate pose estimation under limited temporal context. Even though window-based training is common in the literature, it was not adopted by default, as it led to increased error accumulation during the integration step of complementary approaches;

v) different loss functions, including standard MSE, MSE computed over the shortest rotation arc (*qdist*), and root-segment-relative QAD (*RelQAD*). While QAD was used as the default loss, MSE was included as a baseline despite operating in parameter space. The *qdist* variant accounts for the shortest rotation between quaternions, and *RelQAD* evaluates errors relative to a reference segment

4.2.6 Results

Models' performance

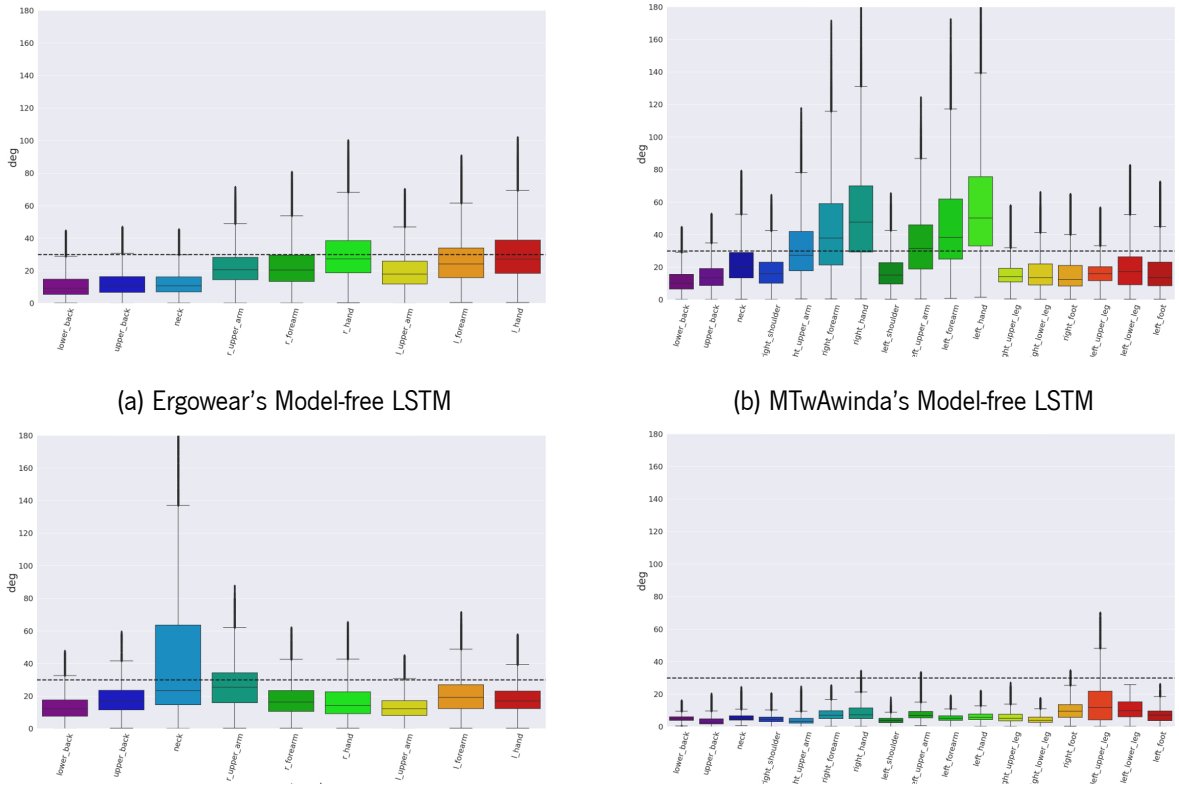
Table 20: Sensor Fusion average results, in QAD°, and inference time (ms) on Ergowear(D1) and MTwAwinda (D2) data.

Experiment	Variant	Ergowear		MTwAwinda	
		QAD°	Inference time (ms)	QAD°	Inference time (ms)
Model-Free-NN	GRU	22.59 ± 15.73	**0.30 ± 0.01**	28.07 ± 23.98	0.33 ± 0.03
	LSTM	21.57 ± 14.97	0.32 ± 0.01	27.97 ± 24.05	0.34 ± 0.02
	TGCN	29.20 ± 21.43	0.50 ± 0.01	37.21 ± 34.13	0.60 ± 0.02
Hybrid-Complementary-NN	GRU	21.48 ± 17.87	0.43 ± 0.01	24.80 ± 23.15	0.47 ± 0.01
	LSTM	21.58 ± 18.89	0.46 ± 0.01	24.30 ± 22.63	0.49 ± 0.02
	TGCN	32.59 ± 23.60	0.63 ± 0.01	38.68 ± 36.13	0.74 ± 0.02
Hybrid-CFF-NN	LSTM-MG-detach	**20.63 ± 16.45**	1.64 ± 0.02	8.08 ± 6.67	1.63 ± 0.03
	LSTM-MG-feedback	diverged	-	diverged	-
	TGCN-MG-detach	21.89 ± 17.00	1.84 ± 0.02	**7.96 ± 6.72**	1.85 ± 0.01
	TGCN-MG-feedback	diverged	-	diverged	-

Experiments with Neural Networks Table 20 depicts the results from the 3 approaches, i.e., model-free NNs, Hybrid-Complementary-NNs, and Hybrid-Classic-Filter-NNs, on both Ergowear and MTwAwinda data. All these models were trained with magnetic data.

The best performance was achieved by the TGCN-MG-feedback, on the MTw Awinda dataset, attaining a QAD error of 7.96° ± 6.72. However, when testing on the Ergowear dataset, the architecture with the best results was the LSTM-MG-detached, with a QAD of 20.63° ± 16.45. Overall, the Hybrid-Classic-Filter-NNs outperformed the model-free-NNs and Hybrid-Complementary-NNs. Notwithstanding, when comparing between datasets, it is possible to observe that the results for Ergowear were more precise (meaning that

result values are closer), ranging from 20.63° ± 16.45 to 32.59° ± 23.60 in contrast to 7.96° ± 6.72 to 38.68° ± 36.13 from MTwAwinda. In specific, the Model-free-NNs performed better on the Ergowear than on MTwAwinda data. This is also supported by the standard deviation which is higher on the MTwAwinda data. Regarding inference time, as expected, it increases with the complexity of the model. Model-free NNs had the lowest inference times (ranging from 0.3 to 0.6 ms), followed closely by the Hybrid-Complementary-NNs. The Hybrid CFF-NNs, due to the added complexity of combining a classic filter with NNs, resulted in longer inference times (1.64 to 1.85 ms) .



(a) Ergowear's Model-free LSTM

(b) MTwAwinda's Model-free LSTM

(c) Ergowear's LSTM-MG-detach

(d) MTwAwinda's LSTM-MG-detach

Figure 45: Boxplots showing the QAD error per segment: (a) Model-free LSTM on the Ergowear test set, (b) Model-free LSTM on the MTwAwinda test set, (c) LSTM-MG-detach on the Ergowear test set, and (d) LSTM-MG-detach on the MTwAwinda test set.

Figure 45 presents the error per segment of model-free LSTM and LSTM-MG-detach, on both Ergowear and MTwAwinda. As observed, the error per segment of the model-free LSTM trained and tested on Ergowear (fig. 45a) is more consistent than the one resulting from MTwAwinda (fig. 45b), meaning that the third quartile, in almost all segments, was bellow the 30° degrees error mark on Ergowear data. The main contributor for the high average error and standard deviation are the upper limbs (i.e., hands,

lower arm, and upper arm) possibly due to the high range of motion of the shoulder joint and consequent movement possibilities for the upper limbs. Also, this high average error is consistent with the pattern observed in the boxplot presented in figure 44a, in terms of acceleration range. In turn, for Ergowear's LSTM-MG-detach (45c), the greatest source of error is the neck segment, while for MTwAwinda's LSTM-MG-detach (45.d) the right hand and the left lower and upper leg were the segments with higher error.

Comparison with classical sensor fusion filters As a baseline for the deep learning–based experiments, the performance of classical sensor fusion filters was first evaluated on both datasets using the QAD metric. Several filter variants were tested to provide a broad reference, namely, Mahony filter without magnetometer data, Madgwick filter with and without magnetometer data, Madgwick filter with magnetometer rejection band, and the Extended Kalman Filter (EKF) with magnetometer data. Table 21 depicts the achieved results.

As expected, lower orientation errors were obtained on the MTw Awinda dataset, with QAD values ranging from 7.76° ± 6.75° to 17.03° ± 17.79°. The lowest error was achieved by the Madgwick filter with magnetometer band rejection, followed closely by the Madgwick filter using magnetometer data (7.87° ± 6.76°). When compared with the NN-based approaches reported in Table 20, none of the learning-based models surpassed the best-performing classical filter. However, two models achieved comparable performance, namely the TGCN-MG-detach (7.96° ± 6.72°) and the LSTM-MG-detach (8.08° ± 6.67°),

Table 21: Average QAD° and inference time (ms) results, on Ergowear and MTwAwinda data, of Classical Sensor Fusion filters and best NN-based sensor fusion method.

Experiment	Ergowear		MTwAwinda	
	QAD°	Inference time (ms)	QAD°	Inference time (ms)
Mahony (w/o mag)	20.73 ± 19.39	0.38 ± 0.01	12.35 ± 14.00	0.38 ± 0.01
Madgwick (w/o mag)	20.71 ± 20.34	0.35 ± 0.01	11.24 ± 13.25	0.36 ± 0.01
Madgwick (w/ mag)	23.23 ± 17.58	1.15 ± 0.03	7.87 ± 6.76	1.16 ± 0.02
Madgwick (w/ mag reject)	22.86 ± 17.50	1.15 ± 0.02	**7.76 ± 6.75**	1.17 ± 0.01
EKF (w/mag)	27.48 ± 20.56	1.09 ± 0.02	10.17 ± 7.11	1.12 ± 0.01
Integral	26.74 ± 27.77	**0.18 ± 0.00**	17.031 ± 17.79	**0.19 ± 0.00**
LSTM-MG-detach (NN-based)	**20.63 ± 16.45**	1.64 ± 0.02	8.08 ± 6.67	1.63 ± 0.03
TGCN-MG-detach (NN-based)	21.89 ± 17.00	1.84 ± 0.02	7.96 ± 6.72	1.16 ± 0.02

For the Ergowear dataset, QAD values were substantially higher, ranging from 20.71° ± 20.34° to 27.48° ± 20.56°. The best and worst classical fusion results were obtained with the Madgwick filter

without magnetometer data and the EKF without magnetometer data, respectively. In this case, the best classical result was slightly outperformed by the LSTM-MG-detach model, which achieved a QAD of $20.63^\circ \pm 16.45^\circ$. Notably, the standard deviation associated with the NN-based models was consistently lower than that of the classical filters, indicating more precise and stable orientation estimates.

A clearer advantage of learning-based methods emerges when considering filters that rely on magnetometer data. For Ergowear, classical fusion filters using magnetometers yielded QAD values between $22.86^\circ \pm 17.50^\circ$ and $27.48^\circ \pm 20.56^\circ$. In contrast, nearly all NN-based models outperformed these classical approaches, with the exception of TGCN-based architectures. Given the strong magnetic disturbances present in the Ergowear environment, this suggests that NN-based fusion strategies may be inherently more robust to magnetic interference.

The inclusion of magnetometer data also affected inference time. For classical fusion filters, adding magnetometer measurements increased inference time from 0.35 ms to 1.15 ms in the case of the Madgwick algorithm. Moreover, comparison between Tables 21 and 20 shows that Hybrid-CFF-NNs exhibit higher inference times than the combined execution times of their individual components, reflecting the additional computational overhead introduced by tightly coupling classical filtering and neural network inference.

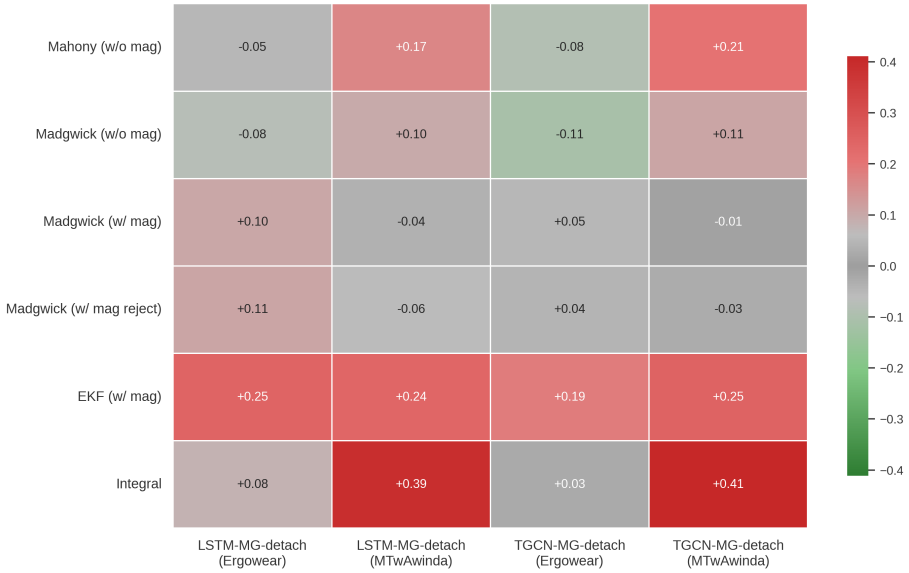


Figure 46: Cliff's δ effect sizes comparing classical filters against NN-based models (negligible: $|\delta| < 0.15$, small: $0.15 \leq |\delta| < 0.33$, medium: $0.33 \leq |\delta| < 0.47$, large: $|\delta| \geq 0.47$). Positive δ indicates classical filters performed worst in relation the NN-based methods.

Figure 46 complements the average QAD values by quantifying the effect-size magnitude of the differences between the classical filters and the best NN-based models. All cases in which the classical filters

showed lower errors than the NN-based models had negligible effect sizes, as observed for Mahony without magnetometer and Madgwick without magnetometer on Ergowear, and for Madgwick with magnetometer and Madgwick with magnetic rejection on MTwAwinda. This indicates that these apparent advantages correspond to highly overlapping error distributions. In contrast, all non-negligible effects were positive, indicating higher errors for the classical filters. Small effects were observed for Mahony without magnetometer on MTwAwinda and for the EKF with magnetometer on both datasets, while medium effects were observed only for the Integral method on MTwAwinda.

Benchmark and Ablation Studies

Table 22: Average QAD° results of the ablation studies on Ergowear data. To ease the results analysis, performance improvements are in bold and underlined, slight performance decreases ($< 1°$) are underlined, and performance decreases are in normal text, all in relation to the baseline results.

Ablation study	LSTM	COMP-LSTM	TGCN	TGCN-COMP	LSTM-MG-detach
Baseline	**21.57 ± 14.97**	**21.58 ± 18.89**	**29.20 ± 21.43**	**32.59 ± 23.60**	**20.63 ± 16.45**
no augmentation	22.20 ± 15.39	21.70 ± 18.17	29.73 ± 22.25	34.85 ± 23.23	**20.54 ± 16.38**
output (6D)	23.04 ± 16.82	23.30 ± 21.44	**23.06 ± 17.17**	**25.18 ± 23.25**	21.21 ± 16.35
no magnetometer	36.97 ± 29.07	33.95 ± 27.79	41.15 ± 31.45	42.41 ± 30.65	-
window (50)	**17.48 ± 11.01**	27.44 ± 29.47	30.08 ± 24.76	46.56 ± 34.47	22.70 ± 16.16
window (5)	46.08 ± 36.07	84.60 ± 43.00	-	-	23.46 ± 17.42
window (2)	69.76 ± 31.47	84.75 ± 47.76	-	-	23.11 ± 16.16
window (1)	109.41 ± 41.29	95.80 ± 43.13	-	-	23.17 ± 17.64
MSE	30.80 ± 25.67	33.46 ± 31.66	-	-	23.53 ± 19.68
qdist	22.12 ± 15.57	**21.08 ± 17.64**	-	-	21.23 ± 16.97
RelQAD	35.28 ± 19.77	**21.37 ± 17.72**	-	-	20.78 ± 16.17

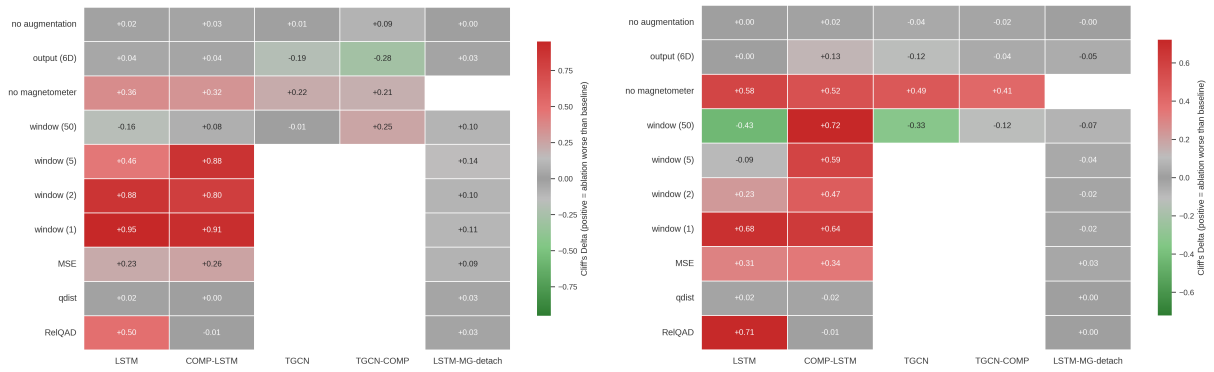
Multiple ablation studies were conducted by removing certain components of the complete pipeline, to identify the contribution of each to the overall results. To ease the analysis, the results were divided by dataset, i.e., Ergowear and MTwAwinda, which are summarized in Table 22 and Table 23. For a question of space and organization, the tables that contain the inference time of these models were added to the appendix (Table 55 and Table 54.The findings are next described.

The Cliff's δ effect-size analysis for the ablation studies is provided in Figure 47. Negative values indicate that the ablation reduced the QAD error relative to the baseline, whereas positive values indicate

Table 23: Average QAD° results of the ablation studies on MTwAwinda data performance improvements are in bold and underlined, slight performance decreases ($< 1°$) are underlined, and performance decreases are in normal text, all in relation to the baseline results.

Ablation study	LSTM	COMP-LSTM	TGCN	TGCN-COMP	LSTM-MG-detach
Baseline	**27.97 ± 24.05**	**24.30 ± 22.63**	**37.21 ± 34.13**	**38.68 ± 36.13**	**8.08 ± 6.67**
no augmentation	**27.65 ± 24.05**	25.08 ± 23.95	**35.20 ± 32.54**	**36.00 ± 32.91**	**8.05 ± 6.85**
output (6D)	29.53 ± 26.90	31.4 ± 30.58	30.23 ± 26.64	**34.73 ± 31.42**	**7.82 ± 6.76**
no magnetometer	62.92 ± 39.69	52.72 ± 37.88	66.87 ± 40.04	65.87 ± 41.60	-
window (50)	**14.29 ± 12.17**	98.18 ± 56.96	**19.56 ± 17.20**	**30.24 ± 25.83**	**7.58 ± 6.72**
window (5)	28.33 ± 30.83	63.00 ± 48.63	-	-	**7.80 ± 6.76**
window (2)	36.26 ± 28.74	47.77 ± 36.35	-	-	**7.87 ± 6.76**
window (1)	69.82 ± 40.09	64.64 ± 40.19	-	-	**7.93 ± 7.92**
MSE	46.53 ± 36.94	41.36 ± 35.59	-	-	8.37 ± 7.00
qdist	28.55 ± 24.24	**23.86 ± 22.61**	-	-	8.15 ± 6.85
RelQAD	77.33 ± 45.35	**24.11 ± 22.40**	-	-	**7.97 ± 6.76**

a worsening effect. Overall, only a limited number of ablations produced non-negligible improvements. In Ergowear, the 6D output representation improved the TGCN-based models with small effects ($\delta = -0.19$ for TGCN and $\delta = -0.28$ for TGCN-COMP), and the 50-sample window improved the model-free LSTM with a small effect ($\delta = -0.16$). In MTwAwinda, the 50-sample window improved the model-free LSTM with a medium effect ($\delta = -0.43$) and the TGCN with a small effect ($\delta = -0.33$). In contrast, the strongest worsening effects were associated with removing the magnetometer and using very short temporal windows. Removing the magnetometer produced small to medium worsening effects in Ergowear ($\delta = 0.21$ to $\delta = 0.36$) and medium to large worsening effects in MTwAwinda ($\delta = 0.41$ to $\delta = 0.58$). Very short windows produced medium to large worsening effects for LSTM and COMP-LSTM, especially with 1-sample windows in Ergowear ($\delta = 0.95$ and $\delta = 0.91$) and MTwAwinda ($\delta = 0.68$ and $\delta = 0.64$). The MSE and RelQAD losses also produced non-negligible worsening effects in specific cases, particularly for the model-free LSTM ($\delta = 0.23$ and $\delta = 0.50$ in Ergowear; $\delta = 0.31$ and $\delta = 0.71$ in MTwAwinda). Most remaining ablations produced negligible effects, including all tested ablations for the LSTM-MG-detach model.



(a) Ergowear

(b) MTwAwinda

Figure 47: Cliff's delta effect-size analysis for the ablation studies. Positive values indicate higher QAD errors for the ablation compared with the corresponding baseline.

Data augmentation and regularization To understand the effects of data augmentation for the NNs training, we conducted an experiment where the methods were trained without any data augmentation and regularization.

Starting by the Ergowear dataset (22), and comparing with the results achieved with the methods trained with data augmentation and regularization (baseline), the removal of these techniques generally led to a slight degradation in performance. Specifically, the average QAD increased between 0.58% and 6.93% for the model-free LSTM and TGCN, as well as for the hybrid-complementary (*COMP*) LSTM and TGCN models. An exception was observed for the LSTM-MG-detach model, for which the average QAD decreased by 0.47%.

For the MTw Awinda dataset (23), the opposite trend was observed. Compared to the baseline, the non-use of augmentation resulted in a improvement in the QAD metric, with a reduction between 0.45% and 6.92%. The exception was the complementary LSTM, where QAD increased 3.22%.

Output Representation Most works use quaternions to represent 3D rotations. However, recently, 6D continuous representations have been proposed suggesting that they outperform the commonly used 4D representation. This ablation test aims to understand if this extra effort, in terms of computation [205] and implementation, is worthwhile.

For Ergowear data, table 22 shows that using a 6D representation only resulted in a performance improvement of the TGCN model, both in model-free and complementary approaches, with a QAD reduction of 21.07% and 22.75%, respectively. In contrast, all remaining architectures did not benefit from the 6D representation, instead exhibiting increased errors, with QAD rising between 2.81% and 7.99%.

For MTwAwinda data, using the 6D output representation resulted in minimal improvement of the LSTM-MG-detach (0.03°, reduction of 3.23 %), and similarly to the ergowear data, the complementary TGCN. All other models showed degraded performance, with QAD increases ranging from 5.56% to 29.35%. Further, as it can be observed in Tables 54 and 55 in Appendix A, the inference time of the models using this output representation increased, being the highest between the ablations tests.

Use of Magnetometer The magnetometer is a sensor typically affected by magnetic interferences, which can increase the pose estimation error. To understand the impact of the use of magnetometer data and if the NNs can handle magnetic interferences, we conducted an ablation study where all the algorithms were trained without using the magnetometer data.

In both datasets, the removal of the magnetometer sensors 's affect all models' performance negatively, with the QAD increasing between 30.12% and 71.38% for ergoware dataset, and between 70.31% and 124.91% for MTwAwinda dataset.

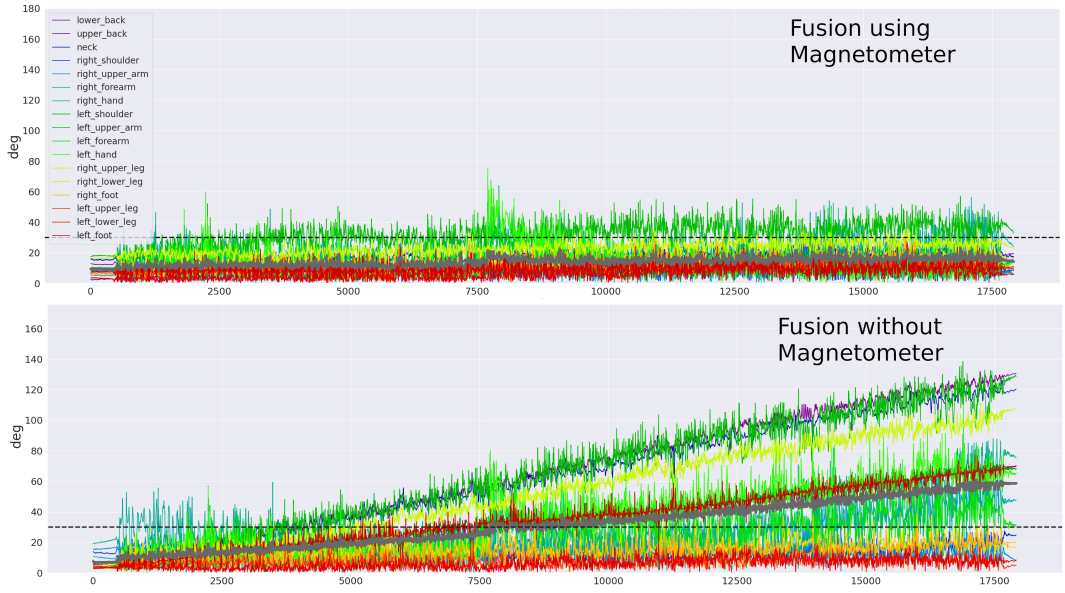


Figure 48: Temporal results on 5min trial with and without magnetometer, for the COMP-LSTM model.

Use of Windows for training To understand the trade-off of using window for training, a experiment was conducted where NNs were trained using windows of 50, 5, 2, and 1 samples.

For the ergowear dataset, table 22 shows that when using windows of 50 samples the LSTM model outperformed considerably the baseline value. Also, when comparing to the classical sensor fusion filters from table 21, it is possible to observe that QAD obtained by the LSTM trained with windows of 50 samples (17.48° ± 11.01°) is smaller than the one obtained by the Madgwick without magnetometer (20.71° ±

20.34°), which was the best method in terms of QAD. However, for the remaining NN models, the effects of these 50 samples windows were not positive. For the smaller window sizes (5, 2, and 1 sample) the resulting QAD value was considerably higher, being the highest achieved by 1 sample windows (109.41° ± 41.29° for LSTM, 407% higher than the baseline). Notwithstanding, the LSTM-madgwick_detach was not so affected by the window size as the other algorithms.

Regarding MTwAwinda dataset, similar to the Ergowear dataset, the use of 50 sample window led to the best results in terms of QAD, in almost all models, to the exception of the complementary LSTM. In fact, the effect was very notorious on the LSTM model, whose QAD lowered from the baseline's 27.97°± 24.05° to 14.29° ± 12.17° (reduction of 48.82%) and on the LSTM-madgwick_detach, which attained the best QAD lowest QAD of all the experiments, with 7.58° ± 6.76°, lower than the QAD achieved with the Madgwick with magnetometer rejection (7.76° ± 6.75°). The use of small windows, with 1,2, and 5 samples, on the other hand, only seemed to benefit the LSTM-madgwick_detach model, with all windows achieving a QAD slightly inferior to the baseline. However, it is possible to observe that smaller the window size, higher the QAD.

Use of different loss functions These experiments were conducted to test the impact of different loss functions, namely MSE, *qdist*, and *RelQAD*, during the NNs training. Overall, no significant QAD improvements were verified.

On Ergowear data, the only cases where QAD decreased was with Complementary LSTM, trained with the *qdist* (21.08° ± 17.64°, reduction of 2.30%) and *RelQAD* (21.37° ± 17.72°, reduction of 0.95%) loss functions, The remaining experiments resulted in higher QADs, with increases between 0.73% and 63.57%. The MSE loss function was the

On the MTwAwinda data, the results pattern was similar, with QAD being decreased when training the Complementary LSTM, with the *qdist* (23.86° ± 22.61°, reduction of 1.81%) and *RelQAD* (24.11° ± 22.40°, reduction of 0.77%) loss function, and, contrary to the Ergowear data, training the LSTM-madgwick_detach with *RelQAD* also resulted in a QAD decrease (7.97° ± 6.76°, reduction of 1.37%). The remaining experiments also resulted in higher QADs, with the increase ranging between 0.79% and 176.42%

Additionally, for both datasets (Tables 22 and 23), the MSE loss consistently resulted in the poorest performance across models. The *RelQAD* loss proved to be ill-suited for the model-free LSTM, leading to degraded accuracy, while the *qdist* loss produced results comparable to those obtained when using the optimizing the QAD loss.

4.2.7 Discussion

This work investigated the applicability of neural network–based methods for inertial human pose estimation and benchmarked them against commonly used classical sensor fusion algorithms using two datasets representative of low-cost (Ergowear) and high-end (Xsens MTw Awinda) sensing systems.

In general, the NN-based models achieved QAD values in the same range as the classical fusion filters, particularly for the hybrid Madgwick-NN variants. On MTwAwinda, the best learning-based models closely matched the best classical filters, whereas on Ergowear they slightly outperformed them. However, this behavior was not consistent across all architectures and conditions.

Data quality and calibration emerged as dominant factors influencing performance. The best results were generally obtained on the MTw Awinda dataset, which was expected given the higher sensor quality and the more reliable sensor-to-segment calibration procedure. In contrast, the Ergowear dataset exhibited substantial and variable sensor-to-segment offsets, which limited achievable accuracy across all methods. This highlights that, for full-body inertial pose estimation, performance depends on the complete processing pipeline, including sensor attachment stability and calibration strategy, rather than on the sensor fusion stage alone.

Regarding the classical filters, these obtained relatively large errors, when compared to values reported in the literature, particularly for the Ergowear dataset. This degradation can be largely attributed to persistent sensor-to-segment offsets and strong magnetic disturbances. Filters operating without magnetometer data achieved lower errors in short trials, especially in magnetically perturbed environments. However, magnetometer data remains necessary to prevent unbounded drift over longer time horizons, as observed in extended trials. The use of a magnetic rejection heuristic improved performance compared to default magnetometer-based filters, consistent with prior findings [182], confirming the presence of magnetic interference during acquisition and the need for mitigation strategies in real-world deployments. NNs demonstrated superior robustness to magnetic disturbances, outperforming classical magnetometer-based filters, especially for the Ergowear dataset. The improvement was lesser in the MTwAwinda dataset, given the lower amount of disturbances, due to the added attention to this issue during acquisition. This interpretation is consistent with the descriptive analysis of the input data, where the Ergowear dataset showed a wider magnetometer range than the MTwAwinda dataset. This difference provides additional context for the lower competitiveness of magnetometer-dependent classical filters on Ergowear, while the more stable magnetometer distribution in MTwAwinda is consistent with the smaller benefit observed for magnetic rejection in that dataset.

Among the evaluated neural network architectures, the LSTM performed best in general, closely fol-

lowed by GRU, while the graph-based TGCN underperformed across most conditions (table 20). Model-free NN approaches were not able to consistently outperform classical fusion filters. On the Ergowear dataset, they achieved slightly better results, suggesting some capacity to compensate for lower data quality and calibration errors. However, on the MTw Awinda dataset, where sensing quality was higher, classical filters remained superior. Withal, the higher error obtained with MTw Awinda (in comparison to ergowear) was not expected, but might be attributed to the higher complexity in fusion the full body sensor data (17 full-body vs 9 upper-body sensors) and lower number of training samples (2.5M vs 1M timesteps).

The complementary NN approach provided modest improvements over the model-free formulation but did not surpass classical fusion filters. Given its higher implementation complexity and training instability, particularly under windowed training, this approach does not appear advantageous in its current form. In contrast, the hybrid Madgwick–NN approach yielded the strongest results among learning-based methods. For both datasets, it achieved performance comparable to the best classical filters and slightly surpassed them on the Ergowear dataset. Nevertheless, absolute orientation errors remained non-negligible, with best-case QAD values of 20.63° for Ergowear and 7.80° for MTw Awinda, indicating that further improvements are required for high-precision applications.

The Cliff's δ analysis shows that the cases where classical filters obtained lower errors than the NN-based models corresponded only to negligible effect sizes. Therefore, these differences have limited practical relevance despite being visible in the average QAD values. In contrast, all non-negligible effects favored the NN-based models, mainly against the EKF and Integral baselines. This supports the conclusion that the best hybrid NN-based models achieved performance comparable to the strongest classical filters, while providing meaningful improvements over less robust classical baselines.

The ablation studies provided additional insights. Data augmentation and regularization did not consistently improve performance, likely due to the large dataset sizes, relatively small network capacities, and simplistic augmentation strategies. Contrary to prior findings [204], the 6D rotation representation did not improve performance over quaternions and increased inference time. Magnetometer data proved critical for learning-based methods, reinforcing its importance for long-term orientation stability.

Window-based training proved problematic for complementary-based models due to drift and the associated error accumulation inherent to orientation integration. When training uses short temporal windows, the accumulated drift within each window remains small and is largely ignored during optimization. Consequently, the model fails to learn long-term correction mechanisms, leading to substantial error accumulation when evaluated over longer sequences at test time.

In contrast, window-based training can be beneficial for non-complementary models, as it enables

greater training parallelization, reduces memory requirements, and often improves convergence. However, window length remains a critical parameter. Very small temporal windows (fewer than five samples) consistently degraded performance across models, indicating that insufficient temporal context limits the learning of meaningful motion dynamics. Finally, loss function choice had a significant impact: standard MSE yielded the worst results, while the *qdist* loss produced performance comparable to direct QAD optimization. Relative root-based losses proved unsuitable for model-free approaches, further indicating their limited ability to integrate orientation over long temporal horizons.

The ablation effect-size analysis clarifies which design choices had practical impact. The only non-negligible improvements were associated with the 6D output representation in some TGCN-based models and with 50-sample windows in selected model-free models. However, these improvements were model-dependent and did not generalize across all architectures. In contrast, removing the magnetometer and using very short temporal windows produced consistent non-negligible worsening effects, confirming the importance of heading-related information and sufficient temporal context for stable orientation estimation. The negligible effects observed for all LSTM-MG-detach ablations suggest that the detached hybrid formulation was comparatively robust to the tested design changes.

Inference times ranged from approximately 0.3 ms for model-free NNs to 2 ms for Hybrid-CFF-NNs on a CPU (Intel^{TM} i9-10940X, 3.30 GHz), with the highest values observed for the LSTM-MG-detach models. These results indicate that deploying full-body fusion algorithms is unlikely to pose computational constraints in most settings, as all components involved, including classical filters and RNNs, support real-time inference and can be efficiently vectorized across multiple sensors. Therefore, future improvements should primarily target robustness to calibration errors, magnetic disturbances, and long-horizon drift rather than execution efficiency.

4.3 Human Activity Recognition

This study is primarly based on the journal paper published in Knowledge-Based Systems journal, entitled "Seeking optimal and explainable deep learning models for inertial-based posture recognition" [206]. HAR has become a valuable tool in ergonomic risk assessment, particularly for linking specific postures to WRMSDs risk. Since each job consists of a sequence of goal-oriented postures, identifying those associated with high-risk tasks enables more targeted interventions. However, manual annotation of posture transitions is impractical in real-world settings. Automated posture recognition using AI provides a viable solution by enabling continuous and unobtrusive monitoring. This study explores the application of these AI models to classify human postures as a basis for ergonomic risk analysis. Grad-CAM will be leveraged to enhance the transparency of the classification process, identifying which sensor signals and time segments most influence the model's decisions. This approach contributes to more interpretable and actionable ergonomic assessments. This work is grounded in the following research question : ***How do architectural design choices and spatio-temporal information from inertial sensors influence the performance and interpretability of deep learning based posture recognition?***

4.3.1 Solution Description

This chapter details the development and validation of the posture recognition module. It encompasses: i) data acquisition following the defined experimental protocol; ii) dataset preparation, including labeling, integrity verification, signal reorientation and resampling, feature selection, and segmentation into windows for model input; iii) the dataset splitting strategy, proposed deep learning architectures, and implementation details; and iv) hyperparameter optimization and evaluation of the final models. An overview of the full development pipeline is shown in Figure 49.

4.3.2 Dataset Acquisition

No publicly available dataset was identified that provides posture-level annotations suitable for generic ergonomic analysis across manual work activities. While a limited number of datasets exist in related domains, such as construction [207], their class definitions typically focus on task specific or highly granular activities rather than fundamental postures that recur across different types of manual work. Consequently, a custom dataset was created to develop the posture recognition models, based on the posture classes identified as most relevant in our previous work [44]:*standing*, *reaching*, *stooping*, *squatting*, *kneeling*, *lifting/lowering*, *carrying* and *others*. Illustration of these postures is shown in Figure 50.

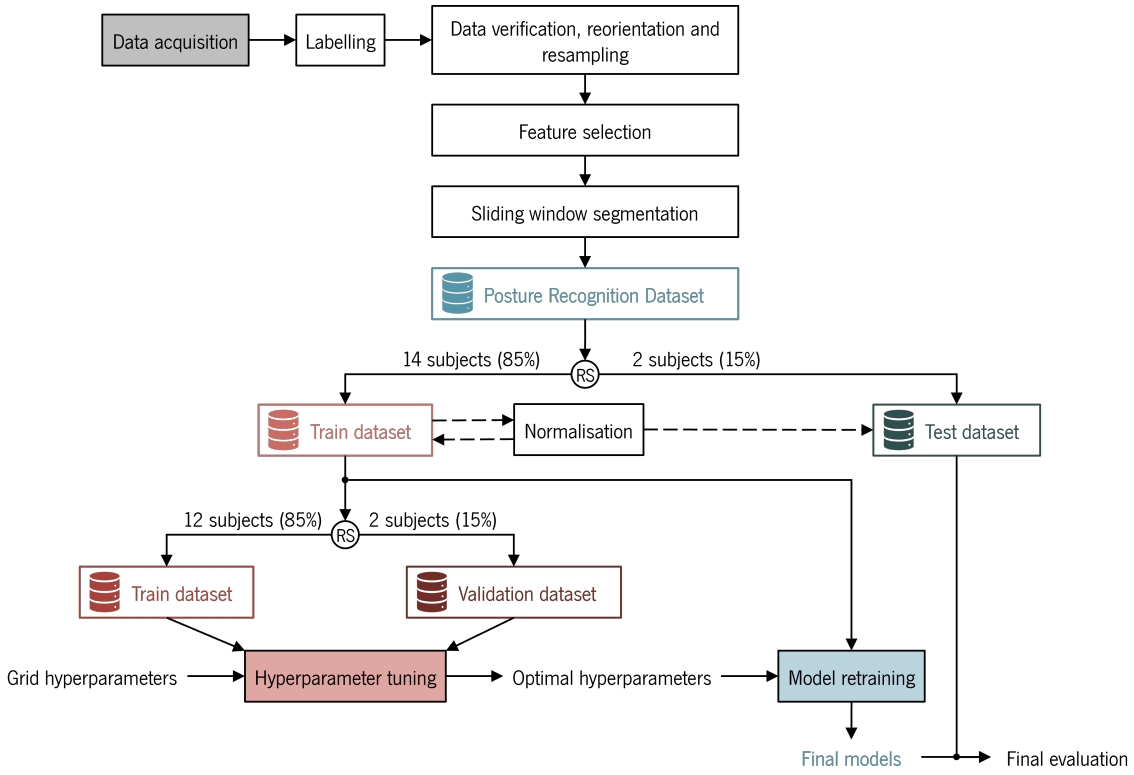


Figure 49: Pipeline of the development of the DL-based posture recognition models, where RS denotes random sampling.



Figure 50: Identification of typical postures.

The data acquisition was conducted inside a laboratory in the School of Engineering of the University of Minho, under the ethical procedures of the Ethics Committee in Life and Health Sciences (CEICVS 147/2021), following the standard set by the declaration of Helsinki and the Oviedo Convention.

Participants

Sixteen healthy subjects (10 males and 6 females, age: 24.2 ± 1.9 years) participated in the data acquisition. Participants were physically diverse (body mass: 66.0 ± 10.9 kg, height: 171.6 ± 10.9 cm, hip height: 96.0 ± 7.5 cm, arm span: 169.8 ± 12.2 cm) to capture variability in posture execution due to body stature. All subjects met the inclusion criteria of having no clinical history or evidence of motor impairments.

Instrumentation

Participants were equipped with two full-body inertial MoCap systems Xsens MTw Awinda (Xsens Technologies B.V., Movella Inc., The Netherlands). The first system streamed raw inertial data (acceleration, angular velocity, magnetic field) via Xsens MT Manager to develop the posture recognition model. The second system was connected to the proprietary software Xsens MVN Analyze, whose virtual representation of the subject's avatar was used for the model GT. The two sets of 17 IMUs were stacked with tape. The MVN system was placed on the bottom, closer to the body. The sensors' placement followed the manufacturer's guidelines, on the head, shoulders, upper arms, forearms, hands, chest, pelvis, thighs, shanks and feet, over bony landmarks and tightened with straps, as shown in Figure 51. Data were recorded at 60 Hz.

Protocol

After the participants' instrumentation, their anthropometric data was collected to adjust the biomechanical model from Xsens MVN Analyze software for them. The GT sensory system was calibrated in N-pose, performing the necessary repetitions until achieving good calibration quality. The other system's IMUs were turned on after that, to avoid interference.

The participants were instructed to perform movement sequences in three different scenarios, aiming at a dataset comprising posture classes performed both solo and with transitions between them, similarly to real worksites. The trials were performed as follows:

1. **Isolated postures**: Seven posture classes were performed without any associated action or task. Thirteen trials were recorded three times each, in the following (sequences of) postures: **i)** *stand-*



Figure 51: Participant instrumented with the two Xsens MTw Awinda systems. Blue circles highlight the sensors used in the DL models, after the feature selection.

ing; **ii)** *reaching*; **iii)** *standing* → *reaching* → *standing*; **iv)** *standing* → *carrying* → *standing*; **v)** *stooping*; **vi)** *squatting*; **vii)** *kneeling*; **viii)** *standing* → *lifting/lowering* → *stooping* → *lifting/lowering* → *standing*; **ix)** *standing* → *lifting/lowering* → *squatting* → *lifting/lowering* → *standing*; **x)** *standing* → *lifting/lowering* → *kneeling* → *lifting/lowering* → *standing*; **xi)** *stooping* → *lifting/lowering* → *squatting* → *lifting/lowering* → *kneeling*; **xii)** *kneeling* → *lifting/lowering* → *squatting* → *lifting/lowering* → *stooping*; **xiii)** *stooping* → *lifting/lowering* → *kneeling* → *lifting/lowering* → *stooping*. *Reaching*, *carrying*, *stooping*, *squatting*, and *kneeling* postures were held for 5 s each, and *standing* posture only for 3 s, as this one was repeated more times.

2. **Agriculture task circuit**: This scenario replicated the harvesting and transportation of crops. The subject should pick a total of 12 Christmas baubles (simulating fruits) from an artificial tree at a height of 90 to 180 cm (Figure 53a), and put them into a bucket on the floor, one by one (Figure 53b), at three harvest sites, located 2.5 m apart, transporting the bucket between each of them (Figures 53c and 53d), during each trial.Six repetitions of this trial (with an average duration

of 2 min) were performed, three going to the floor by *stooping*, and another three by *squatting*. Additionally, *standing*, *reaching*, *lifting/lowering* and *carrying* were the other postures needed to accomplish it.

3. **Construction task circuit**: This trial mimicked bricklaying and consisted of carrying a bucket containing six milk cartons (simulating bricks) over 10 m (Figure 52a), from a static starting position to a work site located one subject-foot from a 90 cm high workbench. The subject placed the bucket on the workbench, where a sheet of paper simulated a mortar tray, and stacked the milk cartons in two vertical layers to emulate wall construction. Each carton was picked up (Figure 52b) and placed on the floor to the right of the workbench (Figure 52c). Using a trowel initially placed on the workbench (Figure 52d), the subject simulated spreading mortar on the floor before starting a new layer and on the sides of each carton before and after placement.Each subject performed nine repetitions (average duration 2 min), adopting a fixed posture to reach the floor in each trial: three with *stooping*, three with *squatting*, and three with *kneeling*, to ensure class balance. The task also involved *standing*, *reaching*, *lifting/lowering*, and *carrying* postures.

The protocol was explained to the participants without imposing a way to perform each posture, to guarantee variability.



(a) (b) (c) (d)

Figure 52: Participant in the experimental setup of the agriculture task circuit.



(a) (b) (c) (d)

Figure 53: Participant in the experimental setup of the construction task circuit

4.3.3 Experimental Methodology

Labeling and Data Preprocessing

After the data collection, the labeling was executed offline and manually, by visualizing the avatar playback in the Xsens MVN Analyze interface, sample by sample, at 60 Hz.

Clipped and missing samples were replaced by second-order polynomial interpolation, with a limit of 30 samples (0.5 s) for both backward and forward directions.

Then, the raw data was reoriented so that all sensors share the same orientation, with all axes aligned with a reference frame. For this, the original orientation of each IMU was assumed to be the same for all subjects. This is considered reasonable since their instrumentation was verified at the beginning of each trial, and, whenever a slippage occurred, the trial stopped and was discarded and repeated.

This data, until this step, was organized in a database, hosted on Zenodo [208], to enable its widespread use within the research community.

The inertial data acquired at 60 Hz was resampled to 20 Hz, since the trials' frequency spectrum revealed that the signals' frequencies lay below 5 Hz. This frequency respects the Nyquist theorem, as it is superior to the human movement frequencies (mostly between 0.3 and 3.5 Hz, with a maximum of 10 Hz [54]). Moreover, in the literature, activity recognition performance is reported not to increase beyond 20 Hz [71]. The downsampling was achieved with a moving average filter, which helps to smooth the signals and reduce random noise while keeping the sharpest step response [209] and whose simplicity makes it suitable for further real-time implementation. The number of samples considered in the filter was equal to the downsampling factor (3). Accordingly, the labels were also downsampled, by taking a label

every 3 timestamps.

After resampling, the dataset contains 768,224 samples (10 h 40 min). Class distribution was evaluated in terms of sample count and percentage. As shown in Figure 54, *standing* and *lifting/lowering* are the most represented classes. *Reaching*, *stooping*, *squatting*, and *kneeling* show similar representation. *Carrying* and *others* are underrepresented, with *others* being the least frequent. The maximum class imbalance ratio is approximately 1:7.

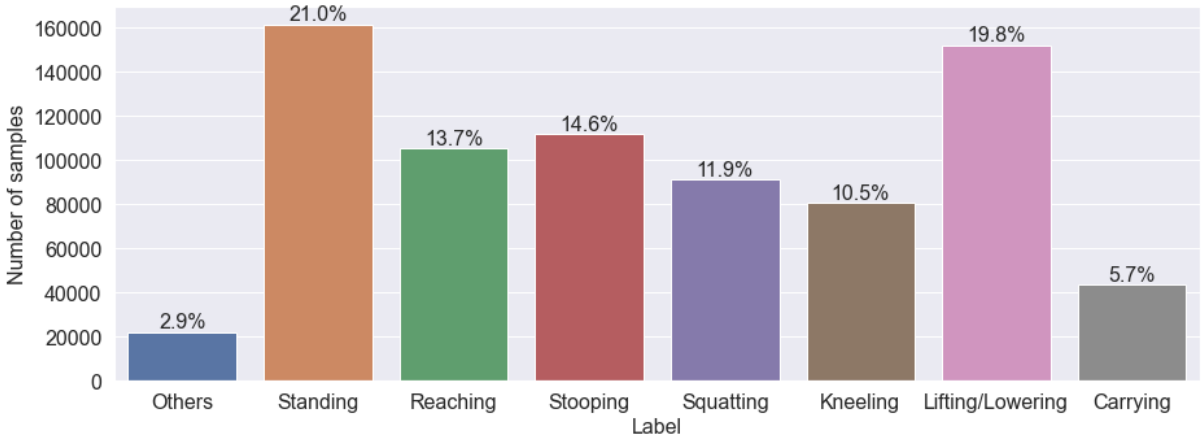


Figure 54: Number of samples and percentage of posture classes in the dataset.

To minimize bias in the neural network, as different sensors may have distinct ranges and sensitivities, MinMax normalization between 0 and 1 was applied to the data, considering the extreme values of the entire training dataset by feature.

Feature Selection

Feature selection was performed to reduce computational load and eliminate redundant inputs. Magnetometer data were excluded because identical postures produce different measurements when orientation relative to the Earth's magnetic field changes, introducing variability that could confuse the DL models.

This dataset is neither normally distributed nor characterized by equal variances. Thus, Kruskal–Wallis H-test [210] was applied to the complete labeled dataset with eight classes to assess feature relevance. Features with p-value < 0.05 were considered significantly different across groups [210]. Following [211], the top 25% of features ranked by Kruskal–Wallis score (H) were retained. From this subset, the six channels per IMU were selected, as 3-axis accelerometer and gyroscope signals are typically coupled. This resulted in the selection of 13 IMUs (head, sternum, pelvis, shoulders, forearms, upper legs, lower legs, and feet) out of 17. Notably, all six channels of each selected IMU exhibited p-values below the threshold, rejecting the null hypothesis of equal class medians.

The correlation between the different channels was also studied, through Spearman's correlation (see Figure 141 in appendices), since data was not normally distributed [212]. Left and right IMUs on the shoulders, upper legs, lower legs, and feet show high correlation, indicating redundant information. Therefore, only one side per segment was retained for posture classification. The selected side corresponds to the higher H value from the previous test, resulting in the retention of the right shoulder, left upper leg, right lower leg, and left foot.

Hence, the final sensor set consisted of seven accelerometers and gyroscopes, on the sternum, right shoulder, right forearm, left forearm, left upper leg, right lower leg, and left foot. These are highlighted with blue circles in Fig. 51.

Furthermore, we conducted a preliminary test with a CNN model and verified that using only the selected sensors' data as input effectively led to better results than using all 17 sensors' data.

Data Input Preparation

Using the sliding window technique, data was divided into fixed-size time segments, with the possibility of overlap between windows. Overlap helps to capture transitional postures between consecutive windows and increases the dataset. Finding the optimal Window Size (WS), i.e., the size of the signal subsequences relevant to extract discriminative features [213], is not trivial since distinct postures, and even the same posture performed by different subjects, have different durations [214].

In this work, all sensor channels were concatenated along the same dimension. During hyperparameter tuning, the models were evaluated using different WSs (0.5 s, 1 s, 1.5 s, and 2 s) and overlap percentages (0%, 15%, 30%, and 50%). Although larger window sizes have been reported in the literature, they are unsuitable here because *lifting/lowering*, the shortest posture class, does not exceed these durations. The maximum overlap was chosen to ensure that consecutive windows shared no more than half of their data.

For the smallest WS of 0.5 s, the number of windows ranged from 76 353 with 0% overlap to 152 149 with 50% overlap. For the largest WS of 2 s, the number decreased to 18 636 without overlap and 36 837 with 50% overlap.

Regarding labeling, each window was assigned the majority label of its samples, which is the most common approach [66].

Dataset Split

The dataset was randomly split by subject into training, validation, and test sets, ensuring that validation and test data were never seen during training to avoid bias.

Of the 16 subjects, 14 were used during grid search, while the remaining 2 were reserved exclusively for final testing using the best-performing models. Among the 14 subjects, 12 were assigned to training and 2 to validation following a leave-two-subjects-out strategy. For testing, models were retrained using data from the 14 subjects without validation, applying the optimal hyperparameters. Test set normalization used statistics from the 14 training subjects, while validation data were normalized using the 12-subject training set, assuming high inter-subject variability.

Although cross-validation would provide stronger evidence of generalizability, it was not applied due to time constraints. Instead, a hold-out approach was adopted, with random but fixed subject assignments.

Ideally, the data split would preserve population characteristics across subsets to avoid biased representations [215]. However, the limited number of validation and test subjects prevented maintaining comparable anthropometric distributions, making stratified sampling infeasible.

DL Architectures

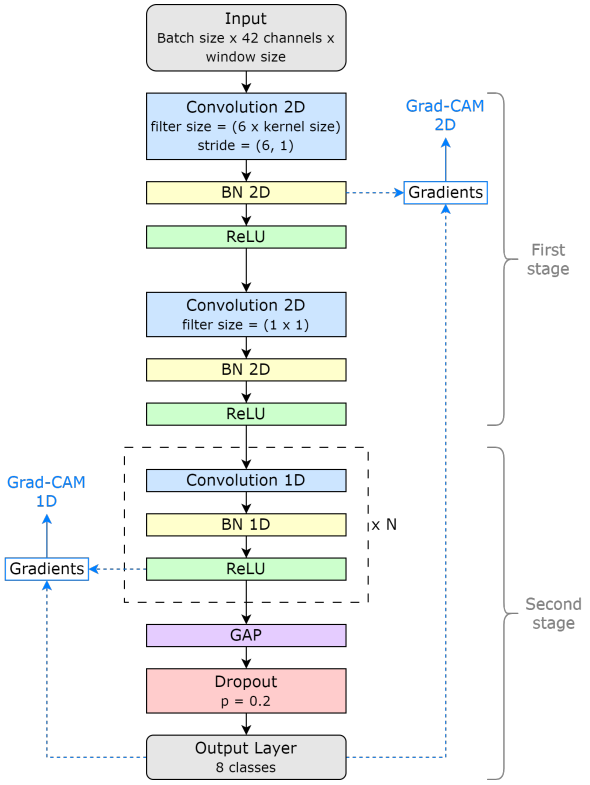
Five DL architectures for classification were implemented: CNNs, LSTMs, hybrid CNN-LSTMs, and transformers.

CNN The CNN used in this work, inspired by [216], was designed to be explainable and allow Grad-CAMs to identify which sensors and time steps contribute most to posture recognition. It consists of two main stages (Figure 55a).

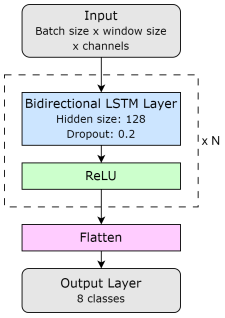
The first stage consists of a 2D convolution with a stride of 6 (channels per IMU) along the feature axis and 1 along the time axis. This ensures the kernel slides over time without mixing data from different body segments, preserving their individuality and enabling 2D Grad-CAM to highlight relevant body segments, as in [216]. Unlike parallel convolutions, the same filters are shared across all segments, reducing the number of weights. The number of filters was treated as a hyperparameter. A subsequent 1×1 convolution compresses the resulting feature maps into a single 2D representation.

In the second stage, a series of 1D convolutions combines features across body segments, with filters sliding over time. A 1D Grad-CAM is then used to identify the most relevant time steps at the global level. The number of layers and filters, which doubled at each layer starting from the first stage, were treated as hyperparameters and tuned accordingly.

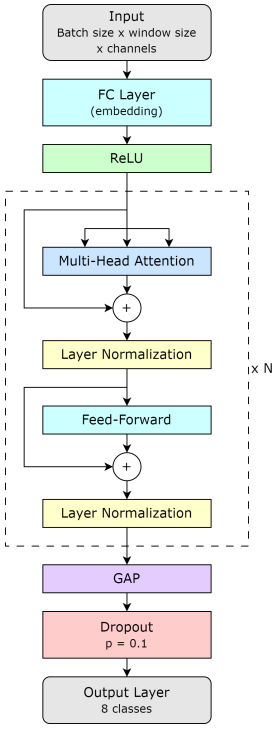
Each convolutional block consisted of a convolutional layer followed by Batch Normalization (BN) [217] and a ReLU activation. Dropout (20%) was applied only after the final convolutional block to avoid the variance shift issue when used before BN [218], while still improving generalization.



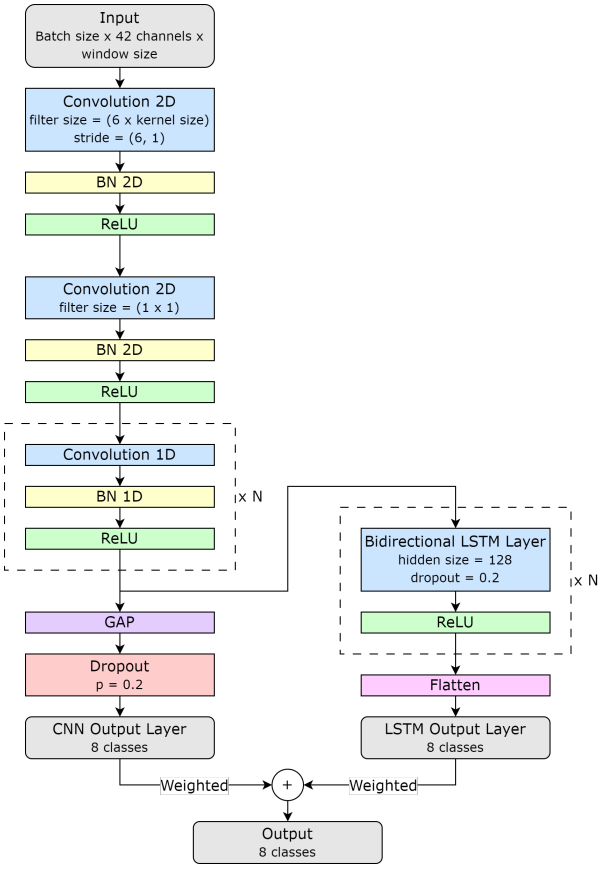
(a) Explainable CNN



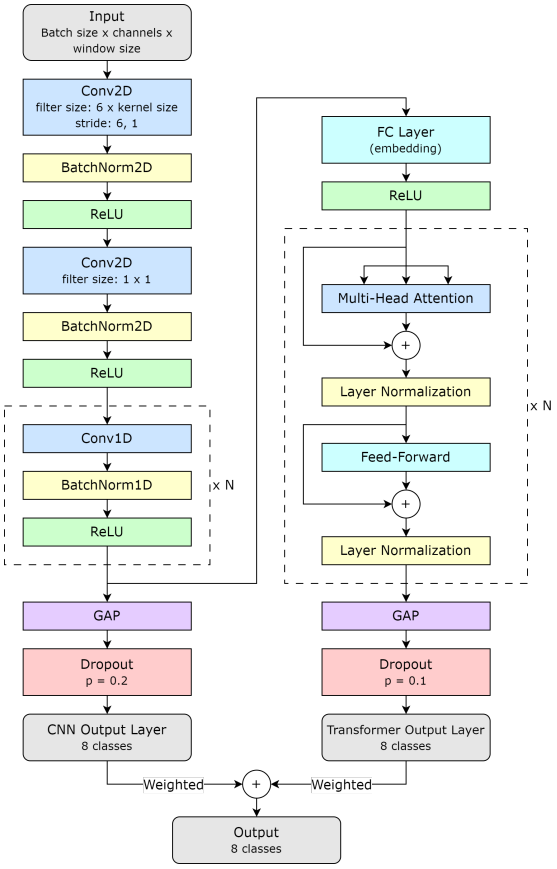
(b) LSTM



(c) Transformer



(d) Hybrid CNN–LSTM



(e) Hybrid CNN–Transformer

Figure 55: Proposed architectures.

Instead of using Fully Connected (FC) layers after the convolutional blocks, Global Average Pooling (GAP) was adopted to reduce overfitting and convert the 2D representation into a 1D feature vector [219]. A preliminary experiment confirmed that FC layers led to overfitting, whereas GAP did not, due to the absence of learnable parameters [220]. Additionally, GAP reduces hyperparameter complexity and experimental cost.

To compute the Grad-CAM for body segment relevance, L^c_{2D}, the gradients of the class score y^c with respect to each feature map A^f are averaged over the sensor (S) and temporal (WS) dimensions, yielding the weights α^c_f in (4.3):

$$\alpha^c_f = \frac{1}{S \times WS} \sum_s \sum_t \frac{\partial y^c}{\partial A^f_{st}} \tag{4.3}$$

These weights are used to compute a class-specific weighted combination of the feature maps, followed by a ReLU to retain only positive contributions, as shown in (4.4):

$$L^c_{2D} = ReLU\left(\sum_f \alpha^c_f A^f\right) \tag{4.4}$$

The time-related Grad-CAM, L^c_{1D}, follows the same principle but is applied to the output of the last 1D convolutional layer. Using the feature maps A^g, it is computed as:

$$\alpha^c_g = \frac{1}{WS} \sum_t \frac{\partial y^c}{\partial A^g_t} \tag{4.5}$$

$$L^c_{1D} = ReLU\left(\sum_g \alpha^c_g A^g\right) \tag{4.6}$$

LSTM The LSTM architecture includes 2 or 3 (hyperparameter for tuning) bidirectional layers. The choice of bidirectional over unidirectional layers was due to the former's superior performance [221], able to process input sequences both forwards and backwards, considering context information before and after each sequence's time step [222]. The hidden size was fixed at 128, like most of the State of the Art (SOA), and a dropout of 0.2 was applied to avoid overfitting, followed by ReLU. Figure 55b presents the model architecture.

Transformer From the original proposal [69], only the transformer encoder was used. Its architecture is presented in Figure 55c. To generate the latent features, before entering the encoder blocks, the input windows pass through an FC layer that embeds the data in a higher dimension, whose value was a hyperparameter for optimization, as well as the number of encoder layers and the number of parallel heads. Their values (in Table 24) were chosen by surveying the ones used in SOA studies for posture/activity recognition [63, 64, 223]. Following these works, the dimensionality of the inner FC layers was always fixed at twice the embedding size. Similarly to the CNNs, before the output, a GAP layer was used instead of FC layers to achieve lightweight models. As in [69], a small dropout of 0.1 was applied to each sublayer output (before adding it to the sub-layer input and normalizing) and after GAP. The decision not to include the sine and cosine-based positional encoding, like the original transformer [69], was based on a small experiment we performed, where it was observed that, with the same hyperparameters, its removal led to better metrics (2.74 percentage points (pp) higher F1-score).

Hybrid CNN-LSTM with Learnable Fusion A hybrid CNN-LSTM with learnable fusion was also implemented. The last convolutional block output is the LSTM input, but CNN and LSTM outputs are also blended by a weighted sum, as depicted in Figure 55d. These weights were set as learnable parameters. The idea behind this model is that the network leverages the advantages of both architectures and learns which output, CNN's or LSTM's, should be more valued, i.e., the network discovers the most effective approach to fusing the subnetworks. Each class score in the output is weighted separately, since there may be classes that CNN identifies better than LSTM and vice versa. These weights, initialized as 1 (same importance to both outputs), pass through a ReLU function to guarantee that the outputs of each subnetwork are not subtracted. The hyperparameters to optimize, regarding the CNN and LSTM blocks, were the same as for isolated models.

Hybrid CNN-Transformer with Learnable Fusion

Similarly, a hybrid CNN-Transformer with learnable fusion was implemented (Fig. 55e). The joining of CNN with Transformer is similar to that of CNN with LSTM, and the hyperparameters to optimize were the same as for the isolated models.

4.3.4 Implementation Details

Model training lasted a maximum of 50 epochs. Early stopping was used, and the criterion was the lack of improvement in validation loss for 10 epochs, i.e., the goal was to find the epoch with the

minimal loss, before overfitting. All models were trained using Cross Entropy Loss in conjunction with AdamW optimizer with a Learning Rate (LR) scheduler, where the LR, whose initial value was considered a hyperparameter for tuning, decayed by a multiplicative factor of 0.2 every 10 epochs. The batch size was another hyperparameter tuned in the grid search, and we tested small values, powers of 2, to fully leverage GPUs' processing, as recommended by [224].

Table 24 summarizes the various hyperparameters tuned during the grid search for all the models and the respective values. In total, 512 experiments were run for CNN, 128 for LSTM, 768 for CNN-LSTM, and 1152 for transformer.

Table 24: Hyperparameters explored in the grid search

Types of hyperparameters		**Hyperparameters**	**Values**
Training		Batch size	32, 64
		LR (initial)	10^{-4}, 10^{-3}
Data Preprocessing		WS (s)	0.5, 1, 1.5, 2[a]
		Overlap (%)	0, 15, 30, 50
Architecture	CNN	Number of convolutional layers	4, 5
		Number of filters	(64 for 2D conv; 64, 128, 256 for 1D conv), (128 for 2D conv; 128, 256, 512 for 1D conv)
		Kernel size	3, 5
	LSTM	Number of LSTM layers	2, 3
	Transformer	Number of encoder layers	2, 5, 8
		Number of heads	4, 8, 16
		Embedding size	128, 256

[a]Value not used for hybrid CNN-LSTM models; conv: convolutions

The whole pipeline was implemented using Pytorch and Pytorch Lightning libraries in a Python environment. Grid search experiments were divided by multiple nodes of the University of Minho's SeARCH cluster. The test phase was carried out on a personal laptop with the following specifications: GPU: 1x Nvidia GeForce GTX 1050; CPU: Intel(R) Core(TM) i7-7700HQ @ 2.80GHz (4 core, 8 threads); RAM: 16 GB.

Model Evaluation Metrics

The models' quality was quantified through accuracy and F1-score metrics, in the epoch with the minimum loss. Despite being the most popular metric, accuracy is misleading for an imbalanced dataset

like this, as it considers every class equally important [225]. The F1-scores of all classes were macro-averaged, since macro F1-score gives equal importance to each class, allowing objective results on imbalanced datasets. Training and inference time are also important for evaluating the models. However, since the grid search was performed on multiple machines, those times are not comparable. Only testing times are presented, as the final evaluation of the models' generalizability was performed on the same computer.

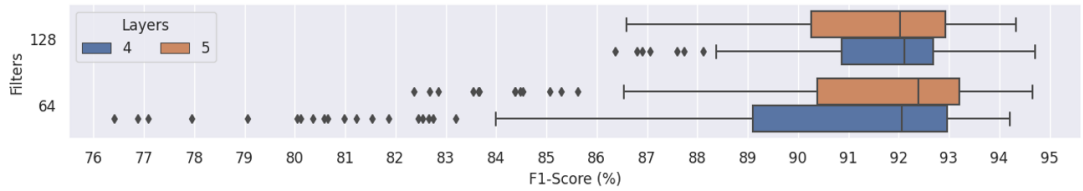
4.3.5 Results

Hyperparameter Optimization Results

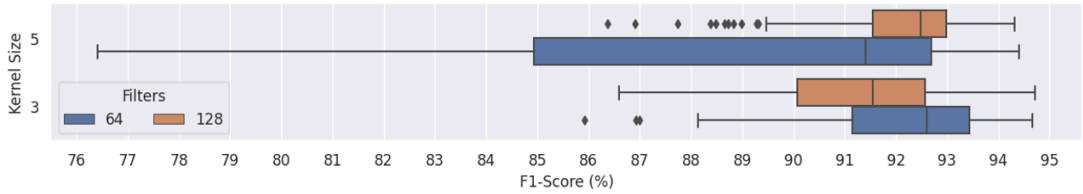
Each architecture's results are presented separately, focusing on their specific hyperparameters. Then, the results of the common hyperparameters study (non-specific for each architecture) are presented and discussed jointly. The results are presented through both summary tables and box plots. The boxplots present the F1-score distribution of the hyperparameter combinations, aiming to identify the more consistent combinations in terms of F1-score, i.e., narrower boxes (less variability) and higher F1-scores, to provide guidelines for the choice of the "safest" hyperparameters. For clarity of presentation and due to space constraints, only the top three hyperparameter combinations are reported in the main body of the thesis, while the complete top ten ranking is provided in Appendix D.2.

CNN All 10 models from the grid search with the highest validation F1-scores reported F1-scores above 94% and accuracies above 96.5% in the epoch with the lowest loss. Specifically, the best model achieved an F1-score of 94.72%. Table 56, in appendix D.2, presents the ten best performing CNN models.

The CNN architecture-specific hyperparameters were analyzed to assess their impact on classification performance. Figure 56 presents the distribution of F1-scores grouped by the evaluated hyperparameter values, which were analyzed jointly due to their interdependence. Figure 56a shows that configurations with a larger number of filters generally exhibited narrower F1-score distributions, indicating more consistent performance. Among the evaluated configurations, the combination of 5 convolutional layers and 64 filters achieved the highest median and third-quartile F1-scores and appeared most frequently among the top ten performing models. Figure 56b shows that models combining smaller kernel sizes with fewer filters and larger kernel sizes with more filters achieved comparable median F1-scores. Overall, models using a kernel size of 3 consistently achieved higher performance metrics.



(a) Filters and layers



(b) Kernel size and filters

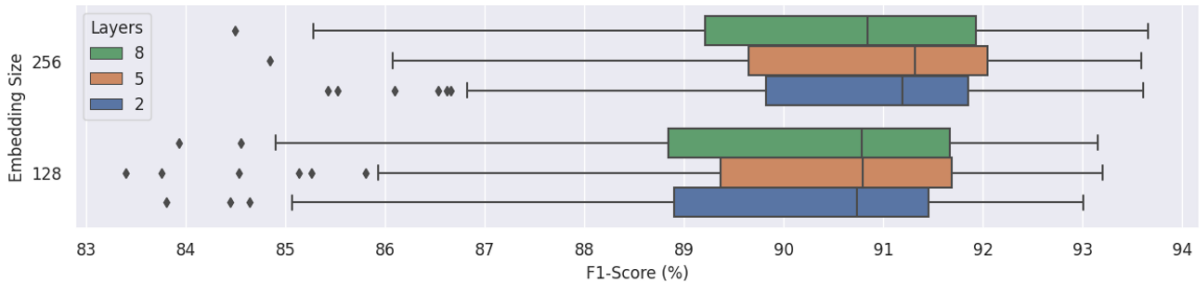
Figure 56: Box plot of validation F1-scores for different combinations of (a) number of filters and layers, and (b) kernel size and number of filters.

LSTM For the LSTM-based models, grid-search experiments showed that the top ten configurations achieved F1-scores above 92%, with a maximum of 92.45%, and accuracies exceeding 95.5%. Table 57, in appendix D.2, presents the ten best performing LSTM models.

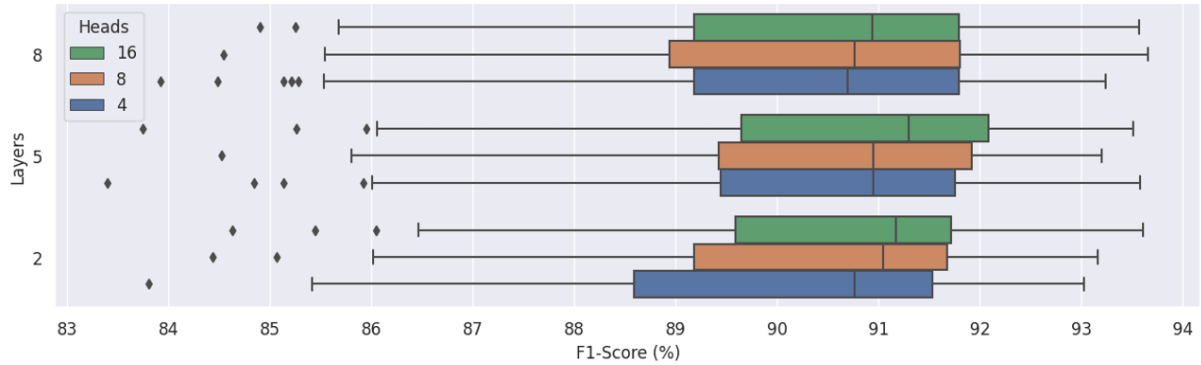
Overall, models with two LSTM layers exhibited more consistent F1-score distributions than those with three layers, although the quartiles were similar. The top ten configurations included four pairs of models differing only in the number of layers, with negligible differences in F1-score. For the best-performing configuration, increasing the number of layers from two to three reduced the F1-score by approximately 0.2 percentage points.

Transformer The top 10 transformers (in terms of F1-score) were able to achieve above 93.2% F1-score and 95.9% accuracy, and the best one was 93.66% F1-score. Table 58, in appendix D.2, presents the ten best performing transformer models.

The influence of architectural hyperparameters, namely embedding size, number of layers, and number of attention heads, is summarized in Figure 57. Figure 57a shows that models with an embedding size of 256 consistently outperformed those using 128, as reflected by higher quartiles and maximum F1-scores. All top ten configurations employed an embedding size of 256. Models with five encoder layers achieved higher F1-scores than those with two or eight layers. The effect of the number of layers was more pronounced for the larger embedding size, while for an embedding size of 128, F1-score distributions were similar across layer counts. Figure 57b further indicates that increasing the number of attention heads was associated with higher F1-scores.



(a) Embedding size and number of layers



(b) Number of layers and heads

Figure 57: Box plot of validation F1-scores for different combinations of (a) embedding size and number of layers, and (b) number of layers and heads, for transformer architecture.

CNN-LSTM Among all evaluated architectures, a hybrid CNN–LSTM model achieved the highest validation F1-score of 95.06%. The top ten configurations all obtained F1-scores above 94.3% and accuracies exceeding 96.2%. Table 59, in appendix D.2, presents the ten best performing CNN-LSTM models.

Figure 58 illustrates the effect of the number of convolutional and recurrent layers on hybrid model performance. Models with fewer convolutional layers (four) consistently achieved higher performance when combined with two LSTM layers. For configurations with five convolutional layers, performance varied; however, the highest overall results were obtained when paired with two recurrent layers.

The fusion weights between the CNN and LSTM subnetworks were also examined. When using a learning rate (LR) of 10^{-4}, the fusion weights assigned to both subnetworks were equal. In contrast, with an LR of 10^{-3}, the hybrid models assigned substantially higher weights to the CNN outputs. This is demonstrated in Table 25, by taking as examples the five models with the highest validation F1-scores, though this phenomenon extended to the remaining models.

CNN–Transformer

Among all evaluated architectures, a hybrid CNN–Transformer model achieved the highest validation F1-score of 95.39%. The top ten configurations all reached F1-scores above 94.9%. Table 60, in appendix

Table 25: Weights learned by the top 5 hybrid CNN-LSTM models (sorted by F1-score) to fuse both sub-networks' outputs, in the validation

Model	LR	Weights \Class	Others	Standing	Reaching	Stooping	Squatting	Kneeling	Lifting/Lowering	Carrying
# 1	10^{-4}	**CNN output**	0.98	0.99	1.00	0.99	0.99	0.98	0.96	0.99
		LSTM output	0.80	0.92	0.89	0.82	0.86	0.78	0.99	0.81
# 2	10^{-3}	**CNN output**	1.15	1.39	1.37	1.45	1.32	1.28	1.20	1.24
		LSTM output	5.37×10^{-6}	1.83×10^{-6}	2.92×10^{-9}	3.01×10^{-8}	4.77×10^{-6}	7.36×10^{-6}	7.43×10^{-6}	6.42×10^{-9}
# 3	10^{-4}	**CNN output**	0.98	1.00	1.01	1.00	1.00	0.99	0.98	0.99
		LSTM output	0.73	0.85	0.84	0.75	0.79	0.72	0.97	0.73
# 4	10^{-4}	**CNN output**	0.96	1.05	1.03	0.97	1.03	1.01	0.94	0.99
		LSTM output	0.77	0.86	0.88	1.02	0.69	0.74	0.99	0.76
# 5	10^{-3}	**CNN output**	1.15	1.51	1.45	1.52	1.42	1.36	1.30	1.29
		LSTM output	1.86×10^{-7}	8.72×10^{-7}	2.10×10^{-7}	4.32×10^{-7}	1.36×10^{-7}	7.87×10^{-10}	1.51×10^{-6}	1.43×10^{-6}

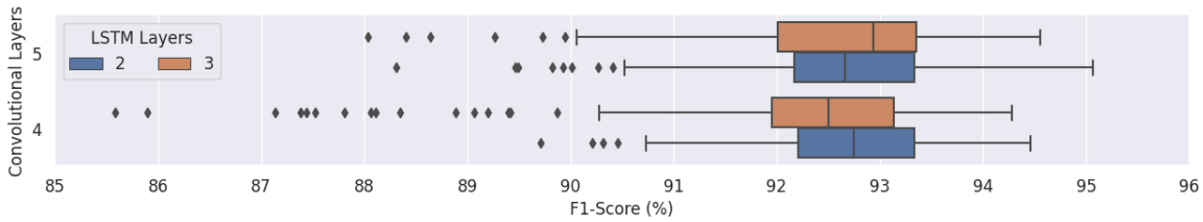


Figure 58: Box plot of validation F1-scores for the different combinations of the number of convolutional layers and LSTM layers for CNN-LSTM architecture.

D.2, presents the ten best performing CNN-transformer models.

Figure 59 shows the effect of varying the number of convolutional and encoder layers on hybrid model performance. Configurations with fewer encoder layers achieved higher validation F1-scores, while the top-performing CNN–Transformer models consistently employed a higher number of convolutional layers (five rather than four).

The fusion weights between the CNN and Transformer subnetworks were also analyzed. For models trained with the highest learning rate, the fusion weights assigned to CNN and Transformer outputs were balanced. In contrast, for models trained with the lowest learning rate, the fusion weights assigned to the Transformer output dominated, with the CNN contribution being nearly suppressed, as illustrated by representative models in Table 26.

Training Hyperparameters and Input Data The following results concern the analysis of neural networks' common hyperparameters. Figure 60 presents the distribution of validation F1-scores obtained for the four combinations of LR and batch size, considering all five architectures. Models trained with an LR

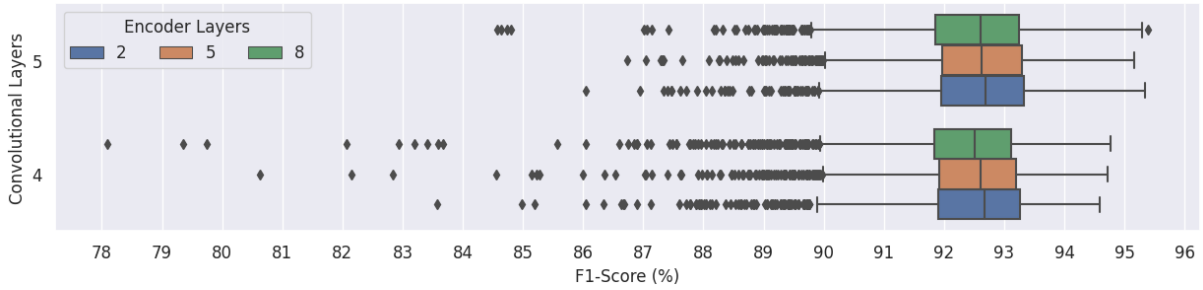


Figure 59: Box plot of validation F1-scores for the different combinations of the number of convolutional layers and encoder layers for CNN-transformer architecture.

Table 26: Weights learned by the top hybrid CNN-Transformer models (sorted by F1-score) to fuse both subnetworks' outputs, in the validation

Model	LR	Weights \Class	Others	Standing	Reaching	Stooping	Squatting	Kneeling	Lifting/Lowering	Carrying
# 1	10^{-4}	**CNN output**	0.95	0.96	0.98	0.96	0.96	0.95	0.98	0.97
		Transformer output	0.94	0.95	0.94	0.94	0.94	0.92	1.02	0.91
# 2	10^{-4}	**CNN output**	0.87	0.85	0.90	0.89	0.87	0.82	0.93	0.90
		Transformer output	0.98	0.98	0.98	0.97	0.99	0.97	1.02	0.97
# 3	10^{-4}	**CNN output**	0.95	0.96	0.98	0.96	0.96	0.95	0.98	0.97
		Transformer output	0.94	0.95	0.94	0.94	0.95	0.92	1.02	0.91
# 4	10^{-4}	**CNN output**	0.86	0.92	0.93	0.94	0.89	0.90	0.93	0.91
		Transformer output	0.99	0.97	0.98	0.96	0.99	0.94	1.05	0.95
# 5	10^{-3}	**CNN output**	7.99×10^{-7}	1.28×10^{-7}	2.63×10^{-8}	3.69×10^{-7}	3.31×10^{-7}	1.43×10^{-7}	6.24×10^{-7}	8.30×10^{-7}
		Transformer output	0.86	0.98	1.00	0.94	1.00	0.97	0.94	0.88
# 14[a]	10^{-3}	**CNN output**	1.63×10^{-6}	2.45×10^{-6}	6.57×10^{-6}	2.94×10^{-6}	9.17×10^{-6}	3.10×10^{-7}	4.83×10^{-7}	3.22×10^{-7}
		Transformer output	0.85	0.97	0.96	0.95	1.01	0.96	0.94	0.91

[a]Second best-performing model with the LR of 10^{-3}.

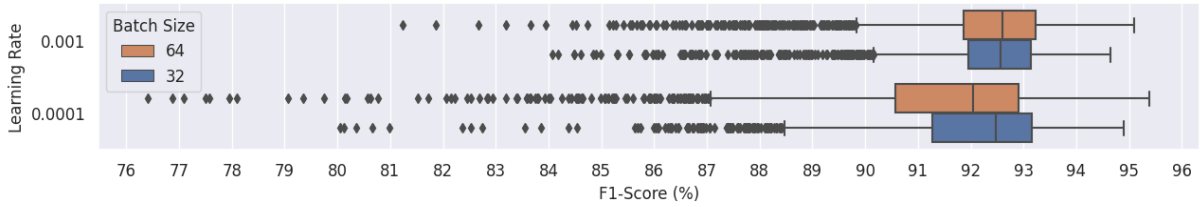


Figure 60: Box plot of validation F1-scores for different combinations of LR and batch size, considering all the models of the five architectures.

of 10^{-3} exhibited more consistent performance, with similar F1-score distributions across both batch sizes. In contrast, models trained with an LR of 10^{-4} and a batch size of 64 showed higher variability, yielding both the highest and lowest F1-scores observed. These trends were consistent across all architectures.

The effects of WS and overlap were also analyzed jointly due to their interdependence. Figure 61 shows

the validation F1-score distributions for different combinations of WS and overlap. A WS of 1 s combined with 50% overlap consistently yielded the best overall performance. Across all architectures, a WS of 1 s resulted in the highest and most consistent F1-scores for all overlap values. The 50% overlap produced the highest median F1-scores and the narrowest, rightmost distributions for all WS values except 0.5 s, whereas the behavior of other overlap values varied depending on the WS.

Despite these general trends, 7 of the 10 models with the highest validation F1-scores used no overlap, including the top three models, all of which were hybrid CNN–Transformer architectures. Conversely, a WS of 2 s led to the lowest and most variable F1-scores. The worst-performing configurations shared large window sizes (2 s) combined with low to medium overlaps (0–30%).

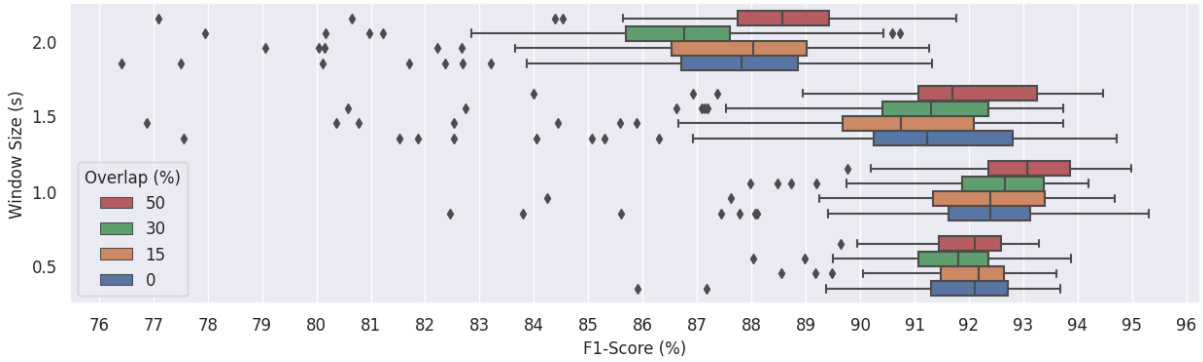


Figure 61: Box plot of validation F1-scores for different combinations of WS and overlap, considering all the models of the four architectures.

Final Test Results

Each architecture's three best models (in terms of F1-score in the validation) were selected for new training and final evaluation using the test set. Table 27 describes these models, as well as the number of training epochs (where the validation loss was minimal in the grid search), and presents the corresponding training time, inference time per window, test F1-score and accuracy.

All 15 evaluated models exhibited a decline in F1-score on the test set compared to validation, and the models that achieved the best validation performance among CNNs and Transformers did not retain their ranking in the test. In the final evaluation, training and validation sets were merged, early stopping was not applied, and training was stopped at the epoch corresponding to the early stopping criterion identified during grid search. As a result, the test evaluation did not necessarily correspond to the optimal training point. Additionally, the use of a fixed hold-out split may have contributed to the observed performance decline, as the validation subjects may have been easier to classify than the test subjects.

Table 27: Hyperparameters, number of training epochs, training and inference times, and test metrics of the three best models of each architecture

Architecture	Model ID	LR	Batch Size	WS (s)	Overlap (%)	Architecture Hyperparameters Tuned	Epoch	Training Time (min)	Inference Time[a] (ms)	F1-Score (%)	ACC (%)
CNN	CNN 1	10^{-3}	64	1.5	0	cL: 4, F: 128, KS: 3	47	10.84	0.20	91.18	93.81
	CNN 2	10^{-3}	64	1	50	cL: 5, F: 64, KS: 3	46	30.98	0.10	90.70	92.28
	CNN 3	10^{-3}	32	1	50	cL: 5, F: 64, KS: 5	46	62.32	0.12	93.87	95.55
LSTM	LSTM 1	10^{-3}	64	1	50	rL: 2	33	22.19	0.08	92.38	93.91
	LSTM 2	10^{-4}	32	1	50	rL: 3	39	56.08	0.15	84.73	87.34
	LSTM 3	10^{-3}	32	1	50	rL: 3	32	45.43	0.13	90.60	93.44
Transformer	TR 1	10^{-4}	64	1	50	eL: 8, H: 8, ES: 256	49	122.35	0.18	92.31	94.00
	TR 2	10^{-3}	64	1	50	eL: 2, H: 16, ES: 256	39	37.32	0.10	93.15	94.78
	TR 3	10^{-4}	64	1	50	eL: 5, H: 4, ES: 256	48	79.85	0.17	92.85	94.65
Hybrid CNN-LSTM	CNN-LSTM 1	10^{-4}	64	1	0	cL: 5, F: 128, KS: 5, rL: 2	46	26.34	0.16	94.20	95.31
	CNN-LSTM 2	10^{-3}	64	1	15	cL: 5, F: 64, KS: 5, rL: 2	27	15.94	0.16	93.76	95.27
	CNN-LSTM 3	10^{-4}	64	1	15	cL: 5, F: 128, KS: 5, rL: 2	46	30.40	0.15	93.97	95.76
Hybrid CNN-Transformer	CNN-TR 1	10^{-4}	64	1	0	cL: 5, F: 128, KS: 5, eL: 8; H: 16; ES: 128	48	60.43	0.29	**94.33**	95.29
	CNN-TR 2	10^{-4}	64	1	0	cL: 5, F: 128, KS: 5, eL: 2; H: 8; ES: 256	48	32.64	0.17	93.85	94.94
	CNN-TR 3	10^{-4}	64	1	0	cL: 5, F: 128, KS: 5, eL: 8; H: 4; ES: 128	48	54.22	0.37	94.23	95.21

[a]Time to load and classify a window of data; cL: number of convolutional layers; F: number of filters in the first convolutional layer; KS: kernel size; rL: number of LSTM layers; eL: number of encoder layers; H: number of heads; ES: embedding size; ACC: accuracy.

Among the evaluated architectures, hybrid models achieved the best overall performance. Only one CNN model attained F1-scores comparable to those of the hybrid architectures, while the remaining CNN models underperformed relative to the hybrids. The higher performance of hybrid models was accompanied by increased inference times compared to single-architecture models.

LSTMs were not able to achieve F1-scores as high as the other neural networks in the test (neither in the validation), but that did not translate into lower training and inference times compared with, e.g., CNNs. The F1-scores obtained by LSTM models ranged between 84.73% and 92.38%. The hidden size of the LSTM layers was fixed at 128 for all experiments.

The proposed CNN models achieved substantially higher test performance than CNN-based approaches reported in other studies [66, 67]. In the present work, the lowest-performing CNN model achieved an F1-score of 90.70% and an accuracy of 92.28%.

Transformer-based models did not exceed an F1-score of 93.15% in the test. Although Transformers performed worse than CNNs during validation, they achieved higher performance than CNNs on the test set. Transformer models exhibited the highest training and inference times, while CNN models had the lowest, consistent with previous observations [64].

The highest test performance was achieved by the hybrid CNN–Transformer 1 model, which reached an F1-score of 94.33% with an inference time of 0.29 ms per window. This inference time is substantially lower than the 1 s window step, satisfying real-time requirements. Per-class test metrics for this model are reported in Table 28, and the corresponding confusion matrix is shown in Figure 63. The *others* class exhibited the lowest performance across models. Some models showed reduced accuracy for the *kneeling* class, which was sometimes misclassified as *lifting/lowering*. The confusion matrices indicate that *lifting/lowering* was misclassified at least once as every other class except *carrying*. Misclassifications were also observed between *reaching* and *lifting/lowering*. Most errors occurred near posture transitions, particularly for short-duration postures such as *lifting/lowering*, as illustrated in Figure 62.

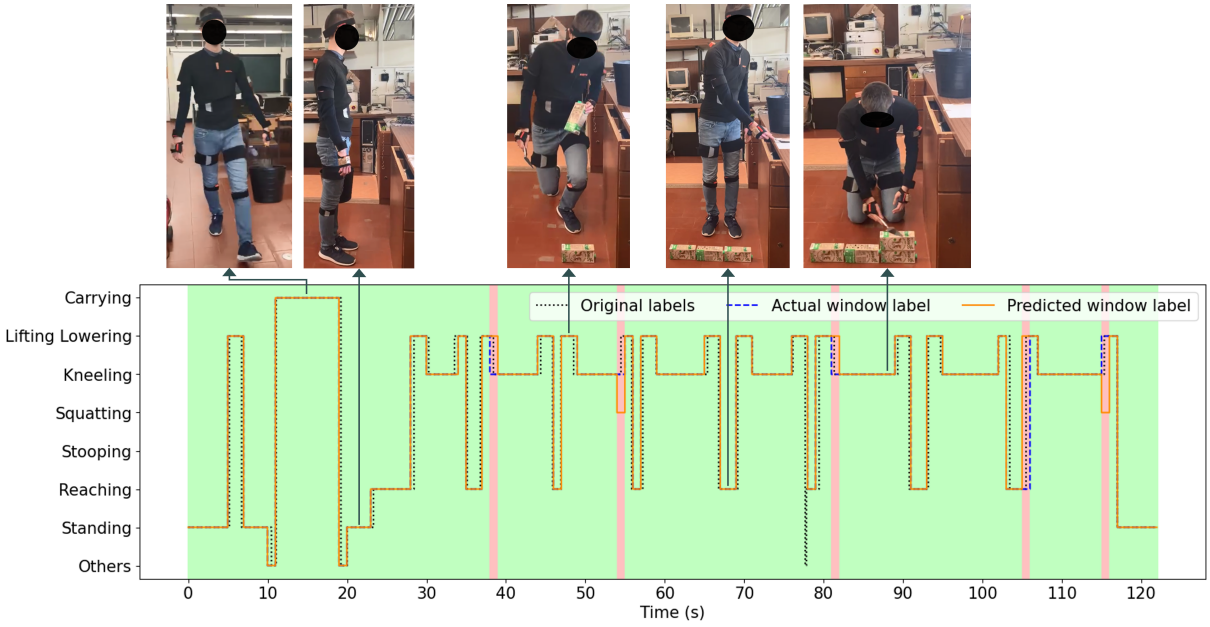


Figure 62: Comparison of the classification output with the actual window labels and the original sample labels in a construction circuit. Green and red stripes highlight, respectively, correct and incorrect classifications by the CNN-TR 1.

Grad-CAM for CNN Explainability

To interpret the CNN models' decisions, Figure 64 presents Grad-CAM visualizations for two test windows of the *stooping* class using CNN 1, together with the corresponding softmax probabilities and labels. Figure 64a shows that the sternum sensor contributed most strongly to correct identification of *stooping*. Figure 64b illustrates a typical misclassification during a posture transition between *stooping* and *lifting/lowering*. Although the sliding window assigns the majority label, both postures are present and

Table 28: Test metrics per class for the hybrid CNN-TR 1

Class	**OT**	**N**	**RE**	**ST**	**SQ**	**KN**	**LL**	**CA**
Accuracy (%)	85.38	98.74	91.85	96.44	98.89	96.76	91.62	93.41
F1-Score (%)	86.05	98.17	94.13	96.32	98.20	98.10	89.92	93.77

OT: *others*; N: *standing*; RE: *reaching*; ST: *stooping*; SQ: *squatting*;

KN: *kneeling*; LL: *lifting/lowering*; CA: *carrying*.

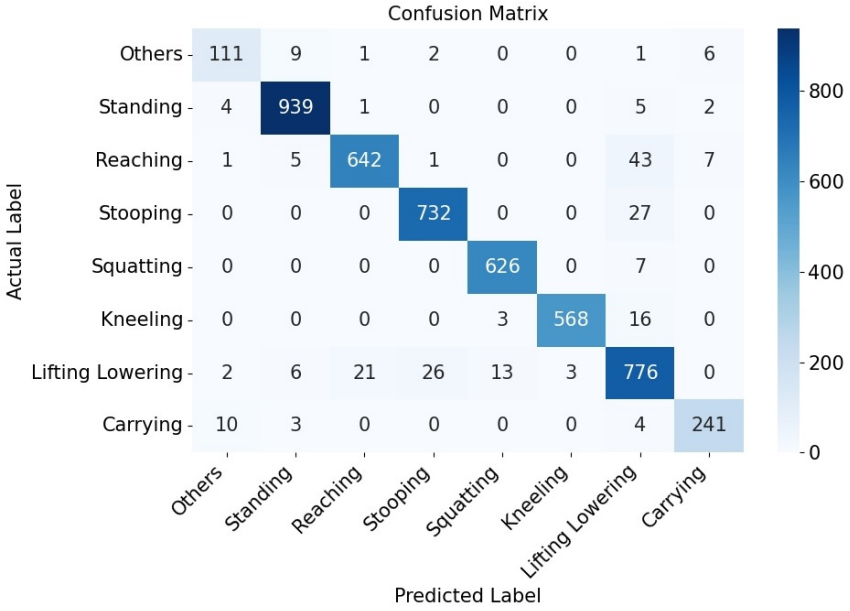
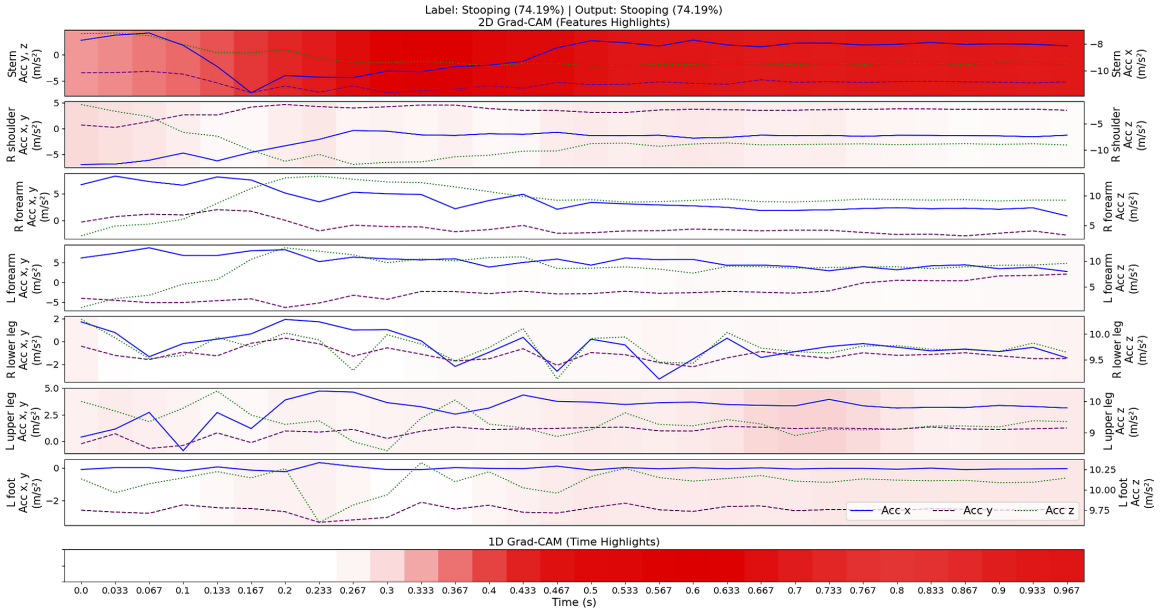


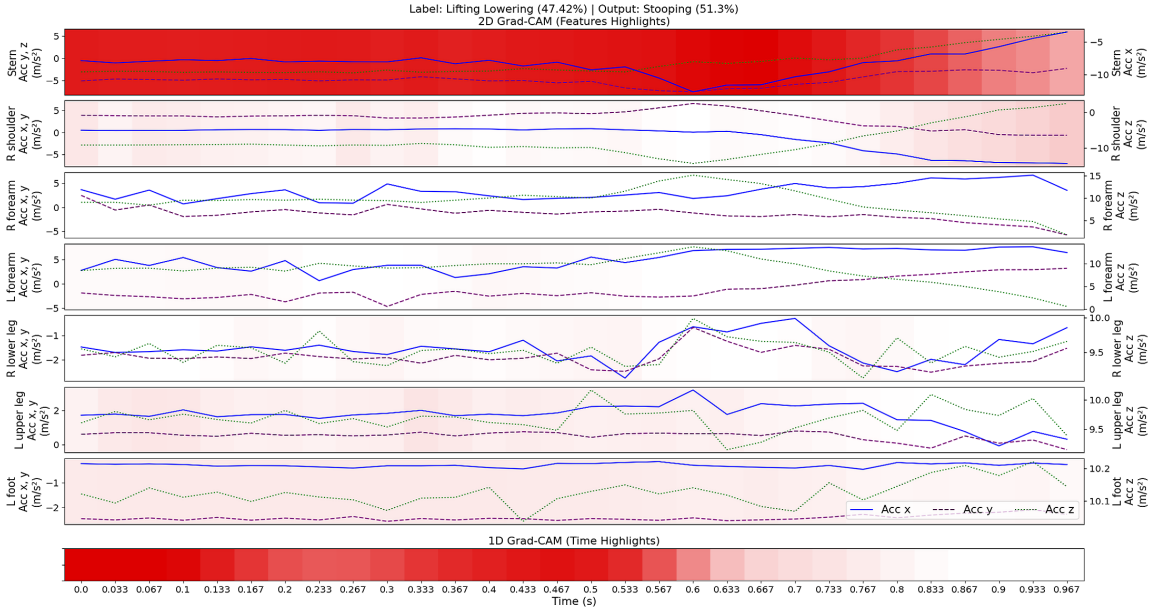
Figure 63: Confusion matrix of the model CNN-TR 1.

yield similar probabilities. In this case, the model focused primarily on earlier time steps, assigning less importance to acceleration changes occurring after $t > 0.6$ s.

Figure 65 illustrates in body maps, for two CNN models, the result of summing the Grad-CAM scores over the time axis and the whole test set (for true positive windows), for every IMU location, which corresponds to the sensor importance. Both global sensor contributions and class-specific contributions are shown. Taking *stooping* as an example again, the sensor that contributed the most to identifying this class was the sternum. Across models, the body segments contributing most to correct classification differed. Some body segments did not contribute much globally but were relevant for identifying a particular class (e.g., the right shoulder was the most important for *squatting* in CNN 1).



(a)

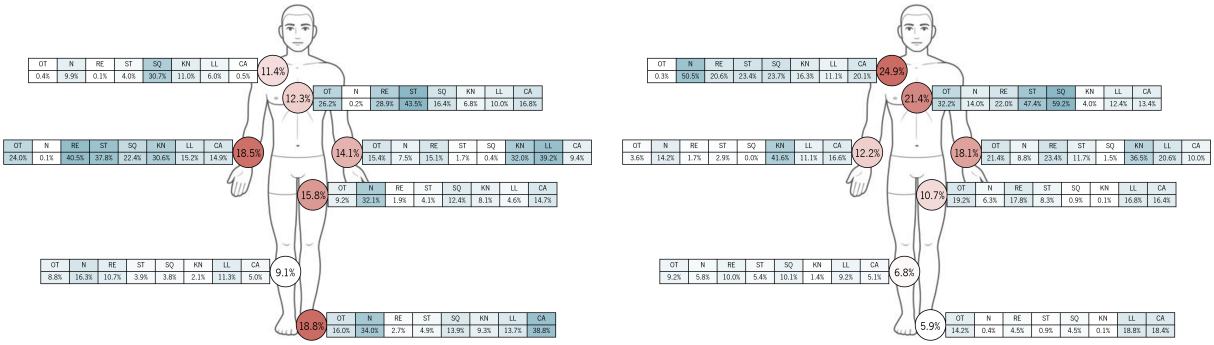


(b)

Figure 64: Grad-CAMs for the class *stooping*. (a) Correct prediction. (b) Misclassified as *stooping* instead of *lifting/lowering*. The redder the pixels, the more important the signal regions are for the model's decision. Only acceleration signals are shown.

4.3.6 Discussion

CNN The observed trends indicate that increasing the number of filters contributes to more stable CNN performance, likely by enabling richer feature representations. This observation is consistent with [226], who reported improved results with higher filter counts and increasing depth. In this study, however, the



(a) (b)

Figure 65: Importance of each IMU considering the Grad-CAMs of all the TP windows in the test set, in (a) CNN 1 and (b) CNN 2. The posture classes are identified as follows: OT - *others*; N - *standing*; RE - *reaching*; ST - *stooping*; SQ - *squatting*; KN - *kneeling*; CA - *carrying*.

best performance resulted from a balanced configuration of five layers and 64 filters rather than maximal depth or width.

Regarding kernel size, the results suggest that smaller kernels, particularly size 3, are well suited for posture recognition from inertial data. Although larger kernels combined with more filters achieved similar median performance, kernel size 3 consistently produced stronger overall metrics. This supports previous findings [226] and indicates that fine-grained temporal feature extraction is advantageous for capturing posture-specific signal patterns.

LSTM The results indicate that increasing the number of LSTM layers beyond two provides limited benefit for posture recognition. The minimal performance difference between two- and three-layer configurations suggests that two layers already offer sufficient model capacity to capture the relevant temporal dynamics. Adding further layers is therefore unlikely to improve performance and may instead reduce generalization capability, as reported in previous work [227].

Transformer The results suggest that richer feature representations, enabled by larger embedding and feed-forward dimensions, are critical for effective transformer-based posture recognition. Increasing model depth improved performance only up to an intermediate number of encoder layers, beyond which additional complexity did not yield further gains, indicating diminishing returns.

The stronger influence of the number of layers observed at higher embedding sizes highlights the interaction between representational capacity and architectural depth: deeper models are beneficial only when sufficient feature richness is available. Similarly, the positive effect of increasing the number of

attention heads suggests that parallel attention mechanisms help capture diverse temporal relationships in inertial data. This behavior is consistent with prior findings [228], which reported improved transformer performance with higher head counts in datasets characterized by greater sample size and intra-class variability.

CNN-LSTM The superior performance of the hybrid CNN–LSTM architecture indicates that combining spatial and temporal feature extraction is beneficial for posture recognition. The consistent advantage of using two recurrent layers suggests that a limited temporal modeling depth is sufficient when paired with a deep convolutional front end.

The analysis of fusion weights highlights the influence of the learning rate on how the hybrid model balances its subnetworks. At higher learning rates, larger weight updates appear to bias the model toward the CNN branch, potentially limiting the contribution of the LSTM component. In contrast, lower learning rates lead to smoother updates, allowing a more balanced contribution from both subnetworks throughout training. This suggests that learning rate selection plays a critical role in stabilizing the fusion mechanism and preserving the intended complementarity between convolutional and recurrent components.

CNN-Transformer The superior performance of CNN–Transformer hybrids indicates that combining convolutional feature extraction with attention-based temporal modeling is effective for posture recognition. The observed advantage of using fewer encoder layers suggests that excessive depth in the Transformer component may introduce unnecessary complexity, which can hinder generalization when combined with a deep convolutional front end.

The analysis of fusion weights highlights the sensitivity of the hybrid architecture to the learning rate. Higher learning rates appear to promote a balanced contribution between CNN and Transformer components, whereas lower learning rates bias the model toward relying predominantly on the Transformer branch. This behavior mirrors observations made for CNN–LSTM hybrids and underscores the importance of optimization dynamics in stabilizing hybrid fusion mechanisms.

Training Hyperparameters and Input Data The results indicate that learning rate primarily affects training stability rather than peak performance. While a lower LR combined with a larger batch size occasionally produced strong results, it also introduced greater variability, suggesting increased sensitivity to optimization dynamics. In contrast, an LR of 10^{-3} promoted more stable convergence across architectures, independently of batch size.

The strong influence of WS highlights the importance of temporal resolution in posture recognition.

A WS of 1 s appears to balance temporal context and data availability, leading to more reliable learning across architectures. Larger windows reduce the number of training samples and may limit the model's ability to generalize, which likely explains the poorer performance observed for WS values of 2 s.

Although higher overlap generally improves performance by increasing the effective dataset size, the observation that several top-performing models used no overlap demonstrates that overlap alone is not a determining factor. Instead, these findings emphasize the importance of comprehensive grid search, as optimal performance depends on the interaction between multiple hyperparameters rather than on isolated choices. Excessive overlap may also reduce data variability, further reinforcing the need for balanced input configurations.

Final Test Results

The systematic reduction in test performance relative to validation suggests limited generalization to unseen subjects. The absence of early stopping during final training and the reliance on a fixed hold-out strategy likely contributed to this effect. Although cross-validation would provide a more robust estimate of generalization, it was not adopted due to the substantial computational cost and the decision to prioritize extensive hyperparameter exploration.

The superior performance of hybrid architectures confirms findings reported for CNN–LSTM models [66] and extends them to CNN–Transformer hybrids. These architectures benefit from combining convolutional spatial feature extraction with temporal modeling through recurrent units or attention mechanisms. The learnable fusion mechanism allows the models to dynamically weight the contributions of each subnetwork, explaining their consistent advantage over convolution-only models. However, this increased performance comes at the cost of higher computational complexity and longer inference times.

The relatively weaker performance of LSTM-only models indicates that temporal modeling alone is insufficient for this dataset. Despite the lack of hyperparameter tuning for the LSTM hidden size, the achieved results still surpass those reported in [66]. Other studies have reported higher metrics [62, 65], but differences in dataset structure, activity segmentation, and evaluation protocols limit direct comparability.

The strong performance of the proposed CNN models relative to prior work [66, 67] highlights the effectiveness of the architectural and training choices adopted in this study, particularly given the inclusion of continuous task circuits.

Transformer models underperformed relative to some prior studies [63, 64], likely due to differences in dataset complexity. Those studies relied on benchmark datasets with segmented activities and limited transitions, whereas the present dataset includes natural posture transitions and higher intra-class

variability. Architectural choices, such as replacing fully connected layers with global average pooling, may have further constrained Transformer performance. Nevertheless, Transformers demonstrated better generalization than CNNs on the test set, consistent with previous comparisons [64, 223].

Class-wise errors were largely driven by class imbalance and transition ambiguity. The *others* class suffered from limited training samples and high movement variability. Confusions involving *kneeling*, *reaching*, and *lifting/lowering* were exacerbated by short posture durations and difficulties in defining precise transition boundaries. Although manual inspection of inertial signals could improve labeling accuracy, the adopted downsampling and sliding-window strategy already mitigates the impact of isolated labeling errors.

Grad-CAM for CNN Explainability

The Grad-CAM analysis confirms that CNN models rely on biomechanically meaningful sensor locations, such as the sternum for trunk-dominated postures like *stooping*. Misclassifications were primarily associated with posture transitions, where overlapping movement patterns reduce temporal separability, particularly for short-duration postures such as *lifting/lowering*. In these cases, the model emphasized earlier posture-consistent signal segments, explaining the observed errors.

The class-specific relevance of sensors highlights the limitation of purely global importance measures and supports retaining sensors that may be critical only for certain postures. While the seven retained body segments were already selected through prior feature selection, Grad-CAM provides complementary insight into sensor relevance from a predictive perspective. This suggests potential for model-driven sensor reduction strategies, as opposed to statistical tests that evaluate signals independently and ignore inter-signal dependencies [216]. Moreover, classical nonparametric tests assume sample independence [229], an assumption that may not hold for temporally correlated inertial data.

Finally, the architectural modifications required to enable Grad-CAM for time-series data may influence model performance. Further comparison with standard 1D convolutional architectures would be necessary to assess whether the gain in explainability justifies potential performance trade-offs.

4.4 Ergo-report

This chapter outlines the role of the posture recognition module within the broader ergonomic assessment framework. For an ergonomic evaluation to be objective and reliable, sensor-based measurements are essential [56]. However, these methods often produce large volumes of data, which can make interpretation burdensome. Therefore, there is a clear need for intuitive and efficient ways to present this information to end users.

Work is structured as sequences of tasks [230], each executed through a series of physical postures. Ergonomic risk emerges from these postures and their temporal accumulation. However, as discussed in Section 2.1, existing methods rarely combine accurate posture classification with quantitative ergonomic assessment.

To address this gap, this work proposes a digital framework that empowers ergonomists, workers, and managers with a clear, accessible, and data-driven analysis of posture-related risk. The framework quantifies instantaneous risk and its accumulation across tasks, enabling the identification of harmful postures and supporting task redesign or posture correction.

The system uses the deep learning models developed in section 4.3 to continuously recognise human postures without handcrafted features. This enables real-time integration of posture recognition and ergonomic scoring, making the analysis scalable and practical.

This study is based on the paper entitled "ErgoReport: A Holistic Posture Assessment Framework Based on Inertial Data and Deep Learning" published at Sensors (2025) [231] and is guided by the following research question: ***How can postural ergonomic risk be continuously quantified and attributed to specific working postures over time in a manner that is both objective and usable for end users?***

4.4.1 Solution Conceptualization

The conceptual design of the proposed system for posture assessment is shown in Figure 66.

Figure 66 presents the conceptual design of the proposed system for posture-based ergonomic assessment and biofeedback. The inertial data, acquired from the Xsens sensory system, serves as the primary input for the system's various modules. It is first processed by the posture recognition module, presented in the previous section, which classifies the user's posture in real time. These classifications are then used in the ergonomic assessment block to estimate the associated risk levels. This information supports a subsequent evaluation of potential joint wear. A graphical user interface (GUI) delivers visual biofeed-

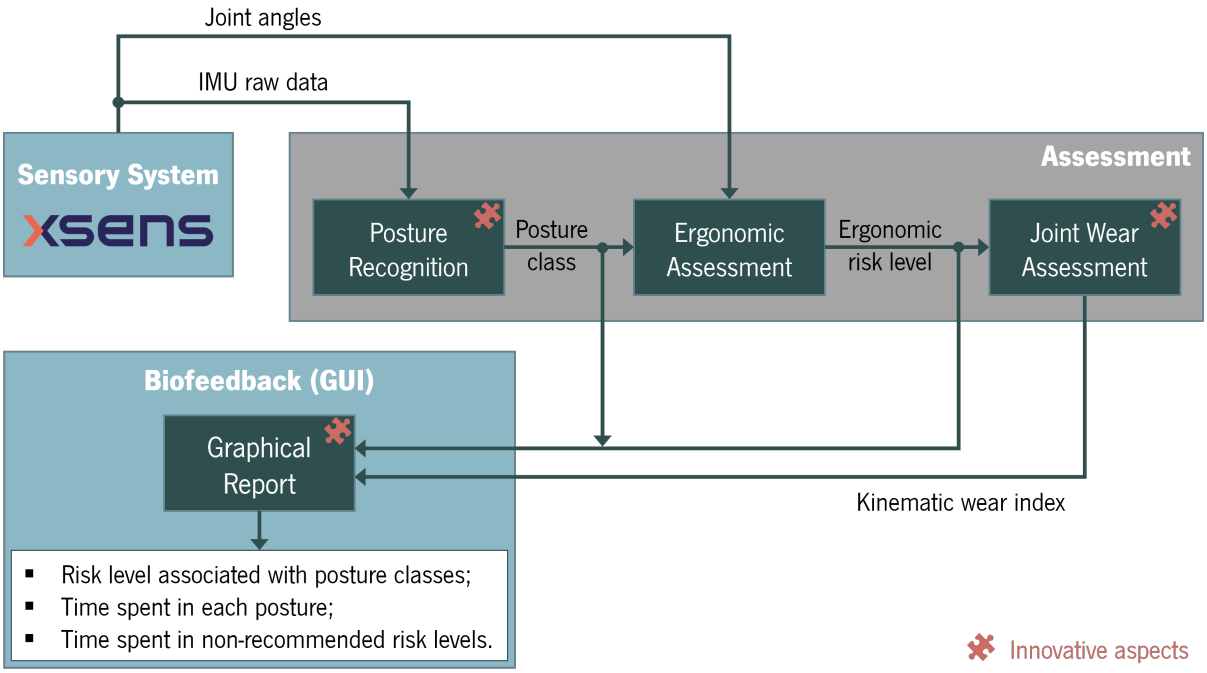


Figure 66: Conceptual design of the system.

back through a report that summarizes the risk level of each posture, the time spent in each posture, and the duration spent in non-recommended risk levels. The posture recognition, joint wear assessment, and graphical reporting components represent innovative contributions of this work.

Ergonomic Assessment

The *ergonomic assessment module* quantifies posture by measuring its deviation from a neutral configuration, defined as body segments aligned with gravity. The module takes joint angles as input and outputs ergonomic risk levels that reflect the distance between the observed posture and the neutral reference. The back, shoulders, and elbows are assessed, as the back and upper limbs are the body regions most affected by WRMSDs [49]. Two ergonomic assessment methods were adapted as evaluation criteria: Agricultural Whole-Body Assessment (AWBA) and LUBA. Users can select either method through the GUI, as they address different ergonomic aspects and provide complementary insights, as detailed in Section 4.4.2.

The AWBA tool [232] integrates the Agricultural Upper-Limb Assessment (AULA) [233] and the Agricultural Lower-Limb Assessment (ALLA) [234]. These methods target postures common in agriculture, such as kneeling and trunk flexion, which also occur in industrial settings. However, they only evaluate sagittal plane motions. Risk levels were defined based on electromyography, heart rate, and self-reported discomfort. Both AULA and ALLA have been shown to estimate risky postures more accurately than REBA,

Table 29: Finite state machine rules for ergonomic assessment using AULA and ALLA

a) AULA, upper body

Back (α)	**Shoulder (β)**	**Elbow (γ)**	**Level**
$\alpha \leq 22.5$	$\beta \leq 22.5$	Any	1
$\alpha \leq 22.5$	$22.5 < \beta \leq 72.5$	Any	2
$\alpha \leq 22.5$	$72.5 < \beta \leq 105$	Any	3
$\alpha \leq 22.5$	$\beta > 105$	Any	4
$22.5 < \alpha \leq 72.5$	$\beta \leq 72.5$	$\gamma \leq 22.5$	2
$22.5 < \alpha \leq 72.5$	$\beta \leq 72.5$	$\gamma > 22.5$	3
$22.5 < \alpha \leq 72.5$	$\beta > 72.5$	Any	3
$\alpha > 72.5$	Any	Any	2

b) ALLA, lower limbs

Posture	**Knee angle (°)**	**Level**
Standing or squatting	$\delta \leq 15$	2
	$15 < \delta \leq 45$	3
	$45 < \delta \leq 135$	4
	$\delta > 135$	3
Kneeling	Any	3

Note: Risk level 1 is not present in ALLA, as it is assigned only to sitting postures, which were outside the scope of this work.

RULA, and OWAS, with higher agreement with expert evaluation [233–236]. AWBA was also applied to construction tasks, as both sectors share similar postural patterns. Since AULA and ALLA were originally observational methods, they define discrete reference angles (e.g. 0°, 45°, 90°, 125°) rather than ranges. In this work, joint angles were partitioned into ranges using thresholds set midway between the original values. Lower limb assessment depends on the *posture recognition module*, as ALLA assigns a fixed risk level to *kneeling* independent of knee angle (Table 29b).

For ergonomic posture analysis, a finite state machine was implemented to discretise sensor derived joint angles of the back, shoulders, elbows, and knees into risk states. Tables 29a and 29b report the state definitions for the upper and lower body, respectively, while Table 30 illustrates the computation of the overall risk level.

Table 30: AWBA rules

		AULA level			
		1	2	3	4
ALLA level	1	1	2	3	3
	2	2	2	3	4
	3	3	3	3	4
	4	3	4	4	4

To capture a wider range of upper-body motions, LUBA [237] was also implemented. Unlike RULA,

which is limited outside the sagittal plane, LUBA defines thresholds for multiple joint motions, including flexion/extension, lateral bending, and axial rotation for the back, and flexion/extension and abduction/adduction for the shoulders. Each motion is assigned a score based on its angular range (Table 31).

Table 31: Finite-state machine for the ergonomic assessment based on the LUBA method

Joint	Joint motion	Angle range (°)	Score
Back	Flexion (+)/extension (-)	$\alpha_{fe} \leq -30$	15
		$-30 < \alpha_{fe} \leq -20$	8
		$-20 < \alpha_{fe} \leq -10$	4
		$-10 < \alpha_{fe} \leq 30$	1
		$30 < \alpha_{fe} \leq 60$	3
		$60 < \alpha_{fe} \leq 90$	6
		$\alpha_{fe} > 90$	12
	Lateral bending	$\lvert\alpha_{lb}\rvert \leq 10$	1
		$10 < \lvert\alpha_{lb}\rvert \leq 20$	4
		$20 < \lvert\alpha_{lb}\rvert \leq 30$	9
		$\lvert\alpha_{lb}\rvert > 30$	13
	Axial rotation	$\lvert\alpha_{ar}\rvert \leq 20$	1
		$20 < \lvert\alpha_{ar}\rvert \leq 60$	3
		$\lvert\alpha_{ar}\rvert > 60$	10
Shoulder	Flexion (+)/extension (-)	$\beta_{fe} \leq -60$	10
		$-60 < \beta_{fe} \leq -45$	6
		$-45 < \beta_{fe} \leq -20$	3
		$-20 < \beta_{fe} \leq 45$	1
		$45 < \beta_{fe} \leq 90$	3
		$90 < \beta_{fe} \leq 150$	6
		$\beta_{fe} > 150$	11
	Abduction (+)/adduction (-)	$\beta_{aa} \leq -30$	8
		$-30 < \beta_{aa} \leq -10$	2
		$-10 < \beta_{aa} \leq 30$	1
		$30 < \beta_{aa} \leq 90$	3
		$\beta_{aa} > 90$	7
Elbow	Flexion (+)/extension (-)	$\gamma_{fe} \leq 45$	1
		$45 < \gamma_{fe} \leq 120$	3
		$\gamma_{fe} > 120$	5

To provide a broader assessment that represents the overall ergonomic risk of the task, besides each joint separately, the entire upper-body posture was also quantified, with a global LUBA score that is the sum of each joint motion score when it is above 1 (the minimum score), as expressed by Equation 4.7:

$$G = \sum_{i=1}^{n} \sum_{j=1}^{m_i} S_{ij} \tag{4.7}$$

where G is the global LUBA score, i denotes the joint (Table 31), j the joint motion (Table 31), n the

number of joints considered (5), m_i the number of joint motions for joint i (3 for the back, 2 for each shoulder, and 1 for each elbow), and S_{ij} the corresponding LUBA score. The global score considers only one arm. Since both arms are involved in the analysed activities, the assessment was performed using the arm with the higher combined shoulder and elbow score at each time step. The resulting scale ranges from 0 to 62.

Following LUBA guidelines, four posture categories regarding the need for corrective actions were defined, according to the calculated G score, as Table 32 clarifies.

Table 32: Posture categories regarding the need for corrective actions defined by the LUBA method

Category	Global score	Corrective actions
I	$G \leq 5$	Not needed
II	$5 < G \leq 10$	In the next regular check (immediate intervention not needed)
III	$10 < G \leq 15$	Redesigning workplaces or working methods soon
IV	$G > 15$	Immediate

Note: to make it more elucidating, colors green, yellow, orange and red are associated with each category from I to IV, in ascending order.

Joint Wear Assessment

Although the ergonomic scores described above provide an estimate of instantaneous risk, they were primarily conceived for discrete assessments. A key limitation of conventional ergonomic methods is their neglect of cumulative exposure, implicitly assuming that past postural risk does not affect the current ergonomic state. To address this limitation, this work adopts the kinematic wear index introduced in [23], which is inspired by the RC circuit-like behaviour of muscle fatigue and enables a more realistic assessment. While the model neglects overloading forces, it retains memory of the time spent in hazardous postures and of posture repetition. The kinematic wear index reflects both current and past postural risk by accumulating ergonomic scores for each joint and motion type during hazardous postures (wear phase), representing stress accumulation, and decreasing during neutral postures (recovery phase). The index ranges from 0, corresponding to an ergonomically ideal condition, to 1, representing the worst case. In this work, unlike [23], each joint motion is assessed independently, and its kinematic wear increases whenever the

associated ergonomic score exceeds the minimum value (wear phase), as formalised in Equation 4.8.:

$$V_{ij}(t) = 1 - (1 - V_{ij}(t_0))e^{-\int_0^t \frac{S_{ij}(q(\tau))}{C_{ij}}\, d\tau} \tag{4.8}$$

where $V_{ij}(t) \in [0, 1]$ represents the kinematic wear index of the joint i for the motion j (from Table 31) at instant t; $V_{ij}(t_0)$ is the kinematic wear index at the instant t_0, where the ergonomic score level ceased to be the minimum; $q(\tau)$ is the joint configuration; $S_{ij}(q(\tau))$ is the ergonomic score associated with that angle, in this case, according to LUBA; and C_{ij} the endurance capacity for that joint motion. When the joint is at the minimum risk level, it is assumed to be in recovery, and the kinematic wear index V_{ij} decreases according to Equation 4.9:

$$V_{ij}(t) = V_{ij}(t_0)e^{-\frac{r_{ij}}{C_{ij}}t} \tag{4.9}$$

where $V_{ij}(t_0)$ is the kinematic wear index at the instant t_0, where the ergonomic score became the minimum; and r_{ij} is the recovery rate associated with that joint motion.

The endurance capacity, C_{ij} is given by inverting Equation 4.8 considering $V_{ij}(t_0)$ = 0, $V_{ij}(t)$ = V_{max} = 0.993 (which corresponds to the kinematic wear at the time step of five time constants, when a capacitor in an RC circuit is typically considered to be fully charged), the average LUBA score $S_{ij,avg}$ of the motion j of joint i (calculated between the minimum and the maximum of the scale), and a T_{max} = 240 s (corresponding to the time until a subject applying a low force feels physical discomfort, in a static configuration, as stated by [238]). Differently from [23], in our work, the endurance capacity was defined for each joint motion individually, as follows:

$$C_{ij} = -S_{ij,avg}\frac{T_{max}}{\ln(1 - V_{max})} \tag{4.10}$$

Regarding recovery rate, r_{ij}, also specific for each joint motion, Equation 4.11 is obtained by inverting Equation 4.9 in order to match the recovery time with the wear time, i.e., a decrease of five time constants, starting at $V_{ij}(t_0) = V_{max}$:

$$r_{ij} = -\frac{C_{ij}}{T_{max}} \ln\left(\frac{1 - V_{max}}{V_{max}}\right) \tag{4.11}$$

Figure 67 exemplifies how the index changes over time, during wear and recovery phases, respectively.

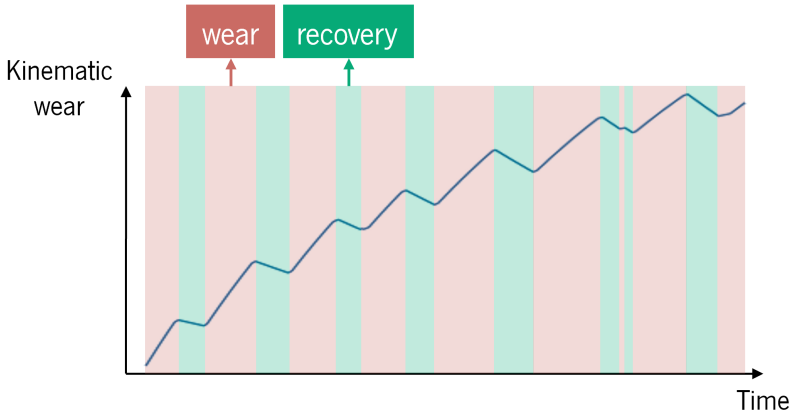


Figure 67: Behaviour of the joint kinematic wear index, V_{ij}, calculated using Equation 4.8 during wear phase and using Equation 4.9 during recovery phase.

This index enables the definition of risk levels that account for cumulative postural load. Two levels were defined using a threshold of 0.7, following [2]. Values below 0.7 indicate low risk and do not require posture adjustment. Values above 0.7 indicate high risk, reflecting sustained exposure to non-neutral postures and requiring a transition to a neutral configuration.

The percentage of time above the threshold was computed to quantify the duration of excessive cumulative load. This kinematic wear model could also be integrated into a real time biofeedback strategy, where exceeding the threshold triggers biofeedback cues, namely haptic cues.

Report for Biofeedback

The ErgoReport framework was designed to support ergonomists in automated and comprehensive posture assessment. An unstructured interview with two ergonomists identified key challenges in postural risk evaluation (Table 33). The recognised postures and the computed ergonomic metrics are integrated into a report that explicitly links each posture to its associated risk score. To the best of our knowledge, this is the first approach that simultaneously performs posture classification using deep learning and assigns a quantitative ergonomic score. The computed information is presented through several representations in the ErgoReport:

- A temporal visualisation of ergonomic risk and, for LUBA only, the kinematic wear index for each joint, enabling analysis of how individual ergonomic scores contribute to joint stress.
- A single time series plot combining the sequence of recognised postures and their corresponding ergonomic scores. This representation acts as a playback of the worker's shift and is particularly useful for analysing time intervals associated with high ergonomic risk.

- A shift level summary reporting the time spent in each posture and the average ergonomic score per posture class. This enables a straightforward visualization of the most hazardous posture classes, which require attention. It is relevant to show these two aspects jointly, as certain postures may have a lower average risk than others, but they may represent a larger time percentage. The report also identifies joint motions exceeding the kinematic wear threshold (0.7) for the longest duration. By computing these metrics both per subject and across all subjects, managers can determine whether ergonomic risk is driven by the task or by individual postural behaviour.

Implementation Details

The complete pipeline was implemented in a Python environment using the models developed in Section 4.3. The flowchart of the proposed framework is depicted in Figure 143, in appendix E. Although the framework was designed to receive data one sample at a time, enabling real-time operation and future integration of haptic biofeedback, in this study, all modules were executed offline, with the full dataset loaded and processed at once. Note that, for the LUBA ergonomic assessment, the posture class is only required for report generation, while for AWBA it is also necessary for ergonomic risk computation. Both the *posture recognition* and *ergonomic assessment* modules operate at 20 Hz; however, posture recognition processes the data in non overlapping 1 s windows, while the ergonomic assessment metrics are computed on a per sample basis. The posture recognition module uses the CNN–LSTM 1 architecture, selected as the best-performing model in Section 4.3 (Table 27). Since posture information is required only for report generation, the posture recognition module is executed after the end of each trial, even in the envisioned real time ergonomic assessment setup.

Figure 68 shows one of the pages of the developed framework's GUI, where the user can upload the data acquired by the wearable sensors and select the type of analysis from the ones mentioned in Section 4.4.1.

Report Usability Assessment To assess ErgoReport's usefulness and usability, ergonomists were invited for an online interview where we presented the system's goal and the use cases. A demonstration of how to use the GUI, with some example trials, was conducted.

At the end, the ergonomists were asked to fill out a 7-point Likert scale questionnaire (1: strongly disagree; 7: strongly agree). This custom questionnaire (Table 34) addresses three main aspects of the report — its usefulness, ease of use and usability — and was adapted from three different questionnaires: Perceived Usefulness and Ease of Use (PUEU) [239], Interface Usability Instrument (INUIT) [240], and

Table 33: Typical ergonomists' difficulties in postural risk assessment and the corresponding ErgoReport's solution to address them

Ergonomists' difficulty	**Framework's solution**
Visualisation of the ergonomic risk along the work shift	Plot showing the temporal evolution of LUBA joint scores and the corresponding kinematic wear index, using colour coding to indicate whether accumulated joint stress is below (green) or above (red) the recommended maximum (Figure 69)
Realising the sequences of postures performed by the workers throughout the shift	Plot with the sequences of the ergonomic scores and performed postures over time, coloured in green, yellow, orange, and red according to LUBA posture categories to indicate corrective actions; analogous colours are used for AWBA scores (Figures 70 and 144)
Realising which postures led to the highest ergonomic risks along the work shift	Analysis of the posture class immediately preceding high-risk scores, highlighting transitions and their impact on cumulative joint load
Realising which postures led to the highest ergonomic risks overall	Graph of the average ergonomic score associated with each posture class, and indication of the joint motions that surpassed the recommended maximum kinematic wear for longer cumulative durations (Figure 71 and Figure 145 in appendix)
Knowing the time each posture was held	Pie chart showing the proportion of time spent in each posture, averaged across subjects or recordings (Figure 71 and Figure 145 in appendix)

Software Usability Measurement Inventory (SUMI) [241]. With this custom questionnaire, new specific directions for improvement can be pointed out by ergonomists, who are end users.



Figure 68: GUI of the developed framework, showing the page with the options to upload the acquired data and perform the posture assessment.

4.4.2 Results

The kinematic wear index was computed for nine joint motions. Figure 69 presents one of the ErgoReport's graphs, with the kinematic wear of one of the most concerning joint motions, the back flexion/extension. Three different cases are exposed, refering to three trials of construction tasks from the same subject, one using *stooping* posture for bricklaying (Figure 69a), and the others using *squatting* (Figure 69b) and *kneeling* (Figure 69c), starting with zero as initial conditions. These figures reflect the temporal dynamics of the tasks performed. In the three cases, as represented in frame B in Figure 69c, the *carrying* of the bucket took place until $t \approx 20$ s, when the subject arrived at the bench (frame C). This did not require any back flexion; thus, the back kinematic wear index did not increase much. It only did when grabbing the bucket from the ground in the beginning, in frame A. Still, the kinematic wear index remained negligible until $t \approx 25$ s. After that, risky postures started to be adopted frequently, first when the subject reached the mortar in the tray (frame D) and then when he headed to the floor to spread it, achieving ergonomic scores above the minimum in those instants, which translated into an increase in the back kinematic wear index. At $t \approx 35$ s in the trial of Figure 69c, when the subject got up to reach a milk carton (frame E), he straightened his back, which led the LUBA score to the minimum (1) and, consequently, induced a decrease (recovery) in the back kinematic wear regarding flexion/extension motion. Then, the participant

lowered (frame F) for *kneeling* to spread the mortar on the milk cartons and lay them on the "wall" (as in frames G and H), which required considerable back flexion, which, for *kneeling*, typically leads to a LUBA score of 3 or 6. With this, the joint kinematic wear escalated every time the subject had to place the simulated brick on the "wall" and decreased every time the participant got up to pick another carton — during this time, the subject's back was able to recover.

Note, in Figure 69a, that, during the last fifth (after t = 90 s) of the trial where *stooping* was used for bricklaying, the back kinematic wear was above the defined threshold of 0.7 for the recommended maximum accumulation of joint stress. Other joints also surpassed that value, for instance, the shoulders, regarding flexion/extension. This resulted from the accumulation of joint stress when the subjects were in the *stooping* posture, where the shoulder flexion angle was above 90°. Accordingly, back and shoulder flexion/extension are the joint motions that should require more attention from the workers.

Association between Ergonomic Risk and Postures

The report associates the ergonomic assessment with the postures recognized by the framework. In a continuous way over time, Figure 70 shows that the highest LUBA scores occurred in the moments of *stooping*. Figure 70b shows, e.g., LUBA score peaks during *stooping* repetitions, which correspond to the act of placing the milk cartons on the floor and spreading mortar on them, where the subjects have to bend the back, almost always corresponding to the most concerning LUBA category (IV) and achieving a maximum global score of 43 out of 62 for that subject specifically. In comparison, *squatting* postures in Figure 70a were associated more often with category II (global score between 5 and 10).

Figure 71 displays an overview of the tasks carried out, encompassing all trials from all subjects, for the agriculture and construction task circuits separately. It transmits information regarding the average percentage of time spent at each posture, as well as the average ergonomic score (here, the global LUBA score) for each posture, and the three joint motions that breached the kinematic wear threshold for more time. *Lifting/lowering* was the posture that was held more time, since the protocol, primarily designed to train the DL models, included a huge number of transitions. Both harvesting and bricklaying tasks were associated with a considerable global ergonomic score. For harvesting, the overall was 8, which, according to LUBA guidelines, would mean no need for immediate corrective actions. For bricklaying, the overall was 10, meaning the need for corrective actions soon. Only *standing* (neutral posture), *carrying* and *others* were free of risk. *Stooping* was the posture with the highest risk score in both task circuits.

For clarity of presentation, only the LUBA based results were included in the main body of the chapter, while the corresponding AWBA based analyses were moved to the appendix due to space constraints.

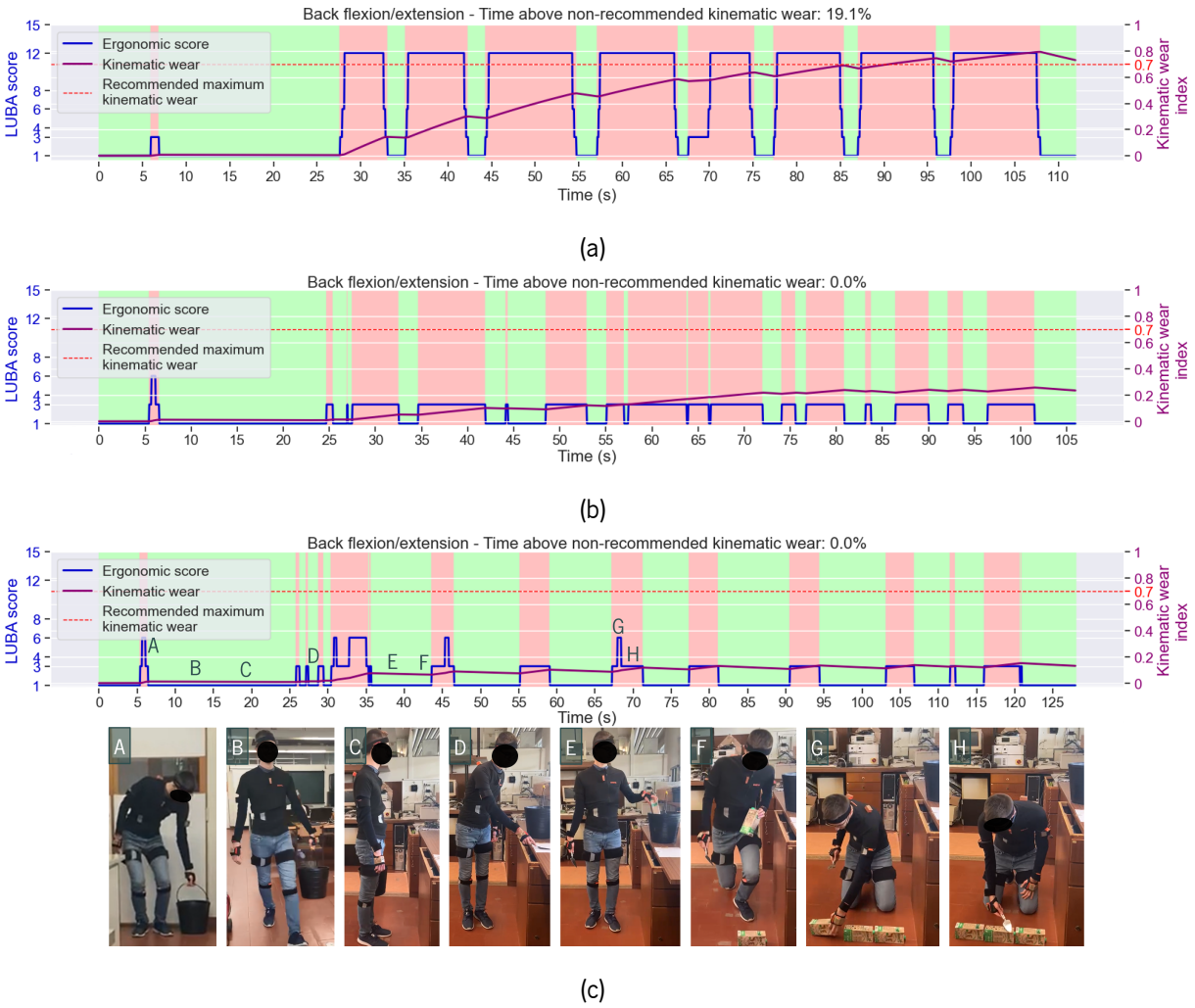
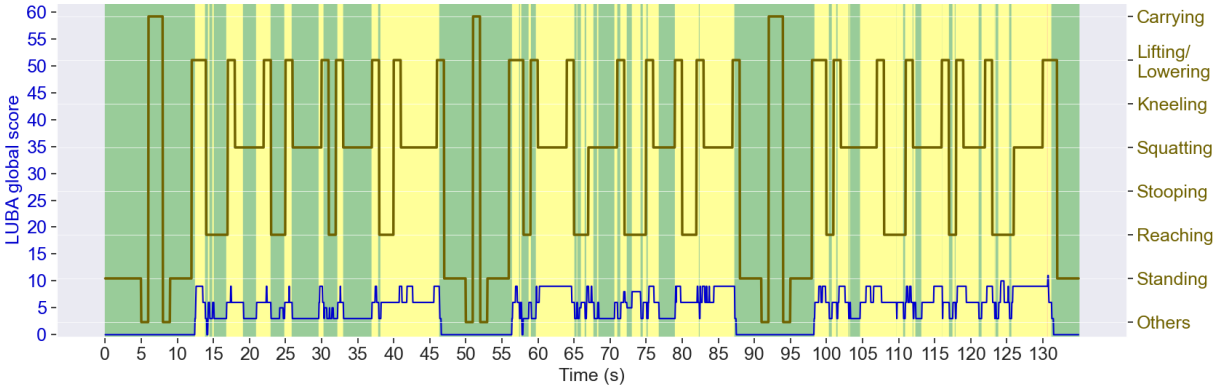


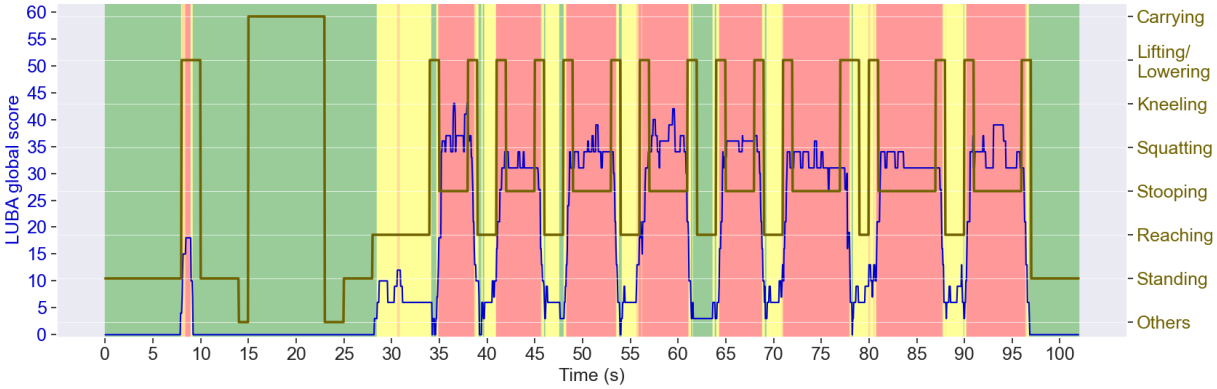
Figure 69: Back flexion/extension LUBA scores and kinematic wear during the circuit based on the construction task, adopting (a) *stooping*, (b) *squatting*, or (c) *kneeling* postures for bricklaying. Red and green stripes highlight, respectively, wear and recovery phases. The red dashed line indicates the kinematic wear threshold of 0.7, suggested by [2]. To illustrate the trials, some frames from the latter example's recording are presented and identified in the plot.

The figures follow the same structure and visual representation, differing in the ergonomic assessment method applied (AWBA versus LUBA). Figure 144, reported in the appendix, presents the AWBA based ergonomic assessment for the same trials shown in Figure 70. When using AWBA, noticeable differences emerge, particularly in the relative risk assigned to postures. In contrast to LUBA, the *squatting* posture is associated with a higher ergonomic risk than *stooping*.

Figure 145, also reported in the appendix, presents the corresponding summary report based on the AWBA assessment. When compared with Figure 71, AWBA generally assigns higher risk levels to the evaluated postures. This behaviour is largely due to the ALLA component of AWBA, which assigns the



(a)



(b)

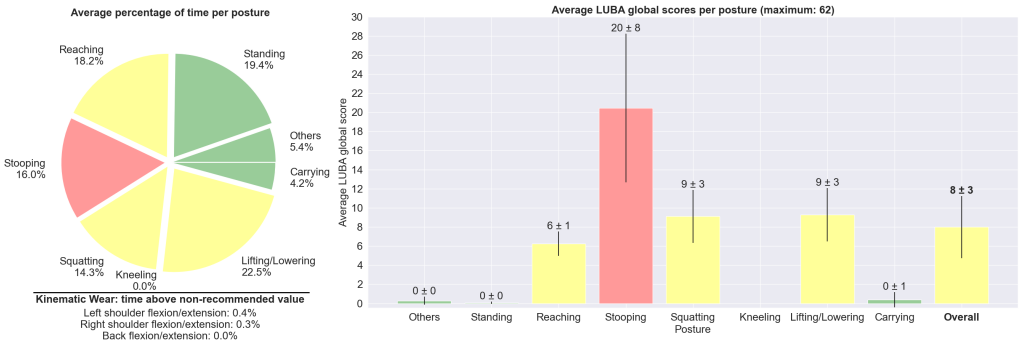
Figure 70: Global LUBA scores (on a scale from 0 to 62) and postures adopted during (a) an agriculture and (b) a construction trial. The background colors refer to the categories regarding the need for corrective actions, defined in Table 32.

lowest risk level only to sitting postures, a condition not considered in this study. Consequently, only the *standing* posture was classified with a risk level below 3. Consistently with the results in Figure 144, the *squatting* posture exhibits the highest average ergonomic risk.

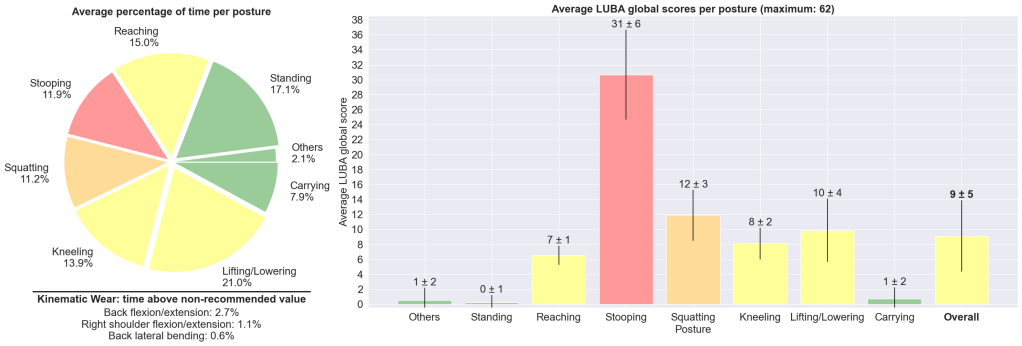
Report Usability and Usefulness

Seven ergonomists were invited to participate in the report usability study. Of these, five agreed to participate. However, one was subsequently unable to participate, and another one was interviewed but was not able to further fill out the questionnaire. Hence, four ergonomists participated in the study but only three completed the questionnaire.

According to the results of the custom questionnaire, in Table 34, the ergonomists agreed that the postural ergonomic report would be useful (Q6) in increasing their productivity (Q1, Q3, Q5) and performance (Q2, Q4) and it would be easy to use (Q7-Q9). They found the desired information (Q15) and were able



(a)



(b)

Figure 71: Summary report with the average percentage of time spent at each posture, averages of the mean LUBA scores (on a scale from 0 to 62) for each posture and 3 joint motions that breached the kinematic wear threshold of 0.7 for the longest time, considering all (a) agriculture and (b) construction trials. Note that the time percentages do not have to sum to 100%, as these are the averages of all trials. The colours refer to the categories regarding the need for corrective actions, defined in Table 32.

to understand it easily (Q10), although some suggested that other options could be added, such as the display of the joint angles over time. Some of them considered that the presentation of the information and the layout were a little confusing (Q11, Q12).

4.4.3 Discussion

This study addressed the problem of continuously quantifying postural ergonomic risk and attributing it to specific working postures over time, while ensuring objectivity and usability for end users. The proposed framework combines posture recognition, joint angle based ergonomic scoring, and cumulative risk modelling to move beyond discrete ergonomic assessments and towards a time aware, posture specific analysis.

A central contribution is the use of the joint kinematic wear index to model the accumulation of er-

Table 34: Results of the 7-point Likert scale questionnaire for ErgoReport's usability assessment

Adapted from	Category	#	Question	Mean	STD
PUEU [239]	Usefulness	1.	Using the system in my job would enable me to perform the postural ergonomic assessment more quickly.	6.00	0.82
		2.	Using the system would improve my job performance, by providing more information and types of postural ergonomic assessment.	6.33	0.94
		3.	Using the system in my job would increase my productivity, since it automates the postural ergonomic assessment.	5.67	0.94
		4.	Using the system would enhance my effectiveness on the job, by providing more information about the workers' postures.	6.00	0.82
		5.	Using the system would make it easier to do my job.	5.67	1.25
		6.	I would find the system useful in my job.	6.33	0.47
	Ease of use	7.	Learning to perform the postural ergonomic assessment with the system would be easy for me.	5.33	1.25
		8.	It would be easy for me to become skilful at using the system.	5.67	0.94
		9.	I would find the system easy to use.	5.67	0.94
INUIT [240]	Usability	10.	I could easily understand the provided information.	5.67	0.94
		11.	I was not confused using the interface/graphical information.	4.67	0.47
		12.	I was not distracted by elements of the interface/graphical information.	4.00	0.82
		13.	Typography and layout added to readability.	5.00	0.82
		14.	The information presented was adequate for the space in the interface.	5.67	0.47
		15.	My desired information was easily reachable.	5.67	0.94
SUMI [241]		16.	I would recommend this software to my colleagues.	6.00	0.82
		17.	I would like to use this software every day.	5.67	0.94

STD: standard deviation.

gonomic risk over time. Conventional ergonomic scores, such as LUBA and AWBA, provide instantaneous estimates of postural risk but do not account for the duration or repetition of hazardous postures. By contrast, the joint kinematic wear model introduced in [23] distinguishes wear phases, during which joint stress accumulates when the ergonomic score exceeds the minimum, from recovery phases, during which stress decreases when the posture returns to a neutral configuration. This enables a continuous representation of joint usage that captures both current and past exposure.

The results show that joint stress can accumulate rapidly even over short task durations. As illustrated in Figure 71, back and shoulder flexion and extension motions exceeded a kinematic wear threshold of 0.7 within approximately two minutes, with the back joint spending 3.3% of the time above this threshold during construction task circuits. This highlights the limitation of relying solely on instantaneous ergonomic scores, which fail to reflect cumulative exposure. Moreover, the model differentiates the effect of distinct ergonomic risk levels. In trials dominated by *stooping*, higher LUBA scores resulted in steeper increases in the kinematic wear index, indicating faster stress accumulation. This behaviour is consistent with Equa-

tion 4.8, which links higher ergonomic scores to increased wear rate. Recovery dynamics also followed the model assumptions, with faster recovery observed when joints started from higher accumulated stress levels, as described by Equation 4.9. For instance, in Figure 69a, recovery beginning at $t = 108$ s with a kinematic wear index of approximately 0.8 is noticeably faster than recovery starting at $t = 33$ s with an index below 0.2.

These results support the suitability of the kinematic wear model for describing joint usage over time. However, as discussed in [23], the model remains a simplified representation of joint stress accumulation and lacks direct physiological validation. In particular, given the short duration of the experimental trials compared with a typical work shift, the rapid exceedance of the wear threshold suggests that the adopted value, following [2], may not be optimal. Thresholds could be made adaptive to posture type or task demands, especially for work activities that inherently require higher ergonomic risk postures. Further investigation is therefore required.

Beyond quantifying accumulation, the framework enables the attribution of ergonomic risk to specific postures. By associating each time step with both the recognised posture and the corresponding ergonomic score, it was possible to analyse posture specific risk profiles. The ErgoReport showed that *stooping* was consistently associated with the highest ergonomic risk in both task circuits, reflecting its large back flexion angles and frequent assignment of the highest LUBA scores. In contrast, *squatting* and *kneeling* were associated with substantially lower risk, particularly *kneeling*, which suggests that these postures may constitute safer alternatives for certain tasks. These findings are consistent with the joint specific kinematic wear patterns observed in Figure 69.

Differences between ergonomic methods further illustrate the importance of posture attribution. While LUBA identified *stooping* as the most hazardous posture, AWBA assigned the highest risk to *squatting*, due to the explicit consideration of lower limb postures through ALLA. Although *stooping* typically involves greater trunk flexion, its knee angles are often classified at lower risk levels. This behaviour aligned with subjects' perceived exertion, suggesting that AWBA may better capture subjective load in tasks involving lower limb engagement. However, AWBA exhibited limitations in dynamic activities such as *carrying*, where rapidly changing leg postures led to oscillating risk levels. Additionally, the minimum risk level in this study was constrained to values above 1, since sitting postures were not included, indicating that adaptations of AWBA would be necessary for broader occupational use.

Usability and interpretability were addressed through the design of the ErgoReport and its evaluation by ergonomists. Feedback indicated that the tool provides clear benefits by accelerating ergonomic assessments and offering objective, posture resolved data. While the interface was generally considered intuitive

after brief explanation, some visualizations were initially perceived as complex. In particular, combining kinematic wear and ergonomic scores in a single plot and displaying posture sequences over time required clarification. Nevertheless, summary visualizations aggregating time spent per posture and average ergonomic risk were consistently highlighted as valuable, especially due to their ability to provide a rapid overview of task level ergonomic performance. Color coding was also considered effective in supporting risk interpretation. Suggestions such as replacing posture labels with representative images provide clear directions for improving comprehensibility.

Despite these contributions, several limitations remain. The experiments were conducted with volunteers simulating agricultural and construction tasks, and validation in real workplace environments is required. The kinematic wear model, although inspired by muscle fatigue behavior, lacks physiological validation and should be correlated with muscle activity measurements such as sEMG. The threshold values adopted for kinematic wear may not be suitable for long duration tasks and could be adapted according to posture type or task demands. Future work should also consider user-specific risk assessment by incorporating individual and psychosocial factors known to influence WRMSDs. Finally, the current implementation relies on a commercial motion capture system, which is intended to be replaced by the Ergowear smart garment, enabling a fully wearable, low-cost, and real time ergonomic assessment pipeline.

4.5 Conclusion

This chapter presented a set of computational methods designed to transform inertial sensor data into meaningful representations of human motion and ergonomic risk. Building upon the sensing technologies introduced in the previous chapter, the focus shifted from data acquisition to algorithmic interpretation, addressing the need for objective, continuous, and interpretable ergonomic assessment methods that can operate on wearable data streams.

The first contribution of this chapter is the investigation of data driven approaches for joint angle estimation from inertial measurements. By framing inertial pose estimation as a learning problem, this work demonstrated that human kinematics can be inferred directly from IMU data while reducing dependence on complex biomechanical models, calibration routines, and proprietary sensor fusion pipelines. This contribution establishes a flexible computational basis for downstream posture and ergonomic analysis, particularly suited for scalable and wearable systems.

The second contribution concerns the development and evaluation of learning based models for human activity recognition, with a specific focus on posture recognition. By leveraging temporal deep learning ar-

chitectures, the proposed methods enabled robust classification of work related postures from inertial data, including during dynamic transitions. This step is critical for contextualizing joint kinematics and linking continuous motion patterns to discrete posture classes that are meaningful in ergonomic assessment.

The third contribution integrated joint angle estimation and posture recognition into a unified ergonomic assessment framework, culminating in the development of the ErgoReport. Recognised postures and estimated joint angles were combined with established ergonomic scoring methods, and cumulative metrics such as the joint specific kinematic wear index were introduced to account for exposure over time. This integration moves beyond instantaneous or snapshot based ergonomic evaluation by enabling the quantification of accumulated postural load, the attribution of ergonomic risk to specific joint motions and postures, and the generation of comprehensive reports that are interpretable by ergonomists and safety practitioners. The results showed that cumulative assessment reveals risk patterns that are not captured by conventional discrete scoring alone.

The contributions presented in this chapter were developed and validated using inertial data acquired from both a commercial sensing system and a previous version of the Ergowear prototype, the latter being used specifically for the development of the deep learning based pose estimation methods. The commercial system was intentionally selected to decouple algorithmic performance from sensing quality and to accelerate methodological development, ensuring that the reported results primarily reflect the effectiveness of the proposed computational approaches rather than hardware specific constraints.

The practical outcome of this chapter is a set of validated computational building blocks for human motion and ergonomics analysis, including joint angle estimation, posture recognition, and cumulative ergonomic risk assessment. These methods form a coherent algorithmic layer that bridges wearable sensing technologies and higher level human state understanding, and they provide a foundation for real time feedback, decision support, and adaptive assistance strategies.

Future work should focus on several practical improvements and integration challenges: (i) improving inertial pose estimation by exploring alternative deep learning architectures and training strategies to increase robustness across movements, sensor placements, and user variability; (ii) adopting the Ergowear system as the primary sensing platform for the complete processing pipeline, enabling end to end validation on a custom wearable system; (iii) validating the proposed posture recognition and ergonomic assessment framework in real industrial environments, including longer work shifts and a broader range of occupational tasks; (iv) physiologically validating the kinematic wear model through its association with muscle activity measurements, in order to assess its relevance as a proxy for joint stress accumulation.

Part II

Human-care Control Strategies

Chapter 5

Ergoaware Framework Overview

5.1 Introductory Insight

This chapter describes the design and functionalities of the Ergoaware framework. Ergoaware was started to be developed during the Factory of the Future (FoF) project, which define some of its based functionalities.

5.2 Conceptual Design and Functionalities

ErgoAware is a human–robot collaboration framework organized as a layered architecture that integrates multimodal human sensing, high-level human-aware control, and robot motion execution. The framework is implemented within the Robotic Operating System (ROS) Noetic middleware and runs on Ubuntu 20.04. It follows a modular and interoperable software design with a clear separation between sensing and processing components, enabling independent development and straightforward extension. Figure 72 provides an overview of the implemented framework.

At a high level, the framework is structured into two main blocks: the sensing systems and the processing station.

The sensing block acquires complementary information about the human operator and the physical interaction with the robot. Upper-body motion is captured using the Xsens MVN Awinda system, while muscle activity is measured through Delsys Trigno Avanti sEMG sensors. In parallel, the collaborative robot UR10e (Universal Robots, Odense, Denmark) is integrated as both an actuator and a sensing device, providing end-effector wrench measurements through its internal force–torque sensor at 500 Hz while enabling closed-loop control via the ROS URDriver over TCP/IP.

Human motion and muscle activity data are acquired on a dedicated Windows host computer using proprietary software. Upper-body kinematics are streamed from the Xsens MVN Awinda system through

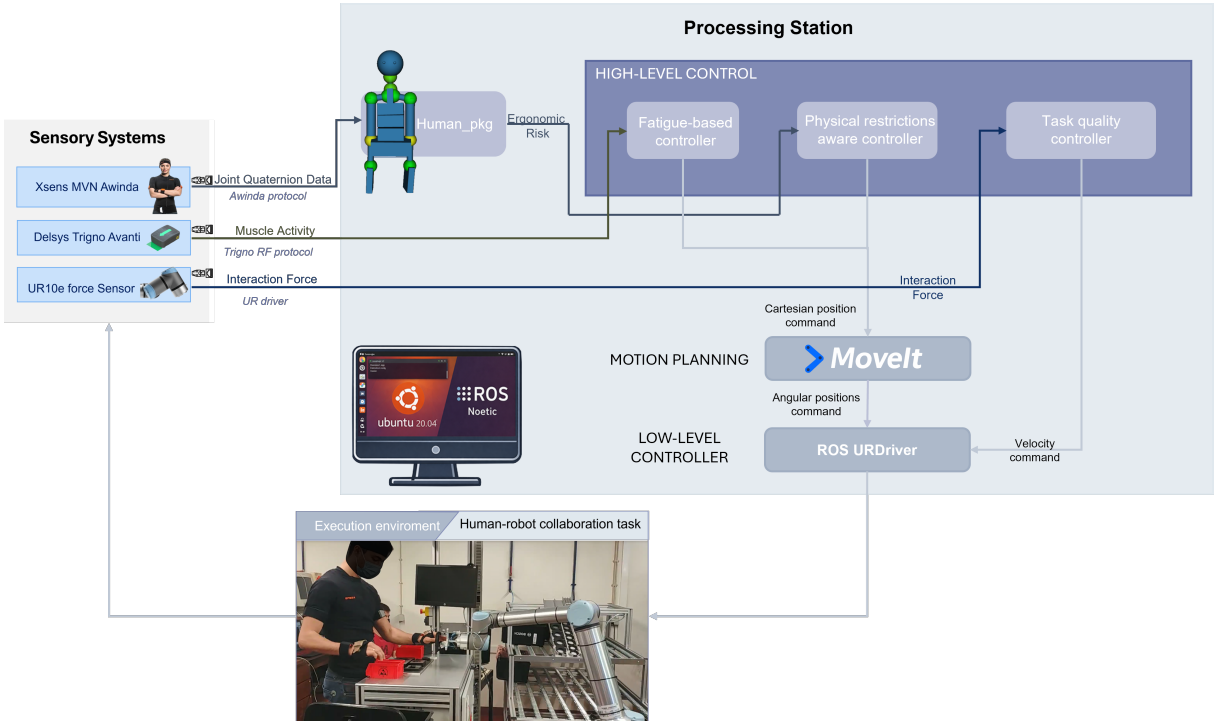


Figure 72: Ergoaware HRC framework overview.

MVN Analyze, while raw sEMG signals are captured using EMGworks acquisition software. These data streams are forwarded to HRC framework, which runs on a separate processing computer under Ubuntu.

Motion data are transmitted from MVN Analyze at 60 Hz using UDP over IP via a wired Ethernet connection [24], while raw EMG data are streamed at 74 Hz using TCP/IP over Ethernet [242]. Although the Xsens MVN system supports multiple data modalities, only segment position and orientation, expressed as quaternions, are used in this work.

The processing station hosts the core of the ErgoAware framework and is responsible for all high-level data processing and control execution. In particular, it processes the Xsens motion data to estimate human joint angles, which serve as the primary input for some human-aware control strategies. All software components run as independent ROS nodes and communicate through well-defined interfaces, ensuring modularity and loose coupling. This design allows sensing modalities and control strategies to be added, replaced, or removed without structural changes to the overall architecture.

The high-level control layer comprises three human-aware controllers. Two of these controllers are position-based and generate reference Cartesian target poses for the robot end-effector based on human state information. Specifically, the fatigue-aware controller described in Chapter 6.4, and the physical restrictions-aware controller presented in Chapter 7 use processed sensing data to compute an updated desired Cartesian position. This reference is then converted into robot joint-space commands through

inverse kinematics and the MoveIt motion planning framework before being sent to the UR10e.

In contrast, the task quality controller, detailed in Chapter 8, leverages the low-level velocity controllers available on the UR10e. It directly receives force measurements from the robot's end-effector and computes a corresponding joint-space velocity command, which is then transmitted to the robot.

5.2.1 Sensory systems

MVN Awinda Integration package

A C++ ROS-based wrapper node was developed to integrate the Xsens MVN data stream into the ErgoAware framework. This node builds on a lower-level networking and parsing module responsible for receiving MVN UDP datagrams and decoding the proprietary binary format into structured motion data. The wrapper continuously receives decoded data, reads the data type identifier contained in each datagram header to determine how the incoming motion information should be interpreted, and converts the resulting samples into ROS messages. These messages are published on dedicated topics, providing access to motion data. This allows to decouple the proprietary MVN data stream from the rest of the framework and enable consistent, modular consumption of motion data.

Delsys integration package

The Delsys Trigno Avanti system is integrated into the ErgoAware framework through a dedicated C++ communication layer and a ROS-based wrapper, enabling real-time acquisition of the sEMG data. The integration relies on an client library that implements the network protocol defined by the Trigno Wireless System SDK [242], developed by the team.

The SDK exposes a configurable TCP server running on the host computer executing the EMGworks software, which streams sensor data in real-time. The client manages the TCP connection, receiving EMG streams and providing a bidirectional command interface for system configuration and status queries.

Incoming data are organized into lightweight structures designed for high sampling rates, including individual samples, synchronized frames across active sensors, and time-indexed sequences representing continuous recordings. This organization supports efficient handling of both raw and processed data. Although the SDK delivers the data in packets received at approximately 74 Hz. Each received packet is unpacked by the communication layer and then published to the corresponding ROS topic

Incoming data are organized into lightweight structures suitable for high-rate streams, including individual samples, frames across active sensors, and time-indexed sequences representing continuous

recordings. This organization provides a practical structure for handling the data received from the Trigno server. The Trigno server transmits data through TCP packets at approximately 74 Hz; each packet contains multiple consecutive samples according to the configured sensor mode. For the Trigno Avanti in Mode 65, corresponding to 1259.259 Hz, each packet contains approximately 17 samples. The client library sequentially unpacks these samples and publishes them as individual ROS messages, preserving their order in the original packet stream.

The ROS wrapper exposes this functionality through standard ROS services and publishers. System-level operations such as starting and stopping data streaming, querying sensor information, issuing server commands, and resetting the network connection are accessed via ROS services. Sensor data are handled by a dedicated data wrapper that publishes incoming streams to ROS topics. Data acquisition and publishing are executed asynchronously to ensure non-blocking operation.

Latency measurement

A latency analysis was conducted to estimate the processing delay introduced by the ROS integration of the two sensory systems. In the implemented acquisition setup, the proprietary acquisition software runs on a Windows computer, with MVN Analyze used for Xsens MVN and the Trigno SDK used for Delsys Trigno. The data are then transmitted through Ethernet to a Linux computer, where they are received and published to ROS topics.

The measured latency does not represent the full delay from physical acquisition at the sensor to data availability in ROS. This end-to-end latency cannot be directly measured in the present setup because the sensor hardware, the Windows host, and the Linux host do not share a synchronized clock. Therefore, the reported values correspond only to the Linux-side delay, measured from data arrival at the Linux computer to publication on the corresponding ROS topic. This metric isolates the part of the pipeline implemented in the ErgoAware framework, namely the communication handling and ROS publication stages.

The tests were performed while the full ErgoAware framework was running, under operating conditions representative of those used during the control experiments. The acquisition frequencies were 1259.259 Hz for the Delsys Trigno system and 60 Hz for the Xsens MVN system. Figure 73 shows the distribution of the measured processing delays. The Delsys Trigno integration presented a mean delay of 2.64 ms and a 95th percentile of 4.84 ms. The Xsens MVN integration presented a mean delay of 5.51 ms and a 95th percentile of 9.81 ms.

These values should be interpreted with respect to how each system delivers data. For Xsens MVN, samples are received at 60 Hz, corresponding to an inter-sample period of 16.67 ms. The measured 95th

percentile delay of 9.81 ms remains below this interval, indicating that the ROS publication step does not introduce delay accumulation at the tested acquisition rate. For Delsys Trigno, the sEMG samples are acquired at 1259.259 Hz, corresponding to an inter-sample period of approximately 0.79 ms. However, the Trigno SDK does not deliver one TCP packet per sample. Instead, it transmits packets at approximately 74 Hz, with each packet containing multiple consecutive samples. Therefore, the relevant timing constraint for the communication layer is the packet arrival period, not the individual sample period. The measured 95th percentile delay of 4.84 ms remains below the packet interval, and no sample loss was observed during the trials. These results indicate that, under the tested conditions, both integrations publish the received data to ROS without measurable backlog or sample loss.

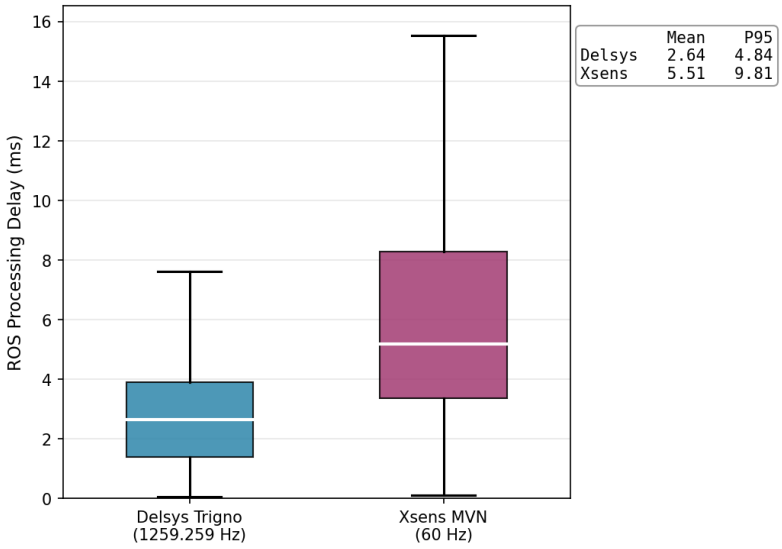


Figure 73: ROS processing delay distributions for Delsys Trigno at 1259.259 Hz and Xsens MVN at 60 Hz.

5.2.2 Actuator

UR10e

The proposed framework employs a Universal Robots UR10e collaborative robot. The UR10e is a six-degree-of-freedom manipulator with a maximum reach of approximately 1.3 m and a nominal payload capacity of 12.5 kg. The robot supports collaborative operation through integrated safety features and is capable of smooth position and velocity control.

The robot is equipped with a pneumatic soft robotic gripper from Rochu (Zhangjiagang, Jiangsu Province, China), composed of three compliant fingers, pneumatically actuated through an external control unit that switches between positive and negative pressure. The gripper supports a maximum operating speed of

approximately 300 open–close cycles per minute [243].

UR ROS Driver The UR ROS driver [244] provides the communication interface between the ErgoAware framework and the collaborative robot Universal Robots UR10e, enabling real-time control and feedback within the ROS ecosystem, relying primarily on the Real-Time Data Exchange (RTDE) protocol for deterministic communication.

The driver exposes multiple control interfaces, such as joint position, joint velocity, allowing different control strategies to be implemented at the ROS level. In addition, it provides direct access to the robot's internal force–torque sensor, enabling high-frequency acquisition of end-effector wrench measurements, which are essential for interaction-based controllers. Trajectories can either be generated and updated directly by ROS controllers or sent to the robot as target waypoints, with the interpolation handled internally by the robot controller. In this work, ROS-based position and velocity controllers were used to continuously update robot commands.

The driver supports seamless interaction with the robot's teach pendant via a URCaps program, allowing programs to be paused, resumed, or stopped without restarting the ROS driver. Safety mechanisms such as speed scaling and protective stops are transparently handled, with ROS controllers adapting to safety-imposed speed limits.

5.2.3 Processing Station

Simulation world

RViz was used to provide a visualization of the workspace and to enable testing of the developed algorithms in a simulated environment. The UR10e provides a offline simulator interface [245] that can be connected to the framework, allowing the same software stack to be executed either with the physical robot or within a simulated robot instance.

The simulation world was designed to represent an industrial workstation developed within the FoF project, a partnership between Bosch and the University of Minho. The workstation includes the table, the robot base, and a feeder system that holds the pieces to be assembled. The corresponding CAD geometry of this workstation was provided as STL files and integrated into the simulation through Xacro descriptions. These files define the static elements of the environment and are organized within the `world` package. They are used to launch a consistent visualization of the workspace in RViz. Figure 75 presents the environment visualized in RViz.

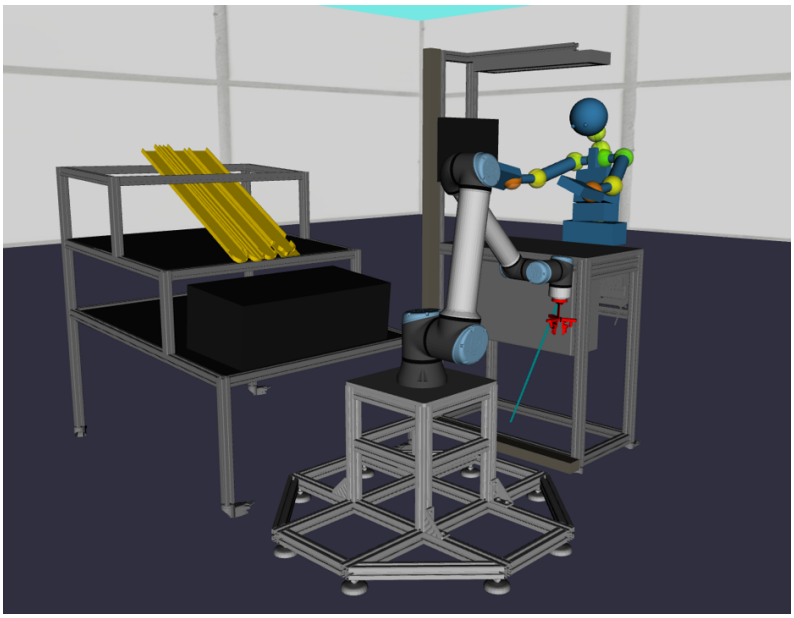


Figure 74: Simulation environment in Rviz.

Collision modeling and motion constraints To ensure safe and environment-aware motion planning, a set of collision objects was explicitly defined within the MoveIt planning scene, as illustrated in red in Figure 75. They were specifically designed to reflect the requirements of the handover task selected within the FoF project, ensuring safe and appropriate robot behavior in the shared human–robot workspace.

Collision objects were defined around the feeder structure, the workstation table, the robot base and lateral structures, the screen mounted on the workstation, and a vertical plane positioned in front of the human operator. Together, these elements constrain the robot's reachable space and prevent trajectories that would result in collision with the environment or unsafe motion close to the human.

The vertical plane in front of the operator plays a dual role. Within the region behind this plane, the robot is allowed to execute joint-space motions (MoveJ), which are faster and more efficient but less predictable from a human perspective, and thus less safe. When planning motions that cross this plane, typically during handover or close human interaction, the robot switches to Cartesian motion (MoveL). This enforces straight-line end-effector trajectories, improving predictability and legibility of the robot's behavior near the human. The vertical plane is deactivated when the robot approaches it.

Human Package

Figure 76 illustrates the ROS nodes, topics, and data flow implemented within the Human package. This package encapsulates the human modeling, kinematic processing, and ergonomic assessment functionalities of the ErgoAware framework, providing a modular and reusable representation of the human

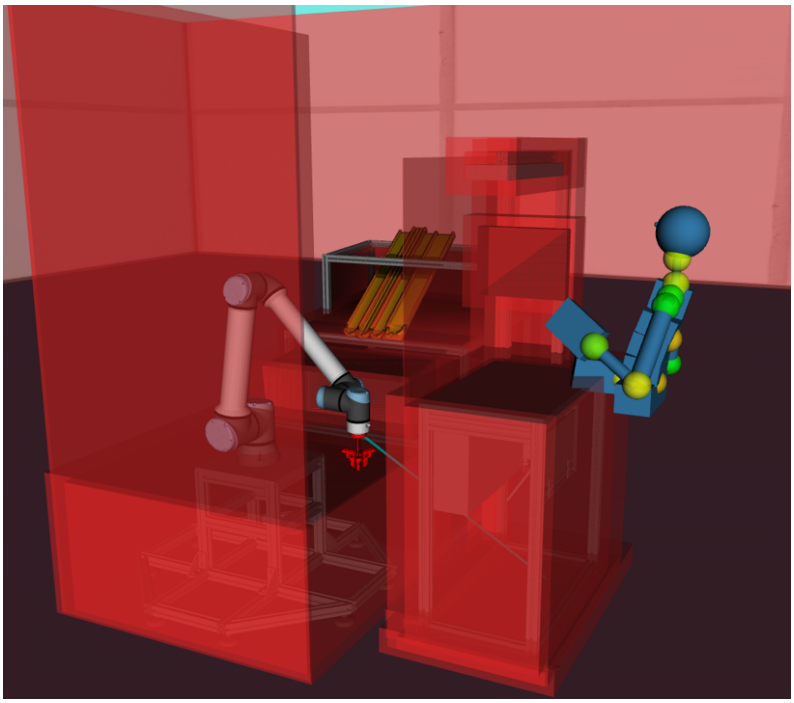


Figure 75: Simulation environment and collision model used for motion planning. Collision volumes are defined in red, around the workstation, feeder, robot structure, and a vertical interaction boundary in front of the human.

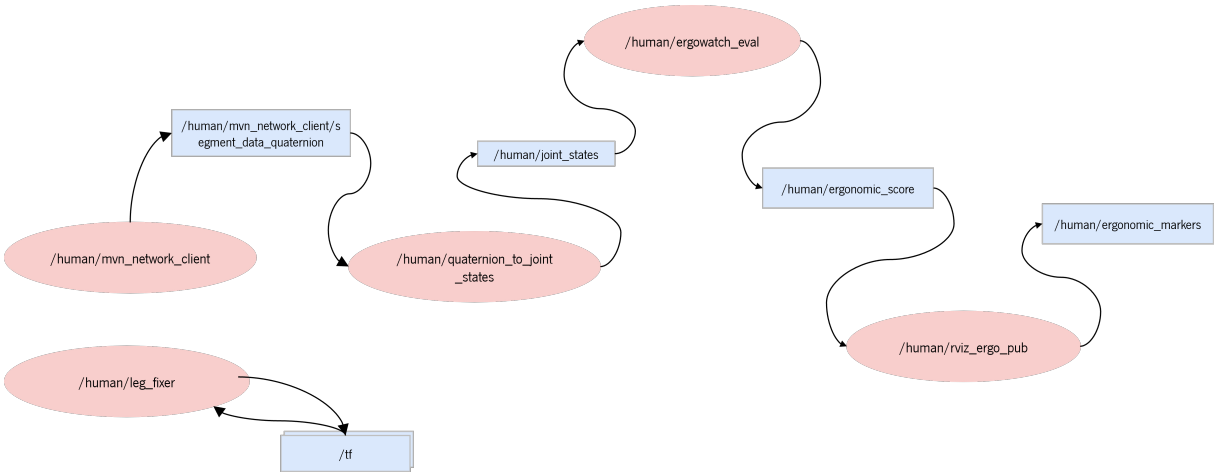


Figure 76: ROS node and topic graph of the Human package, ROS node and topic graph of the Human package.

operator to support motion interpretation, risk evaluation, and visualization.

The package generates a subject-specific human model by constructing a URDF representation from anthropometric and kinematic information extracted from previously recorded Xsens `.mvnx` files. The URDF model used in this work is based on an open-source GitHub package [246] and was adapted to the requirements of this project. Since only upper-body segments are considered, the lower-limb segments

were removed and the pelvis was defined as the root of the kinematic chain. As a result, the pelvis was no longer constrained by lower-limb kinematics, leading to an undefined relationship with the global reference frame. To address this issue, the package includes a dedicated pelvis-fixing node (`/human/leg_fixer`) that anchors the pelvis pose to a fixed workstation reference by publishing a consistent transform, thereby maintaining TF integrity for visualization and downstream modules. This subject-specific kinematic model is used consistently across the framework for kinematic computation and visualization.

To obtain anatomically meaningful joint angles, the package converts Xsens segment orientation data (node (`/human/quaternion_to_joint_states`), expressed as quaternions, into joint-space representations using a human kinematic model implemented with the Kinematics and Dynamics Library (KDL). Joint angles are computed by solving inverse kinematics between connected body segments, enforcing kinematic consistency across joints. This approach was particularly necessary for joints with complex motion, such as the shoulder, where direct derivation of joint angles from segment orientations was found to be unreliable due to multiple degrees of freedom and susceptibility to gimbal lock. The resulting joint angles are published as standard ROS `JointState` messages.

In addition, the Human package integrates ergonomic risk assessment (node `/human/ergowatch_eval`) based on the RULA guidelines [238]. Postural risk scores are computed from the estimated joint configurations and published as custom ROS messages. In particular, all intermediate/local scores that compose the RULA assessment are also made available, enabling their use by higher-level controllers and monitoring modules.

Finally, the package provides real-time visualization of ergonomic risk by augmenting the human URDF skeleton with graphical markers at relevant joints (node `/human/rviz_ergo_pub`). The animated model reflects the user's current posture, while colored spherical markers attached to each joint encode the associated ergonomic risk level. Marker colors range from green to red, indicating increasing RULA scores, with green representing the lowest and red the highest joint-level ergonomic risk. This visualization enables intuitive and immediate assessment of ergonomic load in RViz. These spheres can be observed in figures 74 and 75.

Handover dectector

Envisioned for handover tasks, a dedicated handover detector was developed to trigger the gripper release. The detector is implemented as a ROS node that subscribes to end-effector wrench data published on the `/robot/wrench` topic by the `/robot/ur_hardware_interface` through the ROS UR Driver. End-effector forces and torques are continuously monitored, and consecutive measurements

are compared. A handover event is detected when the absolute change in any Cartesian force component exceeds 4 N or when the absolute change in any Cartesian torque component exceeds 4 N·m. This approach detects abrupt interaction-induced variations rather than absolute force levels, enabling robust identification of intentional physical contact by the human. Upon detection, the gripper is commanded to open, releasing the object to the human operator.

5.2.4 Position-based Control

The position-based control in the ErgoAware framework uses MoveIt as middleware for collision-aware motion planning and execution. To align with industrial practice and simplify task-level integration, MoveIt was encapsulated into two motion primitives, MoveJ and MoveL, which directly mirror the motion options available on the Universal Robots teach pendant.

MoveJ plans motion in joint space by setting a joint target through the MoveGroupInterface, computing a collision-aware trajectory, and executing it on the UR10e controller. This results in smooth joint motions with an unconstrained Cartesian tool path, making MoveJ fast and well suited for free-space repositioning between task-relevant poses. In contrast, MoveL generates a Cartesian path using MoveIt's Cartesian planner and applies time parameterization before execution, enforcing a straight-line end-effector motion that is appropriate for precise approach and handover actions but more sensitive to kinematic constraints.

Motion requests are issued through ROS services, providing synchronous execution. Each service call blocks until planning and execution are completed, ensuring deterministic task sequencing and reliable coordination between task logic and robot motion during assembly and handover operations.

5.2.5 Velocity-based Control

Joint-space velocity control is achieved by publishing directly to the `/robot/joint_group_vel_controller/command` topic, which is the command interface of the underlying velocity controller instance on the UR robot. At runtime, the controller node receives a vector of joint velocities on this topic at a fixed rate of 500 Hz, interprets the array entries as desired velocities for each actuated joint in order, and forwards them through the UR hardware interface to the robot's low-level velocity command channel.

5.3 Conclusions

This chapter presented the ErgoAware human–robot collaboration framework, describing its conceptual design, software architecture, and core functionalities. The framework was developed as an interopera-

ble software stack capable of integrating multimodal human sensing, ergonomic assessment, and robot control within a modular and extensible ROS based architecture, enabling systematic experimentation with human aware collaborative strategies. Emphasis was placed on modularity, interoperability, and the clear separation between sensing, processing, and control components, allowing the framework to support multiple human aware controllers and to evolve with minimal structural modifications.

Chapter 6

Fatigue-Based Control strategy

6.1 Introductory Insight

Monitoring and managing muscle fatigue is essential for sustaining safe and efficient human performance in physically demanding tasks. Muscle Fatigue alters motor unit behavior, reduces force control, and increases the likelihood of injuries and work-related musculoskeletal disorders [73]. This chapter presents the design and development of a HRC control strategy for fatigue management, with the objective of preventing or delaying the onset of muscle fatigue during repetitive tasks. Muscle fatigue is defined here as the exercise-induced decline in the force-generating capacity of specific muscles, resulting from neuromuscular changes at both central and peripheral levels [72, 73]. While fatigue is a multidimensional phenomenon that can also manifest at systemic, cognitive, and psychological levels [247, 248], this framework is intentionally scoped to its neuromuscular expression, as sEMG provides direct observability over muscle activity and the proposed control actions operate exclusively on muscular demand.

Muscle activity signals acquired through surface electromyography are processed by a fatigue detection model, which provides the collaborative robot with continuous awareness of the human partner's local muscle fatigue state and enables adaptive assistance. Within this framework, the chapter investigates the following research question: ***To what extent can a deep learning model accurately estimate muscle fatigue in a subject-independent manner and inform a control strategy to effectively mitigate muslce fatigue progression and enhance task endurance?***

6.2 Solution Conceptualization

6.2.1 Requirements

This section defines the technical and methodological requirements for developing a fatigue-informed control strategy based on sEMG-based muscle fatigue estimation. These requirements structure the subse-

quent design decisions, including signal processing, model formulation, training and validation protocols, and integration within the collaborative control framework.

While the immediate objective of this work is the development of a muscle fatigue estimation model and its integration within the HRC control framework, the system is conceived with scalability as a core design principle, targeting future deployment in wearable-based solutions. This long-term perspective motivates the adoption of a muscle-agnostic modeling strategy, implemented through a single-sensor fatigue classification approach that is compatible with the textile sEMG electrodes introduced in Chapter 3 and suitable for embedded or resource-constrained platforms.

Because the dominant frequency content of voluntary human motion lies below 5 Hz [249], the fatigue estimation update rate can be decoupled from the robot's 500 Hz control loop. This decoupling enables more efficient use of computational resources and allows the use of longer temporal analysis windows, which improves feature stability and reduces sensitivity to short-term signal variability without compromising control responsiveness.

To ensure robust real-time operation and seamless integration with the collaborative control layer, the system must satisfy the following requirements:

- **Classification performance:** overall accuracy above 70% across the three fatigue states (no fatigue, transition-to-fatigue, fatigue). This value was grounded in cross-subject sEMG-based fatigue classification benchmarks, where subject-independent deep learning approaches yield accuracies between 57.94% and 71.68% [80].

- **Inference time:** Model inference must complete within the analysis window duration, ensuring the fatigue estimate is available before the next window is processed.

- **Signal acquisition:** sEMG sampled at $f_s > 1000$ Hz to preserve spectral information relevant to fatigue dynamics [143].

- **Control integration:** compatibility with the ROS middleware and synchronization with the collaborative robot control loop running at 500 Hz.

6.3 Fatigue estimation Model

6.3.1 Experimental Methodology

This section lays out the process and data collection of the dataset.

6.3.2 Dataset Acquisition

Participants

Healthy participants from the University of Minho and Massachussets Institute of Technology academic community were contacted to participate in the study. They were informed about the study's goal and details, protocol, and duration. The inclusion criteria to select the participants were as follows: *i)* over 18 years of age; *ii)* full range of motion; *iii)* no clinical history or evidence of motor injuries, including shoulder or arm pain; and *iv)* no upper-limb exercises performed within the previous 48 hours. Participants' rights were preserved and, as such, personal information that could identify them remained confidential and it is not provided in this dataset.

Dataset 1 Thirteen participants (5 females and 8 males; age: 23.92 ± 3.36 years old, height: 173 ± 10cm, body mass: 66.1 ± 10.52kg), all right-handed, agreed to participate voluntary in this study. The anthropometrics, as well as the exercise habits and caffeine intake of each participant can be consulted in table 61. All participants provided their written and informed consent to participate in the study, according to the ethical conduct defined by the University of Minho Ethics Committee in Life and Health Sciences (CEICVS 006/2020), that follows the standard set by the declaration of Helsinki and the Oviedo Convention.

Dataset 2 Ten participants (5 males and 5 females, age: 25.7 ± 4.27 years old, height: 181 ± 11cm, body mass: 64.1 ± 10kg)), all right-handed, agreed to participate voluntary in this study. Table 62 in appendix F presents the participants antropometrics. All participants provided their written, informed consent prior to participation, in accordance with the ethical standards approved by the MIT Committee on the Use of Humans as Experimental Subjects (2409001415 - Exploring kinematic and physiological signals to develop a muscle fatigue model).

Instrumentation and Data Collection

Participants were instructed to not perform any high intensity physical activity on their upper limbs in the 48 hours prior to the data acquisition, as they would be excluded from the experiment. To avoid upper limb movement constraints and ease the instrumentation, each participant was instructed to wear upper body loose clothing (shirts or strap tops which allowed easy access to upper arm muscles) and sneakers. Data were collected at the School of Engineering of the University of Minho (D1) and at Immersion Lab at MIT.Nano (D2).

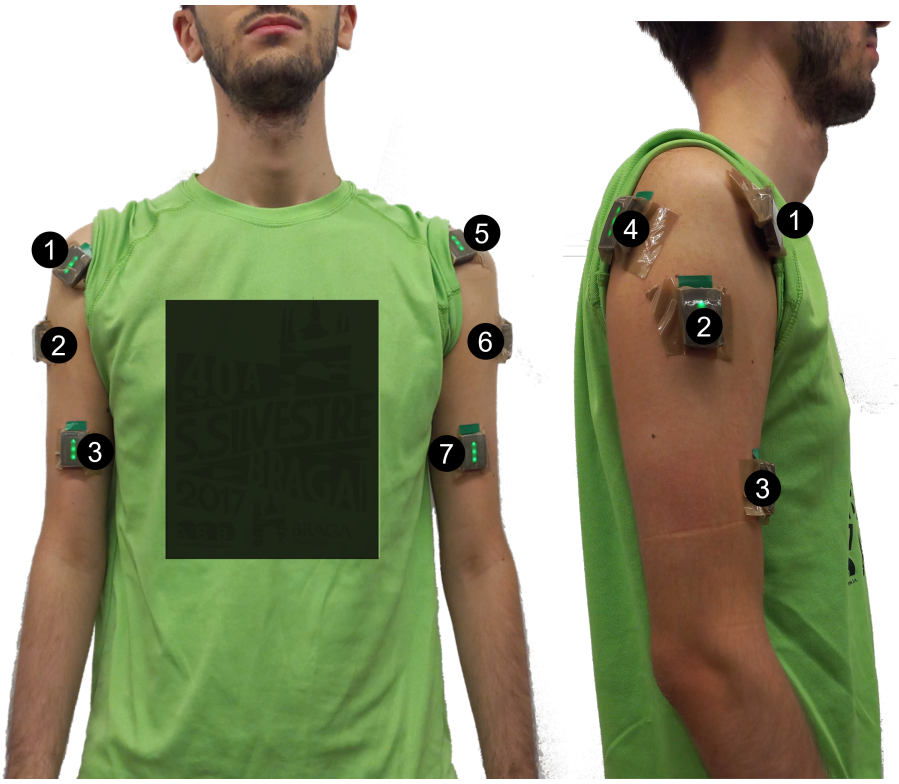


Figure 77: Participant instrumented with 8 wireless sEMG sensors (Trigno Avanti, Delsys), placed on: 1) Right Deltoid Anterior, 2) Right Deltoid Medius, 3) Right Biceps Branchii, 4) Right Deltoid Posterior, 5) Left Deltoid Anterior, 6) Left Deltoid Medius, 7) Left Biceps Branchii. Sensor 8 is placed on the Left Deltoid Posterior. sEMG placement is the same for D1 and D2.

The participant's instrumentation (Figure 77) occurred as following: *i)* Skin was shaved and cleaned with ann abrasive alcohol pad to remove dead skin and natural oils and decrease contact impedance; *ii)* 8 wireless sEMG sensors (Delsys Trigno Avanti, Delsys Incorporations, USA) with dry electrodes were placed on the Biceps Brachii (BB), Deltoid Anterior (DA), Deltoid Posterior (DP), and Deltoid Medius (DM) of both upper limbs, using double-sided adhesive. The sensors were placed over the muscle belly, with the arrow parallel to the muscle fiber orientation, following SENIAM and manufacturer's recommendations [150]. For D2, wireless sEMG sensors with snap-lead wet electrodes were used on the right and left BB, instead of dry electrodes, due to their availability at the acquisition site and their superior signal quality resulting from lower impedance. This electrode type was applied only to the BB given its larger muscle belly, which provided sufficient surface area for proper wet electrode placement. To ensure secure hold and minimize the risk of motion or displacement artifacts, all sensors were taped with *Tesa Pack* tape in D1 and kinesiology tape in D2.

All participants were instrumented by the same researcher to ensure repeatability and minimize errors caused by misplacement. Data collection included: *i)* electrophysiological signals from the 8 muscles,

provided by EMGworks Acquisition from Delsys, at 1259Hz for D1 and 2418.14Hz for D2; *ii)* participants' self perceived rated on a 3-level scale fatigue level (presented in Figure 78) and collected at 50Hz in D1 and 518 Hz in D2.

Scale	Rate of Perceived Exertion
0 (Non-fatigue)	**Light Activity** Perform the task without difficulty. No movement compensation. Feels like you can maintain the task for hours.
1 (Transition-to-fatigue)	**Moderate Activity** It starts to become aware of the muscle addressed. Muscle starts to feel warm. It becomes difficult to maintain exercise intensity.
2 (Fatigue)	**Hard Activity** Very difficult to maintain exercise intensity. Total muscle awareness. Motion compensation. You can no longer execute the movement correctly.

Figure 78: Self-perceived fatigue-level description.

For D1, the self-perceived fatigue level was recorded using a custom system developed by the team, consisting of an Arduino Mega with two push buttons. The system started at level 0 (no fatigue) and was updated to level 1 (transition to fatigue) or level 2 (fatigue) when the researcher pressed the corresponding button.

For D2, self-perceived fatigue levels were recorded using a force-sensitive resistor (Force Sensing Resistor (FSR)) connected to channel 1 of a 4-channel wireless analog sensor from Delsys, which the participant held in the hand not involved in the task. Whenever the participant felt their fatigue state changing, they would press the FSR sensor with its fingers, generating a peak in the analog signal. The definition of the fatigue scale, shown in Figure 78, was based on the modified Borg scale, which has 10 levels. However, since self-perceived fatigue had to be reported in real-time, requiring participants to respond quickly and confidently, a 10-level scale was considered unnecessarily demanding. Moreover, given the subjective nature of perceived fatigue and its dependence on individual exercise background, a finer granularity could increase labeling uncertainty. A 10-level scale would also raise cognitive load, potentially delaying responses when participants are unsure of their fatigue level. These factors could limit the reliability of the data and hinder future applications that require accurate labeling.

This 3-level scale was explained to all participants by the same researcher to ensure a consistent interpretation of muscle fatigue. Throughout the experiment in D1, this researcher regularly asked the participants for feedback on their fatigue state, pressing one of the buttons when the participant reported

a different level of fatigue. In D2, pressing the FSR sensor was entirely the participant's responsibility. All data were synchronized using a hardware trigger.

Experimental protocol

Collecting fatigue data presents several constraints. Once a muscle reaches a truly fatigued state, full recovery may require 24 to 48 hours, which prevents repeated trials under comparable non-fatigued conditions. In addition, multi-day data collection is difficult in practice, as recruiting participants willing to commit to extended protocols is challenging. At the same time, training deep learning models requires a sufficiently large and diverse dataset.

To address these constraints, the protocol was designed such that each trial targets a specific muscle and is performed continuously until exhaustion, yielding a single fatigue progression per muscle and avoiding repeated trials. Since repetitions cannot be reliably performed due to incomplete recovery, we departed from the assumption that fatigue progression exhibits similar patterns across muscles. This is a deliberate design choice, as one of the goals of the model is to generalize across muscles. Bilateral execution (left and right sides) was included to partially mitigate this assumption, providing comparable instances of homologous muscles and introducing controlled variability across subjects.

Each trial was designed to induce fatigue continuously, without rest periods, to isolate fatigue-related changes in the signal from potential confounding effects of recovery. Allowing participants to rest within a trial would have introduced partial recovery dynamics that are difficult to quantify and could obscure the onset and progression of fatigue. By enforcing uninterrupted effort within each trial, the protocol ensures that observed changes can be attributed to fatigue progression alone, at the cost of producing a strictly monotonic fatigue trend.

After the instrumentation of the participants, 3 maximum voluntary contractions of all 8 muscles was acquired, following the SENIAM guidelines [150].

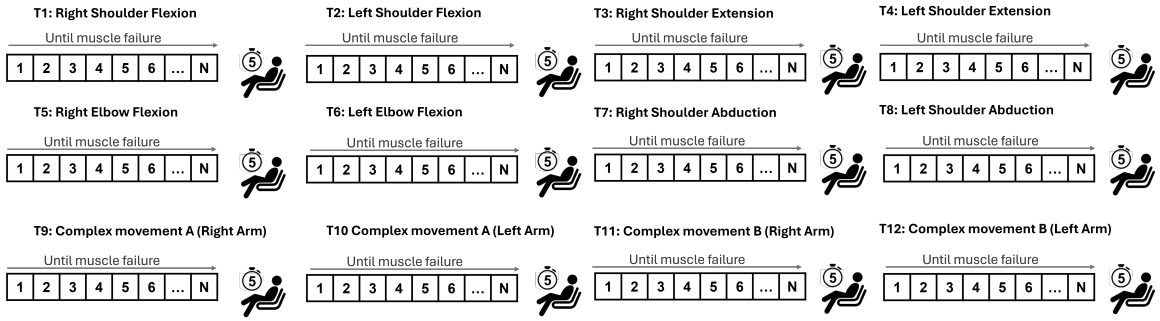


Figure 79: Flowchart illustrating the experimental protocol sequence.

For D1, participants performed 12 movements in total, comprising six per upper limb: four uni-articular

and two compound movements (Figure 79). The compound task was included to approximate industrial actions such as welding, while maintaining a primary focus on muscle fatigue. This task involved drawing a pentagon across the table with alternating reaches to activate the anterior and posterior deltoids, simulating continuous welding-like motion. Simpler unilateral tasks allow clearer observation of fatigue progression. For D2, only the eight uni-articular movements were performed, excluding compound tasks to reduce overall experiment duration.

Participants were instructed to continue each movement until exhaustion. Once fatigue level was reached, they were asked to persist until they could no longer tolerate the task. Only one repetition per movement was recorded, as additional repetitions were expected to be strongly influenced by accumulated fatigue.

The protocol began with uni-articular movements targeting specific muscles. Movements were performed while holding a 1.5 kg dumbbell to accelerate fatigue onset, with a metronome set at 30 bpm to regulate pace. In D1, compound movements followed, using a 2 kg arm-mounted weight that allowed free hand motion. Participants were encouraged to maintain focus on the body–mind connection during execution.

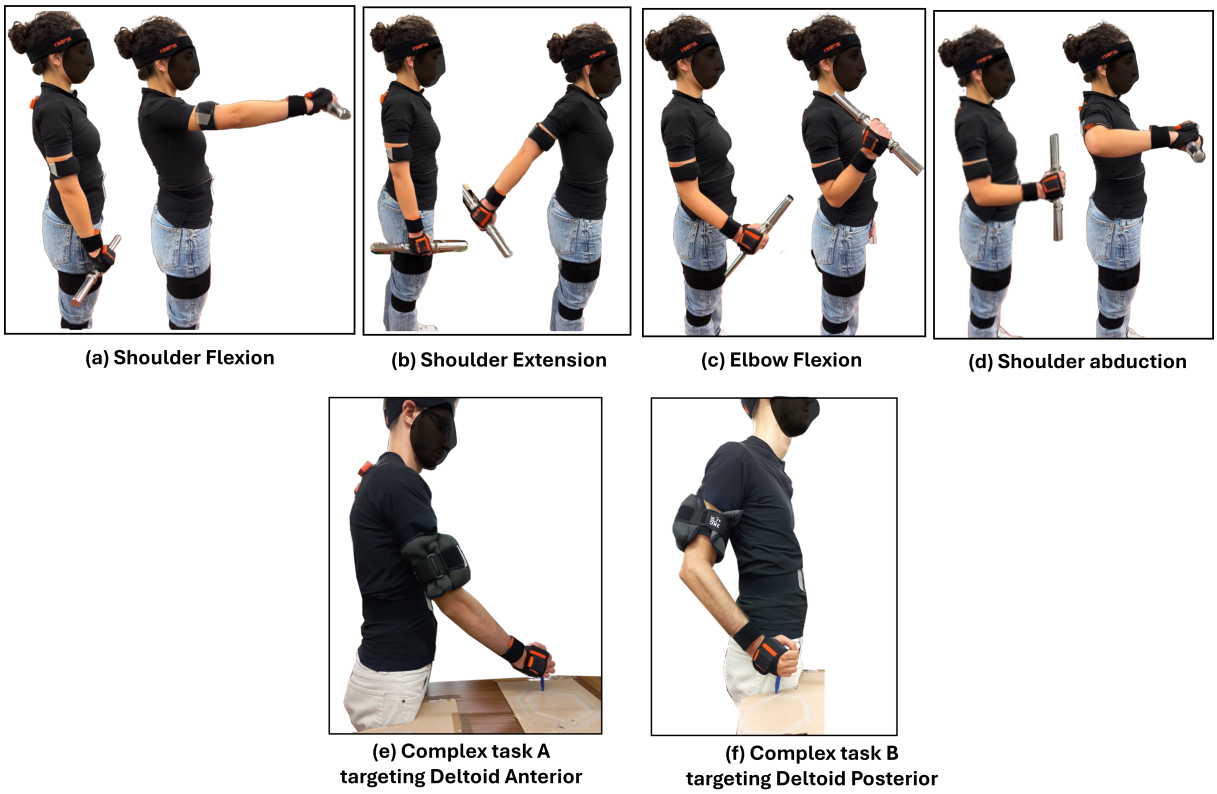


Figure 80: Experimental protocol. All movements were executed, one at once, with both right and left arm. For D2, only movement a, b, c, and d were executed.

Figure 80 presents the executed movements, namely the four uni-articular and two compound motions. For D1, twelve trials were recorded (six for each side), and the sequence of movements performed by the participants is illustrated in Figure 79. Signals were collected alternately from both arms to allow recovery of the arm previously under strain. A 5-minute rest was applied between trials, and the time spent performing the task with the opposite arm further increased recovery, resulting in more than 10 minutes between consecutive recordings of the same arm. For D2, only eight trials were recorded, corresponding exclusively to the same uni-articular movements shown in Figure 80.

6.3.3 Dataset pre-processing and features selection

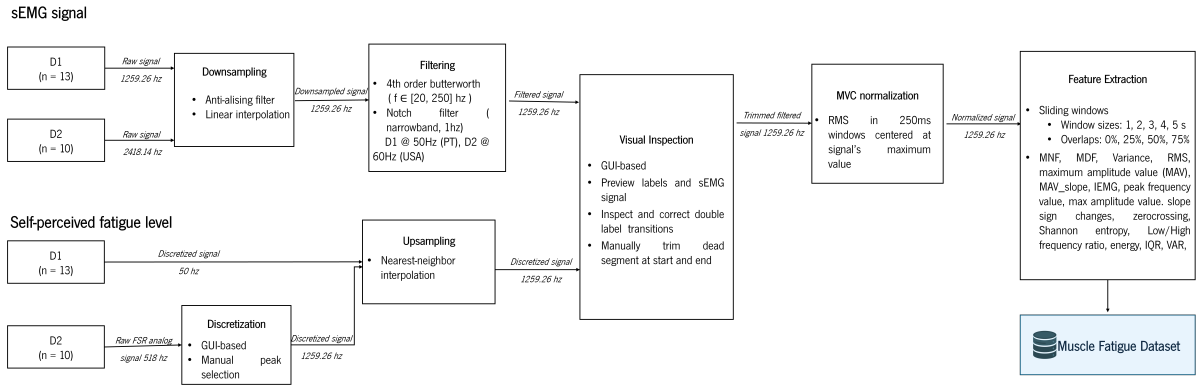


Figure 81: Preprocessing and feature extraction pipeline for the muscle fatigue dataset.

Figure 81 illustrates the pipeline implemented for preprocessing and feature extraction. The raw sEMG data was exported using EMGAnalyze for dataset D1 and Trigno Discovery for dataset D2, both proprietary software from Delsys. The exported signals were then subjected to a series of preprocessing steps. For dataset D2, the data was downsampled from its original frequency to 1259.26 Hz. Prior to resampling, a low-pass filter with a cutoff at 629.63 Hz was applied to prevent aliasing. Resampling was performed using linear interpolation. After resampling, all signals were filtered with a 4th-order Butterworth band-pass filter between 20 Hz and 250 Hz. This range was chosen because the signal spectrum contained little information above 250 Hz, and it is well established that although the EMG signal occupies approximately 0–500 Hz, its dominant components lie within the 50–150 Hz range [250]. In addition, a narrowband IRR notch filter (1 Hz bandwidth) was applied at 50 Hz for dataset D1, collected in Portugal, and at 60 Hz for dataset D2, collected in the United States, to suppress powerline interference.

The raw dataset comprised over 64 million samples, corresponding to a total recording duration of 14.28 hours.

For dataset D1, the labels were upsampled from 50 Hz to 1259.6 Hz by detecting transition times

and assigning preceding samples to the prior label. For dataset D2, trials were labeled using an FSR signal sampled at 518 Hz, which was discretized with a threshold-based algorithm (threshold set at 5 mV) and subsequently upsampled. Minor fluctuations caused by double-pressing the sensors were visually inspected and removed. A GUI was developed to preview both the sEMG signals and the associated labels, enabling visual inspection of signal integrity. The GUI also facilitated the removal of dead signal segments at the beginning and end of trials, which contained no relevant information.

Figure 146 in appendix F presents the filtered sEMG data for all 8 trials corresponding to the uni-articular movements of subject 2: (a) the time-domain signals and (b) the corresponding frequency spectra.

The MVC signal was band-pass filtered, and the RMS was computed over a 250 ms window centered on the signal maximum. This value was stored and used for data normalization.

The data was subsequently subjected to an Exploratory Data Analysis (EDA). Time- and frequency-domain features were extracted using a sliding window approach on the normalized data. The selected features, informed by the literature, included: (i) Mean Frequency (MNF), (ii) MDF, (iii) variance, (iv) RMS, (v) maximum amplitude, (vi) peak value, (vii) Slope Sign Change (SSC), (viii) zerocrossing, (ix) time-domain Shannon entropy, (x) frequency ratio, (xi) signal energy, (xii) Interquartile Range (IQR), (xiii) Integrated Electromyography (iEMG) (xiv) Mean Absolute Value (MAV), and (xv) MAV slope. Table 63, in appendix F, presents the corresponding mathematical definitions used to compute these features. Different window lengths and overlaps were evaluated to investigate their effect on feature extraction. The window sizes tested were 1.0, 2.0, 3.0, and 4.0 seconds, combined with overlap percentages of 0%, 25%, 50%, and 75%. Window lengths were selected relative to the movement pace of 30 bpm, where each beat lasts 2 s and a full contraction–relaxation cycle spans around 4 s, allowing evaluation of segments that capture complete and partial portions of the underlying biomechanical cycle.

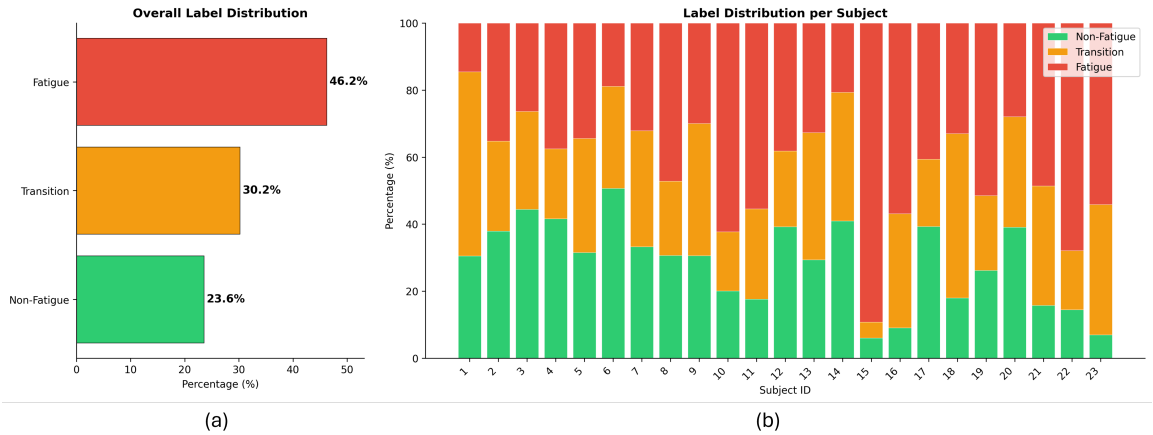


Figure 82: Global (a) and per-subject (b) class distribution of the fatigue dataset.

Figure 82a and 82b presents the global and per-subject class distribution of the dataset. The distribution is imbalanced, with the non-fatigue class being the least represented state (23.6%). Additionally, substantial inter-subject variability in self-perceived fatigue levels is observed, with subject 15 emerging as a outlier compared to the rest of the participants. For this reason, subject 15 was excluded from the subsequent processing and analysis steps.

6.3.4 Exploratory Data Analysis

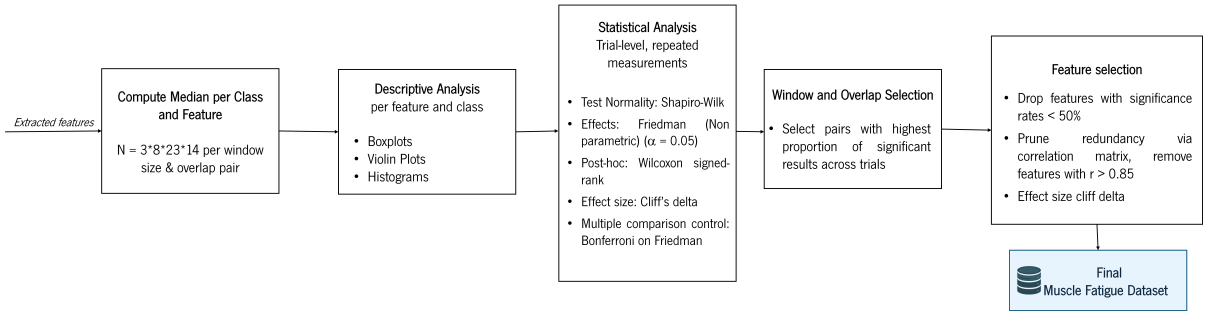


Figure 83: Statistical analysis and feature selection pipeline for the muscle fatigue dataset.

The primary questions that this analysis aims to clarify was: (i) which features carry more information about the fatigue level (ii) whether fatigue is present in the dataset, assessed through statistical testing of sEMG features to identify significant differences across the three fatigue classes (no fatigue, transition-to-fatigue, and fatigue); (iii) Do the features trends match the findings in the literature review in terms of signal trend (increasing /decreasing)? (iv) How long does each fatigue phase typically last and are these durations consistent across muscles?

The exploratory analysis aimed to characterize the collected sEMG data and to inform subsequent modeling choices. The analysis was structured around the six research questions defined above, each addressed through a targeted combination of statistical and descriptive methods designed to evaluate feature relevance, class separability, and temporal fatigue dynamics. An overview of the analysis pipeline is provided in Figure 83.

To quantify the discriminative strength of the extracted features across fatigue levels, effect sizes were first estimated using Cliff's delta. Magnitudes were interpreted as negligible ($|\delta| < 0.147$), small ($0.147 \leq |\delta| < 0.33$), medium ($0.33 \leq |\delta| < 0.474$), or large ($|\delta| \geq 0.474$) [203]. This analysis was performed across all tested window sizes and overlap configurations.

Next, to verify if muscle fatigue is really present in the dataset (question ii), a statistical analysis was conducted to determine whether significant differences existed among the three fatigue classes per feature:

non-fatigue, transition-to-fatigue, and fatigue. Since the data consisted of time series and the individual samples were not independent, the median value of each class was extracted per trial, resulting in three representative values per trial per feature. Independence was then assumed at the trial level, as each trial corresponded to a specific muscle with a specific sensor. In addition, the analysis was treated as a repeated-measures design, acknowledging that multiple observations originated from the same experimental subjects. Data normality was assessed using Shapiro–Wilk tests for each feature across fatigue levels. Because several features did not meet the assumption of normality, a conservative non-parametric approach was adopted. The Friedman test, which does not require normality, was applied to all features to test for overall differences between fatigue levels at a significance level of $\alpha = 0.05$. Benjamini-Hochberg (FDR) correction was applied to the Friedman p-values to control the false discovery rate, which increases power to detect true effects when many comparisons are tested. To assess the contribution of each trial type, statistical significance was additionally evaluated on a per-trial basis.

To address question (iii), when statistically significant differences between fatigue levels were identified, post-hoc pairwise comparisons were performed using one-sided Wilcoxon signed-rank tests. The directionality of these tests was defined a priori based on trends consistently reported in the literature. Specifically, the following features were expected to decrease with fatigue: MDF, MNF, zerocrossing, SSC, peak frequency value, frequency ratio [72, 75]. Conversely, amplitude maximum, RMS, Shannon entropy in the time domain, MAV, MAV slope, variance, energy, were expected to increase with fatigue [75, 251]. Window–overlap pairs were evaluated according to the configuration that yielded the highest proportion of significant results across trials. Features with a significance rate below 50% were excluded. For the retained set, inter-feature redundancy was assessed through a correlation matrix, and highly correlated features (r>0.85) were removed to avoid redundancy.

To address question (iv), classes durations were also computed across subjects and trials and by muscle groups, with left and right muscles combined.

6.3.5 Data Input Preparation

Since fatigue is an ordinal, time-dependent, and cumulative process, where a fatigued state can only occur after a transition-to-fatigue state, and transition-to-fatigue can only follow a non-fatigued state, the data were provided to the neural network in trial-wise sequences. Trials were presented in random order, but within each trial the samples were fed sequentially to preserve the temporal progression of fatigue.

As the architectures operate on features rather than raw sEMG signals, two distinct temporal windowing stages are required: feature windows and model input windows. First, the continuous sEMG signal is

segmented using sliding feature windows with fixed length and overlap. For each feature window, time- and frequency-domain features are computed, yielding one feature vector per window. This step transforms the raw signal into a lower-dimensional, noise-robust representation while preserving short-term muscle activation characteristics. Second, to account for the temporal evolution of fatigue, consecutive feature vectors are grouped into model input windows, which form the actual inputs to the neural network. Therefore, each input window represents a sequence of feature vectors spanning a longer time horizon.

All models were formulated as window-level classifiers. Each model input window was associated with one fatigue label, and the network produced one prediction per input window. Therefore, no architecture was trained to predict fatigue labels at the raw-sample level or at each individual feature vector. During inference, the same representation was preserved: a prediction was generated only after the complete input window had been acquired. As consequence, feature windows control the time resolution and spectral fidelity of the extracted features, while input windows determine the temporal context available to the model for fatigue estimation.

A range of window lengths (0.2, 0.5, 1, 2, and 4 s) and overlaps (25%, 50%, and 75%) was evaluated to assess their impact on feature stability and temporal resolution. The resulting feature update rates ranged from 20 Hz (0.2 s window with 75% overlap) to 0.33 Hz (4 s window with 25% overlap).

Regarding the NN input widows, longer temporal contexts were considered. This choice was motivated by three factors. As a cyclic, dynamic movement, the extracted features retain the cyclic structure of the executed movement, meaning that their values oscillate over time following the movement pattern. In contrast, self-perceived fatigue level evolves much more slowly and can be described as a sequence of plateaus, where the fatigue level remains constant for extended periods and changes only during short transition phases (e.g., non-fatigue for several seconds, followed by transition-to-fatigue, and finally fatigue). Due to this mismatch in time scales, short input windows mainly capture movement-related variability rather than fatigue progression. For this reason, longer input windows were evaluated (10 and 15 seconds), including the use of the full trial as model input, to provide sufficient temporal context for the network to associate cyclic feature patterns with stable fatigue states.

The label of each feature window was defined as the mode of the sample-level labels contained within the corresponding window. Therefore, each model prediction represents the dominant fatigue state rather than an instantaneous fatigue estimate at a specific sample.

To minimize bias in the neural network, MinMax normalization between 0 and 1 was applied to the data, considering the extreme values of the entire training dataset by feature.

6.3.6 Dataset Split

The dataset was split by subject into training, validation, and test sets to endure that validation and test data were never seen during training to avoid bias. The split was random but constrained at the subject level to achieve the best possible class balance across subsets. As aforementioned, subject 15 was removed from the analysis. Of the remaining 22 subjects, 18 were used during grid search, while the remaining 4 were reserved exclusively for final testing using the best-performing models. Among the 18 subjects, 15 were assigned to training and 3 to validation. For testing, models were retrained using data from the 18 subjects without validation, applying the optimal hyperparameters. Test set normalization was performed using statistics from the 18 training subjects, while validation data were normalized using the 15-subject training set. The hold-out approach was adopted due to time constraints, in which subject assignments were random but fixed.

DL architectures

The resulting fatigue labels exhibit a piecewise-constant, monotonic progression, consisting of successive plateaus corresponding to increasing fatigue levels.

The experimental protocol was designed to induce continuous fatigue within each trial, targeting a single muscle without intra-trial rest. A direct consequence of this design is that the self-perceived fatigue level is slowly accumulating, consisting of a 3-level piecewise-constant, monotonic signal, with subjective and uncertain thresholds as it is based on human perception and participants were sometimes in doubt if what they were feeling meant fatigue. The sEMG, in turn, is non-stationary and noisy, varying continuously [252]. Short-term fluctuations in sEMG amplitude or spectral content do not immediately imply a clear change in the perceived fatigue level. Further, in dynamic tasks (which require dynamic contractions), the activation patterns reflect the biomechanics of the movement, while the long-term trend captures the progressive physiological adaptation or degradation associated with fatigue accumulation. Also, the physiological response is highly dependent on the participant, namely its muscular composition and exposure to exercise. All of this makes the fatigue estimation problem intrinsically challenging, particularly in real-time and across subjects.

Several alternative model architectures were evaluated before selecting the final approach. Initial experiments employed time–frequency representations obtained through the STFT as 2D inputs to a CNN, using both single and stacked configurations. In addition, hybrid architectures, including CNN-LSTM models and a variant with a temporal attention mechanism, were explored to capture both spatial patterns in the time–frequency domain and longer-term temporal dependencies. All models were evaluated using

a subject-wise split, in contrast to the random splits commonly adopted in the literature. However, the performance achieved by these architectures was not satisfactory (F1 < 0.55) and these approaches were abandoned.

Four DL architectures were implemented: CNN, LSTM, an LSTM with time-attention and a Time–Frequency LSTM with Attention. Recurrent state handling differed between architectures. For the plain LSTM, hidden and cell states were propagated across consecutive input windows within each trial and reinitialized at the beginning of each new trial. For the attention-based recurrent models, hidden and cell states were reinitialized at each forward pass and were not propagated between consecutive input windows. Therefore, these models captured temporal dependencies only within the current input window.

CNN The proposed architecture adopts a compact, fully convolutional design tailored to time-series feature inputs.

The first stage comprises three one-dimensional convolutional layers operating along the temporal axis, each with a kernel size of 5. Given feature update rates as low as 0.33–1 Hz, input windows of 10–15 s contain relatively few samples. Under these conditions, a kernel size of 5 spans a meaningful portion of the available temporal context, enabling effective modeling of temporal patterns without increasing model complexity.

Each convolutional layer is followed by Batch Normalization to stabilize training and improve convergence, and a Rectified Linear Unit (ReLU) activation to introduce non-linearity. This repeated Conv–BatchNorm–ReLU structure progressively refines temporal representations while keeping the number of parameters controlled.

After the final convolutional block, Global Average Pooling (GAP) is applied along the temporal dimension, aggregating each feature map into a single scalar by computing its temporal mean. This replaces fully connected layers, reducing the risk of overfitting and minimizing the number of learnable parameters.

A dropout layer with a rate of 0.2 is applied after GAP to further improve generalization. Dropout is introduced only at this stage to avoid interference with Batch Normalization, while still providing regularization before classification.

Finally, the pooled representation is passed to the output layer, which predicts three fatigue states: no fatigue, transition-to-fatigue, and fatigue.

LSTM The LSTM architecture consists of two unidirectional layers. Since hidden and cell states were propagated across consecutive windows within each trial, unidirectional recurrence was used to preserve causal temporal context across the trial, ensuring that each prediction depended only on the current and

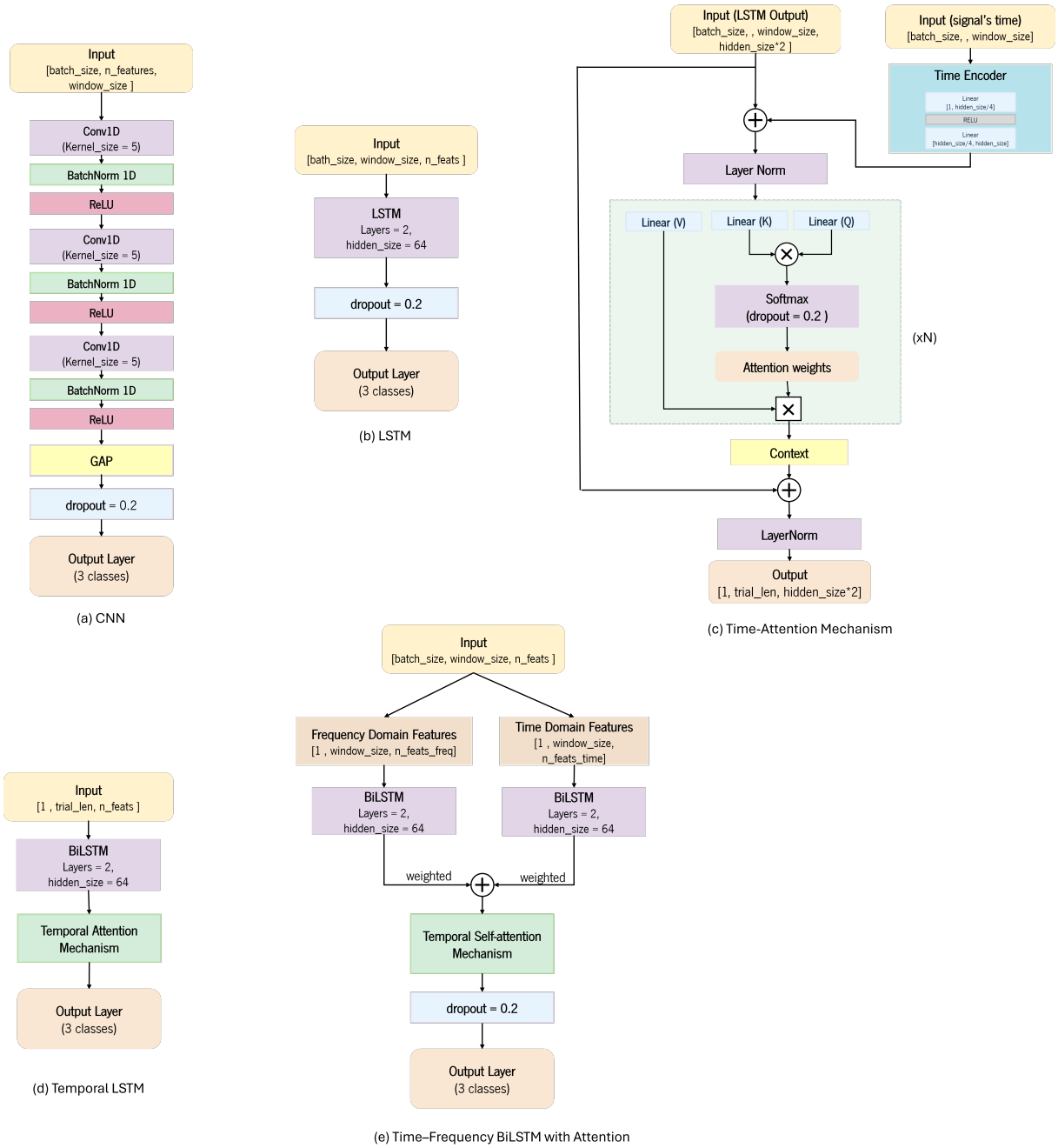


Figure 84: Proposed Architectures

previous windows. Hidden and cell states were reinitialized at the beginning of each new trial to prevent information leakage across trials or participants. The hidden size was fixed at 64 to provide sufficient modeling capacity without overparameterizing the network, and a dropout rate of 0.2 was applied between layers to mitigate overfitting. Figure 84b illustrates the resulting architecture.

LSTM with time-attention The proposed model consists of a bidirectional recurrent backbone followed by a temporal attention mechanism and a final classification layer, as illustrated in Figures 84c and 84d. This model was also applied at the window level, but hidden and cell states were not propagated between consecutive windows. Bidirectional layers were selected based on preliminary experiments, which consistently showed improved performance, likely because the model could exploit both past and future temporal context within each completed input window.

The architecture (Figure 84d) is composed by a two-layer bidirectional LSTM with a hidden size of 64 per direction. The hidden states outputted by the LSTM outputs are then passed to a temporal attention mechanism designed to identify the most informative time steps for fatigue estimation. The temporal attention module (Figure 84c) augments the Bidirectional Long Short Term Memory (BiLSTM) output with explicit time information through a learned time encoding derived from the normalized sequence index. This encoding is added to the recurrent features and normalized before computing query, key, and value projections. Scaled dot-product attention is then applied to weight the temporal contributions of the sequence, allowing the model to emphasize time intervals that are most informative for fatigue estimation. Dropout and residual connections with Layer Normalization are used to improve generalization and training stability.

The output of the attention mechanism preserves the temporal structure of the input and is finally passed to the output layer, which maps the learned representation to the three fatigue classes. The output layer is implemented as a linear projection, followed by a softmax operation during training to obtain class probabilities.

Time–Frequency LSTM with Attention The proposed network (Figure 84e) adopts a dual-branch recurrent architecture to jointly exploit complementary time- and frequency-domain information extracted from the sEMG signal. After feature extraction, time-domain and frequency-domain features are processed in parallel by two independent BiLSTM branches, each composed of two layers with a hidden size of 64. This design allows each branch to model temporal dependencies specific to its feature set, avoiding premature fusion at the input level.

The outputs of the two BiLSTM branches are then combined through a learnable fusion mechanism.

Rather than concatenating features or applying fixed weights, the network learns how to balance the contribution of time- and frequency-domain representations via trainable parameters. This adaptive fusion enables the model to emphasize the feature domain that is most informative for fatigue estimation, potentially in a class-dependent manner.

The fused representation is subsequently processed by a temporal self-attention module, which reweights the sequence along the time axis to highlight temporally relevant segments for classification. To incorporate temporal information without directly coupling absolute time with the feature representations, time is injected exclusively within the attention mechanism using a lightweight multi-layer perceptron–based time encoder.

Because trial durations vary substantially across subjects and tasks, the time index x is normalized using a logarithmic transformation,

$$t_{\text{norm}} = \log(0.1\,x + 1), \tag{6.1}$$

which bounds the time encoding to a compact range while preserving temporal ordering and reducing sensitivity to large inter-trial duration differences.

Finally, a dropout layer with a rate of 0.2 is applied after the attention module and the resulting representation is passed to a linear output layer that produces class scores for the three fatigue states.

6.3.7 Implementation Details

Model training lasted a maximum of 50 epochs. Early stopping was used, and the criterion was the lack of improvement in validation loss for 10 epochs.

All models were optimized using the AdamW optimizer. Two loss functions were evaluated to address class imbalance: weighted cross-entropy and focal loss. In the weighted cross-entropy setting, class weights were computed from the training set to upweight underrepresented fatigue states. Focal loss further reduced the contribution of easy samples while emphasizing difficult or misclassified ones, using a focusing parameter $\gamma = 2$, with α set to balanced class weights derived from training set frequencies. A Reduce-on-Plateau learning rate scheduler was used, monitoring validation performance and reducing the learning rate by a factor of 0.5 when no improvement was observed for 8 epochs, down to a minimum threshold of 10^{-7}. The initial learning rate was set to 10^{-3}.

Batch size was also included in the hyperparameter search and limited to small powers of two to ensure efficient GPU utilization [224]. A batch size of 1 corresponds to providing the full trial, *i.e.*, the complete sequence of windows from a single trial to the LSTM. For larger batch sizes, each mini-batch contains

multiple windows, which may originate from different trials and may be shuffled at the trial level. However, windows belonging to the same trial are kept in their original temporal order to preserve the temporal structure of the signal. Table 35 summarizes the explored hyperparameters.

Model Evaluation Metrics

The models were evaluated using accuracy, macro-averaged F1-score, and weighted F1-score, computed at the epoch selected by early stopping. Accuracy is reported for completeness but is not sufficient on its own, as it can be misleading in imbalanced datasets by being dominated by the majority class [225].

The macro F1-score assigns equal weight to all classes, independently of their frequency, and was therefore used as the primary metric during training and model selection to ensure balanced performance across fatigue states. In contrast, the weighted F1-score accounts for class prevalence and was used in the final analysis to better reflect overall performance under the observed class distribution. Reporting both metrics provides a complementary view of class-balanced performance and real-world behavior.

Training and inference times were also considered. However, because the grid search was executed across multiple machines, training times are not directly comparable. For this reason, only test-time inference durations are reported, as all models were evaluated on the same hardware, allowing a fair comparison of deployment efficiency.

Table 35: Hyperparameters explored in the grid search

Category		Hyperparameter	Values
Training		Batch size	1[a], 16, 32
		Loss function	Weighted cross-entropy, focal
Data preprocessing	Feature extraction	Feature window size (s)	0.2, 0.5, 1, 2, 4
		Feature overlap (%)	0, 25, 50, 75
	Model input	Input window size (s)	10, 15
		Input overlap (%)	0, 25, 50, 75
Architecture		Bidirectional[b]	True, False[c]
		Dropout	0.2

[a] Value used for full-trial experiments [b] Applied only to LSTM-based models,[c] Applied only to standard LSTM model.

The whole pipeline was implemented using Pytorch and Pytorch Lightning libraries in a Python environment. Grid search experiments were divided by multiple personal computers. The test phase was carried

out on a personal laptop with the following specifications: GPU: 1x Nvidia GeForce RTX 4060; CPU: Intel(R) Core(TM) Ultra 9 185H @ 2.5GHz (16 core, 22 threads); RAM: 16 GB.

6.3.8 Results

Exploratory Data Analysis

Feature relevance and information content Feature ranking based on average absolute Cliff's delta (Figure 85a) revealed that frequency-domain features carried more information about fatigue level than purely amplitude-based measures. Eight features exhibited non-negligible effect sizes: MDF, MNF, peak frequency, frequency ratio, time-domain Shannon entropy, zero-crossing, SSC, and MAV slope. In contrast, amplitude-related features including RMS, MAV, variance, maximum amplitude, iEMG, IQR, and energy showed negligible effects and were excluded from further analysis. This ranking indicates that fatigue-related information in this dataset is primarily encoded in spectral shifts, signal complexity, and temporal variation patterns rather than absolute activation magnitude. To reduce redundancy, inter-feature correlation was then assessed among the retained features. Since MDF and MNF were highly correlated ($r = 0.96$), only MDF was retained due to its larger effect size. The final feature set used for neural network training therefore consisted of MDF, peak frequency, frequency ratio, time-domain Shannon entropy, zero-crossing, SSC, and MAV slope. Figure 147 in Appendix F presents the trial ranking based on the feature significance rate computed across all window lengths and overlaps. The least significant trials corresponded to the movements targeting the deltoid posterior, namely shoulder extension.

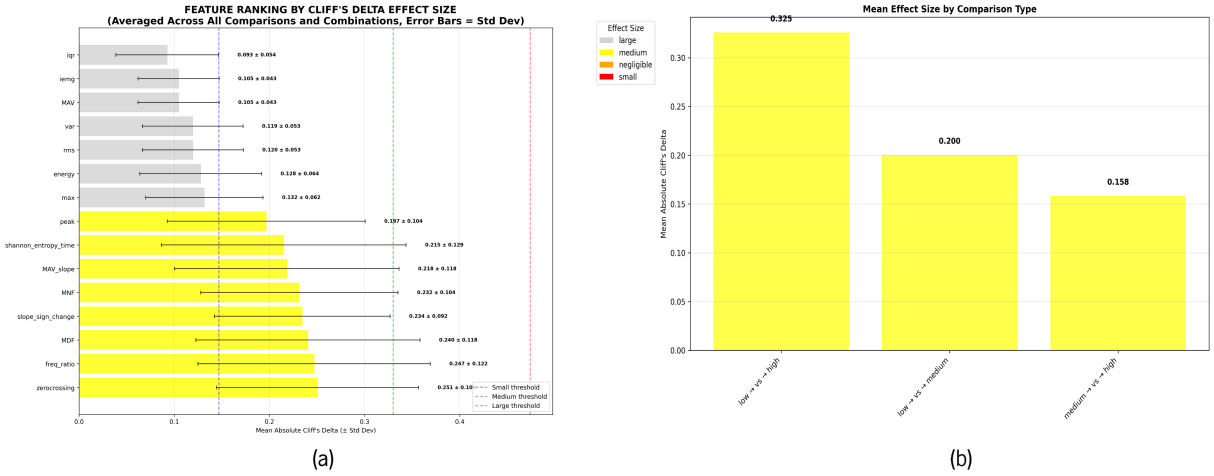
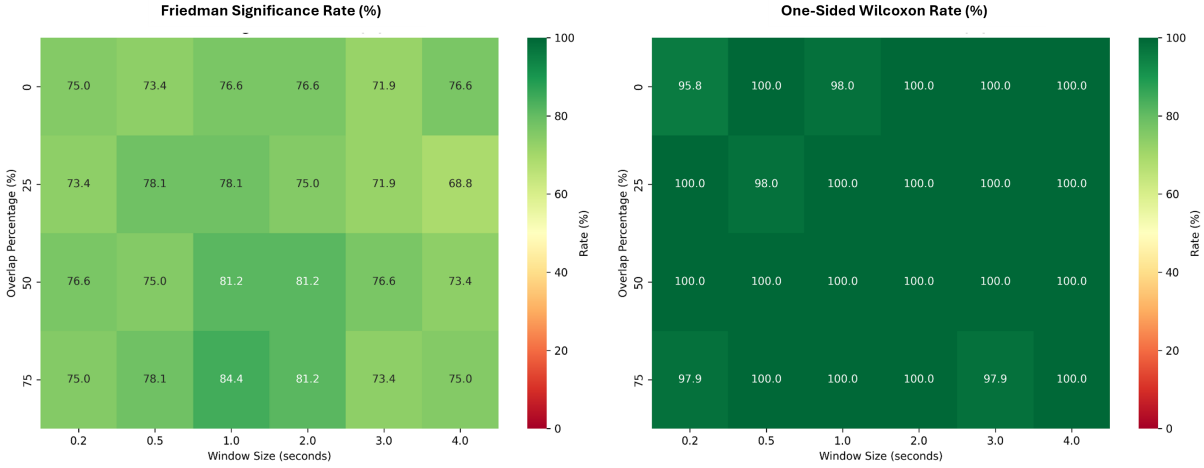


Figure 85: Mean absolute Cliff's delta effect size computed across subjects and trials. (a) Feature ranking obtained by averaging across all trials and combinations, with error bars indicating the standard deviation; (b) average Cliff's delta across sEMG features for each fatigue-level comparison.

Presence of fatigue and separability of fatigue classes Figure 86a shows the Friedman significance rate across feature window sizes and overlaps. The results show that fatigue-related changes are present in the dataset, in a rate of 70 to 85%, depending on the window overlap combination. The effect size analysis based on Cliff's delta, presented in figure 85b confirmed that these differences are also associated with non-negligible, medium, effect sizes ($0.33 \leq |\delta| < 0.474$). Separability was highest between the no-fatigue and fatigue classes (mean $|\delta| = 0.325$). The no-fatigue and transition-to-fatigue class comparison showed lower separation (mean $|\delta| = 0.2$), while the transition-to-fatigue and fatigue class comparison yielded the smallest effect size (mean $|\delta| = 0.158$), indicating overlapping feature distributions. This reflects a gradual fatigue progression rather than a sharp class boundary.



(a) (b)

Figure 86: (a) Heatmap of statistical significance across window length and overlap combinations; (b) Significance rate heatmap based on a one-sided Wilcoxon test, confirming whether the observed differences follow the expected direction.

Consistency of feature trends with the literature Figure 86b shows that post-hoc one-sided comparisons confirm that feature changes consistently follow trends reported in the literature (significance rate > 95.8%). This means that frequency-related features predominantly decreased with increasing fatigue, while complexity- and time-related measures tended to increase.

Duration of fatigue phases and variability of transitions Figure 87 presents the distribution of the self-perceived fatigue levels across all dataset (Figure 87a) and across muscles/trials (Figure 87b). The classes durations revealed that fatigue phases vary considerably across subjects and muscles/trials but follow relatively consistent patterns across muscle groups. Non-fatigue phases tended to be shorter

and less variable, while transition-to-fatigue and fatigue phases exhibited longer durations and increased dispersion. Transition-to-fatigue class is the one with the highest number of outliers, followed by the fatigue class, suggesting variability in how subjects perceive and report fatigue onset. Across muscles, the ones with higher variability are the trial concerning the shoulder extension (deltoid posterior, T3 and T4). Figure 148, in appendix F, shows the duration distribution across trials, highlighting substantial variability across movements, with trials T3 and T4 (shoulder extension) presenting the longest and most variable durations, whereas T1, T2, T7, and T8 are shorter and more consistent, indicating differences in task demand and fatigue progression.

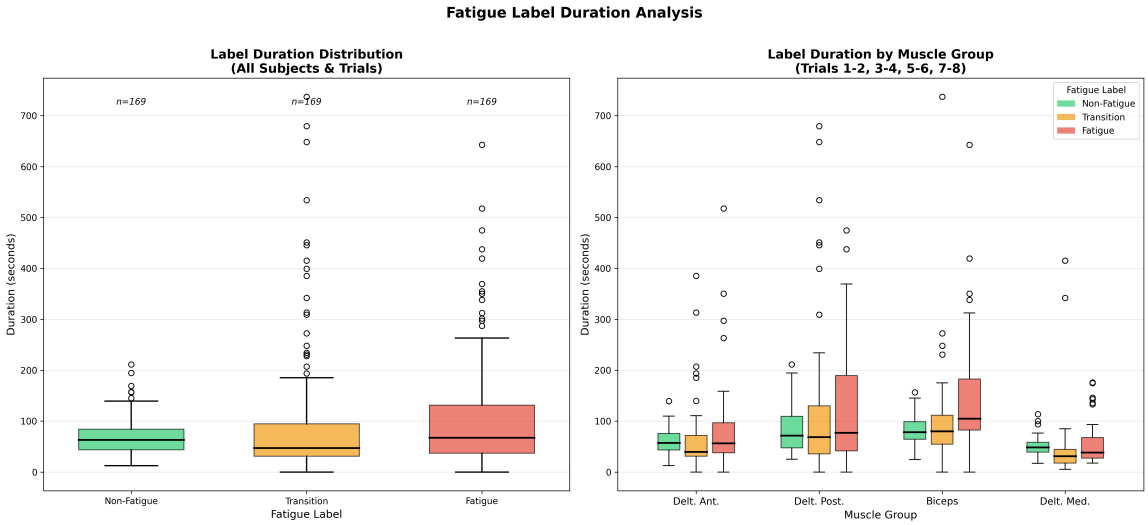


Figure 87: Distribution of self-perceived fatigue label durations across subjects and trials, shown overall and stratified by muscle group.

Hyperparameter Optimization Results

Each architecture is analyzed separately, with results reported as a function of its specific hyperparameter configurations. Performance is presented using both summary tables and box plots. The box plots illustrate the distribution of weighted F1-scores across hyperparameter combinations, allowing a clearer understanding of how performance varies with different design choices and how sensitive each architecture is to hyperparameter selection, as reflected by both central tendency and dispersion.

Table 36 reports the best-performing models, ranked by validation weighted F1-score. For clarity and space constraints, only the top three configurations per architecture are shown.

The time–frequency LSTM with attention achieved the highest performance overall, with the top three configurations attaining similar weighted F1-scores around 0.77. All top-performing models were trained with a batch size of 1, corresponding to full-trial inputs, and used weighted cross-entropy (WCE) as the loss

Table 36: Top three models per architecture ranked by validation weighted F1. $F1_m$ and $F1_w$ denote the macro-averaged and weighted F1-scores, respectively.

Arch.	Rank	Batch	Loss	$Feat_w/Feat_{ov}$	Win	Ov	Drop	Bi	$F1_m$	$F1_w$	Acc
TF LSTM with attention	#1	1	WCE	1.0/0.0	10	0.5	0.2	T	0.731	0.778	0.766
	#2	1	WCE	0.2/0.25	10	0.5	0.2	T	0.733	0.775	0.760
	#3	1	WCE	0.5/0.75	15	0.5	0.2	T	0.738	0.775	0.764
	#52	16	WCE	2.0/0.0	10	0.5	0.2	T	0.663	0.738	0.734
LSTM with time-attention	#1	1	WCE	0.2/0.0	15	0.5	0.2	T	0.729	0.770	0.771
	#2	1	WCE	2.0/0.25	10	0.5	0.2	T	0.729	0.764	0.752
	#3	1	WCE	1.0/0.0	15	0.5	0.2	T	0.729	0.762	0.754
CNN	#1	32	WCE	0.1/0.25	15	0.5	0.2	F	0.443	0.518	0.544
	#2	16	WCE	0.5/0.75	10	0.75	0.2	F	0.432	0.500	0.509
	#3	16	WCE	0.1/0.5	10	0.5	0.2	F	0.375	0.492	0.522
LSTM	#1	1	WCE	4.0/0.25	10	0.5	0.2	T	0.651	0.709	0.727
	#2	1	WCE	4.0/0.25	10	0.5	0.2	F	0.651	0.709	0.727
	#3	1	Focal	2.0/0.0	15	0.5	0.2	T	0.654	0.704	0.717

function. In contrast, the optimal combinations of feature window size ($Feat_w$), feature overlap ($Feat_{ov}$), and input window length (Win) varied across architectures, suggesting that these hyperparameters interact with the model structure rather than exhibiting a single optimal configuration. The worst performing architecture was the CNN. In addition to the top-ranked models, the LSTM with time-attention ranked 52 was also added to the results, as it represents the best-performing configuration whose input was not a full trial and was therefore considered worth evaluating for control-oriented use.

CNN The CNN-based models achieved low performance across all evaluated configurations, with weighted F1-scores consistently below 0.55 and macro F1-scores clustering around 0.35. Figure 88 reports the distribution of accuracy, weighted F1-score, and macro F1-score.

Given the uniformly low performance observed across all hyperparameter combinations, no further detailed analysis is reported for the CNN architecture.

LSTM Figure 89 shows the distribution of accuracy, weighted F1-score, and macro F1-score obtained across all LSTM experiments. Performance varies substantially across configurations, with accuracy and

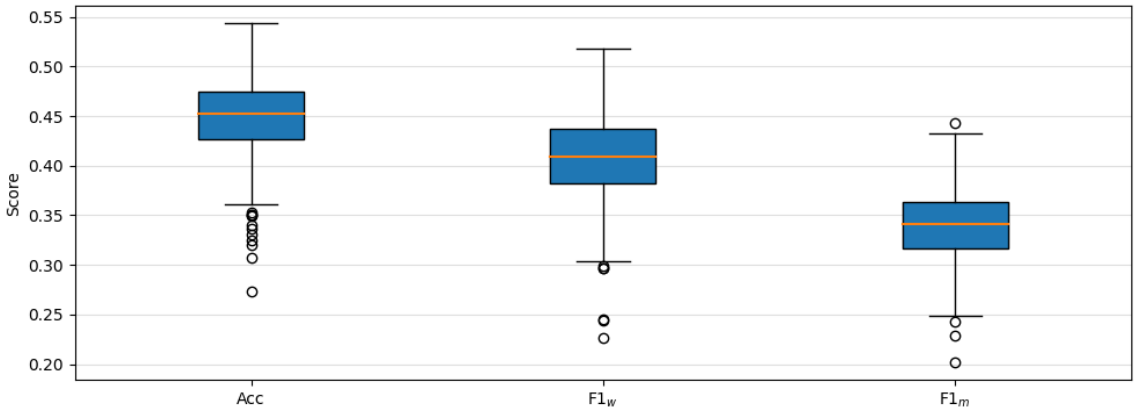


Figure 88: Distribution of performance metrics (accuracy, weighted F1-score, and macro F1-score) across all CNN experiments.

weighted F1-scores reaching 0.7 in several cases.

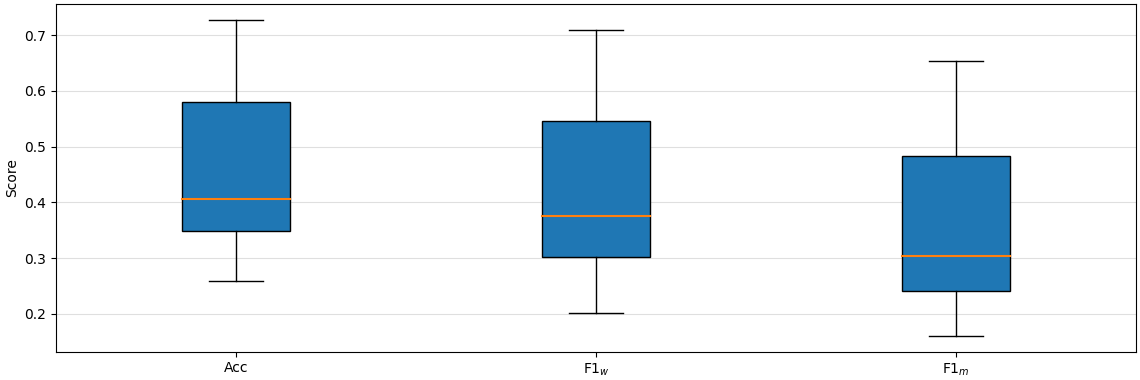


Figure 89: Distribution of performance metrics (accuracy, weighted F1-score, and macro F1-score) across all LSTM experiments.

Among all evaluated hyperparameters, batch size has the strongest impact on performance. As shown in Figure 90, models trained with a batch size of 1 achieve the highest validation scores, with some configurations exceeding a weighted F1-score of 0.7. In contrast, batch sizes of 16 and 32 lead to substantially lower performance, with most results falling below 0.4. The choice of loss function has a comparatively minor effect: weighted cross-entropy and focal loss yield similar score distributions, although weighted cross-entropy shows better results in the batch size 1 setting.

Since the results obtained with batch sizes of 16 and 32 were close to random performance, the remaining analysis for this architecture focuses exclusively on models trained with a batch size of 1.

Figure 91 shows the effect of feature window length and overlap on the weighted F1-score for the LSTM models trained with a batch size of 1. The best-performing configuration uses a 1 s feature window

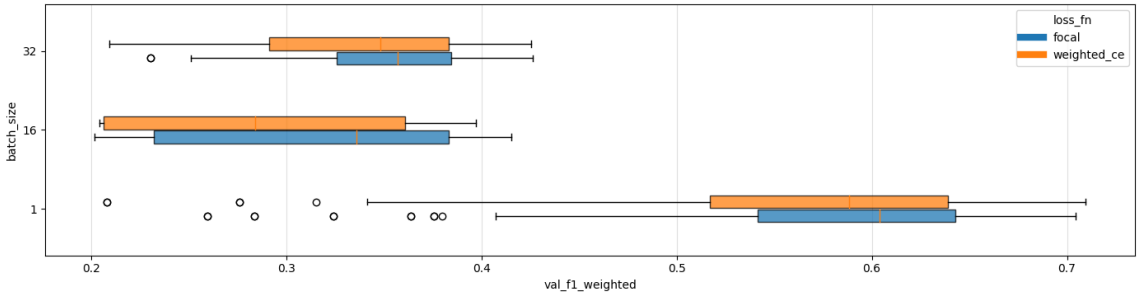


Figure 90: Distribution of validation weighted F1-scores for LSTM experiments across batch sizes and loss functions (weighted cross-entropy and focal loss)

with 0% overlap, followed by a 4 s window also with 0% overlap. Interestingly, the lowest performance is observed for the same 4 s window when combined with a higher overlap of 75%. Overall, overlap appears to influence performance more strongly than feature window length, with lower overlaps generally yielding better results. However, the effects are not consistent across configurations, and performance remains spread across window–overlap combinations, suggesting that these parameters play a secondary role compared to other factors.

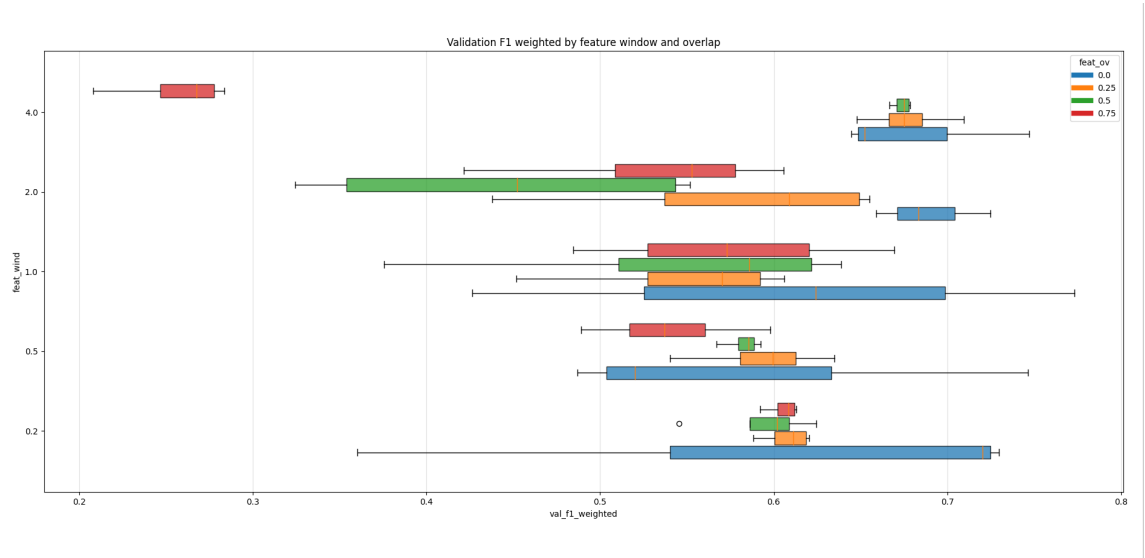


Figure 91: Distribution of validation weighted F1-scores for LSTM experiments across feature windows and overlap for batch size 1.

LSTM with time-attention Figure 92 shows the distribution of accuracy, weighted F1-score, and macro F1-score obtained across all LSTM experiments. Performance varies substantially across configurations, with accuracy and weighted F1-scores varying between 0.4 to 0.75.

As depicted in Figure 93 batch size equal to 1 yielded the best results, while a batch size of 16 followed

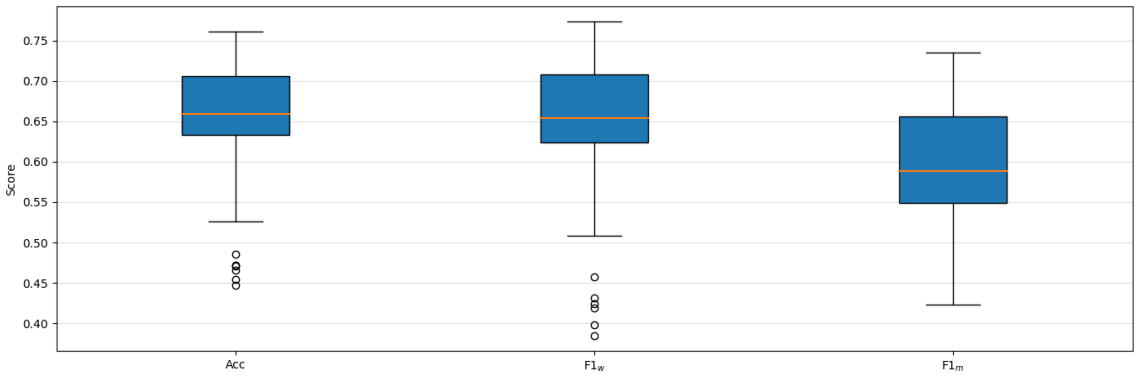


Figure 92: Distribution of performance metrics (accuracy, weighted F1-score, and macro F1-score) across all LSTM with time-attention experiments.

with performances ranging between 0.6 and 0.74. In contrast, a batch size of 32 resulted in the lowest performance, with all models remaining below 0.7. Overall, weighted cross-entropy consistently led to the best-performing models across both batch sizes of 1 and 16, outperforming focal loss in this setting.

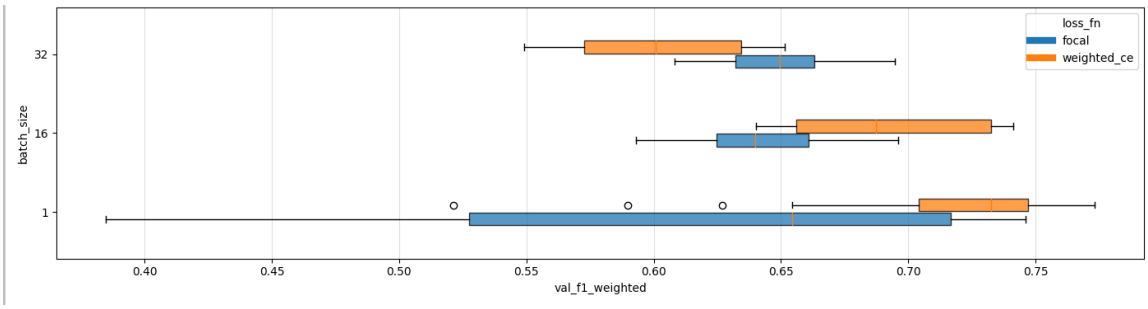


Figure 93: Distribution of validation weighted F1-scores for LSTM experiments across batch sizes and loss functions (weighted cross-entropy and focal loss)

Regarding feature preprocessing, for a batch size of 1 (Figure 94) the best-performing configuration corresponds to a 1 s feature window with 0% overlap. Configurations using a 25% overlap with feature windows of 0.5, 1, and 2 s also achieved relatively good performance. However, performance remains spread across window–overlap combinations, suggesting that these parameters play a secondary role compared to other factors. A similar variability in performance was observed for the batch size of 16, as it can be depicted in Figure 149 in appendix F.2.

Time-Frequency LSTM with Attention Figure 92 reports the distribution of accuracy, weighted F1-score, and macro F1-score across all LSTM experiments, showing considerable variability across configurations, with accuracy and weighted F1-scores spanning approximately 0.60 to 0.778.

Similar to the previously evaluated architectures, weighted cross entropy consistently outperformed

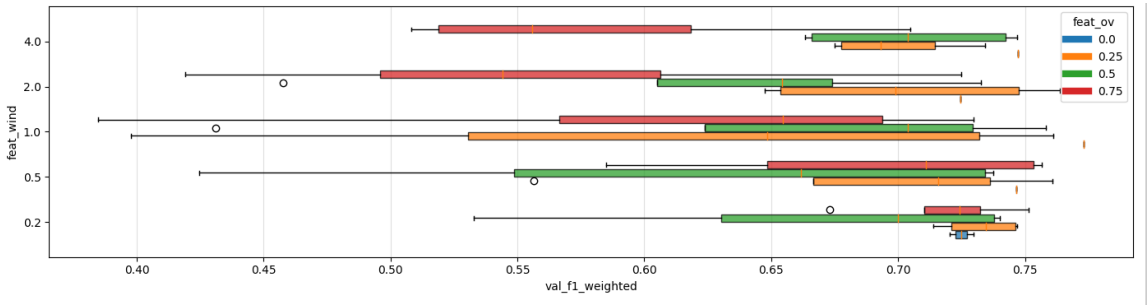


Figure 94: Distribution of validation weighted F1-scores for the LSTM with time-attention (batch size = 1) across feature window sizes and overlaps.

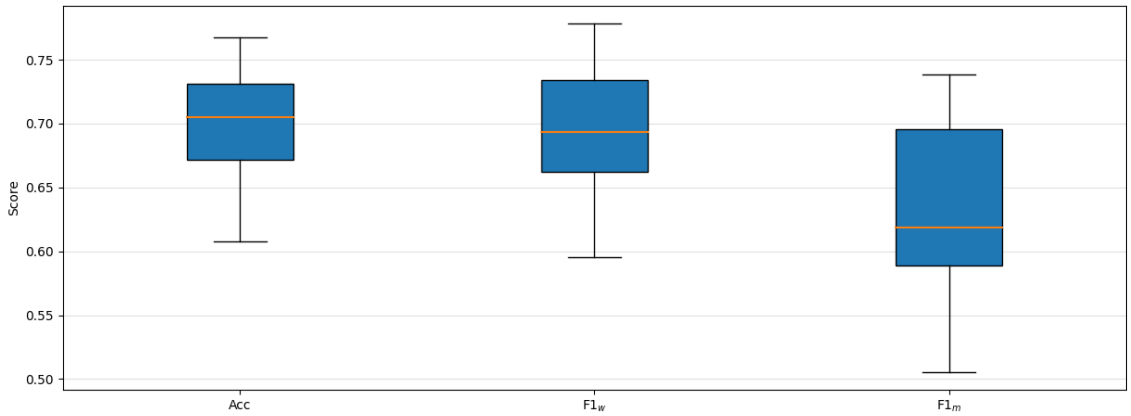


Figure 95: Distribution of performance metrics (accuracy, weighted F1-score, and macro F1-score) across all Time-Frequency LSTM with Attention experiments.

focal loss, leading to superior training outcomes, as illustrated in Figure 96. The figure also shows that a batch size of 1 yields the highest overall performance. Nevertheless, batch sizes of 16 and 32 were still able to produce several models achieving weighted F1-scores above 0.7.

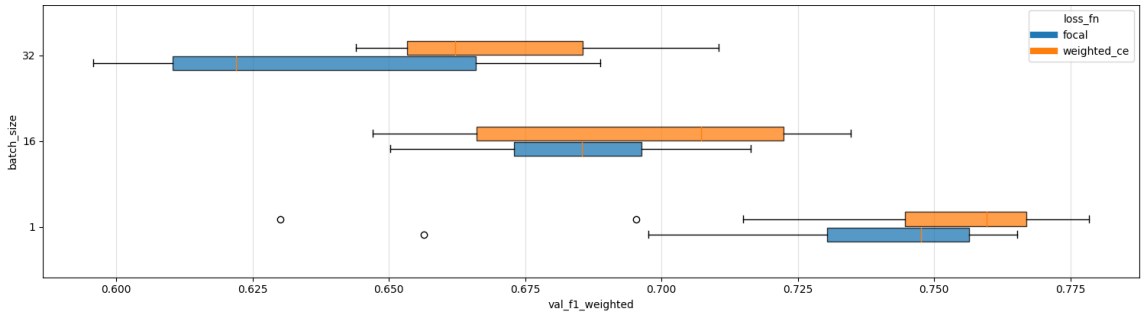


Figure 96: Distribution of validation weighted F1-scores for Time-Frequency LSTM with Attention experiments across batch sizes and loss functions (weighted cross-entropy and focal loss)

Figure 97 illustrates the effect of feature computation window size and overlap across all batch con-

figurations. No clear or consistent relationship between window overlap and weighted F1-score can be identified. Figure 150, reported in Appendix F.2, provides the corresponding batch-wise F1-score distributions, confirming the same observation with greater detail.

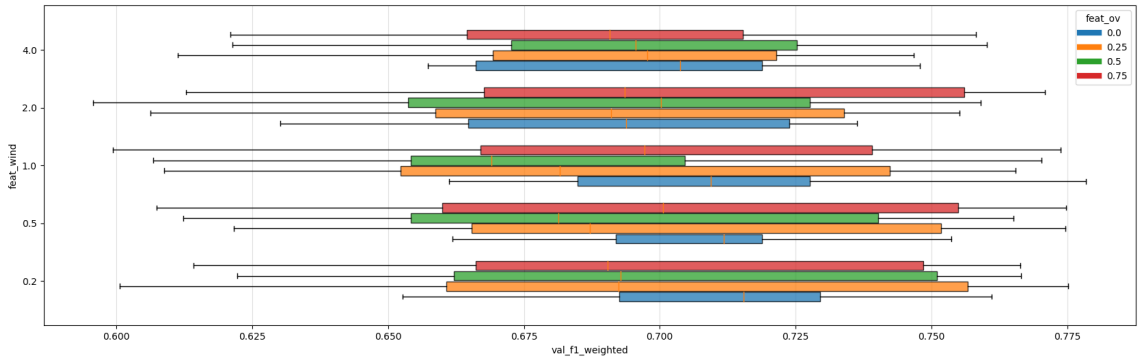


Figure 97: Validation weighted F1-score distributions of the Frequency–Time LSTM with attention as a function of feature window size and overlap for all batch sizes

Final Test Results

Since results for CNN were close to random performance, no test was conducted for this architecture as it does not bring relevant information. Therefore, Table 37 presents only the test performance for the LSTM, LSTM with time-attention, and Time-Frequency LSTM with Attention architectures. The configurations evaluated on the test set correspond to those identified as optimal during validation, allowing assessment of their generalization capability on unseen data.

The Time-Frequency LSTM with Attention achieved the best overall test performance. Its top configuration reached an accuracy of 0.790 and a weighted $F1$-score of 0.792, confirming that the performance gains observed during validation translated consistently to the test set. The macro $F1$-score of 0.766 indicates balanced performance across fatigue classes, with no evidence of overfitting to the dominant class. Precision and recall remained high and well aligned, further supporting stable generalization.

The LSTM with time-attention showed a clear improvement over the plain LSTM. Its best test configuration achieved a weighted $F1$-score of 0.732 and an accuracy of 0.730. However, validation weighted $F1$ ranged from 0.762 to 0.770, indicating a small performance decrease on the test set.

Standard LSTM models exhibited the lowest test performance among the recurrent architectures. Weighted $F1$-scores remained around 0.60 and accuracy around 0.63, confirming that the limitations observed during validation persisted on the test set. These results indicate insufficient modeling of temporal relevance when attention mechanisms are absent, even when optimal hyperparameters are selected.

Table 37: Test performance and inference time for the evaluated architectures.

Architecture	**Rank**	**Batch**	**Feat. win / ov**	**Acc.**	**Prec.**	**Rec.**	$F1_{\mathbf{ma}}$	$F1_{\mathbf{w}}$	**Inf. time[a] (ms)**
TF LSTM + Attention	#1	1	1.0 / 0.00	0.761	0.767	0.810	0.747	0.771	1.40
	#2	1	0.2 / 0.25	**0.790**	**0.796**	**0.815**	**0.766**	**0.792**	2.00
	#3	1	0.5 / 0.75	0.736	0.770	0.796	0.735	0.741	2.30
	#52	16	2.0 / 0.00	0.629	0.648	0.670	0.597	0.596	0.70
LSTM + time-attention	#1	1	0.2 / 0.00	0.693	0.731	0.750	0.685	0.674	1.87
	#2	1	2.0 / 0.25	0.717	0.762	0.785	0.720	0.717	1.20
	#3	1	1.0 / 0.00	0.730	0.773	0.786	0.728	0.732	0.98
LSTM	#1	1	4.0 / 0.25	0.632	0.670	0.670	0.600	0.596	0.64
	#2	1	4.0 / 0.25	0.632	0.670	0.670	0.600	0.596	0.65
	#3	1	2.0 / 0.00	0.629	0.648	0.670	0.597	0.596	0.59

[a] Batch = 1 reports time per full trial. Batch > 1 reports time per NN input window.

Inference time was not a limiting factor for any evaluated model. However, real-time deployment was constrained by the temporal context required by the best-performing configurations, since the highest-performing models used full-trial inputs and were therefore unsuitable for online control. Standard LSTM models were the fastest, with inference times below 0.7 ms per trial. Attention-based models required between 1.0 and 2.3 ms per trial.

Figure 98 shows the confusion matrix of the best performing model on the test set, the Time Frequency LSTM with Attention, for the three fatigue classes. Predictions are concentrated along the diagonal, but off-diagonal errors remain substantial, indicating that the model captures the overall fatigue ordering while still making frequent confusions between adjacent states. Importantly, all misclassifications occur between neighbouring classes, with no sample jumping directly from No Fatigue to Fatigue or vice versa, suggesting the model respects the underlying fatigue progression even when it errs. Within this pattern, the transition-to-fatigue class is the main source of ambiguity, accumulating errors from both directions due to its overlap with the neighbouring states. The No Fatigue and Fatigue classes are each identified more consistently, though both show non-negligible confusion with the transitional state. Notably, Fatigue samples are never misclassified as No Fatigue, meaning severe fatigue is never fully missed, only underestimated.

The LSTM with time-attention ranked 52 was also evaluated on the test set to assess its suitability for the fatigue-aware control strategy. Unlike the best-performing model, this architecture used reduced temporal context and is therefore more compatible with online deployment. As expected, this constraint

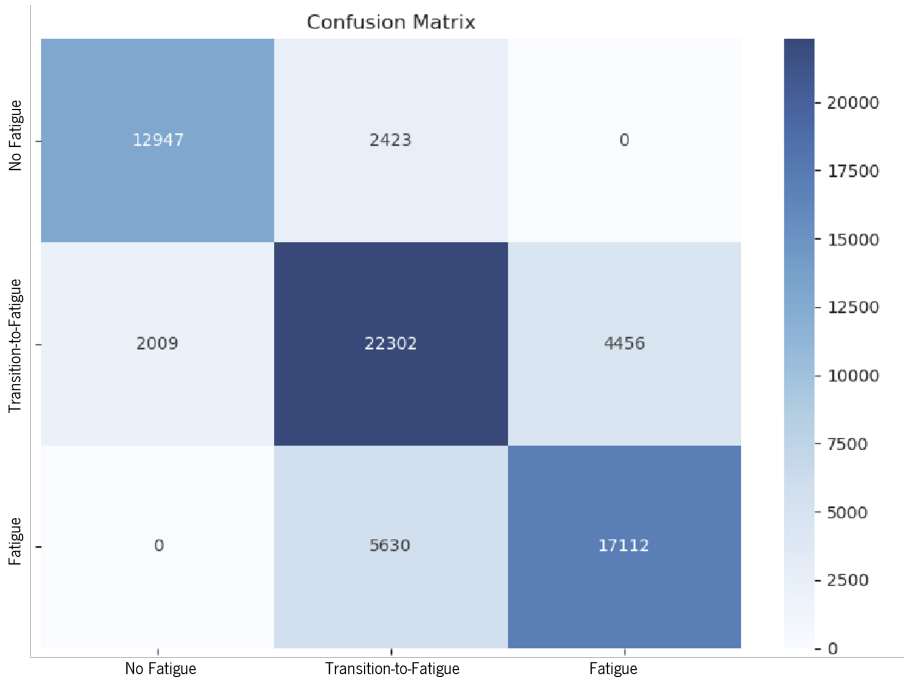


Figure 98: Confusion Matrix for the best performing model (TF LSTM with attention).

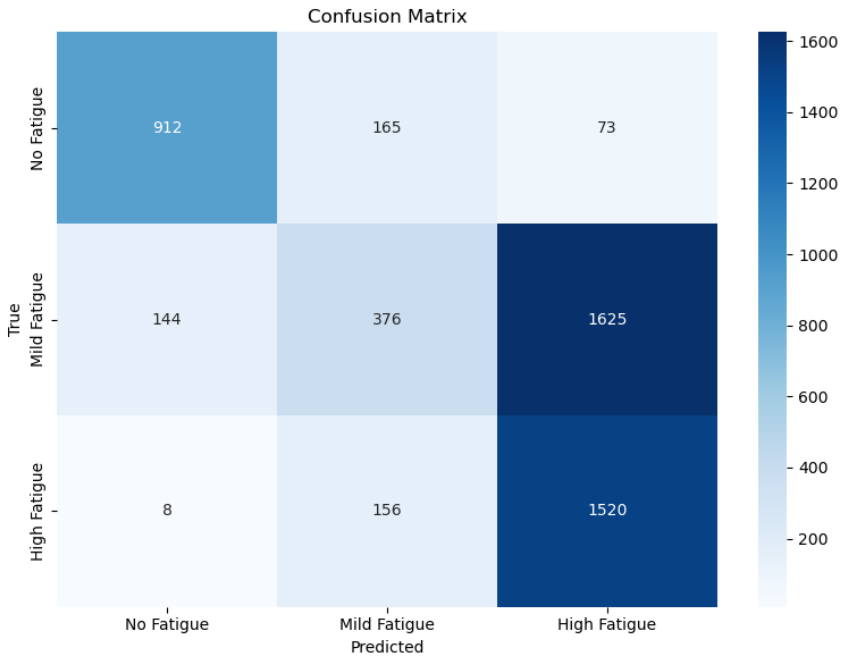


Figure 99: Confusion matrix of the #52 TF LSTM with attention evaluated.

reduced its three-class classification performance. The model achieved an accuracy of 0.629, a macro $F1$-score of 0.597, and a weighted $F1$-score of 0.596, indicating a clear degradation relative to models using longer temporal information.

Figure 99 presents the corresponding confusion matrix. The model correctly identified 80.9% of No

Fatigue samples and 77.5% of Fatigue samples, but only 30.1% of Transition-to-Fatigue samples. Most Transition-to-Fatigue samples were classified as Fatigue, showing that the intermediate state was not reliably separated under the reduced temporal context. Therefore, this model should not be interpreted as a reliable three-class fatigue estimator.

6.3.9 Discussion

Exploratory Data Analysis

Feature relevance and information content In dynamic, cyclic movements, amplitude-based sEMG features are influenced by task biomechanics rather than fatigue alone [253]. Because muscle force production varies across the movement cycle, sEMG amplitude changes systematically with movement execution [72], even in non-fatigued conditions. As a result, amplitude-related features such as RMS, MAV, variance, or iEMG largely reflect mechanical and kinematic effects, possibly explaining the negligible effect sizes observed for these features in the Cliff's delta analysis, as observed in figure 85a.

In contrast, frequency-domain features are more directly linked to physiological mechanisms of fatigue, such as changes in muscle fiber conduction velocity, motor unit synchronization, and recruitment strategies [72]. Although these features are not immune to dynamic effects, their relative temporal trends remain more consistent during cyclic tasks, making them more robust markers of fatigue progression than amplitude-based measures in dynamic conditions.

Regarding the trial ranking results, observed in figure 147 in appendix F, Trials 3 and 4, corresponding to shoulder extension, exhibited the lowest statistical significance. This reduced significance is likely related to increased inter- and intra-subject variability. Shoulder extension tasks may had induce compensatory movement strategies, which can be amplified by pre-existing postural patterns like forward head posture, common within the population with desk jobs. In addition, sensor placement may have also contributed to this variability. The posterior deltoid is relatively underdeveloped in many individuals, making its muscle belly harder to consistently localize and increasing the likelihood of suboptimal electrode positioning.

Presence of fatigue and separability of fatigue classes The small effect sizes observed between the transition-to-fatigue and fatigue classes, in figure 85b were expected. Although efforts were made to consistently explain the meaning of each fatigue level to participants and to limit the number of classes, self-perceived fatigue is inherently subjective and strongly influenced by individual background factors, such as training history, habitual physical activity, and tolerance to exertion. As a result, similar physiological states may be labeled differently across subjects, introducing uncertainty to the models' classification.

Despite this limitation, the Friedman tests (figure 86a) revealed statistically significant differences across fatigue classes in 70–85% of the evaluated window–overlap configurations, confirming that fatigue-related changes are consistently present in the data.

Consistency of feature trends with the literature The agreement between the observed feature trends and those reported in the literature supports the physiological validity of the extracted fatigue signatures (figure 86b). Frequency-related features decreased with fatigue, consistent with known reductions in muscle fiber conduction velocity and changes in motor unit synchronization, while complexity-related measures increased as neuromuscular control became more irregular. As described in the Methods section, these results refer only to features with non-negligible effect sizes, meaning that most time-domain amplitude features are not represented in this analysis. In addition, the high significance rates should be interpreted in light of the adopted aggregation strategy to ensure statistical independence. By representing each fatigue class within a trial using median values, short-term fluctuations and local trend reversals visible in the raw temporal trajectories are attenuated. Consequently, the analysis captures robust shifts in central tendency between fatigue states rather than strictly monotonic temporal behavior.

Duration of fatigue phases and variability of transitions The observed variability in the durantions of the transition-to-fatigue and fatigue phase, as seen in figure 87 reflects the combined influence of task effort, muscle function, and individual perception of effort. Non-fatigue phases are generally short, likely because initial exertion rapidly engages the musculature. In contrast, the transition-to-fatigue and fatigue phases tend to last longer, as fatigue develops gradually over time. Habituation to the task and pacing strategies adopted by participants may also contribute to prolonged transition and fatigue phases, as individuals adjust their effort to sustain performance despite increasing fatigue. Differences across muscles further suggest that task biomechanics play a central role in fatigue progression. Shoulder extension tasks exhibited greater variability, likely due to compensatory movement strategies and postural adaptations that redistribute load across muscle groups, delaying or masking localized fatigue. In contrast, movements with more constrained kinematics and clearer muscle activation patterns produced more consistent fatigue timelines.

Hyperparameter Optimization Results

CNN The low performance of the CNN-based models was expected given the nature of the task. While one-dimensional convolutions can be extended by increasing kernel size or stacking layers, CNNs fundamentally model temporal information through fixed local receptive fields. In fatigue estimation, however,

relevant patterns emerge from long-range temporal dependencies and gradual state transitions rather than from localized temporal motifs. Even with larger kernels, a purely convolutional architecture would probably still struggle to capture the global temporal context and progression inherent to fatigue. These results suggest that, in this dataset, purely convolutional architectures struggle to capture the gradual and history-dependent characteristics of fatigue progression, resulting in performance close to chance level across all tested configurations.

LSTM The standard LSTM establishes a strong baseline but shows clear limitations. In the best configurations, accuracy and weighted F1-scores approach 0.7, while macro F1-scores remain notably lower. This gap reflects uneven class-wise performance driven by class imbalance, with the model favoring dominant fatigue states and struggling to discriminate less frequent ones.

Batch size is the main driver of performance. Training with batch size 1 consistently yields the best results, as each batch corresponds to a full trial and preserves the complete temporal evolution of fatigue. This access to long-term temporal context appears critical for fatigue modeling, as fatigue develops gradually rather than instantaneously. Larger batch sizes imply the use of windows, which segment the temporal structure into shorter sequences, limiting the model's ability to integrate slow fatigue dynamics. Weighted cross-entropy outperforms focal loss, suggesting that explicit class reweighting is sufficient, while focal loss may be less effective under noisy, self-reported labels.

Feature window length and overlap play a secondary role. Configurations approximating a full movement cycle (4s windows) tend to perform better, likely because they reduce variability introduced by phase-dependent dynamics. Shorter windows can still be effective when overlap is low, indicating that fatigue-related information can be captured locally but benefits from temporal aggregation.

LSTM with Time attention The LSTM with time attention exhibits similar sensitivity to training configuration, but introduces a meaningful refinement. As in the standard LSTM, weighted F1-scores consistently exceed macro F1-scores, again reflecting class imbalance and difficulty in modeling minority fatigue states. Batch size remains the dominant factor, with batch size 1 yielding the highest performance by preserving full-trial temporal context.

However, time attention partially mitigates the degradation observed at larger batch sizes. By encoding elapsed time, the model retains information about fatigue progression even when trained on shorter sequences. This leads to improved performance for batch sizes 16 and 32 compared to the standard LSTM. The improvement is modest but consistent, suggesting that time awareness helps compensate for reduced temporal context, though it cannot fully replace access to full trials. Weighted cross-entropy remains the

most effective loss, while focal loss does not provide clear advantages.

Feature window size and overlap again show no strong or systematic trends. In this architecture, smaller feature windows below 2 s combined with low overlap seem slightly favored, but overall variability remains high, reinforcing the idea that temporal modeling at the network level dominates over fine-grained feature preprocessing.

Time-Frequency BiLSTM with Attention The time–frequency BiLSTM with attention achieves the best overall performance. By separating frequency-related features from time or complexity-related features and modeling them in independent branches, the architecture preserves domain-specific temporal patterns before fusion. This separation appears beneficial for fatigue estimation, suggesting that different feature domains encode complementary aspects of fatigue progression that are diluted by early fusion.

As with the other architectures, batch size 1 yields the highest scores, confirming the importance of full-trial temporal context. Batch size 16 produces lower but still competitive performance, with weighted F1-scores reaching approximately 0.73. While inferior to batch size 1, this configuration offers a practical advantage by enabling online operation, unlike batch size 1 which is restricted to offline use. Weighted cross-entropy again emerges as the most reliable loss function.

No clear relationship is observed between feature window size and overlap. The best-performing configurations generally rely on shorter feature windows between 0.2 and 1.0 s, with varying overlap. This supports the interpretation that, once sufficient temporal structure is captured by the architecture itself, detailed feature windowing choices play a secondary role in determining performance.

Final Test Results

The test results highlight both the potential and the current limitations of fatigue classification based on a single sEMG channel. Attention-based recurrent architectures clearly outperform plain LSTM models, confirming that explicitly modeling temporal relevance is necessary to extract fatigue-related information from noisy and highly variable sEMG signals. Nevertheless, even the best-performing configuration reaches a weighted $F1$-score below 0.80 on the test set, consistent with the validation results and indicating that generalization remains limited in a subject- and muscle-independent setting. Importantly, the best-performing model achieved a test accuracy of 79.0%, thereby satisfying the predefined fatigue classification requirement of accuracy above 70%.

The confusion matrix (Figure 98) further clarifies the nature of these limitations. Misclassifications are not uniformly distributed but are concentrated between adjacent fatigue states. In particular, the

transition-to-fatigue class is the dominant source of uncertainty, being confused with both No Fatigue and Fatigue. This behavior is consistent with the gradual and continuous nature of fatigue development, and with exploratory data analysis conducted. Notwithstanding, errors like predicting No Fatigue with Fatigue and vice-versa seem to be rare. This structured error pattern suggests that the model captures the ordinal relationship between fatigue states, even if precise class boundaries remain difficult to learn.

The evaluation of the lower-ranked model operating on reduced temporal context further reinforces this interpretation. Although its performance degrades compared to full-trial inputs, it still maintains a weighted $F1$-score close to 0.60, indicating that meaningful fatigue-related information is present even in shorter windows. However, the observed performance gap confirms that long-term temporal aggregation remains critical, and that reducing context leads to a tangible loss in discriminative power. The corresponding confusion matrix (Figure 99) further clarifies this degradation, showing that reduced temporal context mainly compromises the separation of the Transition-to-Fatigue class, while the direct confusion between No Fatigue and Fatigue remains comparatively low. This suggests that, despite its limited reliability as a three-class estimator, the model still preserves the distinction most relevant for fatigue-aware assistance: separating non-fatigued from clearly fatigued states.

Although the absolute performance remains moderate, the obtained results are superior to those reported in [80] when evaluation is conducted under a per-subject split, which achieved accuracies of 71.68% and 71.24% on datasets D1 and D2, respectively. Under this setting, the authors report a comparable drop in classification performance relative to per-sample splits, reflecting the increased difficulty of subject-independent fatigue recognition. In contrast, many results reported in the literature are not directly comparable, as they rely on per-sample or randomly shuffled splits that allow subject-specific patterns to leak into both training and testing sets, leading to substantially optimistic estimates.

Notwithstanding, the results still indicate that relying on a single sEMG channel is likely insufficient for robust fatigue estimation. The strongest performance was obtained when bidirectionality was enabled and full trials were provided as input, which inherently constrains the models to offline prediction. While this limits real-time applicability, such predictions can still be integrated into offline ergonomic reporting tools, such as the ErgoReport (chapter 4, to support post-task analysis.

To move toward online operation, one possible direction is the use of knowledge distillation, where a student model trained on shorter windows learns to estimate fatigue under the supervision of a full-trial teacher model. A similar approach was explored by Das et al. [254] in the context of trajectory forecasting, where a student constrained to shorter observations is supervised by a teacher with access to longer sequences, exploiting the reduced uncertainty of the teacher's task to guide the student. However, given

the already limited performance of the full-trial models, further degradation in accuracy and F1 is likely and should be carefully evaluated.

Notwithstanding, although overall classification performance remains moderate, the explicit inclusion of a transition-to-fatigue class provides an important safety margin. During this phase, participants become aware of increasing effort without being fully fatigued, and misclassifications in this region are less critical than confusing non-fatigue with full fatigue. As such, this intermediate class reduces the risk of premature or delayed fatigue detection and contributes to a more conservative and safer model behavior.

Future work should therefore prioritize richer input representations. One promising direction is the inclusion of synergistic muscles, which may provide complementary fatigue information and improve robustness, albeit at the cost of reduced scalability and increased sensor requirements for wearable deployment. Another relevant direction is the fusion of sEMG with kinematic data, which could help disambiguate fatigue-related changes from movement-induced variability.

6.4 Fatigue-based Control Strategy

The proposed control strategy aims to prevent or delay muscle fatigue during collaborative tasks by actively redistributing muscular effort through an agonist–antagonist mechanism, following the hypothesis that imposing alternative movements by recruiting the antagonist muscle group enables recovery of the fatigued agonist during task execution [2]. Accordingly, the approach integrates real-time muscle fatigue estimation into the human–robot collaboration (HRC) control loop to identify when a muscle is entering a fatigued state and trigger a change in task configuration that shifts muscular demand. At this stage, the implementation should be regarded as a proof-of-concept of the developed sensing, processing, and control pipeline. A more robust controller depends on improving fatigue estimation, as the current models suitable for online deployment have limited performance.

6.4.1 Methods

Control Strategy description

The proposed control strategy relies on a trained DL model that estimates muscle fatigue from real-time sEMG signals and provides this information as a high-level input to the collaborative robot controller. Rather than directly modifying low-level control parameters, the estimated fatigue state is used to trigger discrete adaptations of the task execution configuration.

Inspired by [2], when the model indicates that the currently active muscle is entering a fatigue state, the

robot updates its end-effector pose. This pose change alters the task biomechanics, deliberately shifting the load from the fatigued muscle to its antagonist. As a result, the previously active muscle is unloaded and allowed to recover, while task execution continues uninterrupted.

In this implementation, the anterior and posterior deltoid muscles are monitored, as they constitute a well-defined agonist–antagonist pair and are selectively activated by the two task configurations. The robot alternates between two predefined workstation poses, each designed to predominantly engage one of these muscles. Pose transitions are driven exclusively by the fatigue state estimated online. Figure 100 illustrates the proposed control strategy.

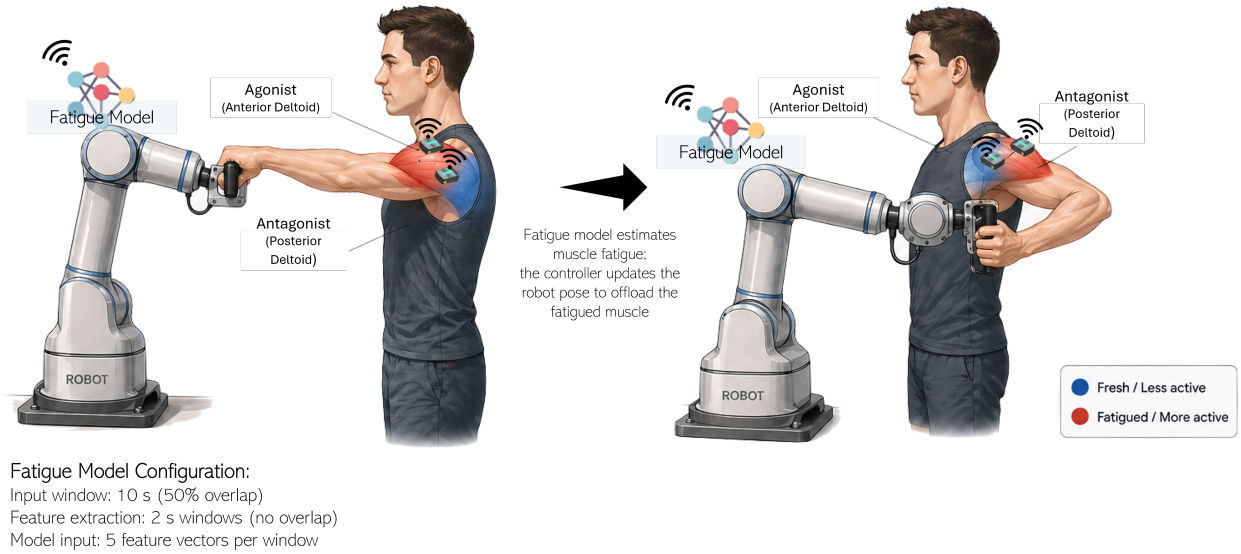


Figure 100: Conceptualization of the control strategy.

During operation, sEMG signals are continuously acquired using the Delsys system and processed by the previously selected fatigue estimation model, namely the TF LSTM + Attention configuration trained with a batch size of 16, a neural network input window of 10 s with 50% overlap, and features computed over 2 s windows with no overlap. This configuration results in an effective update rate of one fatigue prediction every 5 s (0.2 Hz). Although not the highest-performing configuration overall, this model was selected as it does not require a full trial as input, making it suitable for online deployment.

This model showed a conservative error pattern, with fatigue-related states more often overestimated than missed. Therefore, the controller adopted a preventive activation rule, where pose adaptation was triggered when the active muscle was classified as Transition-to-Fatigue. In this context, overestimating fatigue is preferable because it triggers earlier load redistribution, whereas underestimating fatigue may delay assistance and prolong loading of the fatigued muscle.

The online feasibility of the pipeline was verified by measuring both feature extraction and model inference times. Feature extraction was evaluated on 23,928 windows of 2 s, equivalent to 2518 samples at 1259 Hz, from 156 sEMG recordings across 13 subjects. Each window was processed 10 times, totaling 239,280 measurements. Extracting the seven selected features, namely MDF, peak frequency, frequency ratio, time-domain Shannon entropy, zero-crossing, SSC, and MAV slope, required 0.95 ± 0.15 ms per window. Frequency ratio was the most expensive feature, requiring 0.37 ± 0.10 ms, while MAV slope required less than 0.02 ms. Since each prediction uses five feature windows, feature extraction and model inference required approximately 5.45 ms in total, corresponding to 0.11% of the 5 s control update period.

The resulting fatigue predictions govern the timing of pose transitions, forming a closed-loop, fatigue-aware control strategy. By dynamically redistributing muscular load, the controller limits prolonged activation of a single muscle group and mitigates fatigue accumulation, contributing to safer and more sustainable human–robot collaboration.

6.4.2 Implementation Details

The ROS graph depicted in Figure 101 summarizes the nodes, topics and services implemented to develop the control strategy. A detailed description of each topic present in ROS graph is outlined in Table 38.

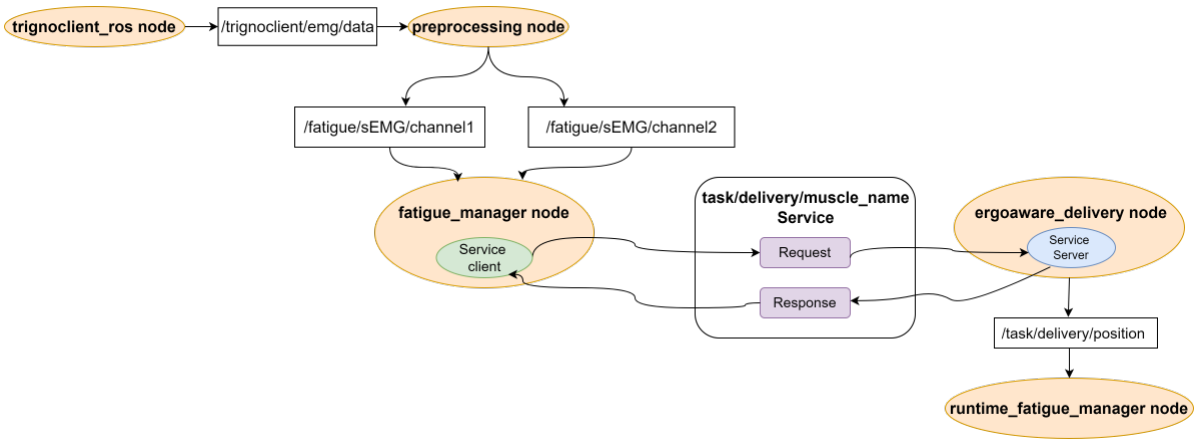


Figure 101: Visualization of ROS computation graph. Schematic of the communication between nodes through topics and services.

The task is initialized in a workstation configuration that predominantly activates the anterior deltoid. Accordingly, the cobot end-effector is positioned at predefined spatial coordinates corresponding to this configuration at the start of each trial.

Table 38: Description of each topic present in ROS graph.

Topics	Description
/trignoclient/emg/data	Contains the data(time+raw data) sent by the sEMG sensor (Trigno Avanti™ system). In this case it only contains data from 2 channels corresponding to the anterior and posterior deltoids.
/fatigue/sEMG/channel1	Contains feature values computed from channel 1 after filtering and normalization.
/fatigue/sEMG/channel2	Contains feature values computed from channel 2 after filtering and normalization.
/task/delivery/position	Contains Pose (position+orientation) where the cobot will move.

Control execution begins with the launch of the **runtime_fatigue_manager** node. This node first calls the **trigno/start** service to initialize the Trigno Avanti™ system (Delsys), enabling real-time acquisition of sEMG signals from the anterior and posterior deltoid muscles. It then triggers the **srv_ preprocessing_EMG1** service, which initializes the **preprocessing node** and subscribes to the **trignoclient/emg/data** topic. Incoming messages from channel 1, corresponding to the anterior deltoid, are buffered for processing.

To support parallel execution, a multithreaded architecture is employed. While channel 1 is active, a dedicated thread preprocesses the sEMG signal by applying filtering, normalization with respect to the subject's MVC values, and subsequent feature computation. The preprocessing pipeline is identical to that used during model training (Section 6.3.5), and feature normalization is performed using the min–max statistics derived from the training set. The resulting feature data stream is published on the **/fatigue/sEMG/channel1** topic.

The **runtime_fatigue_manager** then calls the **start_ergoaware_fatigue_manager** service, which activates the **fatigue_manager** node and subscribes to the processed feature data stream. Incoming samples are accumulated in a sliding window and classified online using the trained LSTM model, with the same window size, overlap, and hyperparameters identified during training.

When the model detects fatigue in the active muscle and the internal flag indicates the anterior deltoid, a pose transition is triggered. The flag is updated to *posterior deltoid*, and the **task/delivery/muscle_name** service is called with request "P". The corresponding service server in the **ergoaware_delivery** node publishes the target pose that activates the posterior deltoid on the **/task/delivery/position** topic. Upon receiving this command, the **runtime_fatigue_manager** executes the position controllers, moving the cobot to the new configuration.

At this point, preprocessing switches to channel 2. The **srv_preprocessing_EMG2** service is called,

the preprocessing thread for channel 1 is terminated, and a new thread is initialized for channel 2. The posterior deltoid sEMG signal is processed using the same pipeline, and the resulting feature values are published on **/fatigue/sEMG/channel2**. The **fatigue_manager** node then classifies fatigue for this muscle.

When fatigue is detected in the posterior deltoid, the flag is reset to *anterior deltoid*, and the **task/delivery/muscle_name** service is called with request "A". The cobot is moved back to the configuration that reactivates the anterior deltoid, which has by then recovered. This alternating process continues in a closed loop until the control strategy is deactivated.

The proposed fatigue-aware control strategy was first validated in simulation using the ROS environment and RViz for 3D visualization of the robot's adaptive behavior. It was subsequently deployed and evaluated in real-time on a UR10e collaborative robot under laboratory conditions to assess its effectiveness in preventing muscle fatigue.

Experimental Protocol

For this proof-of-concept study, five participants (2 male and 3 female; height 1.67 ± 0.11 m; weight 57.76 ± 14.75 kg) were recruited. All participants were right-handed. Each participant was instrumented with two Delsys sEMG sensors placed on the anterior and posterior deltoid of the dominant arm, following the [150] guidelines. These muscles form an agonist–antagonist pair.

To assess the impact of the fatigue based control strategy, the validation protocol simulated an industrial task executed under two conditions: with and without the proposed fatigue management strategy. Each participant performed exactly one trial per condition, resulting in a total of two trials per participant. The task consisted of painting a surface held by the collaborative robot, as illustrated in Figure 102.

Participants performed a collaborative task together with the robot, with the fatigue-aware control strategy enabled. Muscle activity was acquired in real time using the Trigno Avanti™ system and streamed to the HRC framework via TCP/IP communication. The trained DL model continuously estimated the fatigue state of the active muscle. When fatigue was detected, the cobot changed its pose, thereby shifting task execution to the antagonist muscle. The task consisted of a repetitive movement performed in a fixed posture and it was considered complete once the cobot returned to its initial configuration.

Two predefined robot configurations were used to selectively activate either the anterior (Figure 102a) or posterior deltoid (Figure 102b). These configurations were adapted to each participant's anthropometrics.

After completing the task with fatigue management enabled, participants rested for five minutes. They then repeated the same task without fatigue management. In this baseline condition, the cobot remained

fixed in the initial configuration activating the anterior deltoid, chosen as it represents the most natural posture for task execution. The task duration was matched to that of the fatigue-aware condition or terminated earlier if the participant could no longer endure the task.



Figure 102: Collaborative task and the two workstation configurations used for fatigue prevention. (a) Anterior deltoid activation. (b) Posterior deltoid activation. The locations of the sEMG sensors are visible in the figures.

To further assess the perceived usefulness and effectiveness of the proposed control strategy, participants completed a post-experimental questionnaire after finishing both task conditions. The questionnaire was used to benchmark the fatigue-aware control against the baseline condition without fatigue management and is reported in Appendix F.

6.4.3 Results and Discussion

The implemented control strategy was validated from a quantitative and qualitative perspective. In this section the results obtained from the evaluation conducted, are outlined and discussed.

Quantitative Results sEMG signals acquired using the Trigno Avanti™ system were processed by computing the RMS over a 250 ms sliding window with 50% overlap to obtain the muscle activation envelope,

which was subsequently normalized to maximum voluntary contraction. The analysis focused on the anterior and posterior deltoid muscles under two experimental conditions: with HRC and without HRC .

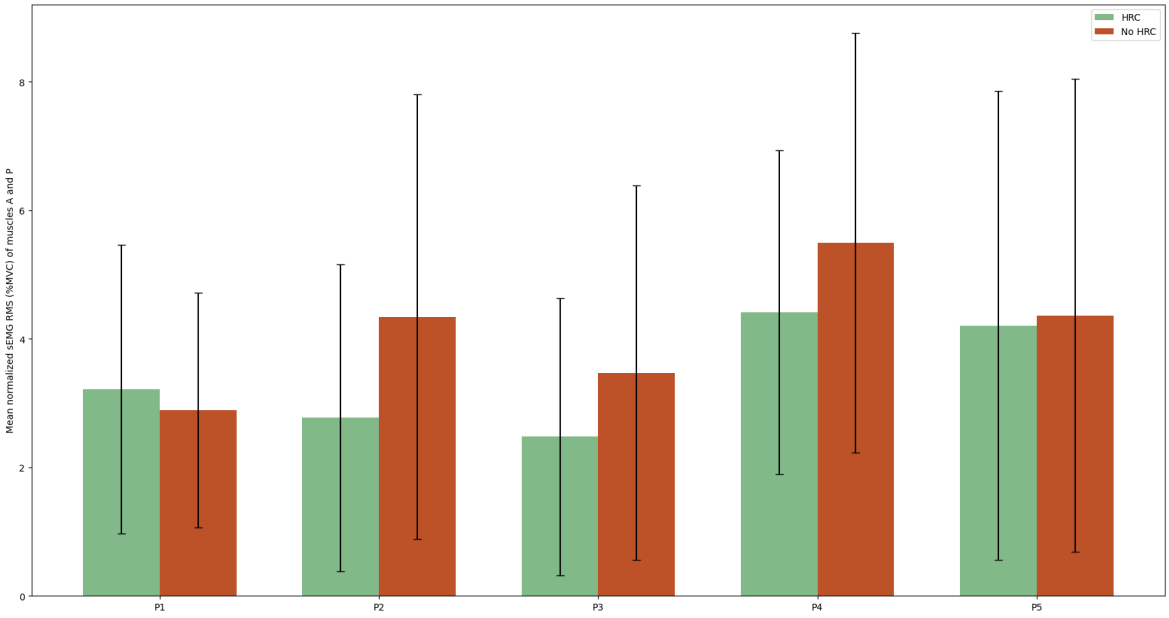


Figure 103: Mean normalized sEMG RMS (%MVC) of the anterior (A) and posterior (P) deltoid muscles across five participants (P1–P5), comparing HRC and No HRC conditions. Error bars represent the combined standard deviation of both muscles, computed as the root mean square (RMS).

Figure 103 summarizes the average normalized muscle activation across participants. When HRC was active, mean deltoid activation was consistently lower than in the No HRC condition for four out of five participants. The magnitude of reduction varied across individuals, reflecting inter-subject variability in movement strategy and muscle recruitment. Participant P1 showed slightly higher activation in the HRC condition, whereas P2, P3, P4 exhibited substantial increases in activation when assistance was removed. P5 had comparable activation between conditions. This trend is quantitatively confirmed in Table 39, where the No HRC condition resulted in mean activation increases ranging from 25.0% to 105.1% relative to HRC.

Time-domain muscle activation profiles further support these findings. As shown in Figure 104, in the HRC condition, following the collaborative robot configuration change, the RMS envelope of the anterior deltoid decreased to lower values, while posterior deltoid activation increased; These transitions coincided with fatigue state predictions generated by the deep learning model, indicating that the control strategy effectively redistributed muscular load rather than merely reducing overall effort. In contrast, during the No HRC condition, anterior deltoid activation remained elevated throughout the task, with higher variability and no observable redistribution of effort between antagonist muscles.

Table 39: Mean normalized sEMG RMS (%MVC) of the anterior and posterior deltoid muscles for each participant under No HRC and HRC conditions. The reduction percentage is computed relative to the No HRC condition, where positive values indicate a decrease in muscle activation with HRC.

Table 40: Mean normalized sEMG RMS (%MVC) of the anterior and posterior deltoid muscles for each participant under No HRC and HRC conditions. The reduction percentage is computed relative to the No HRC condition, where positive values indicate a decrease in muscle activation with HRC.

Participant	No HRC Mean (%)	HRC Mean (%)	Relative difference (%)
P1	2.89	3.21	-11.1
P2	4.34	2.77	36.2
P3	3.47	2.47	28.7
P4	5.50	4.42	19.7
P5	4.36	4.21	3.6

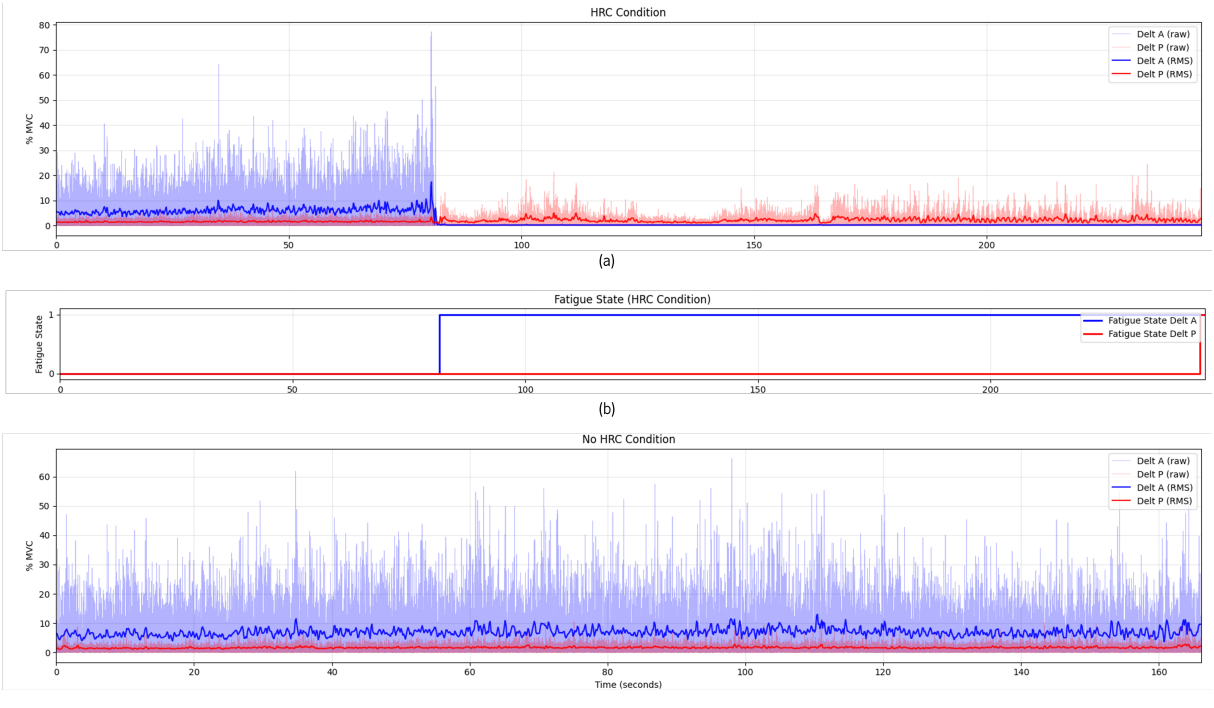


Figure 104: Validation results for participant P2. (a) Normalized muscle activity and envelopes of the anterior and posterior deltoids with the control strategy enabled. (b) Estimated fatigue levels, where posterior deltoid estimation starts after fatigue is detected in the anterior deltoid and the cobot changes configuration. (c) Anterior deltoid activity during task execution without the control strategy.

The results demonstrate that the proposed fatigue-aware collaborative strategy can reduce the average muscle activation and enables dynamic load sharing between antagonist muscles. The effect is not uniform across participants, which is expected given individual differences in biomechanics and task execution. Nevertheless, the consistent increase in muscle activation observed in the No HRC condition indicates that the control strategy provides a measurable ergonomic benefit. Importantly, this benefit is achieved without interrupting task execution, supporting the feasibility of continuous fatigue mitigation through adaptive human–robot collaboration rather than through task cessation or enforced rest periods.

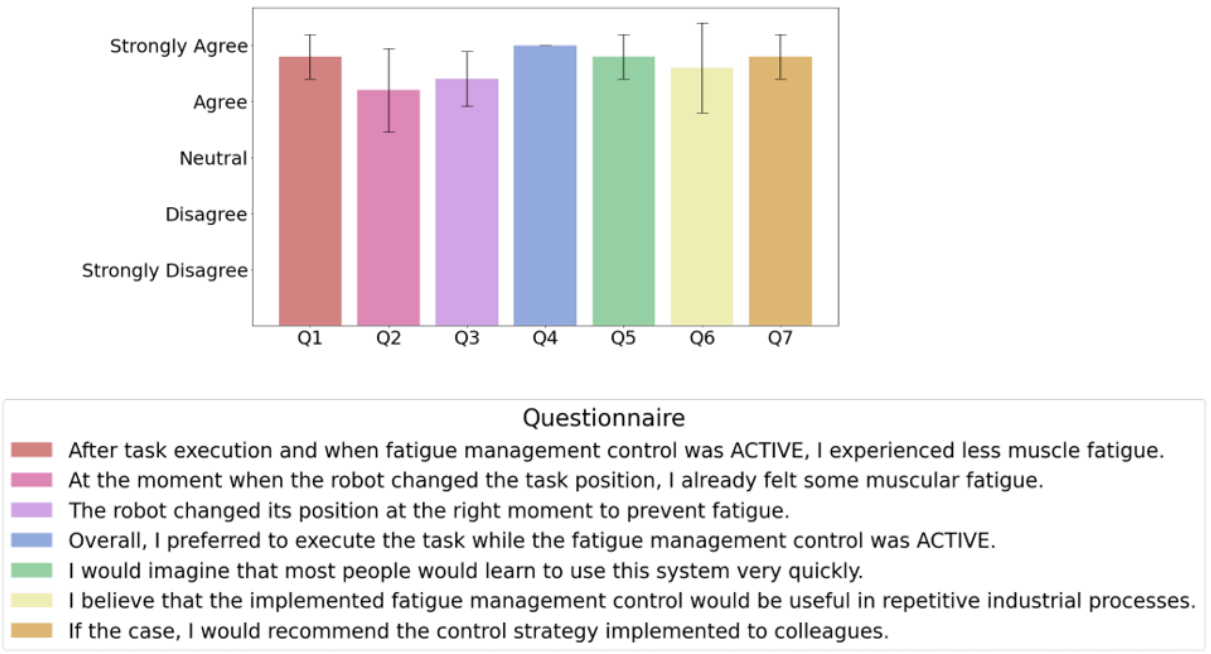
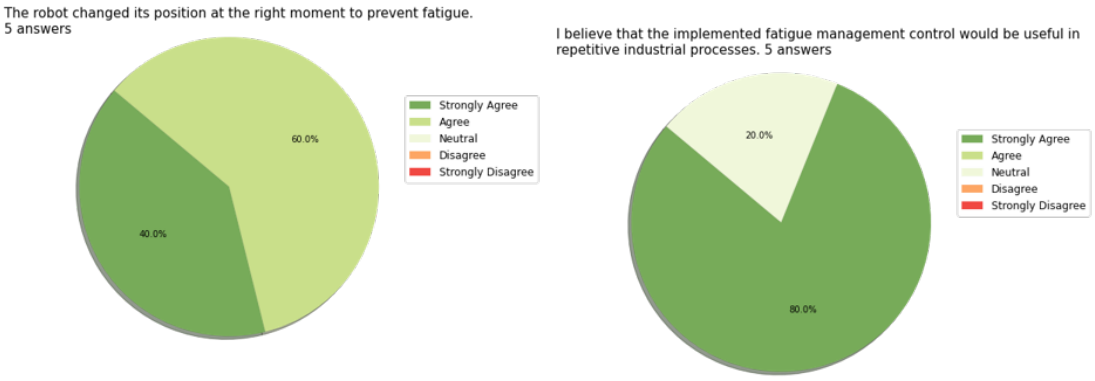


Figure 105: Results of the questionnaire applied to evaluate the implemented work. The bars indicate mean subjective scores given by the subjects for each question. The black lines indicate the standard deviation of mean.

Questionnaire Results The results of the questionnaire applied to each subject after the experiments are depicted in Figure 105. All subjects strongly agreed or agreed that less muscle fatigue was experienced while performing the task in the case where the fatigue management control was active. Likewise, they responded that already felt some muscle fatigue when the cobot changed its configuration, only 1 subject remained neutral in his opinion. Furthermore, all subjects strongly agreed or agreed that they would recommend the implemented control strategy to prevent muscle fatigue, in case they were employed in a work that required repetitive movements of the same muscle. Also, all participants preferred to execute the task while the control strategy was active, validating the implemented work.

The results of the efficiency and usability of the implemented control are outlined in Figure 106. It

is possible to verify that 40% (2 participants) strongly agreed and 60% (3 participants) agreed that the cobot changed its workstation configuration at the right time to prevent fatigue. These results demonstrate that the DL model was able to detect fatigue in time, allowing the active muscle to relax, activating its antagonist to continue the task. Moreover, 80% (4 participants) strongly agreed that the developed control strategy would be useful in repetitive industrial processes. The 20% who responded neutral corresponds to one participant, who justified their answer by stating that they could not envision the selected task fitting into an industrial context. Considering that the main problem highlighted in this work was the WRMSDs, which had a significant incidence in the industry, it is possible to observe from these results that this work successfully accomplished the first steps towards addressing this problem.



(a) Performance of the implemented control strategy

(b) Usability in industrial tasks

Figure 106: Results of the subjective evaluation related to (a) the performance of the implemented control strategy and (b) usability in industrial tasks.

6.5 Conclusions

This chapter investigated the estimation and exploitation of muscle fatigue from surface electromyography signals in dynamic, cyclic tasks, with the objective of enabling fatigue-aware human–robot collaboration. The focus was on modeling fatigue as a gradual, temporally evolving process rather than as an instantaneous state, and on assessing whether fatigue-related information extracted from a single muscle channel could support adaptive collaborative behaviors. Experimental results demonstrated that fatigue-related signatures are consistently present in the data, but are strongly entangled with task biomechanics and inter-subject variability, particularly under dynamic conditions.

The main contribution of this chapter is the systematic analysis and validation of learning-based fatigue

estimation models under subject- and muscle-independent conditions, combined with their integration into a collaborative control loop. Through extensive exploratory analysis and benchmarking, the chapter showed that amplitude-based sEMG features are weak fatigue indicators in dynamic movements, while frequency- and complexity-related features provide more physiologically meaningful information. Attention-based recurrent architectures emerged as the most effective modeling approach, confirming that explicit temporal reasoning is essential to capture gradual fatigue progression. Although classification performance remains moderate, the structured nature of the errors and the rarity of severe misclassifications indicate that the models capture the ordinal structure of fatigue states rather than arbitrary patterns. The fatigue KPI define in 1.3 was met by the best-performing model, which achieved 79.0% test accuracy, above the required 70%. However, this model used the full trial as input, so it was not the model used for real-time control.

The practical outcome of this work is a proof-of-concept fatigue-aware assistance mechanism that uses learned fatigue predictions to redistribute muscular load during task execution.

Validation experiments showed that the fatigue-aware collaborative strategy generally reduced average muscle activation compared to the non-adaptive condition, while promoting antagonist muscle alternation after fatigue detection. This redistribution was achieved without interrupting task execution, supporting the feasibility of adaptive fatigue mitigation through HRC rather than through enforced rest or task cessation. However, the predefined fatigue exposure KPI was only partially fulfilled: mean muscle activation decreased in four out of five participants, but only one participant exceeded the 30% reduction threshold, while one participant showed increased activation with HRC. The objective results were consistent with the subjective feedback, as all participants reported reduced perceived fatigue, preferred the assisted condition, and considered the timing and usefulness of the fatigue-driven workstation reconfiguration appropriate.

Future work will focus on improving robustness and generalization by enriching the input representation. The inclusion of synergistic muscles and the fusion of sEMG with kinematic information are expected to help disambiguate fatigue-related changes from movement-induced variability. From a modeling perspective, knowledge distillation and lightweight temporal models will be explored to move toward online fatigue estimation while preserving the benefits of long-term temporal context. Finally, broader validation across tasks, muscles, and populations will be required to consolidate fatigue-aware collaboration as a reliable ergonomic intervention rather than a task-specific solution.

Chapter 7

Learning Human Specificities based on Ergonomics and Physical restrictions

This chapter presents the development of a RL-based control strategy to personalize cobotic assistance to the human user, considering ergonomic metrics and physical constrains. It is primarly based on the published conference paper entitled "Towards a Human-Sensitive Controller: Learning Human Specificities in Ergonomics and Physical Constraints" [255] and the preprint "A Human-Sensitive Controller: Adapting to Human Ergonomics and Physical Constraints via Reinforcement Learning" [256], which is currently under review.

7.1 Introductory Insight

Most industrial workstations are designed for healthy workers, without physical impairments and are not adequate for people with MSDs. Developing human-sensitive controllers that adapt cobotic assistance to individual functional capacity pose as a solution to this problem. Although HRC increasingly considers ergonomic [18, 22] and fatigue [2, 16] related objectives, such adaptation to workers with musculoskeletal disorders remains underexplored.

This chapter proposes a Human-Sensitive Controller that integrates RL algorithms to learn and respond to human-specific ergonomic needs and physical constraints. By embedding awareness of factors such as range of motion or prior injuries, the controller adapts the robot's behavior to deliver appropriate and personalized support during collaborative manipulation tasks. The central research question guiding this work is: *How to design and train an intelligent, RL-based system capable of learning human specificities in a safe, ergonomic, and pain-free manner during physical collaboration?*

7.2 Knowledge acquisition: Reinforcement Learning

This section provides a critical analysis of the reinforcement learning literature, with the aim of extracting methodological insights to guide its application to the problem addressed in this work. The analysis focuses on how previous studies formulated the Markov Decision Process (MDP), selected learning algorithms, designed reward functions, and structured the learning phase. These elements are examined to inform the design and implementation of an RL based control framework suitable for human robot collaborative scenarios.

7.2.1 Application Domain Considerations

The reinforcement learning studies reviewed in this section span a wide range of application domains. For clarity, they can be grouped according to the type of human–robot interaction they address, while still sharing methodological principles relevant to HRC. A first group of works explicitly targeted ergonomic optimization. In particular, Lagomarsino et al. [257] and Xie et al. [94] formulated RL objectives around ergonomic criteria, but did not incorporate physical limitations such as reduced range of motion or chronic injury. These studies illustrate how RL can encode human centered goals, while also delimiting the scope of the modeled human factors.

A second group of studies applied RL to interaction intensive tasks, such as surgical assistance [258], obstacle avoidance [259], and automation support [260]. Although these application contexts differ from HRC, these works provide transferable insights into reward shaping, action safety, and continuous control, which are central challenges in physically assistive robotic systems. Across this group, real-world validation remains limited, with most approaches evaluated primarily in simulation. This trend highlights the practical constraints associated with deploying RL in human–robot settings and motivates learning strategies that combine simulation based training with cautious real-world deployment.

7.2.2 Insights from MDP Design

The design of the MDP, specifically the state and action space, is critical in defining how the agent perceives the environment and makes decisions. Across the reviewed literature, continuous state spaces were commonly used to represent robotic poses [258], human configurations [261], or sensor outputs [262]. These are more representative of real-world dynamics and offer finer control resolution, but also increase the complexity of the learning process.

Several studies demonstrated the effectiveness of shaping the state space to accelerate convergence

and reduce unsafe behaviors [259, 263–265]. For example, Puthuveetil et al. [261] employed randomized human body configurations to promote generalization, while Long et al. [258] enforced kinematic constraints to ensure safe learning in surgical applications. Such strategies are directly translatable to collaborative tasks, where dynamic human posture and joint safety must be considered.

With respect to action spaces, discrete actions were typically employed in simplified interaction scenarios [257, 259]. In contrast, continuous action spaces were adopted in applications requiring precise motion generation, such as surgical robotics [258] and ergonomic pose adjustment [94]. In designing a controller for physically constrained users, a hybrid action space or parameterized actions may provide the necessary flexibility while maintaining safety.

7.2.3 Reward Function Design and Human Considerations

Across the reviewed studies, reward functions were used as the primary mechanism to encode human related objectives within the learning process. In particular, Xie et al. [94] and Lagomarsino et al. [257] used ergonomic evaluation scores directly in the reward, assigning penalties to postures associated with higher ergonomic risk. This led the learned policies to avoid unfavorable postures during interaction. The effectiveness of this approach depends on the reliability of the ergonomic metrics and on the relative weighting of the reward terms, as poorly calibrated rewards can result in unintended behaviors.

In other application domains, reward functions were defined to reflect task completion, efficiency, or safety objectives. For example, Singh et al. [260] adjusted reward contributions according to task priority to guide automation behavior, while Puthuveetil et al. [261] employed multi term reward formulations that explicitly penalized occlusion of safety critical regions, such as the human head. These examples show how reward decomposition can be used to enforce specific interaction constraints by assigning explicit penalties to undesirable outcomes.

Across the reviewed literature, reward design was predominantly based on handcrafted formulations. This approach requires substantial domain knowledge and careful tuning, and can lead to unintended behaviors if the reward does not accurately reflect the intended task objective. Alternative methods, including Inverse Reinforcement Learning and Preference Based RL, have been proposed to infer reward structures from demonstrations or user feedback, thereby reducing reliance on manual specification. These approaches are outside the scope of this work and are not considered further in this thesis.

7.2.4 Algorithm Selection and Learning Strategies

Algorithm selection was directly determined by the structure of the underlying MDP. Studies operating with discrete domains consistently adopted value-based methods, such as Q-learning and SARSA [259, 263, 264]. These methods are computationally efficient and straightforward to implement, but are typically restricted to low-dimensional settings and cannot scale to high-dimensional or continuous state and action spaces. Deep Q-Networks extended value-based approaches by incorporating learned state representations, yet the reported improvements remained marginal when applied to more complex environments [264].

In contrast, continuous control tasks favored policy-based or actor-critic methods, such as PPO [261], DDPG [258], and SAC [262].These algorithms provided superior flexibility, sample efficiency, and learning stability in complex robotic systems. Their successful deployment demonstrates the applicability of modern deep RL methods to human-centered control applications, contingent upon the integration of safety constraints.

Notably, Lagomarsino et al. [257] circumvented full MDP modeling by framing the task as a Multi-Armed Bandit (MAB) problem, where actions do not affect state transitions and only immediate rewards are considered. This simplification was feasible due to the low task complexity and offers an interesting paradigm for scenarios where only a few discrete strategies need to be evaluated. However, it lacks the temporal modeling capacity necessary for adaptive assistance in dynamic tasks.

7.2.5 Learning Settings and Transferability

All reviewed studies employed online learning, typically in simulation. Training in the real-world was limited to a few studies [94, 257, 261], with high time costs and safety concerns as primary challenges. One effective strategy for bridging this gap was observed in Puthuveetil et al. [261], who trained policies in simulation and deployed them in real-world settings without additional fine-tuning. This transfer was successful due to careful domain randomization and robust policy generalization.

7.3 Solution Conceptualization

7.3.1 Solution requirements

This work focus on learning the optimal cobot's displacement during the object's transport which met Human specificities. It was designed to meet three primary requirements:

- **pain-free operation:** the control system must be designed to ensure that its actions do not induce pain or discomfort for the worker.
- **Ergonomic assistance:** the system must be capable of learning to support the worker in a manner that adheres to ergonomic principles, particularly by reducing awkward or non-neutral postures.
- **Time efficiency:** the system should perform the assigned task within the shortest time frame necessary, without compromising safety or quality. To limit fatigue accumulation during training with human participants, the system must complete its adaptation within 30 minutes, while minimizing pain risk and ensuring safe ergonomic postures.

7.3.2 Elbow Contracture and Use Case Scenario

Although the long-term goal is to develop a broadly applicable solution for a variety of MSDs through the use of digital pain biomarkers, this exploratory study focuses on a representative use case: elbow contracture. This condition limits elbow range of motion [266], typically restricting extension beyond 30° and flexion below 120°, and can cause pain within the affected arc as illustrated in Fig. 107.

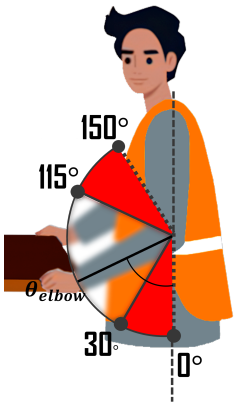


Figure 107: Elbow contracture ROM. Red represents the painful elbow's ROM.

As a proof of concept, this study focuses on a co-manipulation task, where a human and robot jointly transported an object vertically. The task endpoints were tailored to the participant's anthropometric limits.

7.3.3 Conceptional Overview of the RL-based Control

To promote safe and ergonomic collaboration between human workers and robots, a RL control system was developed to guide the movements of a cobot. In the considered shared object transportation task, the cobot leads the motion, while the human worker follows its displacements, allowing the robot to regulate

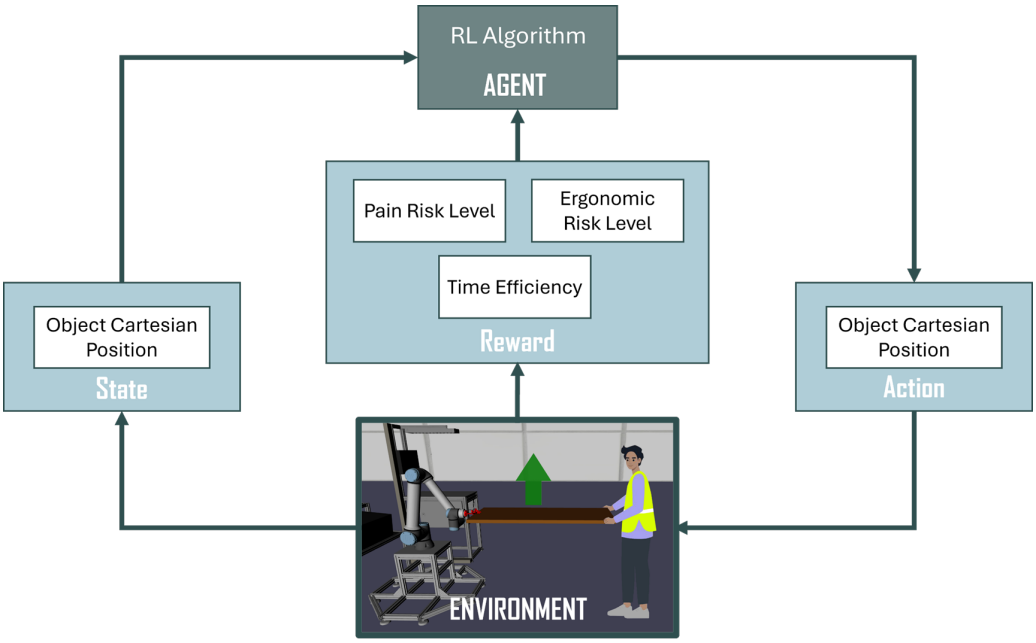


Figure 108: Conceptual diagram of the RL-based control system for co-manipulation transport task.

the interaction in order to support pain-free and ergonomically sound postures. As shown in Figure 108, the system is built on the RL framework and interacts continuously with its environment during policy learning and execution.

The environment consists of a human and a cobot jointly handling an object. The RL agent observes the current state, represented by the Cartesian position of the object, and selects an action corresponding to a new desired object position. After the action is executed, the environment responds with a new object position and a scalar reward based on three factors: (i) ergonomic risk, (ii) pain risk, and (iii) task efficiency. This feedback loop enables the agent to iteratively learn how to guide the object in a way that improves human posture and comfort while maintaining efficiency.

RL Formulation and MDP Definition

This problem is modeled as a MDP, defined by a tuple (S, A, P, R, γ), where:

- **State** $s \in S$**:** The state is the object's Cartesian position, defined as the midpoint between the end-effectors of the cobot and the human. This position is bounded within a two-dimensional Cartesian area (width w_{ss}, height h_{ss}), represented in the shared reference frame $\sum_{sso}$. This RL problem was formulated as a fully observable MDP, as the state representation can be determined with certainty at each time step, ensuring complete state information throughout the task.

- **Action** $a \in A$**:** The action corresponds to the cobot selecting a new desired position for the shared

object, effectively guiding its movement within the workspace. Since the object position is defined by its midpoint, defined by between the human and the robot, it can only be achieved if both agents follow the intended motion. As the human response cannot be directly controlled, the resulting state depends on their interaction, making the environment's transition function stochastic:

$$P(s_{t+1} \mid s_t, a_t) < 1$$

- **Reward $r(s, a)$:** The reward function encodes three learning objectives: avoid postures associated with pain risk, minimize ergonomic risk, and maintain time efficiency. Actions that increase ergonomic or pain risk are penalized, while those that support safety and progress the task are rewarded. This multi-objective reward structure encourages the agent to prioritize both human safety and task efficiency

- **Episodes:** Each episode begins with the object in an initial position and ends when the object reaches a defined target zone, typically at the top of the Cartesian area.

The formulation satisfies the Markov property, as the current object position provides sufficient information to predict the next state given the action.

7.4 Experimental Methodology

7.4.1 Reinforcement Learning Design

Figure 109 provides an overview of the main components of the RL framework and their interactions during training.

Training Methodology

Training RL agents is time-consuming. Thus, relying only on real-world training is unfeasible as it is unpractical and extremely difficult to find a volunteer willing to withstand such lengthy training experiments. On the other hand, human behavior is stochastic and unpredictable, meaning that relying only on simulations could result in an agent that fails to align with human specificities.

To address these limitations, a two stage training pipeline was implemented that integrates simulation and real-world learning. The agent is first pre-trained in a simulated environment to acquire fundamental behaviors and reduce the need for extensive real-world data collection. A kinematic human model, scaled to participant anthropometry, is used to perform inverse kinematics and estimate ergonomic and pain

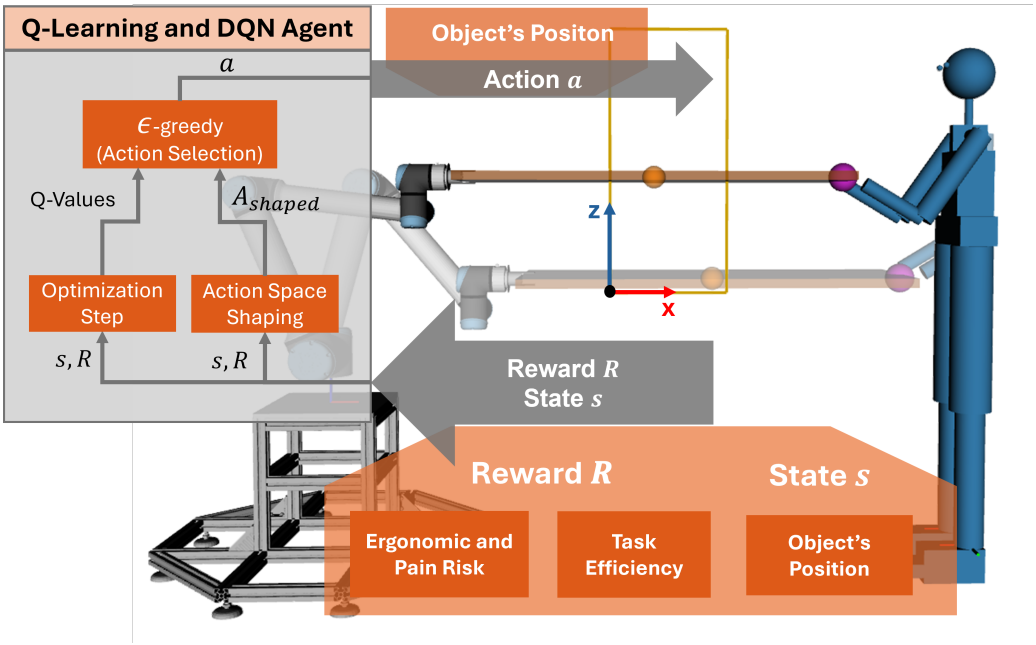


Figure 109: RL framework overview for learning human specificities, in ergonomics and physical constraints.

related risks from simulated postural responses. The learned policy is then deployed in the real-world for fine tuning and personalization, where a sensory system monitors the human collaborator's posture to support real time adaptation.

Algorithm Selection

Two reinforcement learning algorithms were selected for this work: Q-Learning and Deep Q-Networks (DQN), to balance conceptual simplicity, computational efficiency, and scalability across different problem complexities. An ϵ-greedy action selection strategy was adopted for both algorithms to balance exploration and exploitation, ensuring sufficient state–action exploration while progressively favoring actions with higher expected rewards.

Q-Learning is a model-free, tabular RL algorithm that estimates the expected cumulative reward for each state–action pair. It requires discretization of the state and action spaces and assumes a well-defined MDP. The algorithm was selected for its simplicity, low computational cost, and fast convergence in simple environments, which are advantageous when interaction time with human participants is limited.

The algorithm iteratively updates the Q-table based on observed transitions, as shown in Algorithm 1, adapted from [267]. At each step, the agent selects an action using an ϵ-greedy policy, executes it, observes the reward and next state, and updates the corresponding Q-value.

Algorithm 1 Q-Learning pseudocode adapted from [267]

Parameters: Learning rate $\alpha \in]0, 1]$, Discount factor $\gamma \in [0, 1]$
Initialize $Q(s, a)$ for all $s \in \mathcal{S}$, $a \in \mathcal{A}$
for each episode until termination **do**
    Initialize state s
    while s not terminal **do**
        Select action a using an ϵ-greedy policy
        Execute a, observe r and s'
        Update:

$$Q(s, a) \leftarrow Q(s, a) + \alpha \left[r + \gamma \max_{a'} Q(s', a') - Q(s, a) \right]$$

        $s \leftarrow s'$
    end while
end for

However, Q-Learning is constrained by its reliance on discrete representations. In high-dimensional or continuous state spaces, the tabular approach does not scale efficiently and may fail to generalize. To overcome this limitation, a DQN was also implemented.

DQN extends Q-Learning by replacing the Q-table with a neural network that approximates the Q-function, enabling learning in larger and continuous state spaces while preserving the original update structure. This provides a practical balance between representational capacity and computational efficiency, making DQN suitable for real-world robotic applications.

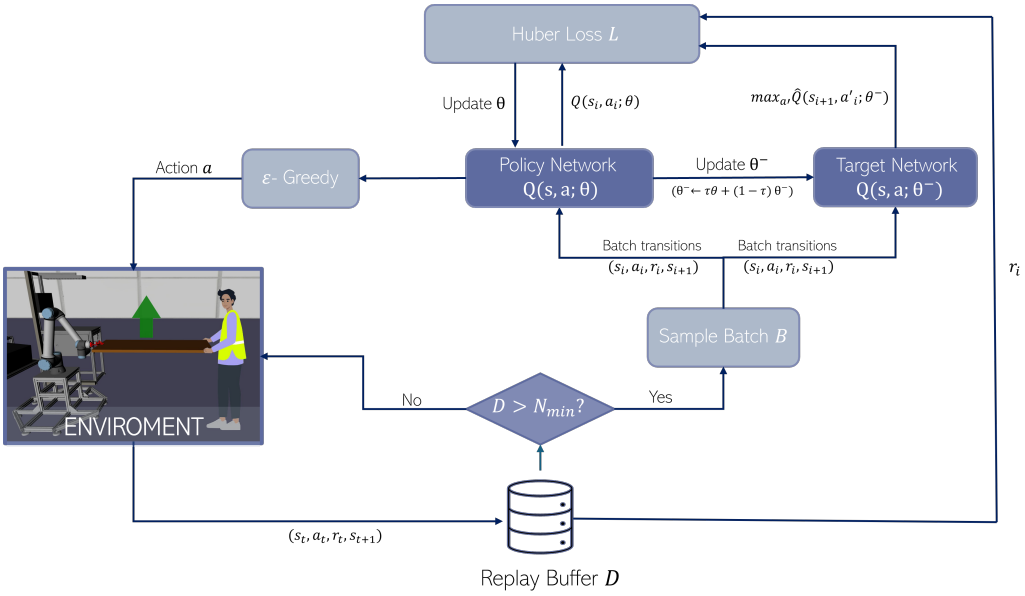


Figure 110: DQN diagram illustrating the main functional components of the algorithm.

Figure 110 illustrates the internal flow of the DQN algorithm, detailed in Algorithm 2, adapted from [268].

At each interaction step, the agent observes a transition (s_t, a_t, r_t, s_{t+1}), where s_t is the current state, a_t the selected action, r_t the resulting reward, and s_{t+1} the next state. This transition is stored in a replay buffer $\mathcal{D}$. To improve learning stability, the policy network is updated using batches $\mathcal{B}$ sampled from the replay buffer, reducing correlation between samples.

The network is trained using the Huber loss $\mathcal{L}$:

$$\mathcal{L} = \frac{1}{|\mathcal{B}|} \sum_{(s_t, a_t, r_t, s_{t+1}) \in \mathcal{B}} L(\delta_t) \tag{7.1}$$

where

$$L(\delta_t) = \begin{cases} \frac{1}{2}\delta_t^2 & \text{if } |\delta_t| \leq 1 \\ |\delta_t| - \frac{1}{2} & \text{otherwise} \end{cases} \tag{7.2}$$

and the temporal-difference error is defined as:

$$\delta_t = r_t + \gamma \max_{a'} \hat{Q}(s_{t+1}, a'; \theta^-) - Q(s_t, a_t; \theta)$$

where γ is the discount factor, $Q(s_t, a_t; \theta)$ denotes the policy network parameterized by θ, and $\hat{Q}(s_{t+1}, a'; \theta^-)$ denotes the target network parameterized by θ^-. The target network shares the same architecture as the policy network and is initialized with identical parameters ($\theta^- = \theta$), providing stable targets during training and mitigating divergence caused by rapidly changing Q-value estimates.

The policy network parameters θ are updated by minimizing the loss:

$$\theta \leftarrow \theta - \alpha \nabla_\theta \mathcal{L} \tag{7.3}$$

To further stabilize training and mitigate instability in the Q-value estimates $Q(s, a; \theta)$, the target network $\hat{Q}(s, a; \theta^-)$ is updated using a soft update rule:

$$\theta^- \leftarrow \tau\theta + (1 - \tau)\theta^- \tag{7.4}$$

where τ is a small constant controlling the update rate.

State space

The space state S was defined as the Cartesian area where the object can be displaced.

Algorithm 2 DQN Algorithm

Initialize replay buffer $\mathcal{D}$ with capacity N
Initialize minimum replay size N_{min} (warmup threshold)
Initialize policy network $Q(s, a; \theta)$
Initialize target network $\hat{Q}(s, a; \theta^-)$ with $\theta^- = \theta$
for each episode **do**
    Initialize state s_t
    while s_t not terminal **do**
        Select action a_t using an ϵ-greedy policy
        Execute a_t, observe r_t and s_{t+1}
        Store (s_t, a_t, r_t, s_{t+1}) in $\mathcal{D}$
        if $|\mathcal{D}| \geq N_{\text{min}}$ **then**
            Sample batch $\mathcal{B}$ from $\mathcal{D}$
            Compute targets:

$$y_i = \begin{cases} r_i & \text{if } s_{i+1} \text{ is terminal} \\ r_i + \gamma \max_{a'} \hat{Q}(s_{i+1}, a'; \theta^-) & \text{otherwise} \end{cases}$$

            Update θ by minimizing $\mathcal{L}$ (Eq. 7.1)
            Update target network: $\theta^- \leftarrow \tau\theta + (1 - \tau)\theta^-$
        end if
        $s_t \leftarrow s_{t+1}$
    end while
end for

Q-Learning The state space is discretized using a resolution factor Δ_d. Each state s corresponds to the center of a discretized cell and is represented by the object position (x_{obj}, z_{obj}).

The coordinates are bounded as $x \in [1, x_{\text{max}}]$ and $z \in [1, z_{\text{max}}]$, where $x_{\text{max}} = \frac{w_{ss}}{\Delta_d}$ and $z_{\text{max}} = \frac{h_{ss}}{\Delta_d}$, with w_{ss} and h_{ss} denoting the workspace dimensions.

The discretization level Δ_d determines the size of the state space. Finer discretization increases the number of states, enabling more precise positioning and potentially improving performance by allowing access to more favorable configurations. Figure 111a illustrates the discretized workspace and examples of state coordinates.

DQN state space consists of all the Cartesian positions of the object within a rectangular area. This position was defined as the midpoint between the end-effectors (EE) of the Cobot and the Human and encoded as a two-dimensional vector $(x_{\text{obj}}, z_{\text{obj}})$, where x and z denote the horizontal and vertical coordinates, respectively, according to the reference frame in Figure 112. These coordinates are limited to the Cartesian rectangular area, where $x \in [0, x_{\text{max}}]$ and $z \in [0, z_{\text{max}}]$, with x_{max} and z_{max} corresponding directly to the width and height of the state space, w_{ss} and h_{ss} respectively.

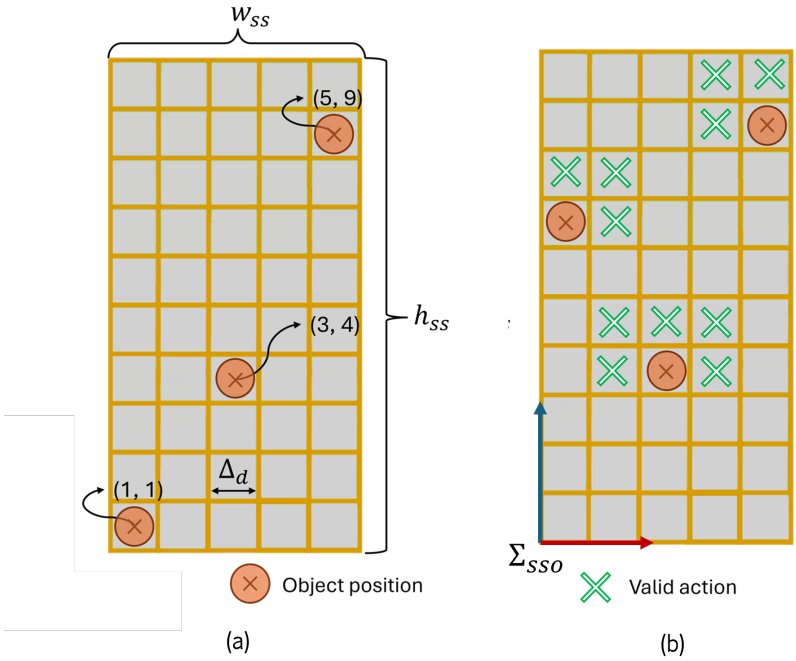


Figure 111: Discrete state and action space representation used in the Q-Learning formulation. (a) Discretized workspace of size $w_{ss} \times h_{ss}$, where each cell represents a possible object position (x, z) with discretization step Δ_d. (b) Examples of valid action sets, where green markers indicate admissible transitions within workspace boundaries and orange circle markers denote the current object position. Σ_{sso} represents the state space frame.

Action space

The action space A defines how the agent interacts with the environment by modifying the object's position. Both Q-Learning and DQN employ discrete action representations.

Q-Learning action space was defined by movements between grid cells. To allow applicability to a continuous state space domain, First, the action space is constrained to ensure that the agent does not move downwards and solely one discretized area at a time. This avoids movement in the opposite direction of the task movement. To accomplish this, the action space is constrained based on the agent's current state. Considering the agent is at state $(x_{\text{obj}}, z_{\text{obj}})$, the action space will be composed by $[(x_{\text{obj}} \pm 1,\ z_{\text{obj}}),\ (x_{\text{obj}},\ z_{\text{obj}} + 1),\ (x_{\text{obj}} \pm 1,\ z_{\text{obj}} + 1)]$. According to the state space limits, the action space could have fewer than five actions if any of these fall outside the valid region (some cases are represented at the right side of Figure 111).

DQN action was represented as a pair (α, d), where α denotes the movement direction and d represents the distance traveled. The possible values for α are {0, 30, 60, 90, 120, 150, 180} degrees, and for d are

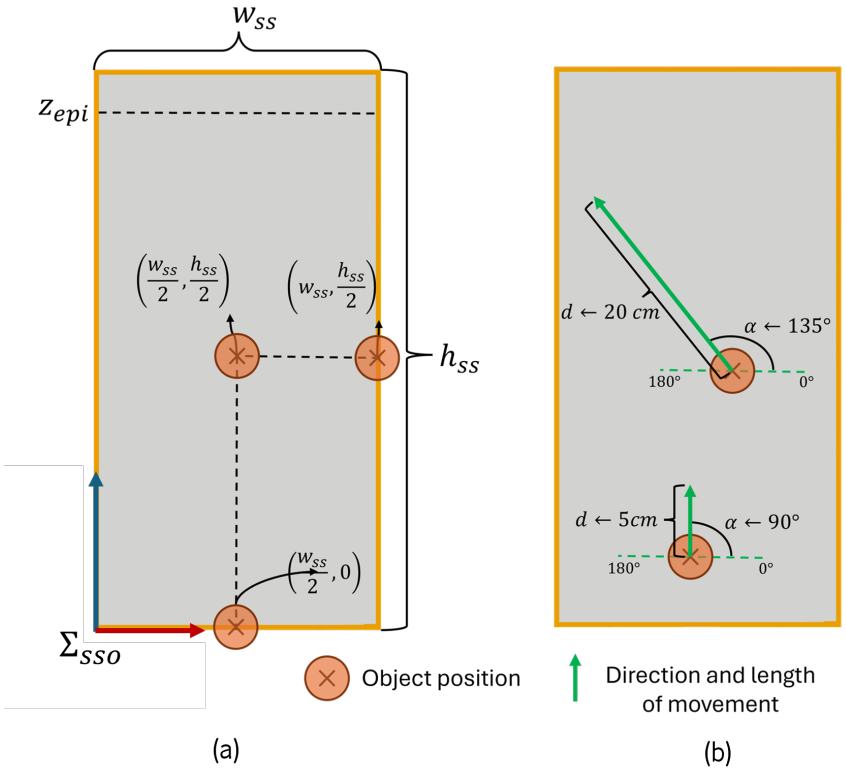


Figure 112: Continuous state and action space representation used in the DQN formulation. (a) Continuous workspace of size $w_{ss} \times h_{ss}$, where object positions are defined in the state space frame Σ_{sso} using Cartesian coordinates (x, z). (b) Actions defined by movement direction α and displacement magnitude d.

{0.02, 0.03, 0.05, 0.2, 0.4} meters. These values offer a balance between directional flexibility and step size variability. The chosen angles avoid backward movements, while multiple distance options allow the agent to plan more efficient transport trajectories by potentially reducing the number of required steps.

The reward function R was designed to guide the agent's behavior by balancing three key objectives: minimizing pain risk, reducing ergonomic risk, and promoting task efficiency. Actions that lead to high ergonomic or pain risks are penalized, while those that improve ergonomic conditions, mitigate pain risks, and contribute to task progress are rewarded.

An episode consists of a sequence of state-action transitions, beginning with the object in its initial position and ending when it reaches a predefined target area (the top of the cartesian area).

7.4.2 Simulation Environment and Human Posture Modeling

The challenge in the simulation environment is to accurately model the Human body configuration after the robot starts the object displacement. To address this, a kinematic model scaled according to the

participant's anthropometry was defined.

Given that the vertical transport task primarily involved shoulder and elbow motion, each arm was modeled as a planar 2-DOF kinematic chain. Assuming symmetric limb movement in the sagittal plane, the model captures the main motion components of the task, namely shoulder and elbow flexion-extension. The same kinematic structure was therefore used for both arms, with the index $i \in \{\text{left}, \text{right}\}$ identifying the corresponding limb.

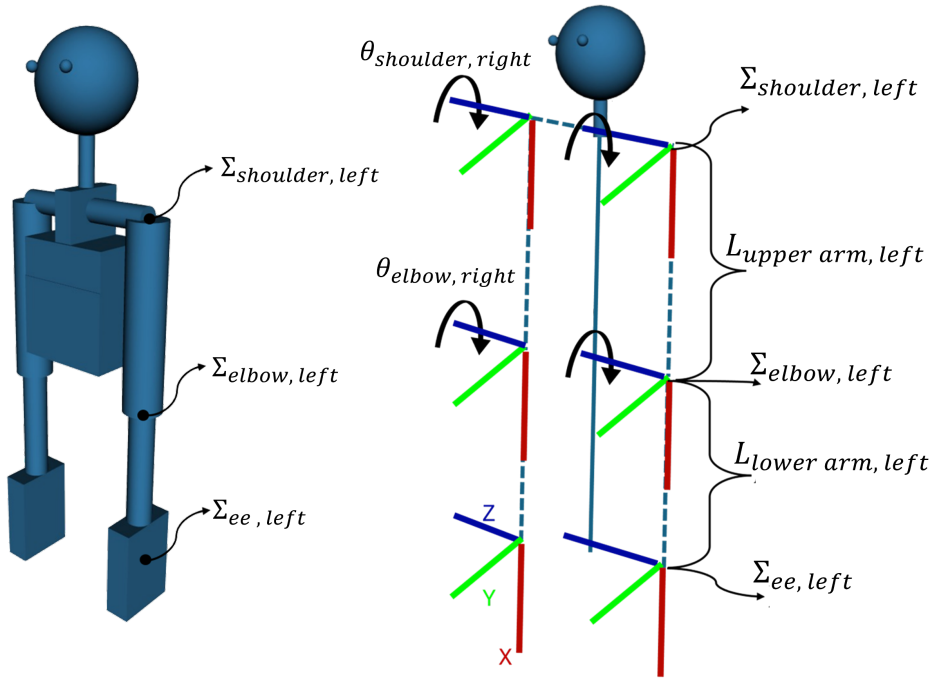


Figure 113: Human kinematic model used to compute human joint angles in the pre-training phase.

Figure 113 depicts the proposed human kinematic model. For each arm i, three reference frames were defined: the shoulder frame $\Sigma_{\text{shoulder},i}$, the elbow frame $\Sigma_{\text{elbow},i}$, and the end-effector frame $\Sigma_{\text{ee},i}$. The kinematic chain was modeled using two revolute joints located at the shoulder and elbow joint centers. The upper-arm link $L_{\text{upper arm},i}$ connects $\Sigma_{\text{shoulder},i}$ to $\Sigma_{\text{elbow},i}$, while the forearm link $L_{\text{forearm},i}$ connects $\Sigma_{\text{elbow},i}$ to $\Sigma_{\text{ee},i}$. The joint variables $\theta_{\text{shoulder},i}$ and $\theta_{\text{elbow},i}$ represent shoulder and elbow flexion-extension angles, respectively.

Since the computed human end-effector position is initially expressed in the global cobot-base frame $\Sigma_{\text{cobot base}}$, it is first transformed into the corresponding shoulder frame $\Sigma_{\text{shoulder},i}$. The transformed end-effector position is then expressed in the sagittal plane as $(x_{\text{obj},i}, z_{\text{obj},i})$. This position is checked to determine whether it falls within the 2-DOF workspace. This is verified by ensuring that the following condition holds:

$$(L_{\text{upper arm},i} - L_{\text{forearm},i})^2 \leq x_{\text{obj},i}^2 + z_{\text{obj},i}^2 \leq (L_{\text{upper arm},i} + L_{\text{forearm},i})^2 \,. \tag{5}$$

After the reachability condition is validated, the Inverse Kinematics (IK) solution is computed using a geometric method. Given the 2-DOF kinematic model and the computed human end-effector position, inverse kinematics provides two possible solutions for each angle, $\theta_{\text{shoulder},i}$ and $\theta_{\text{elbow},i}$. However, only one of these solutions is compatible with the human posture expected during the task. The selected solution is computed using Equations (7.5) and (7.6):

$$\theta_{\text{shoulder},i} = \text{atan2}\left(z_{\text{obj},i}, x_{\text{obj},i}\right) + \arccos\left(\frac{x_{\text{obj},i}^2 + z_{\text{obj},i}^2 + L_{\text{upper arm},i}^2 - L_{\text{forearm},i}^2}{2L_{\text{upper arm},i}\sqrt{x_{\text{obj},i}^2 + z_{\text{obj},i}^2}}\right) \tag{7.5}$$

$$\theta_{\text{elbow},i} = \pi - \arccos\left(\frac{L_{\text{upper arm},i}^2 + L_{\text{forearm},i}^2 - \left(x_{\text{obj},i}^2 + z_{\text{obj},i}^2\right)}{2L_{\text{upper arm},i}L_{\text{forearm},i}}\right) \tag{7.6}$$

Joint limits were defined according to normative shoulder and elbow flexion-extension Range of Motion (ROM) [269, 270]. The shoulder angle was constrained to $\theta_{\text{shoulder},i} \in [-60°, 180°]$, while the elbow angle was constrained to $\theta_{\text{elbow},i} \in [0°, 150°]$.

7.5 Implementation Details

7.5.1 Experimental Setup

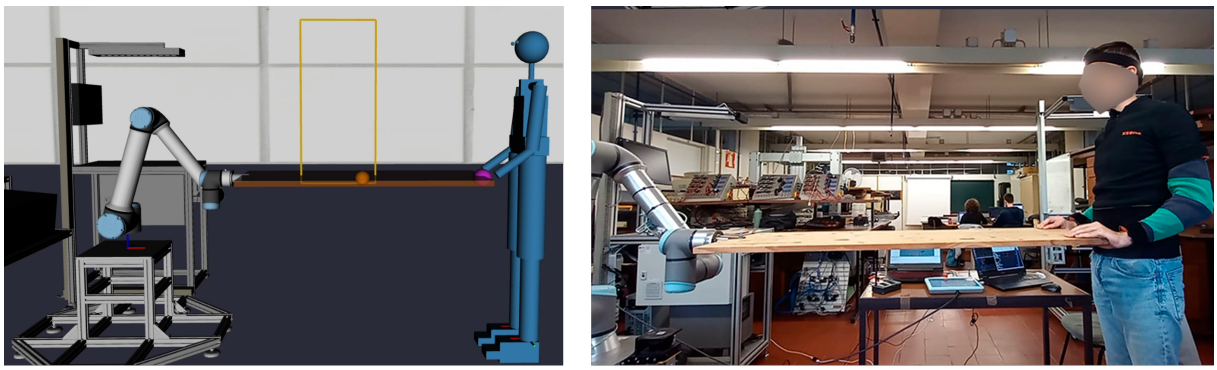
The state space is defined in a 90 cm (height) × 40 cm (width) Cartesian area, located 1 meter from the robot base. The user is assumed to be at a distance of 2.72 meters from the base with fixed foot positions. Figure 114 presents the experimental setup used in both simulation and real-world environments. For each participant, the initial and target positions for object transport are personalized based on their anthropometric measurements to ensure full reachability.

Table 41 summarizes the initial and target conditions for each participant and algorithm. In Q-Learning, states are represented by grid positions, whereas in DQN, they are defined in Cartesian coordinates (x, z in meters). To ensure reachability for all participants, both initial and target conditions are personalized based on individual anthropometry. In Q-Learning, success is defined as reaching a specific target height (z_{obj}). In DQN, success is achieved when $z_{\text{obj}} \geq 90 - \Delta z$, where Δz is a participant-specific height margin.

Table 41: Initial States and End Episode Conditions

	Initial State		End Condition	
Participant's Height	Q-Learning	DQN	Q-Learning (z_{obj})	DQN (Δz)
1.62	(4, 1)	(1.3, 0.434)	9	0.20
1.69	(4, 1)	(1.35, 0.454)	9	0.15
1.79	(4, 1)	(1.3, 0.434)	10	0.03
1.83	(5, 2)	(1.35, 0.474)	10	0.03

Each RL action is executed by the UR10e robot. The EE target position is computed from the desired Cartesian position of the object, $(x_{\text{des}}, z_{\text{des}})$, and horizontally offset by 0.65 meters (half the object's length), resulting in a final EE target of $(x_{\text{des}} - 0.65, z_{\text{des}})$.



(a) (b)

Figure 114: Simulation (a) and real-world (b) environments used for control strategy validation. In the simulation, the yellow rectangle defines the state-space boundary, and the orange sphere represents the object midpoint, used as the object position. In the real-world setup, the participant performs the task while equipped with the Xsens motion capture system.

7.5.2 Action Space Shaping

Action space shaping was used to improve learning efficiency by restricting the agent's available actions at each step. For the Q-Learning agent, actions that moved the object backward were explicitly excluded. In the proposed DQN formulation, this directional constraint is inherently enforced through the definition of the action space, with the angle parameter limited to $\alpha \in [0°, 180°]$, and therefore does not require additional checks.

Independently of this directional constraint, DQN actions must be validated to ensure that the resulting

object motion remains within the state-space boundaries. After an action is selected, the desired position $(x_{\text{des}}, z_{\text{des}})$ is computed from the current object position $(x_{\text{obj}}, z_{\text{obj}})$ and the action vector $(d \cdot \cos\alpha, d \cdot \sin\alpha)$. The action is considered valid only if $x_{\text{des}} \in [0, x_{\text{max}}]$ and $z_{\text{des}} \in [0, z_{\text{max}}]$. Actions that violate these bounds are removed from the available action set A.

In addition, all actions must comply with the kinematic model. This includes verifying that the simulated human End Effector (EE) position, derived from the desired object's position, remains within the kinematic workspace. An action is deemed valid only if it has a feasible IK solution and adheres to joint limit constraints. This validation process is applied in both simulated and real environments, resulting in a shaped action space, A_{shaped}, which guarantees the selection of only safe and efficient actions. This approach is particularly beneficial in real-world applications, as it prevents scenarios where selected actions cannot be physically executed by the human.

Further, an episode may terminate before reaching target condition if the shaped action space A_{shaped} becomes empty (e.g., no valid actions satisfy the IK constraints), or if a pain risk is detected in the real environment.

7.5.3 Pain and Ergonomic Assessment

Ergonomic risk is assessed using the RULA method [238]. As the task primarily involves shoulder and elbow movement (see 7.4.2), the RULA ergonomic score typically falls within the integer set 1, 2, 3.

Pain assessment aims to eventually integrate a digital pain biomarker. In this study, however, pain is estimated based on elbow ROM constraints, which are key indicators of discomfort (Figure 107). A state machine monitors the elbow joint angle and returns a binary pain state: a pain state of "1" is assigned when the angle falls within [0°, 30°] or [115°, 150°], and "0" otherwise. The module is encapsulated to allow easy replacement with a digital biomarker in future work.

In the real environment, avgErg and avgPain are computed by averaging the ergonomic and pain risk levels received at the sensory system's update rate during object movement. In simulation, the UR10e's linear motion is discretized along the path from the initial to the desired object position. At each discretized point, inverse kinematics is applied to compute the corresponding risk levels. The final avgErg and avgPain are obtained by averaging the values across all points along the trajectory, starting from the initial state.

7.5.4 Reward Function

Both Q-Learning and DQN share a base reward structure (ergRew and xMovRew), with the DQN adding an extra component (zStepRew).

The ergonomic reward term (ergRew), which penalizes high ergonomic risk levels, is defined as:

$$\text{ergRew} = -50\,\text{avgErg} + 50 \tag{7.7}$$

where avgErg $\in [1, 3]$ represents the average ergonomic risk level after action execution.

The sideward movement penalty xMovRew, introduced in the reward function (equation 7.8), discourages consecutive lateral displacements $(x \pm 1, z)$ by applying a linearly increasing penalty based on the number of repeated horizontal steps (countX). This component was motivated by the observation that, although the task involves lifting the object vertically, where increased ergonomic risk is inevitable due to posture constraints, the agent may attempt lateral deviations to smooth the long-term ergonomic cost. However, these sideward movements introduces unnecessary movements, reducing time efficiency. The xMovRew term counterbalances this tendency, encouraging the agent to remain vertically focused despite the unavoidable ergonomic burden.

$$\text{xMovRew} = \begin{cases} -20\,\text{countX}, & \text{countX} \leq 5 \\ -100, & \text{countX} > 5 \end{cases} \tag{7.8}$$

The component zStepRew is exclusive to the DQN reward structure and is only applied when the pain risk is zero. This component rewards the agent based on the magnitude of the vertical movement (z-axis), encouraging time-efficient behavior and discouraging unnecessary adjustments that offer minimal ergonomic benefit. The reward for vertical movement is given by:

$$\text{zStepRew} = \frac{\text{d} \times \sin\alpha \times 100}{0.4} \tag{7.9}$$

This component is also normalized between 0 and 100.

The complete reward functions are defined in Equation 7.10 (Q-Learning) and Equation 7.11 (DQN). The weights of each term were selected through trial and error.

$$\text{r} = \begin{cases} -100, & \text{avgPain} = 1 \\ 0.3\,\text{ergRew} + 0.2\,\text{xMovRew}, & \text{avgPain} = 0 \end{cases} \tag{7.10}$$

$$\text{r} = \begin{cases} -100, & \text{avgPain} = 1 \\ 0.15\,\text{ergRew} + 0.3\,\text{zStepRew} & \\ +0.05\,\text{xMovRew}, & \text{avgPain} = 0 \end{cases} \tag{7.11}$$

This reward structure ensures that the agent is penalized for high pain and ergonomic risks, while incentivizing movements that reduce these risks and advance the transport task. Components such as ergRew, xMovRew, and the total reward r are normalized between -100 and 0 to maintain consistent scaling. These elements follow the generic RL framework depicted in Figure 109.

In DQN, the Q-table is replaced by a fully connected neural network. The input layer receives the object position, x_{obj} and z_{obj}, normalized to $[-1, 1]$ using the state space limits. The network includes one hidden layer with ReLU activations. The output layer has 35 neurons, corresponding to the discrete action space, and returns one Q-value per action.

7.5.5 Training Termination Criteria

For the training settings, the termination criteria are different between the simulation and the real-world environment. In simulation, training concludes when the moving average of the reward per episode remains constant over 100 episodes. In the real-world environment, training ends when the average of the ergonomic risk level remains below 2.5 with zero pain risk for 10 consecutive episodes. The ergonomic risk level threshold reflects RULA guidelines, which recommend further investigation and possible changes to the workspace for scores above 3.

7.5.6 Hyperparameter Optimization

Hyperparameter Optimization (HPO) seeks to identify the best-performing model configuration but is computationally expensive, particularly when repeated for each participant. To reduce computational overhead, HPO was performed only once using a human kinematic model with the participant with 1.79 cm of height. The selected hyperparameter set was then reused for pre-training and fine-tuning across all participants.

Q-Learning

For Q-Learning, HPO was conducted using a grid search across the parameter space detailed in Table 42. The learning rate (α) and discount factor (γ) were incremented by 0.1 across the ranges $\alpha \in$ (0, 1] and $\gamma \in$ [0, 1]. The epsilon decay duration was evaluated for different episode counts—[100, 200, 300] episodes with 10 cm discretization and [200, 300, 400] episodes with 6.25 cm discretization—allowing exploration of both short and long decay schedules.

To ensure efficient training, an early stopping criterion was applied: training was halted if the number of episodes exceeded twice the selected epsilon decay duration. This parameter controls the linear decay

of ϵ in the ϵ-greedy strategy, gradually shifting from exploration to exploitation.

Table 42: Q-Learning Hyperparameter Search Space

Hyperparameter	**Search Space**
Learning Rate (α)	[0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]
Discount Factor (γ)	[0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]
Epsilon Decay Duration (No. of Episodes)	[100, 200, 300] (10 cm) [200, 300, 400] (6.25 cm)

DQN

For DQN, a grid search over the space defined in Table 43 was conducted, resulting in 2187 combinations. As with Q-Learning, an early stopping condition was enforced: training was terminated if the number of episodes exceeded twice the epsilon decay duration. This hyperparameter governs the linear decay of ϵ, defining the number of episodes needed to fully transition from exploration to exploitation.

Table 43: DQN Hyperparameters Search Space

Hyperparameter	**Search Space**
Learning Rate	1e-5, 1e-4, 1e-3
Discount Factor	0.5, 0.9, 0.999
Epsilon Decay Duration	1500, 2000, 2500
Soft Update Rate	1e-4, 1e-3, 1e-2
Buffer Size	5000, 10000, 20000
Batch Size	64, 128, 256
Neurons No.	128, 256, 512

7.5.7 ROS Implementation

The RL-based controller was implemented as a ROS package and integrated with the UR10e control framework. The overall ROS architecture is shown in Figure 115.

The agent node, along with the Q-Learning and DQN, were implemented using Python. The remaining nodes were implemented in C++. The DQN implementation also relied on the PyTorch library with GPU acceleration to improve computational efficiency and speed up training. PyTorch utilities were used for

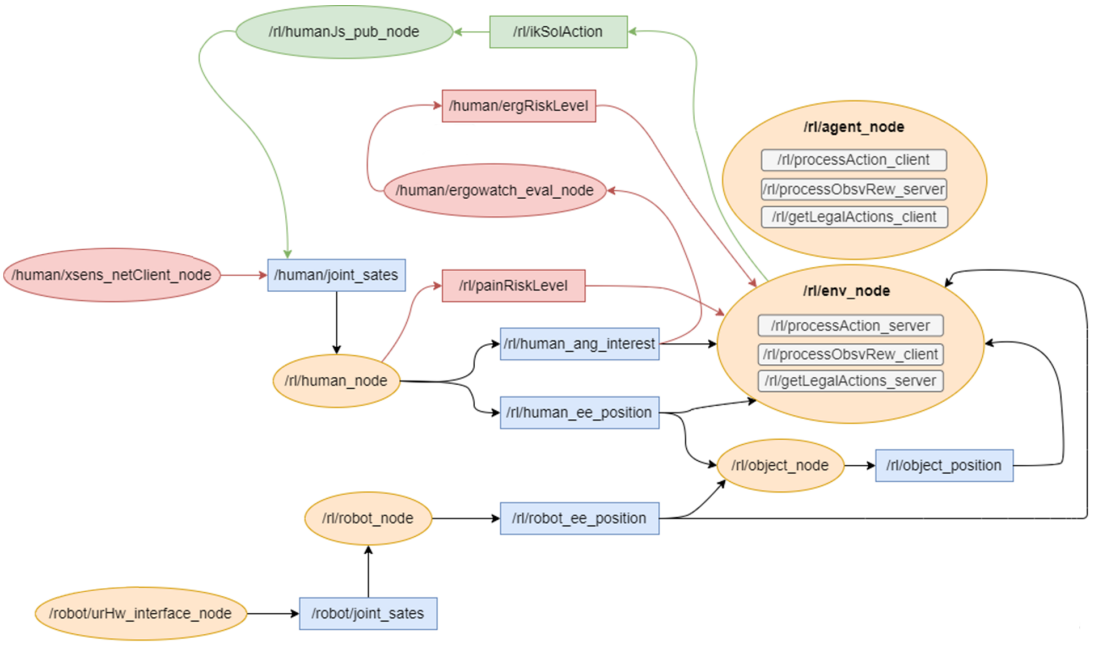


Figure 115: Schematic of communication between nodes through topics and services (Nodes and topics depicted at red are only used in real-world environment settings and the ones at green solely in simulation environment).

Q-value evaluation and Huber loss computation, and the Adam optimizer was employed to update the policy network parameters.

The RL environment required the implementation of five essential nodes: `/rl/human_node`, `/rl/robot_node`, `/rl/object_node`, `/rl/env_node`, and `/rl/agent_node`. Each node processes information related to the human, the UR10e robot, the manipulated object, the environment, and the agent. These nodes subscribe to ROS topics, process incoming messages such as Cartesian positions and risk levels, and publish relevant data.

The `/rl/human_node` computes the human end-effector position and publishes it to `/rl/human_ee_position`. This requires transforming the arm end-effector frame into the base frame. The node subscribes to `/human/joint_states`, extracts the shoulder and elbow joint angles and publishes the joint angles to `/rl/human_ang_interest`.

The `/rl/robot_node` computes the UR10e end-effector Cartesian position by transforming the robot end-effector frame relative to the base frame. It subscribes to `/robot/joint_states`, which are published by the `/robot/urHw_interface_node`, and publishes the computed position to `/rl/robot_ee_position`.

The `/rl/object_node` estimates the object Cartesian position using the human and robot end-effector positions. It subscribes to `/rl/human_ee_position` and `/rl/robot_ee_position`, com-

putes their midpoint, and publishes the result to /rl/object_position.

The /human/ergowatch_eval_node was modified to subscribe to /rl/human_ang_interest instead of /human/joint_states. Based on these joint angles, the RULA ergonomic score is computed and published to /human/ergRiskLevel.

The /rl/env_node enables interaction with both simulated and real environments. It subscribes to /rl/human_ee_position, /rl/human_ang_interest, /rl/robot_ee_position, and /rl/object_position to build the observation space. Ergonomic risk is evaluated by /human/ergowatch_eval_node, and pain risk by /rl/human_node, which are also subscribed by /rl/env_node.

In the real-world configuration, human joint angles are obtained from the Xsens system via the /human/xsens_network_client, which publishes to /human/joint_states. In simulation, joint angles are generated from inverse kinematics solutions corresponding to the agent's selected actions. These solutions are published to /rl/ikSolAction and converted into full joint state messages by the /rl/humanJs_pub_node, which publishes to /human/joint_states. This mechanism is used exclusively in simulation.

To process actions and ROS callbacks concurrently, the /rl/env_node launches a dedicated thread running the main control loop. This loop resets the environment, executes actions, and checks termination conditions.

The /rl/agent_node communicates with the environment through ROS services. It sends reset and action requests to /rl/processAction_server and receives observations and rewards via /rl/processObsvRew_client. The agent also queries /rl/getLegalActions_server to obtain the shaped action space. Based on the received data, the agent updates either the Q-table or the DQN policy network.

7.5.8 Experimental Validation

Participants

This exploratory study was conducted with four healthy participants (3 male and 1 female). They were recruited to provide variation in height to enable evaluation across different anthropometric profiles. The participants ranged in age from 22 to 48 years (29.25 $\pm$ 12.53) and had an average height of 1.713 $\pm$ 0.107 meters. The 2-DOF model was scaled for each individual using their height, shoulder span, upper-arm length, and forearm length.

All participants were instrumented using an upper-body motion capture system. A total of nine inertial



a) b) c)

Figure 116: Participant instrumented with Movella's MVN Awinda in the "upper body and no hands" configuration. For illustration purposes, IMUs are shown on the outside of the straps.

wireless trackers were placed according to the manufacturer's guidelines [116], specifically on the head, shoulders, chest, upper arms, forearms, and waist, and secured using Velcro straps. The system was calibrated in the "upper body, no hands" configuration. Calibration was repeated as needed until visual inspection confirmed satisfactory tracking accuracy. Figure 116 illustrates the participants' instrumentation. This study followed the ethical guidelines of the CEICVS (147/2021), the Declaration of Helsinki, and the Oviedo Convention.

Experimental Protocol

To ensure optimal performance, the following procedure was followed:

1. **Hyperparameter Optimization (HPO):** Models were pre-trained and evaluated in simulation based on reward, pain and ergonomic risk, and time efficiency. The best-performing model was selected.
2. **Real-World Deployment:** The selected model was tested over 10 episodes in the real environment, with key metrics recorded.
3. **Fine-Tuning:** The model was adapted to real-world dynamics through fine-tuning.
4. **Final Evaluation:** The fine-tuned model was re-tested in the real environment to assess performance gains.

This protocol was repeated for each participant and both algorithms, except for the HPO step, which is performed only for DQN. At the end, each participant completed an usability questionnaire. It consisted

in the following questions:

- **Q1**: Which experiment was more comfortable?
- **Q2**: I felt that, after fine-tuning the model, the transport became more comfortable.
- **Q3**: I felt that the fine-tuning phase helped keep the elbow angle within safe limits, avoiding excessive flexion or extension.
- **Q4**: I believe this controller would be useful in industrial settings, enabling adaptation to individual physical needs, including WRMSD workers.

Evaluation Metrics

The following metrics were recorded to evaluate the performance of the algorithm during testing:

1. **Reward**: Total reward accumulated by the agent throughout the episode.
2. **Ergonomic and Pain Risk Levels**: Average ergonomic and pain risk levels observed during the episode.
3. **Steps**: Total number of actions executed by the agent.
4. **Object Travel Distance**: Cumulative distance traveled by the object, computed from the sequence of target Cartesian positions associated with the agent's actions.
5. **Episode Where Convergence Started**: Episode in which the accumulated reward began to stabilize (i.e., converge to a constant value).
6. **Average Loss per Episode (DQN only)**: Mean Huber loss computed across each episode.
7. **Transport Time (Real-world tests only)**: Duration from the start of transport until the episode termination condition was met.

7.6 Results

7.6.1 Hyperparameter Optimization

For Q-Learning, hyperparameter optimization (HPO) was executed on Machine A, with the following specifications: GPU—NVIDIA GeForce GTX 1650; CPU—AMD Ryzen 7 4800H (8 cores, 16 threads); RAM—16 GB. The total runtime for the procedure was approximately 29 minutes for the 10 cm discretization level and 52 minutes for the 6.25 cm discretization level. The average training time per experiment was 5.1 minutes for the 10 cm setting and 9.4 minutes for the 6.25 cm setting.

For DQN, HPO was distributed across two machines. Machine A, previously described, and Machine B,

equipped with a NVIDIA GeForce RTX 3080 Ti (12 GB dedicated memory), an Intel Core i9-10940X CPU at 3.30 GHz (14 cores, 28 threads), and 62 GB of RAM. Machine A trained 1207 experiments over 387 hours, while Machine B trained 980 experiments in 336 hours.

Tables 44 and 45 present the best performing models resulting from the HPO. All models successfully completed training with zero pain risk. For the 10 cm discretization, the top five models achieved identical performance, with a reward of -70.37, a step count of 9, an ergonomic risk level of 2.009, and a traveled distance of 1.066,m. Among these configurations, the fastest converging model reached convergence at episode 195 and was trained using a learning rate of 0.9, a discount factor of 0.7, and an epsilon decay duration of 100 episodes.

For the 6.25 cm HPO, the results showed improved performance compared to the Learning Verification model. The best-performing configuration achieved a reward of -112.31. While the top models exhibited similar overall performance, the model trained with a learning rate of 1.0, a discount factor of 0.6, and an epsilon decay duration of 200 converged faster than the others, requiring 32% fewer training episodes than the next best configuration., reason why it was selected. Minor differences in ergonomic risk level and traveled distance were observed but were considered negligible.

Table 44: Best-performing models from Q-Learning's HPO (10 cm discretization). Selected model is highlighted in bold

Learning Rate	**Discount Factor**	**Epsilon Decay Duration**	**Reward**	**Pain Risk**[1]	**Ergo Risk**[1]	**Steps**	**Distance**[2] **(m)**	**Episode Conv.**[3]
0.9	**0.7**	**100**	**-70.37**	**0**	**2.009**	**9**	**1.066**	**195**
0.2	0.4	100	-70.37	0	2.009	9	1.066	208
0.2	0.7	200	-70.37	0	2.009	9	1.066	293
0.5	0.6	200	-70.37	0	2.009	9	1.066	293
0.1	0.0	200	-70.37	0	2.009	9	1.066	294

[1]Average pain and ergonomic risk levels; [2]Object's traveled distance; [3]Episode at which reward convergence started.

Table 46 presents the results for the best performing DQN models. The best performing configuration achieved a reward of 66.98 and completed the task in 5 steps. This model was trained using a learning rate of 1.0, a discount factor of 0.6, and an epsilon decay duration of 200 episodes.

The reward values obtained by the DQN model are not directly comparable to those of Q-Learning, as they are based on different reward functions (Eq. 7.11). However, when comparing other performance metrics, both models achieved zero pain risk. Although DQN exhibited a slightly higher average ergonomic

Table 45: Best-performing models from Q-Learning's HPO (6.25 cm discretization). Selected model is highlighted in bold

Learning Rate	**Discount Factor**	**Epsilon Decay Duration**	**Reward**	**Pain Risk**[1]	**Ergo Risk**[1]	**Steps**	**Distance**[2] **(m)**	**Episode Conv.**[3]
0.1	0.4	300	-112.31	0	1.992	15	1.093	487
0.1	0.4	200	-112.31	0	1.992	15	1.093	494
0.1	0.3	200	-113.10	0	2.016	15	1.119	453
0.8	0.7	300	-113.88	0	2.039	15	1.145	394
0.1	0.5	200	-113.88	0	2.039	15	1.145	502
1.0	**0.6**	**200**	**-115.40**	**0**	**2.039**	**15**	**1.145**	**298**

[1]Average pain and ergonomic risk levels; [2]Object's traveled distance; [3]Episode at which reward convergence started.

risk level (2.124) than Q-Learning Q-Learning (2.039 for a discretization of 6.25 cm and 2.009 for a discretization of 10 cm), the difference was minimal. A key advantage of the DQN model is its ability to find a solution in just 5 steps, substantially fewer than the Q-Learning model. This efficiency is attributed to the variable step size (d) in DQN action space, which allow the task to progress more efficiently. Notwithstanding, this came at the cost of requiring significantly more episodes to reach convergence.

Table 46: Best-performing models resulting from DQN hyperparameter optimisation. The selected model is highlighted.

α[1]	γ[2]	τ[3]	**Neurons**[4]	**Batch**[5]	**Buffer**[6]	**Eps. Dec.**[7]	**Reward**	**Pain**[8]	**Erg. Risk**[9]	**Steps**	**Distance**[10]	**Ep. Conv.**[11]
10^{-3}	0.999	10^{-3}	512	64	5000	2500	67.60	0	2.071	6	0.923	3320
10^{-3}	0.999	10^{-3}	512	128	5000	2500	67.18	0	2.102	6	0.921	3361
10^{-3}	0.999	10^{-3}	512	64	5000	1500	66.98	0	2.124	5	0.921	1609
10^{-3}	0.999	10^{-3}	512	128	5000	1500	66.98	0	2.124	5	0.921	1672
10^{-3}	0.999	10^{-3}	512	128	20000	1500	66.98	0	2.124	5	0.921	1721

[1]Learning rate, [2]Discount factor, [3]Soft target update rate, [4]Number of neurons in the FCN hidden layer, [5]Batch size, [6]Replay buffer size, [7]Epsilon decay duration, [8]Average pain level, [9]Ergonomic risk level, [10]Object travelled distance, [11]Episodes to convergence.

Hyperparameter Influence

To analyze the influence of individual hyperparameter values on the performance metrics obtained during the HPO experiments, a Pearson correlation analysis was performed using the `pandas` library.

The resulting correlation matrices are shown in Figure 117, presenting both discretization levels for Q-Learning algorithm.



(a) 10 cm

(b) 6.25 cm

Figure 117: Correlation matrix for a) 10 cm and for b) 6.25 discretization level, correlating hyperparameters and performance metrics. The hyperparameters include Learning Rate, Discount Factor and Epsilon Decay Duration. The performance metrics include reward per episode, average ergonomic risk level, steps taken per episode and the episode in which the reward started to converge. Matrix values correspond to Pearson correlation coefficients, denoted as r_{corr}.

At the 10 cm discretization level (Figure 117a), Epsilon Decay Duration showed a strong positive correlation with the episode at which convergence began ($r_{\text{corr}} = 0.77$), indicating that longer decay durations delay convergence. Learning Rate showed a moderate negative correlation with episode reward ($r_{\text{corr}} = -0.42$), and weak positive correlations with ergonomic risk ($r_{\text{corr}} = 0.30$) and number of steps ($r_{\text{corr}} = 0.34$). This suggests that higher learning rates may destabilize learning and lead to less optimal behavior. Discount Factor showed weak correlations with all performance metrics, indicating limited influence at this discretization level.

At the 6.25 cm discretization level (Figure 117b), Epsilon Decay Duration remained positively correlated with convergence time ($r_{\text{corr}} = 0.34$), although less strongly than at 10 cm. Learning Rate maintained weak negative correlations with reward ($r_{\text{corr}} = -0.33$) and episode to convergence ($r_{\text{corr}} = -0.42$), and a weak positive correlation with number of steps ($r_{\text{corr}} = 0.25$). This supports the observation that aggressive learning rates can reduce training efficiency. In the finer discretization, Discount Factor had a stronger role, showing a positive correlation with reward ($r_{\text{corr}} = 0.35$) and a negative correlation with ergonomic risk ($r_{\text{corr}} = -0.34$). This suggests that higher discount factors help the agent balance immediate and long-term ergonomic objectives.

Figure 118 presents the correlation results for DQN. Most hyperparameters showed very weak cor-

relations with the performance metrics, suggesting limited influence on performance. The main exception was the strong positive correlation between Epsilon Decay Duration and episodes to convergence ($r_{\text{corr}} = 0.81$), indicating that longer decay periods slow convergence, likely due to prolonged exploration. Learning Rate also showed a weak positive correlation with reward ($r_{\text{corr}} = 0.37$), suggesting a limited but measurable effect on performance.

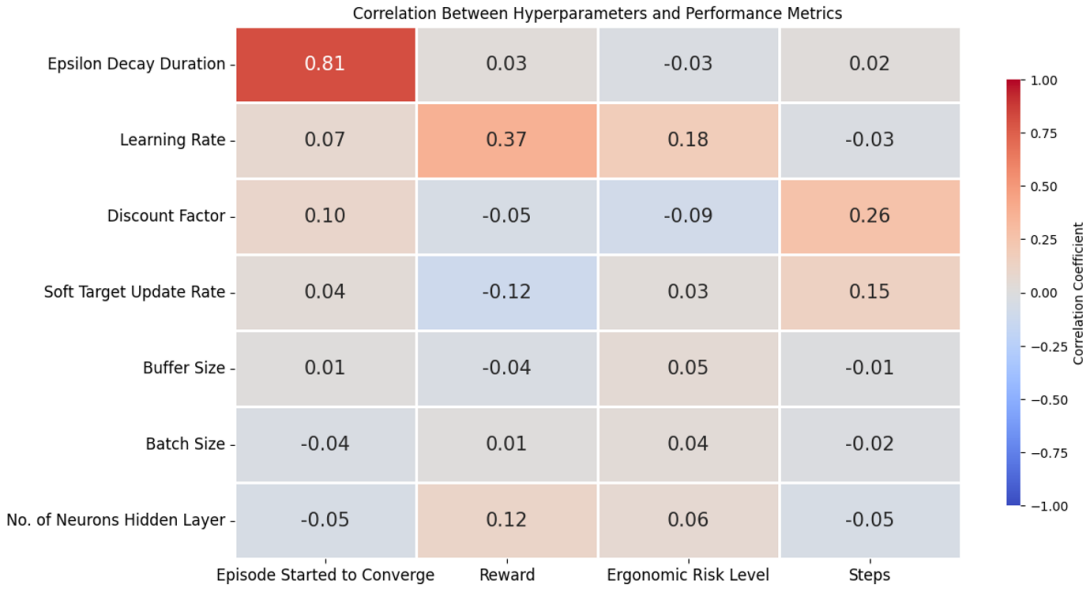


Figure 118: Correlation matrix relating DQN hyperparameters and performance metrics for successful experiments. Matrix values correspond to Pearson correlation coefficients, denoted as r_{corr}.

7.6.2 Real Environment Tests

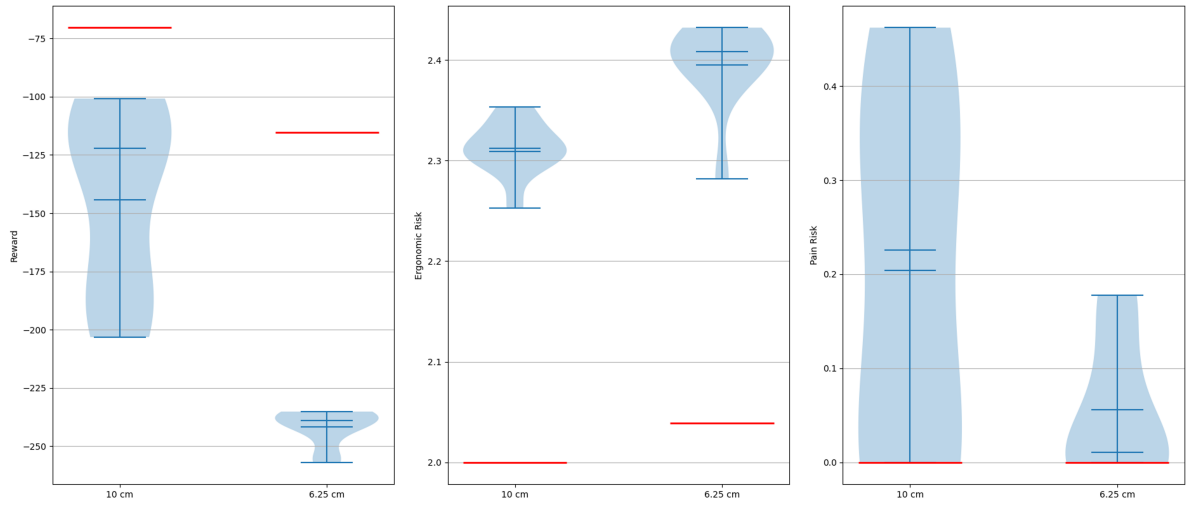
The best-performing models identified during the HPO were deployed in the real-world environment using the same hyperparameters adopted during their pre-training phase. Each model was executed 10 times per participant. If a model failed to complete the transport with pain-free motion or resulted in increased ergonomic risk, it was fine-tuned with the participant in the real-world setting and re-evaluated.

To gain deeper insight into agent behavior and refine the experimental protocol, a pilot study was conducted with Participant 1. This pilot study served two objectives: first, to analyze the agent's behavior in depth and better understand its decision-making process; and second, to improve the time efficiency of the experimental procedure by evaluating whether a 6.25 cm or 10 cm action space discretization was more appropriate for benchmarking DQN against Q-learning. The following subsections present the results of this pilot study.

The pre-training phase successfully produced policies that enabled efficient transport, maintained ergonomic risk below 2.5, and avoided pain risk. However, transitioning from simulation to real-world execu-

tion revealed a significant simulation-to-real gap, with reward values dropping across all experiments and most of the trials failing in preventing pain.

Q-Learning Figure 119 summarizes the performance metrics obtained for Participant 1 across the ten real-world trials. The detailed performance metrics for each trial are reported in Table 64 in the Appendix G.



Discretization (cm)	Avg. Steps	Avg. Distance (m)	Avg. Transport time (s)
10	9	1.066	24
6.25	15	1.15	33

Figure 119: Metrics obtained from the Q-Learning (both discretization levels) pre-trained models' execution in the real-world environment for P1. Red line represents metrics achieved in simulation.

Execution of the pre-trained Q-Learning models resulted in non-zero pain risk in 60% of the trials for the 10 cm discretization and in 50% of the trials for the 6.25 cm discretization. For the 10 cm case, ergonomic risk increased by 15.6% relative to the simulation baseline, while the reward degraded by 104.6%. For the 6.25 cm discretization, reward deterioration reached 116%, accompanied by a 17.9% increase in ergonomic risk.

These results indicate that policies trained with the finer 6.25 cm discretization are more sensitive to real-world variability, resulting in greater degradation of both reward and ergonomic performance. However, this configuration exhibited lower overall pain risk during execution. Figures 120 and 121 illustrate how the Q-Learning policy evolved from the simulation solution to the final fine-tuned trajectory for this participant under the 10 cm and 6.25 cm discretization settings, respectively. The reward maps shown are representative of simulation conditions only. In the real-world setting, human motion introduces stochasticity, such that revisiting the same state may result in different postural responses and, consequently,

different reward values.

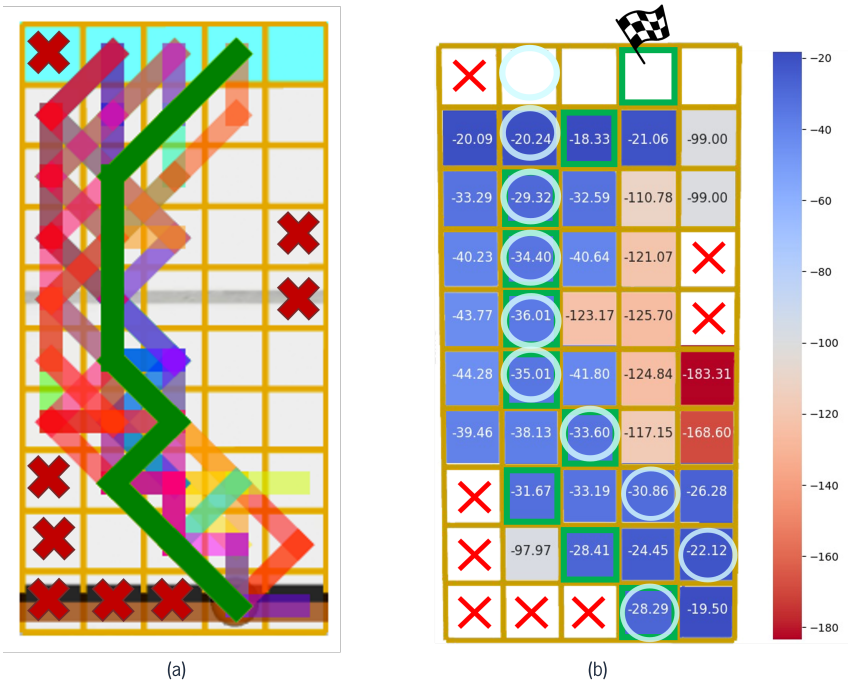
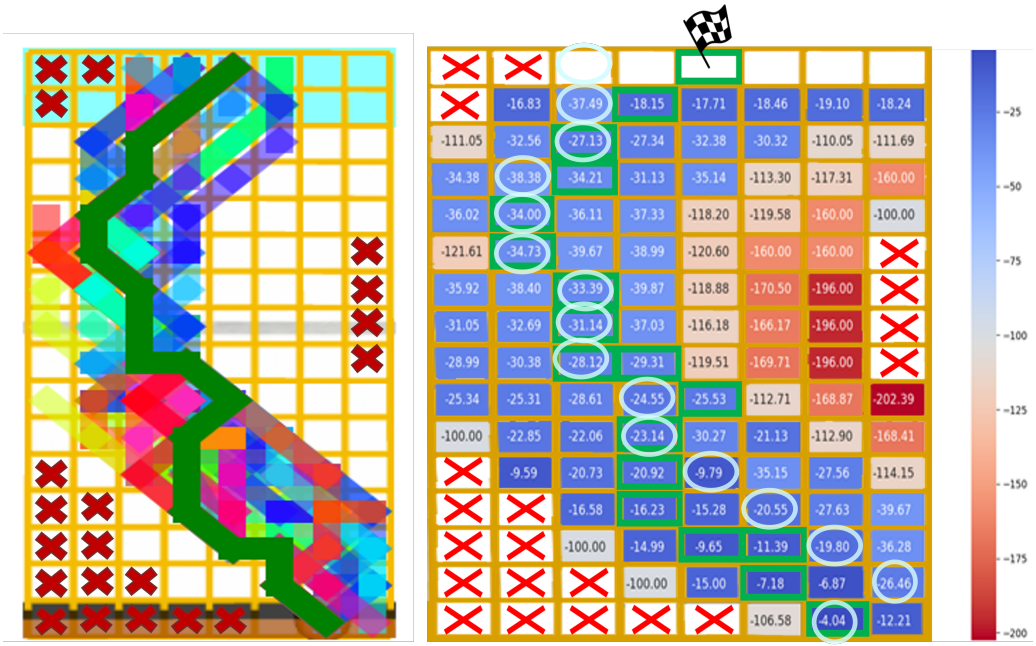


Figure 120: Q-Learning trajectories for Participant 1 under the 10 cm discretization during real-world fine-tuning. (a) Executed trajectories across trials. (b) Reward map from simulation, highlighting the final fine-tuned trajectory with green squares and the simulation pre-trained trajectory with light-blue circles.

Figures 120 and 121 show that, in simulation, the agent learned to avoid high-risk regions and to generate trajectories confined to areas identified as safe in the simulated reward map. However, this behavior did not always transfer reliably to the real-world setting. In practice, the real human exhibits additional degrees of freedom beyond those modeled in simulation, as well as stochastic movement behavior. Moreover, posture estimation in the real environment relies on inertial sensing, which is subject to calibration sensitivity and electromagnetic interference. As a result, the reward expected by the agent at a given state did not always match the reward observed during execution.

One illustrative simulation-to-real mismatch occurred when the agent attempted to reach state [10,2] (light-blue circle in Figure 120).. In simulation, the inverse kinematics solution yielded a safe posture with an elbow angle of 48.6 degrees. In the corresponding real-world trial, the participant's elbow angle reached 28.6 degrees, falling below the safe threshold of 30 degrees and triggering pain risk. **As consequence, during fine-tuning the model moved to state [10,4] (green square in Figure 120).** A similar discrepancy was observed for the 6.25 cm discretization at state [16,3], where the simulated IK solution produced an elbow angle of 45.1 degrees, while real-world execution resulted in an elbow angle of 27.3 degrees during Trial 1. These examples illustrate how modeling simplifications and sensing uncertainty



(a) (b)

Figure 121: Q-Learning trajectories for Participant 1 under the 6.25 cm discretization during real-world fine-tuning. (a) Executed trajectories across trials. (b) Reward map (from simulation) highlighting the final fine-tuned trajectory (green) and the simulation pre-trained trajectory (light blue).

jointly contribute to mismatches between predicted and actual rewards, thereby leading to the simulation-to-real gap.

DQN Figure 122 summarizes the performance metrics obtained for Participant 1 across the ten real-world trials. The detailed performance metrics for each trial are reported in Table 65 in the Appendix G.

The real-world execution of the pre-trained model shows a clear simulation-to-real gap. The reward is approximately 193% lower than the simulated baseline. The ergonomic risk is slightly higher than in simulation, as shown by the upward shift relative to the reference value, but remains within a comparable range. In contrast, the pain risk is non-zero in all trials, which is not acceptable. The number of steps required to complete the task is comparable to simulation, while the traveled distance is many times lower.

Figure 124 shows the object positions corresponding to transitions that triggered pain risk across the ten trials. The zoomed view in Figure 124a reveals that these positions are tightly clustered, identifying a specific region of the grid where human postural responses are more likely to induce pain. This problematic region is further illustrated in Figure 123. In Trials 8 and 10, the agent did not reach the upper grid area, indicating early episode termination following the onset of pain risk.

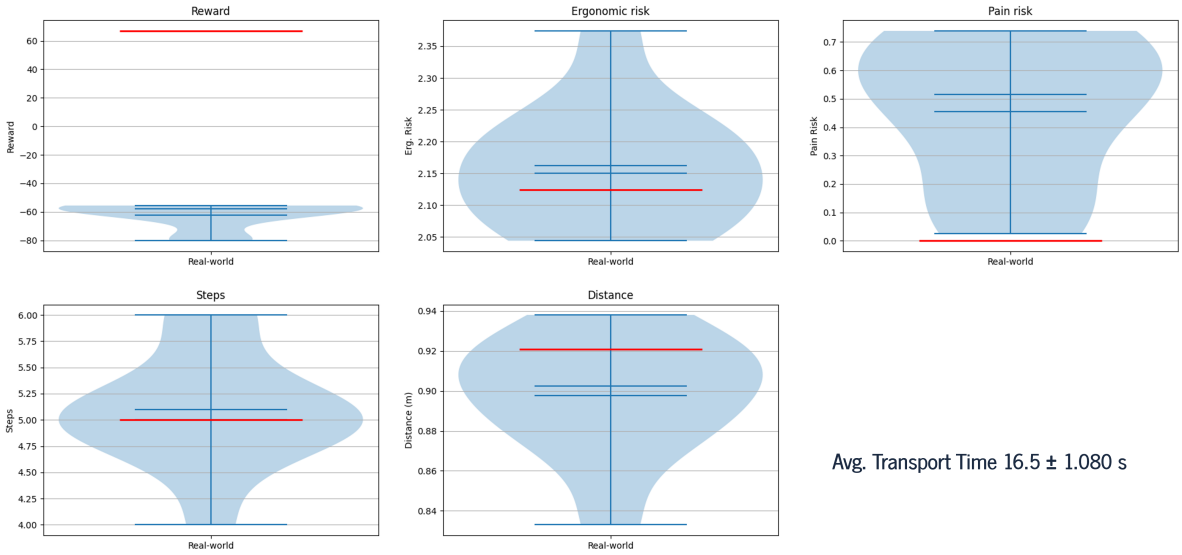


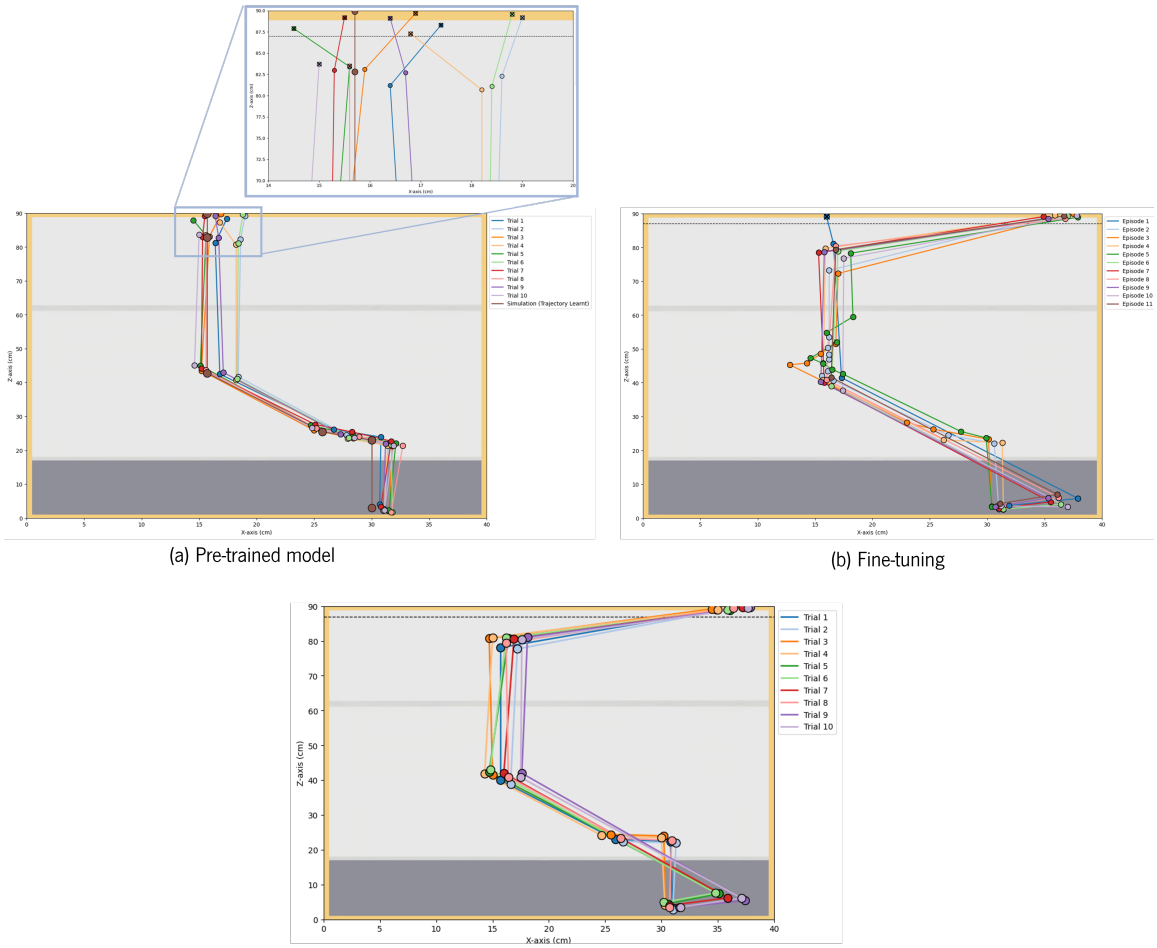
Figure 122: Metrics obtained from the DQN pre-trained model's execution in the real-world environment for P1. Red line represents metrics achieved in simulation.



(a) (b)

Figure 123: Human posture at the end of (a) pre-trained policy execution and (b) episode 2 during fine-tuning of the DQN model. In (a), the elbow is highly extended (≈13°), indicating pain risk, whereas in (b) the elbow remains flexed (≈56°), within a safe range.

Figure 124a) shows the trajectories executed by the pre-trained model in the real-world environment. The zoomed-in view highlights that the terminal positions are closely clustered, indicating a problematic region near the end of the transportation, where human postural responses are more likely to trigger pain risk in the real-worl. in Trials 8 and 10 the episode terminated early, preventing the agent from reaching the upper grid area. Figure 124b) illustrates the trajectories executed during the fine-tuning, where its possible to observe that the agent rapidly starts to convert its trajectory. Figure 152 in Appendix 7 illustrates the evolution of the key metrics during real-world fine-tuning. After the third episode, the loss decreased steadily, indicating stable learning, while the ergonomic risk level fluctuated within a narrow range between 2.191 and 2.384. By the end of training, the agent achieved a final reward of 57.79 with an ergonomic risk



(a) Pre-trained model

(b) Fine-tuning

(c) Fine-tuned model

Figure 124: Executed trajectories by the DQN model: (a) pre-trained model. black crosses represent state transitions that resulted in pain; (b) fine-tuning of the model. (c) fine-tuned model.

level of 2.384. The resulting trajectory required four steps and covered a total distance of 1.055 meters.

After the fine-tuning, As shown in Figure 124c), the trajectories were largely consistent, with variations occurring primarily in the first-step action selection. In Trials 5–7, 9, and 10, the agent chose a different initial action and completed the transport in four steps, whereas the remaining trials required an additional step, totaling five. Despite this variation, the agent successfully completed all trials without triggering pain risk.

Overall results

Table 47 summarizes the performance of Q-Learning and DQN across all participants and anthropometric profiles. The comparison includes simulation-only evaluation (Sim/Sim), direct real-world execution of models trained in simulation (Sim/Real), and real-world performance after fine-tuning (Real/Real). For the latter two conditions, all metrics are averaged over ten real-world trials.

A preliminary benchmark comparing Q-Learning with 10 cm and 6.25 cm discretization revealed consistently superior performance for the 10 cm discretization, including higher cumulative rewards and shorter execution times. Based on these results, only the Q-Learning model with 10 cm discretization was trained and evaluated for the remaining participants. This explains why participant P1 includes results for three models, whereas the other participants present results for only two models.

Across all participants and both algorithms, deploying pre-trained policies directly in the real environment resulted in a clear simulation-to-real gap, characterized by degraded rewards and the frequent emergence of pain risk. This gap was more pronounced for Q-Learning, which showed higher reward variance and less consistent real-world behavior.

Real-world fine-tuning systematically eliminated pain risk for all participants and both algorithms, while restoring ergonomic risk levels close to those observed in simulation. For Q-Learning, fine-tuning led to moderate performance recovery. In contrast, DQN consistently achieved superior real-world performance after fine-tuning, completing the task in fewer steps and shorter time across all participants while maintaining comparable ergonomic risk levels.

These trends are reflected in the participant-level results. For example, the Q-Learning trajectories for Participant 1 exhibit noticeable variability during fine-tuning, consistent with the higher reward standard deviation reported in Table 47. In contrast, the fine-tuned DQN model for the same participant achieved a stable ergonomic risk level of 2.28 ± 0.08. Across the ten trials, the transport task was completed in an average of 12.1 ± 1.6 seconds, covering a distance of 0.75 meters.

Usability Questionnaire

To assess perceived comfort, safety, and potential industrial relevance, participants were asked to respond to a usability questionnaire. Figure 125 presents the distribution of participants' responses to the usability questionnaire.

All participants reported higher comfort during transport with the DQN-based controller compared to Q-Learning. This perception is consistent with the objective results, where DQN produced smoother trajectories, shorter execution times, and more stable ergonomic risk during real-world execution.

Most participants agreed that fine-tuning improved comfort by keeping the elbow within safer limits, aligning with the observed elimination of pain risk after real-world adaptation. Responses also emphasized the importance of fine-tuning, supporting the quantitative evidence of a significant simulation-to-real gap.

Finally, all participants indicated that the proposed controller would be suitable for industrial use, particularly for applications involving workers with work-related musculoskeletal disorders.

Table 47: Performance Comparison of Q-Learning and DQN Algorithms for Different Human Anthropometries

ID	Body Height	Discretization	Train / Exec[1]	Reward	Pain Risk[2]	Erg. Risk[3]	Steps	Distance (m)[4]	Time (s)[5]
P1	1.79	QL (10 cm)	Sim/Sim	-70	0	2.00	9±0.0	1.07±0.0	–
			Sim/Real	-144±40	1	2.31±0.03	9±0.0	1.07±0.0	24±0.0
			Real/Real	-107±24	0	2.33±0.12	9±0.0	1.15±0.0	26±0.0
		QL (6.25 cm)	Sim/Sim	-115	0	2.039	15±0.0	1.145±0.0	–
			Sim/Real	-249.26±26.51	1	2.404±0.049	15±0.0	1.145±0.0	33±0.0
			Real/Real	-99.35±7.469	0	2.030±0.031	17±0.0	1.270±0.0	37±0.0
		DQN	Sim/Sim	66.98	0	2.12	5±0.0	0.92±0.0	–
			Sim/Real	-62±9	1	2.17±0.09	5.3±0.5	0.91±0.02	16.5±1.1
			Real/Real	62±2	0	2.22±0.07	4.5±0.5	1.06±0.01	18.4±1
P2	1.83	QL (10 cm)	Sim/Sim	-57.39	0	2.09	8±0.0	1.05±0.0	–
			Sim/Real	-106±21	0	2.43±0.08	8±0.0	1.03±0.0	24.3±0.2
			Real/Real	-87±5	0	2.23±0.04	9±0.0	1.02±0.0	26.3±0.5
		DQN	Sim/Sim	65.76	0	2.15	3±0.0	0.854±0.0	–
			Sim/Real	-89±13	1	2.45±0.07	3.9±0.3	0.77±0.04	13.5±1.4
			Real/Real	92±4	0	2.28±0.08	4±0.0	0.75±0.01	12.1±1.6
P3	1.69	QL (10 cm)	Sim/Sim	-68	0	2.12	8±0.0	0.924±0.0	–
			Sim/Real	-145±24	1	2.51±0.06	8±0.0	0.932±0.01	22.1±0.1
			Real/Real	-80±12	0	2.31±0.08	8±0.0	0.916±0.02	21.4±0.4
		DQN	Sim/Sim	45	0	2.05	3±0.0	0.83±0.0	–
			Sim/Real	-65±15	1	2.29±0.06	3.7±0.5	0.852±0.02	15.2±1.7
			Real/Real	37±12	0	2.33±0.02	3±0.0	0.843±0.01	14.1±1.4
P4	1.62	QL (10 cm)	Sim/Sim	-97.17	0	2.25	9	1.07	–
			Sim/Real	-145±49	1	2.48±0.21	6.6±3.9	0.776±0.47	19.8±11
			Real/Real	-106±6	0	2.42±0.032	8±0.0	0.954±0.03	23.7±0.0
		DQN	Sim/Sim	41	0	2.14	5	0.80	–
			Sim/Real	-93±48	1	2.19±0.13	7.4±3.5	0.763±0.02	16.3±1.8
			Real/Real	25±6	0	2.39±0.04	4.5±1.2	0.728±0.06	15.6±2.0

[1]Training and execution environments.

[2]Average pain risk level.

[3]Average ergonomic risk level.

[4]Distance covered by object.

[5]Execution time in real environment.

7.7 Discussion

7.7.1 Discussion

Hyperparameter Optimization and Discretization

As expected, the finer 6.25,cm discretization led to longer training times. Hyperparameter search led to improved cumulative reward while reducing ergonomic risk and travel distance. At 10 cm, the similar

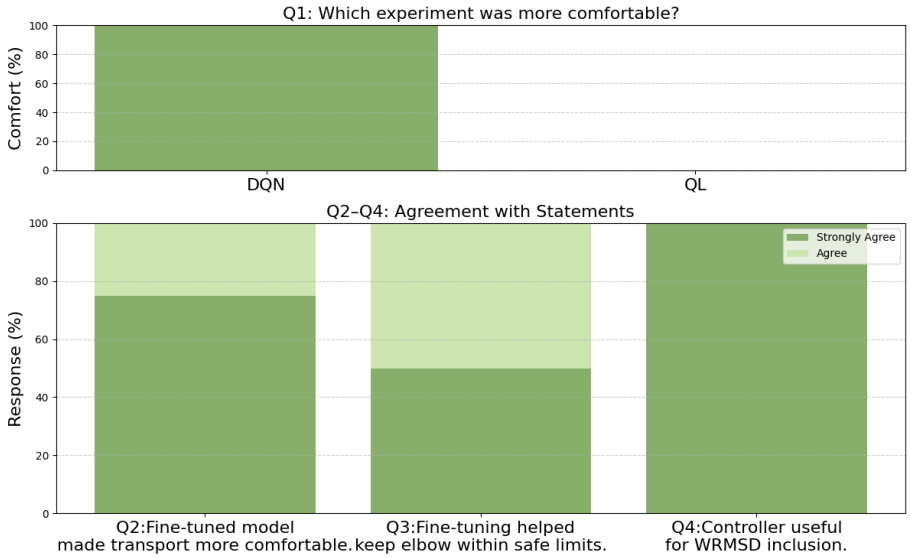


Figure 125: Usability questionnaire responses

performance of the top models indicates robustness of the Q-Learning solution at higher discretization, with convergence speed emerging as the main differentiator.

Correlation analysis highlights discretization-dependent effects. At 10 cm, convergence timing is largely dominated by epsilon decay duration ($r_{\text{corr}} = 0.77$), while the learning rate influences reward and movement efficiency. At 6.25 cm, the discount factor becomes more relevant to reward ($r_{\text{corr}} = 0.35$) and ergonomic risk ($r_{\text{corr}} = -0.34$), reflecting the increased importance of long-term planning in a larger state space. For DQN, correlations were generally weak, with the exception of epsilon decay duration with convergence time ($r_{\text{corr}} = 0.81$), reflecting the dominant role of exploration scheduling in determining convergence speed while suggesting lower sensitivity to hyperparameter tuning compared to Q-Learning.

Despite the finer discretization, the 10 cm model outperformed the 6.25 cm model in simulation, achieving lower ergonomic risk, shorter paths, and fewer steps. This confirms that increased discretization does not necessarily translate into better performance. In real-world deployment of the pre-trained policy, however, the 6.25 cm model exhibited fewer pain-risk events (50% vs. 60%), indicating a modest advantage in avoiding critical boundary conditions. After fine-tuning, the 10 cm model remained preferable, achieving comparable ergonomic risk while completing the task faster and with fewer steps.

Training Efficiency and Real-World Adaptation

Q-Learning converged significantly faster than DQN during the pre-training phase. On average, DQN required 15 minutes more than Q-Learning to converge in simulation (19.2 min vs. 4.2 min), reflecting the difference between direct Q-table updates and neural network optimization with experience replay.

However, DQN exhibited faster adaptation in real-world deployment, converging on average in 5.4 minutes compared to 11.4 minutes for Q-Learning, resulting in a 6-minute reduction that partially offsets the higher simulation training cost. This behavior suggests that the representations learned by DQN transfer more effectively from simulation to physical environments.

The simulation-to-real gap manifested consistently across all participants. For Q-Learning, transferring the policy from simulation to real-world execution resulted in an average reward degradation of 84.6%, with pain risk emerging for all participants except P2, likely due to anthropometric differences associated with greater body height. In contrast, DQN exhibited a larger absolute reward reduction (241%), with pain risk observed for all participants during initial real-world deployment. Despite these degradations, fine-tuning proved essential and effective, eliminating pain risk for all participants and for both algorithms, while restoring ergonomic performance to levels comparable to those observed in simulation.

This simulation-to-real gap can be attributed to two main factors: (1) real human behavior exhibits motor variability and additional degrees of freedom that are not fully captured by the digital human model, and (2) IMU-based posture estimation might introduce calibration drift and measurement noise that affect the reward computation during real-world execution.

Q-Learning vs. DQN for Deployment

After fine-tuning, DQN consistently outperformed Q-Learning in task efficiency. DQN completed transport in 3-5 steps versus 8-9 for Q-Learning (10 cm), achieving execution times of 14-18 seconds compared to 22-26 seconds. This efficiency results from DQN's variable step-length action space, enabling larger task progress per decision.

DQN also maintained slightly better ergonomic performance than Q-Learning, with a lower average ergonomic risk (2.28 vs. 2.43), while covering shorter distances on average (0.82 m vs. 0.95 m).

The usability questionnaire validated these findings. All participants reported higher comfort with DQN and agreed the controller would be suitable for industrial use, particularly for workers with musculoskeletal disorders. Most participants indicated that fine-tuning improved comfort by keeping the elbow within safer limits, consistent with the elimination of pain risk.

Methodological Considerations

Classical Q-Learning convergence guarantees rely on strong assumptions, including a stationary MDP and sufficient exploration of the state–action space. These assumptions are systematically violated in real-world physical human–robot interaction, where human motor variability, sensor noise, and human

adaptation introduce non-stationarity into the learning process. As a result, Q-values learned during real-world deployment should not be interpreted as globally optimal solutions, but rather as empirical estimates that are valid for the specific interaction distribution observed during training. This limitation reflects an inherent trade-off in human-in-the-loop reinforcement learning. Incorporating real human interaction is essential to obtain policies that are transferable and meaningful in physical collaboration, but it comes at the cost of sacrificing formal convergence guarantees. In this context, learning prioritizes practical effectiveness in physical collaboration, accepting that formal convergence guarantees may not fully apply.

For DQN, formal convergence guarantees are not present by design, as the use of function approximation replaces tabular value updates. While this enables generalization across states and users, it also introduces potential sources of instability. Experience replay and target networks are employed to mitigate these effects, but they do not fully restore theoretical guarantees. In this study, the fine-tuning phase was limited to 10 trials per participant, which was sufficient to demonstrate short-term adaptation but not to assess long-term stability. Extended testing would therefore be necessary to evaluate robustness to rare edge cases and gradual behavioral changes due to factors such as fatigue or user adaptation.

7.8 Conclusions

This chapter presented the development and evaluation of a RL-based control strategy for personalizing cobotic assistance during shared object transportation tasks. The proposed approach enables robots to lead collaborative tasks while continuously adapting their motion to respect human ergonomic constraints and physical limitations, demonstrating how simulation-based training can be efficiently transferred to real-world human–robot interaction targeted fine-tuning with human users.

The first main contribution is the development of a human-sensitive RL framework that personalizes assistance based on ergonomic metrics and physical constraints. Rather than relying on predefined assistance rules, the system embeds ergonomic risk and pain-related considerations directly into the reward structure, allowing the cobot to learn appropriate interaction behaviors through experience. Both tabular Q-Learning and Deep Q-Networks were shown to acquire safe and ergonomically plausible behaviors in simulation, and to recover comparable performance after brief real-world adaptation, validating RL as a viable control paradigm for human-sensitive collaboration.

The second main contribution addresses a key deployment challenge through a training strategy that combines simulation-based pre-training with targeted real-world fine-tuning. Training exclusively through physical human interaction would require impractically long and potentially unsafe sessions; therefore,

simulation is used to efficiently acquire core task competencies. Brief fine-tuning with real users then adapts the learned policies to individual movement patterns and compensates for modeling simplifications. This hybrid strategy achieved strong real-world performance while eliminating pain risk, demonstrating that extensive real-world training is unnecessary when simulation provides suitable initial behaviors.

Additionally, the systematic comparison between tabular Q-Learning and DQN under identical conditions revealed important implementation insights. While both algorithms met the predefined safety and efficiency requirements after fine-tuning, DQN demonstrated superior real-world task efficiency, completing tasks with fewer actions and shorter execution times. This advantage was attributed to DQN's ability to learn more compact and transferable representations, resulting in reduced sensitivity to discrepancies between simulated and real-world reward landscapes. Interestingly, finer state-space discretization in Q-Learning models failed to improve performance while increasing convergence time, suggesting that more resolution does not automatically translate to better control policies.

Several limitations of this study must be acknowledged: (i) the experimental validation involved only four healthy participants, preventing statistical analysis and limiting conclusions about the approach's effectiveness for workers with musculoskeletal disorders; (ii) pain risk assessment relied on machine-state proxies rather than direct physiological measurements, potentially missing nuanced aspects of human discomfort; (iii) the simplified human kinematic representation, while enabling efficient learning, oversimplified postural variability and contributed to inaccurate ergonomic risk estimation in certain configurations, directly contributing to the observed simulation-to-real gap; (iv) the fine-tuning phase was limited in duration, preventing assessment of long-term stability and robustness to gradual human adaptation or fatigue.

Future work should address these limitations through: (i) extended validation studies with larger participant groups, including individuals with work-related musculoskeletal disorders; (ii) integration of physiological sensing to replace proxy-based pain estimation with digital biomarkers derived from wearable systems; (iii) integration of higher-fidelity human models that balance computational efficiency with accurate posture-dependent risk assessment; (iv) longer-term deployment studies to assess stability under prolonged interaction and evolving user behavior;

Chapter 8

Task Quality Based Control Strategy

8.1 Introductory Insight

HRC is a transformative paradigm that combines human flexibility with robotic precision and strength, enhancing overall efficiency and task execution. In manufacturing, HRC has already demonstrated clear benefits, including increased productivity, improved task performance, and higher product quality [42]. Beyond efficiency gains, robots equipped with appropriate control strategies can provide active support to workers by compensating for errors or fatigue, thereby reducing production defects and ensuring more consistent outcomes [42].

A critical advantage of HRC lies in its potential to accelerate the training and integration of non-expert workers. By interacting with collaborative robots, novice operators can acquire task-specific skills more rapidly, while the robot provides real-time feedback, guidance, and correction. In this way, robots do not simply automate tasks but also function as training partners, enabling a more inclusive and skilled workforce.

This chapter introduces a proof-of-concept of a task quality management module designed to support the training of non-expert workers, enabling robots to ensure consistent task execution while enhancing both productivity and skill acquisition. The work builds upon the manuscript “An Admittance-based Control Strategy for Human–Robot Collaboration Task Quality Management” (ICARSC 2024) [271]. Although the long term goal of this research is to evaluate whether a control strategy based on admittance control can effectively train novice subjects, the present study constitutes a preliminary investigation to address the research question: ”How can an admittance-based control strategy be designed to support novice training and task quality management, and how do users interact with and experience this control approach?”

8.2 Knowledge acquisition

The development of a task quality management module in HRC primarily requires addressing how robots interact safely and adaptively with human partners during task execution. This problem falls within the research domain of physical Human-Robot Interaction (pHRI), which studies control strategies that enable effective physical collaboration between humans and robots.

pHRI research focuses on developing safe and intuitive control strategies that enable physical collaboration between humans and robots. Such strategies allow robots to guide, support, and correct human actions in real time, thereby improving task performance and worker training. Most pHRI methods rely on impedance control, where displacement is the input and force is the output [272]. However, this approach requires integrated force/torque controllers, which are not universally available across collaborative platforms. In these cases, admittance control offers a practical alternative by using force as input and displacement as output, enabling effective and safe interaction even when force/torque control is not built into the system [273].

Recent studies have explored different variations of admittance control in robotic manipulators with distinct objectives: Han *et al.* [274] proposed an adaptive controller where inertia and damping were adapted online according to estimated operator intent, reducing human effort while preserving safety and accuracy; Ferraguti *et al.* [275] designed a variable admittance strategy to suppress oscillations and enhance stability during interaction; Li *et al.* [276] proposed a variable admittance controller with a haptic stability observer and model predictive tuning, using admittance to both enable compliance and actively suppress oscillations during high-stiffness or high-frequency interaction; and Kang *et al.* [277] developed a controller that integrates direct and indirect human intention to improve intuitiveness and provide task guidance. In Kang's formulation, direct intention refers to instantaneous robot responses to human-applied forces, while indirect intention reflects responses aligned with the longer-term goal of the motion. Notably, as in other works, the system dynamics were modeled with virtual stiffness set to zero, effectively disregarding the spring component of the mass–spring–damper system. This design attenuates deviations in undesired directions but does not actively correct the trajectory, placing the burden on the worker to infer the correct path. By contrast, including the spring component in the model allows convergence toward an equilibrium point, meaning that the robot can not only resist undesired deviations but also guide the operator back toward the intended trajectory. This transforms the controller from a passive feedback mechanism into an active collaborator capable of correcting human error and ensuring higher task accuracy.

8.3 Solution Conceptualization

The core idea of this controller is to develop a module that allows to train inexperienced workers using trajectories demonstrated by experienced workers aiming for fewer errors and higher productivity. The controller must operate on a UR10e, which does not provide torque control, making impedance control unfeasible. Therefore, a variable admittance control was adopted to modulate compliance based on how well the worker follows the reference. When the worker aligns with the trajectory, the robot stays compliant; when the worker deviates, the robot increases damping and inertia and adds stiffness to pull back toward the reference.

Figure 126 presents an overview of the control strategy. In the offline phase, expert demonstrations are recorded and saved in a file. During deployment, the admittance controller uses this reference to guide inexperienced operators. Please note that, as a proof of concept, "expert demonstrations" here refer simply to a reference trajectory. Future work should validate this controller using demonstrations from actual experts and novice workers.

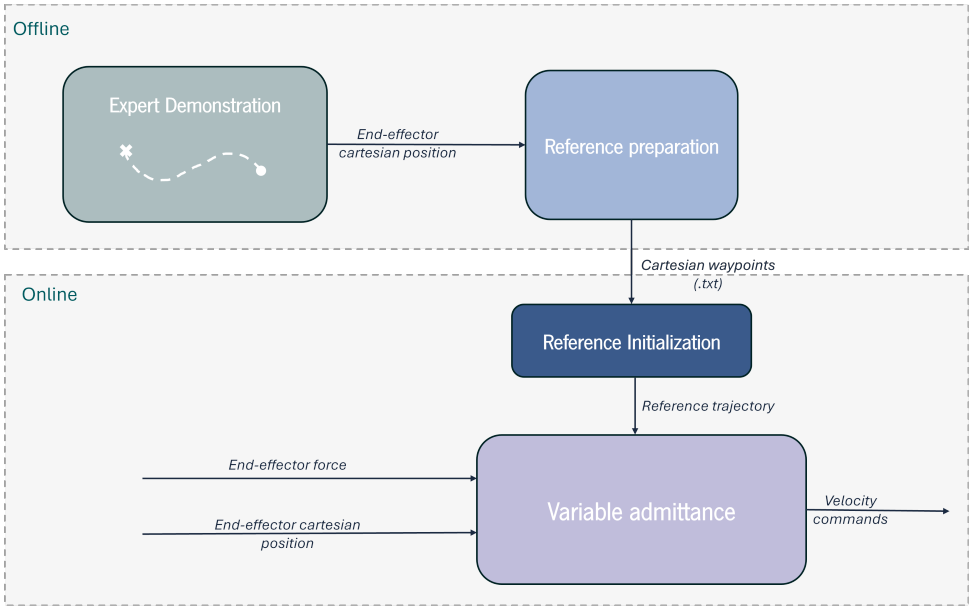


Figure 126: Conceptualization of the task quality control strategy.

8.4 Methods

The admittance controller developed in this work regulates the end-effector cartesian velocity to guide the human operator along a reference trajectory and minimize task error. Two modes were implemented on a Universal Robots UR10e: a free-drive mode and a guided-drive mode. The free-drive mode corresponds

to the direct intention strategy of [277], in which the robot responds to human-applied forces, remaining fully compliant and allowing the operator to move freely without trajectory correction. By contrast, the guided-drive mode extends the indirect intention strategy of [277] by introducing a virtual spring term into the admittance model. This addition results in nonzero stiffness that drives the end-effector back toward the reference trajectory, enabling the robot to actively correct deviations rather than merely resist them as implemented in [277]. Figure 127 illustrates the control scheme implemented for the two developed modes.

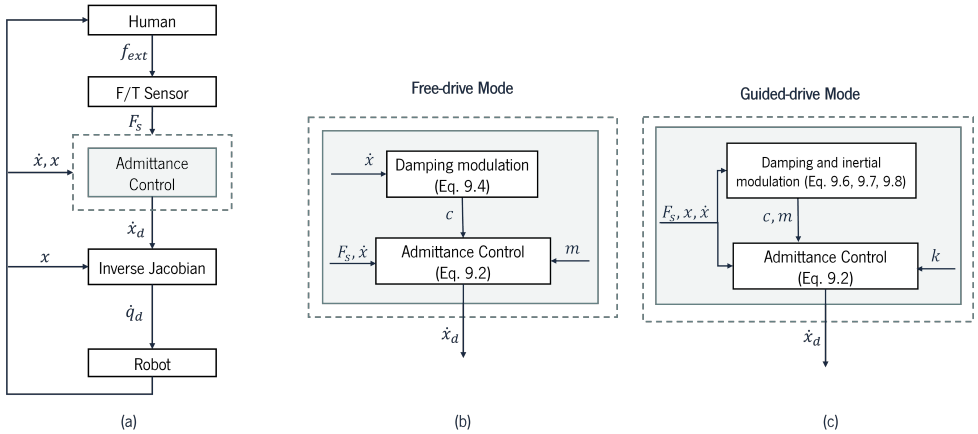


Figure 127: Block diagram of the admittance controller. (a) shows the general control architecture, common to both modes, where human-applied forces are measured through the F/T sensor and processed by the admittance control to generate desired end-effector velocities. In free-drive mode (b), compliance is achieved through variable damping c while inertia m is predefined and kept constant. In guided-drive mode (c), damping c and inertia m are varied according to the alignment between human-applied force and the reference trajectory, while stiffness k is predefined.

8.4.1 Admittance Control Formulation

Admittance control is a control strategy that controls the displacement of the system according to the force applied to it. This relationship between the force applied and the displacement produced is given by the equation that models a spring-mass-damper system, expressed as:

$$m(\ddot{x} - \ddot{x}_0) + c(\dot{x} - \dot{x}_0) + k(x - x_0) = F_{ext} \tag{8.1}$$

where m, c, and k denote inertia, damping, and stiffness parameters, respectively; x, $\dot{x}$, and $\ddot{x}$ are the position, velocity, and acceleration; x_0, $\dot{x}_0$, and $\ddot{x}_0$ represent equilibrium conditions; and F_{ext} is

the applied external force. The discrete admittance formulation takes different forms depending on the selected mode of operation:

- Free-drive mode (Equation 8.2) assumes that the robot remains compliant in all directions. In this mode, the robot follows the force applied by the operator without imposing resistance or moving toward a preferred position. Therefore, no reference trajectory or equilibrium point is defined. For this reason, the stiffness is set to zero, $k = 0$, and the equilibrium terms x_0, $\dot{x}_0$, and $\ddot{x}_0$ are also set to zero. The resulting motion is then governed only by inertia m and damping c, as they are sufficient to achieve the desired behavior:

$$\dot{x}[n] = \frac{F_{ext}[n]\Delta n + m\dot{x}[n-1]}{m + c\Delta n}. \tag{8.2}$$

  where n represents the current time step, $n-1$ the previous time step, and Δn the time interval, corresponding to the inverse of the controller's frequency.

- Guided-drive mode (equation 8.3) introduces a nonzero stiffness ($k > 0$), creating a restoring term that pulls the end effector back toward the reference trajectory whenever it deviates from it. In this mode, the equilibrium position x_0 is defined at each control cycle as the nearest point on the reference trajectory. This equation was derived assuming zero initial acceleration and velocity ($\ddot{x}_0 = \dot{x}_0 = 0$):

$$\dot{x}[n] = \frac{F_{ext}[n]\Delta n + m\dot{x}[n-1] - k\Delta n[x[n] - x_0]}{m + c\Delta n}. \tag{8.3}$$

In both cases, the measured variable is the applied force F_{ext}, and the controlled variable is the desired velocity $\dot{x}$.

8.4.2 Free-drive mode

Admittance control combines three responses to external forces. Inertia, governed by m, resists changes in acceleration. Damping, governed by c, scales the system velocity in response to the applied force. Stiffness, governed by k, generates a restoring force proportional to displacement from the equilibrium position. This ability to combine inertial, damping, and stiffness responses allows the robot to adapt its assistance to human intentions, thereby reducing trajectory errors and enabling personalized support.

In free-drive mode, since $k = 0$, the parameter of greatest practical relevance is damping c, as it directly influences system velocity. As shown in Eq. 8.2, higher damping reduces velocity, while lower damping increases it. Consequently, damping should be adjusted dynamically, increasing for precise, low-velocity movements and decreasing for fast, high-velocity movements. Furthermore, damping must be calibrated to the specific hardware and the operator's force capacity. This calibration is expressed by the reference force F_{std}, which serves as a normalization factor. The relationship governing damping is given in Eq. 8.4, where c is directly proportional to F_{std} and inversely proportional to the magnitude of the previous system velocity $|\dot{x}[n-1]|$:

$$c(\dot{x}) = \frac{F_{std}}{|\dot{x}[n-1]|} \tag{8.4}$$

In a hypothetical case, if the force applied to the robot (F_{ext}) is constant and $k = 0$, the system's velocity will converge to a maximum and stabilize. This makes acceleration $\ddot{x}$ zero, resulting in Eq. 8.5, deduced from Eq. 8.1:

$$\dot{x}_{max} = \frac{F_{ext}}{c} \tag{8.5}$$

As c directly affects the system's maximum velocity (Eq. 8.5), bounds c_{min} and c_{max} must be defined, chosen to correspond to the maximum ($\dot{x}_{max}$) and minimum ($\dot{x}_{min}$) velocities suitable for the task.

8.4.3 Guided-drive mode

The guided-drive mode was designed to support and correct the operator during task execution. When the operator deviates from the reference trajectory, the robot increases its resistance, making off-trajectory motions heavier and more difficult to sustain (Fig. 128). At the same time, a restoring force is generated, pulling the end-effector back toward the reference path. The complete control loop implementing this behavior is summarized in Algorithm 3, while the adaptive modulation of admittance parameters is detailed in Algorithm 4. Unlike the free-drive mode, where damping varies only with force and velocity, guided-drive adapts damping c, inertia m, and stiffness k according to the direction of the applied force F_{ext} relative to the reference trajectory. Prior to parameter adaptation, F_{ext} is conditioned through thresholding to suppress sensor noise, and residual motions within a 5 mm dead-zone of the reference are suppressed to prevent jitter (steps 1 and 4 of Algorithm 3).

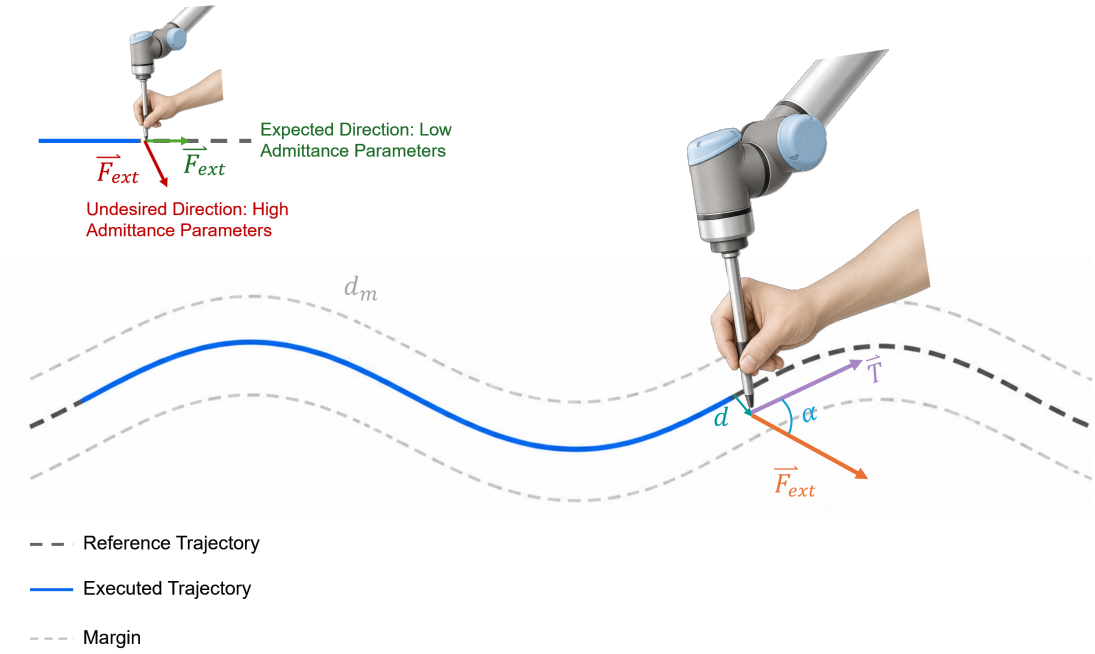


Figure 128: Guided-drive modulation strategy. The applied force $\vec{F}_{ext}$ is evaluated using its angular deviation α from the local trajectory tangent $\vec{T}$ and the distance d from the reference trajectory. Forces aligned with or directed toward the reference path reduce resistance, whereas forces driving the end-effector away from the reference increase resistance.

The angular deviation α (Eq. 8.6) quantifies the alignment between the applied force vector $\vec{F}$ and the local trajectory tangent $\vec{T}$, and is computed as:

$$\alpha = \arccos\left(\frac{\vec{F_{ext}} \cdot \vec{T}}{\|\vec{F_{ext}}\|\|\vec{T}\|}\right) \tag{8.6}$$

When α is close to zero, the applied force is aligned with the reference trajectory, and the robot operates with high compliance, *i.e.*, stiffness is set to $k = 0$ and c and m are maintained at their minimum values, enabling the operator to progress along the reference with minimal resistance. As α increases, the applied force becomes increasingly misaligned with the trajectory, and c and m are modulated according to a scaling factor $\lambda \in [0, 1]$, which grows as α approaches $90°$ (Eqs. 8.7, 8.8):

$$c = c_{min} + \lambda(c_{max} - c_{min}) \tag{8.7}$$

$$m = m_{min} + \lambda(m_{max} - m_{min}) \tag{8.8}$$

The modulation does not depend only on the angular deviation α. It also considers the distance d between the end-effector and the closest point on the reference trajectory. A distance-dependent tolerance

α_d defines the admissible angular deviation for the current distance from the reference:

$$\alpha_d = \begin{cases} \dfrac{90^\circ}{\exp\left(\dfrac{d_m}{d} - 1\right)} & \text{if } d \leq d_m \\ 90^\circ & \text{if } d > d_m \end{cases} \tag{8.9}$$

where d_m is a predefined distance threshold. When the end-effector is close to the reference trajectory, α_d is small, and misaligned forces are penalized more strictly. As d increases, α_d also increases, allowing larger angular deviations when the end-effector is farther from the reference path. This relaxation prevents the controller from over-penalizing corrective forces and gives the operator more freedom to guide the robot back toward the reference trajectory. The difference between the actual and admissible angular deviations is then normalized into a scalar error e:

$$e = \frac{|\alpha_d - \alpha|}{90^\circ} \tag{8.10}$$

This error is mapped to an intermediate scaling factor λ_g through an exponential function:

$$\lambda_g = \frac{\exp(\beta\, e)}{90} \tag{8.11}$$

where β is a tuning parameter that controls how rapidly the damping and inertia increase as the angular deviation exceeds the admissible tolerance.

The final scaling factor λ is selected according to the distance from the reference trajectory and the predicted effect of the applied force (see Algorithm 4):

$$\lambda = \begin{cases} 1 & \text{if the force is predicted to diverge outside the admissible margin} \\ \lambda_g & \text{if the force is aligned with or moves toward the reference} \\ 1 - \lambda_g & \text{if the force is diverging within the admissible margin} \end{cases} \tag{8.12}$$

When the force is aligned with the reference or is predicted to reduce the distance to the reference trajectory, the controller uses $\lambda = \lambda_g$. When the force is predicted to increase the deviation but the end-effector is still within the admissible margin d_m, the controller uses $\lambda = 1 - \lambda_g$, allowing the robot to remain compliant while still limiting divergence. If the end-effector is already outside d_m and the applied force is predicted to further increase the deviation, λ is set to 1, which maximizes damping and inertia

and therefore minimizes compliance. The prediction is obtained by projecting a short displacement in the force direction and checking whether it increases the distance to the reference trajectory, as described by the away test in Algorithm 4.

Algorithm 3 Adaptive Admittance Control Strategy (Guided-drive)

Require: Current applied force $\mathbf{F}$, end-effector position $\mathbf{x}_{ee}$, reference trajectory $\mathcal{T}$, Jacobian $\mathbf{J}$, timestep Δn

Ensure: Joint velocities $\dot{\mathbf{q}}$

(1) Force conditioning:
$\mathbf{F} \leftarrow \text{ThresholdForce}(\mathbf{F})$ ▷ Removes sensor noise
(2) Reference equilibrium:
$\mathbf{x}_{eq} \leftarrow \text{NearestRefPoint}(\mathbf{x}_{ee}, \mathcal{T})$
(3) Adaptive parameters:
$(c, m) \leftarrow \text{ComputeAdmittanceParams}(\mathbf{F}, \mathbf{x}_{ee}, \mathbf{x}_{eq}, \mathcal{T})$ ▷ See Algorithm 4
$c \leftarrow \text{Clamp}(c, c_{\text{min}}, c_{\text{max}})$
(4) Dead-zone stabilization near equilibrium:
if $\|\mathbf{x}_{ee} - \mathbf{x}_{eq}\| < 5\,\text{mm}$ **then**
    $\mathbf{x}_{ee} \leftarrow \mathbf{x}_{eq}$ ▷ Prevents jitter near the reference trajectory
end if
(5) Admittance update:
$\dot{\mathbf{x}} \leftarrow \dfrac{\Delta n\,\mathbf{F} + m\,\dot{\mathbf{x}}_{\text{prev}} - \Delta n\,k(\mathbf{x}_{ee} - \mathbf{x}_{eq})}{m + c\,\Delta n}$
(6) Safety:
$\dot{\mathbf{x}} \leftarrow \text{LimitVelocity}(\dot{\mathbf{x}})$ ▷ Ensures bounded Cartesian velocities
(7) Joint mapping:
$\mathbf{J}^{\dagger} \leftarrow \mathbf{J}^T(\mathbf{J}\mathbf{J}^T + \mu^2\mathbf{I})^{-1}$ ▷ Damped least-squares inverse
$\dot{\mathbf{q}} \leftarrow \mathbf{J}^{\dagger}\dot{\mathbf{x}}$
$\dot{\mathbf{x}}_{\text{prev}} \leftarrow \dot{\mathbf{x}}$
return $\dot{\mathbf{q}}$

8.4.4 Implementation Details

Figure 129 presents the experimental setup for the implemented strategy. A six-DOF collaborative robot (UR10e, Universal Robots) was equipped with a custom tool at its end effector (Fig. 129a). The control framework was implemented in ROS and executed at 500 Hz, exchanging data through ROS topics. The built-in UR10e force/torque sensor provided the external force signal F_{ext}, which was processed using admittance control block to compute the desired Cartesian velocity. This velocity was mapped to joint space using the inverse Jacobian, $\dot{q}_d = J^{-1}\dot{x}_d$, and sent to the UR ROS Driver for execution.

Figure 130 presents the ROS implementation diagram. The hardware interface node `/robot/urHw_`

Algorithm 4 Computation of adaptive admittance parameters $(c,\ m)$

Require: Applied Force $\mathbf{F}$, position $\mathbf{x}_{ee}$, closest reference point $\mathbf{x}_{eq}$, reference trajectory $\mathcal{T}$, margin distance d_m

Ensure: Damping c and inertia m

(1) Local reference direction: $\mathbf{T} \leftarrow \text{LocalTangent}(\mathbf{x}_{eq}, \mathcal{T})$ ▷ Estimated from neighboring points

(2) Angle to reference: $\alpha \leftarrow \arccos\left(\dfrac{\mathbf{F}\cdot\mathbf{T}}{\|\mathbf{F}\|\|\mathbf{T}\|}\right)$

$a \leftarrow \sin(\alpha)$ ▷ Angle proxy, initially $a \in [0, 1]$

$d \leftarrow \|\mathbf{x}_{ee} - \mathbf{x}_{eq}\|$ ▷ Distance to reference trajectory

(3) Distance-dependent tolerance:

if $d > d_m$ **then**

  $\alpha_d \leftarrow 90^\circ$ ▷ Far from path: allow large angular deviations for recovery

else

  $\alpha_d \leftarrow \dfrac{90^\circ}{\exp\left(\dfrac{d_m}{d} - 1\right)}$ ▷ Near path: small tolerance

end if

(4) Predict effect of force on deviation:

$\mathbf{x}' \leftarrow \mathbf{x}_{ee} + \Delta\mathbf{x}(\mathbf{F})$ ▷ Short displacement along force direction

$d' \leftarrow \|\mathbf{x}' - \mathbf{x}_{eq}\|$ ▷ Distance between projected point to reference trajectory

Away $\leftarrow (d' > d + \delta)$ ▷ *Away*=true if applied force increases deviation

(5) Veto for strongly diverging forces outside the margin:

if $d > d_m$ **and** *Away* = true **then**

  $a \leftarrow -a$ ▷ Flag as harmful: forces $\lambda = 1$ below

end if

(6) Map to scaling factor $\lambda \in [0, 1]$**:**

$e \leftarrow \dfrac{|\alpha_d - \arcsin(a)|}{90^\circ}$

$\lambda_g \leftarrow \dfrac{\exp(\beta\, e)}{90}$

if $a < 0$ **then**

  $\lambda \leftarrow 1$ ▷ Maximum resistance (minimum compliance)

else

  if *Away* = false **then**

    $\lambda \leftarrow \lambda_g$ ▷ Penalize angular deviation normally

  else

    $\lambda \leftarrow 1 - \lambda_g$ ▷ Inside margin but diverging; soften penalty to allow recovery

  end if

end if

$\lambda \leftarrow \text{Clamp}(\lambda,\ 0,\ 1)$

(7) Update parameters:

$c \leftarrow c_{\text{min}} + \lambda\,(c_{\text{max}} - c_{\text{min}})$

$m \leftarrow m_{\text{min}} + \lambda\,(m_{\text{max}} - m_{\text{min}})$

return $(c,\ m)$

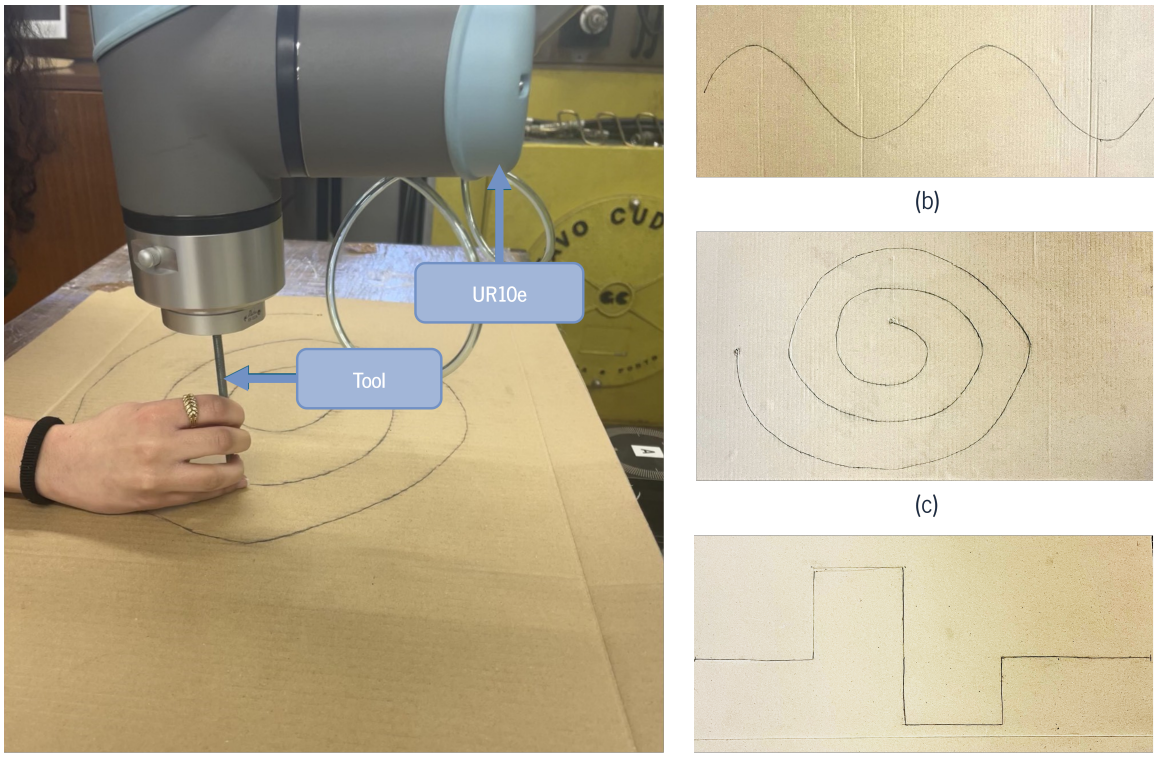


Figure 129: Experimental setup and trajectories. (a) UR10e collaborative robot with tool, (b–d) cardboard trajectories used during experiments: square wave, sine wave, and spiral.

`interface_node` publishes joint states and wrench measurements, which are used by the admittance controller node `/admittance/admittance_controller_node`. In addition, the controller receives the end-effector pose from the `/tf` tree, published on the `/ee_position` topic. Based on these inputs, the admittance controller computes joint velocity commands that are published to `/robot/joint_group_vel_controller/command` and executed by the UR ROS Driver. Services used to initialize the controllers (trigger-based) are not depicted. All code was implemented in C++.

Three bidimensional reference trajectories were used during the experiments: square wave, sine wave, and spiral (Figs. 129b-d). These were recorded using the free-drive mode by hand-guiding the robot along cardboard templates, hand drawn by the research team.

The reference trajectory is provided as a discrete set of Cartesian waypoints loaded from a text file and indexed online using a kd-tree, enabling real-time nearest-neighbor queries. At each control cycle, the controller retrieves the closest reference waypoint to the current end-effector position, which defines the instantaneous equilibrium position of the admittance controller. The local reference direction is then estimated as a tangent vector computed from neighboring waypoints around this closest point.

Controller parameters were tuned empirically. In free-drive mode, the standardization force F_{std} was

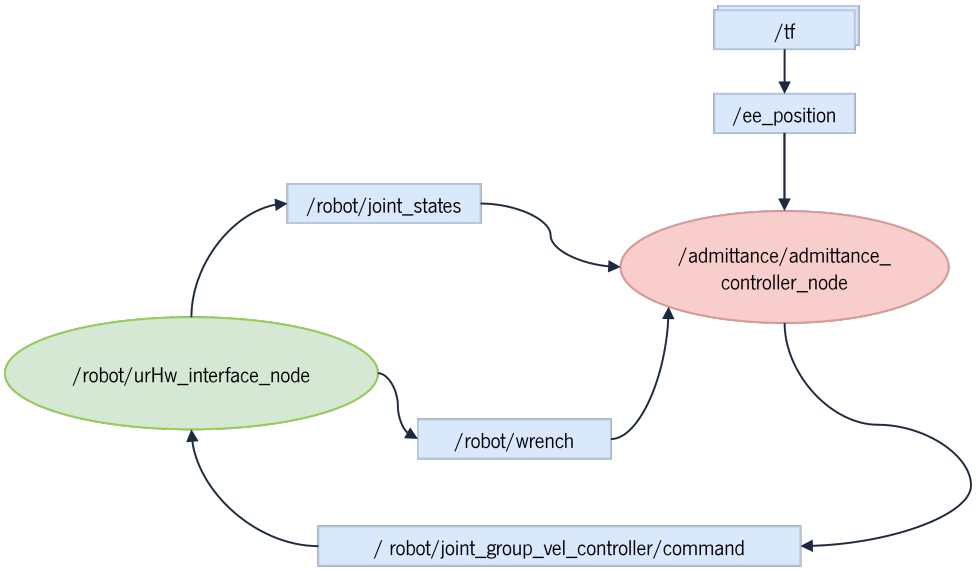


Figure 130: ROS node and topic diagram for the admittance controller architecture. Services used to initialize the controller (trigger-based) are not depicted.

set to 7.5 N. The admissible velocity range was defined between $\dot{x}_{min} = 5$ mm/s and $\dot{x}_{max} = 160$ mm/s, resulting in damping limits of $c_{min} = 70$ N·m/s and $c_{max} = 2000$ N·m/s (Eq. 8.5). The inertia parameter was fixed at $m = 10$ kg.

In guided-drive mode, the parameters varied according to the angle α between the applied force $\vec{F}_{ext}$ and the trajectory reference $\vec{T}$. When $\alpha \approx 0°$, $c = c_{min} = 70$ N·m/s and $m = m_{min} = 10$ kg. As α approached $90°$, c and m increased to $c_{max} = 2000$ N·m/s and $m_{max} = 90$ kg, respectively, following Eqs. 8.7 and 8.8. The stiffness was fixed at $k = 100$ N/m, generating a restoring force toward the reference trajectory. The parameter $\beta = 4.498$ was determined empirically, controlling how rapidly the robot increases resistance as the operator's applied force deviates from the reference trajectory direction.

All code was validated in simulation using RViz and the UR offline simulator before deployment to the physical robot to ensure stability and safe operation. Additionally, trajectories were linearized such that only force data from the F/T sensor was required, and the torque readings were ignored.

8.4.5 Experimental protocol

For a proof-of-concept, 10 participants, 3 females and 7 males with ages between 22 and 27 (23.8 ± 1.9) years old and body mass between 47 and 101 (72.4 ± 15.3) kg participated in this study. All participants were right-handed. These participants were selected based on the following inclusion criteria: healthy subjects without clinical history or evidence of motor injuries, namely shoulder or arm pain. All experiments were conducted at the School of Engineering, University of Minho, in Guimarães. All participants

Table 48: Admittance controller parameters for free-drive and guided-drive modes.

Parameter	**Free-drive mode**	**Guided-drive mode**
Reference force F_{std}	7.5 N	–
Damping c	$c_{min} = 70$ N·m/s, $c_{max} = 2000$ N·m/s	$c(\alpha) \in [70, 2000]$ N·m/s (Eq. 8.7)
Mass m	10 kg	$m(\alpha) \in [10, 90]$ kg (Eq. 8.8)
Stiffness k	0	100 N/m (fixed)
Velocity limits	$\dot{x}_{min} = 0.5$ cm/s, $\dot{x}_{max} = 16$ cm/s	–
Angle α	–	$[0°, 90°]$ (Eq. 8.6)
Margin distance d_m	–	0.08 m

provided their written and informed consent to participate in the study, according to the ethical conduct defined by the University of Minho Ethics Committee in Life and Health Sciences (CEICVS 147/2021), that follows the standard set by the declaration of Helsinki and the Oviedo Convention.

To assess the effort exerted during task execution, participants were instrumented with four Trigno Avanti sEMG sensors from Delsys, as shown in Fig. 131. Since the sEMG signal provides a quantitative measure of muscle recruitment, it serves as an effective proxy for the physical effort exerted by the participant. The sensors were positioned on the Deltoid Anterior (DELTA), Deltoid Medius (DELTM), long head of the Biceps Brachii (BIC), and lateral head of the Triceps Brachii (TRIC), following SENIAM guidelines [278]. The monitored muscles were selected because they are major contributors to the shoulder and elbow movements involved in manually guiding the cobot's end-effector. Specifically, they capture muscle activity associated with pushing, pulling, and lateral arm displacement in the sagittal and coronal planes. For normalization, the MVC of each monitored muscle was recorded prior to trial execution, also in accordance with SENIAM guidelines [278].

The experimental protocol consisted on the execution of the 3 reference trajectories (Square Wave, Sine Wave and the Spiral line) using 3 different modes: 1) manufacturer free-drive mode, 2) the implemented free-drive mode, and 3) the guided-drive mode. Each trajectory and operating mode combination was performed once by each participant, resulting in a total of nine trials per participant. These trajectories were selected to represent different path-following demands, including abrupt direction changes, smooth periodic motion, and continuous curvature. The manufacturer free-drive mode, also designated by the community as teaching mode, is a built-in passive free-drive mode available on UR robots. It allows the participant to move the robot by applying force to the robot joints. This mode served as a benchmark to evaluate the hand-guiding performance and effort required by Mode 2, the implemented free-drive mode.

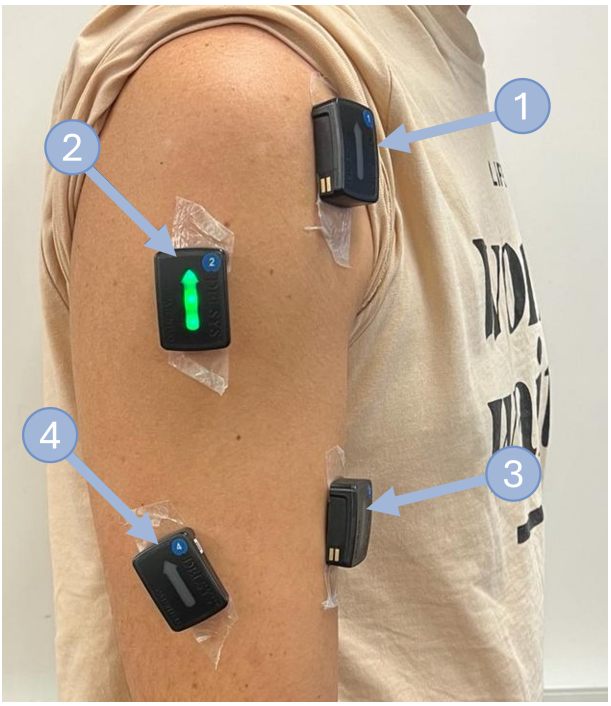


Figure 131: sEMG electrode placement on the dominant hand. (1) Deltoid Anterior, (2) Deltoid Medius, (3) Biceps Brachii (long head), and (4) Triceps Brachii (lateral head).

During the trials, participants were instructed to guide the UR10e using their dominant hand placed on the tool while following the reference trajectories (Fig. 129a). They were not informed which of the tested modes had been developed by the research team. To assess task performance and required effort, force data from the F/T sensor, end-effector position, and execution time were recorded using ROSbag. Simultaneously, sEMG data were collected with EMGworks Acquisition software (Delsys) and synchronized with the robot recordings using the Delsys hardware trigger. The experiments were conducted on a conventional table with a height of 70 cm.

After each participant completed all trials, they were asked to answer 3 anonymous questionnaires. Since the main goal of this study is validate a control strategy that guides the user during the task execution, a SUS questionnaire [279] was used to assess the usability of the guided-drive mode. The other 2 questionnaires were conducted to compare the user-experience between the free-drive mode against the manufacturer's free-drive; and to understand the impact in the user-experience of the guided-drive mode against the free-drive mode.

8.4.6 Data Analysis

The collected data was processed, and the performance metrics such as average exerted force, trajectory error, and execution time were computed using python 3.10. The average interaction force was

computed as the mean magnitude of the force vector measured by the robot F/T sensor during each trial. The sEMG data was processed using EMGAnalyze software. For each trial, the raw sEMG signal of each monitored muscle was normalized by the corresponding MVC value, allowing muscle activity to be expressed as a percentage of MVC. The RMS envelope was then computed using a 0.125 s moving window with 50% overlap, corresponding to a 0.0625 s step size. The final muscle activation value reported for each trial and muscle was obtained as the arithmetic mean of this MVC-normalized RMS envelope over the full trial duration.

8.5 Results

8.5.1 Performance Metrics

Figure 132 provides a compact visual summary of the main performance and muscle activation trends across control modes and trajectories, while the corresponding mean and standard deviation values for the ten participants are reported in Table 66 in Appendix H. Across the three trajectories, the mean exerted force (Figure 132a) ranged from 20.99 to 27.03 N in the UR10e free-drive mode, from 5.74 to 6.56 N in the implemented free-drive mode, and from 7.22 to 7.69 N in the guided-drive mode. This indicates that both implemented modes substantially reduced the physical interaction force required to execute the task across all trajectories. The guided-drive mode required higher force than the implemented free-drive mode, but remained far below the UR10e free-drive condition.

The trajectory error (Figure 132b) ranged from 5.9 to 7.2 mm in the UR10e free-drive mode, from 5.3 to 5.8 mm in the implemented free-drive mode, and from 3.9 to 4.3 mm in the guided-drive mode, confirming that guidance improved trajectory accuracy across all trajectories. Execution time (Figure 132c) ranged from 40.54 to 58.47 s in the UR10e free-drive mode, from 22.54 to 44.33 s in the implemented free-drive mode, and from 33.40 to 57.72 s in the guided-drive mode. Therefore, the implemented free-drive mode generally supported faster task execution, whereas the guided-drive mode improved trajectory accuracy but introduced additional execution time.

Regarding muscle activation, the Deltoid Medius (DELTM) and Triceps Brachii (TRIC) were consistently the most active monitored muscles (Figure 132e). The average monitored muscle activation (Figure 132d) was highest during the square-wave trajectory, particularly in the UR10e free-drive mode (10.91%), followed by the implemented free-drive mode (10.11%) and the guided-drive mode (8.64%). In specific, DELTM activation decreased from 20.08% in the UR10e free-drive mode to 18.85% in the implemented free-drive mode and 15.39% in the guided-drive mode (Table 66). For the sine-wave and spiral trajecto-

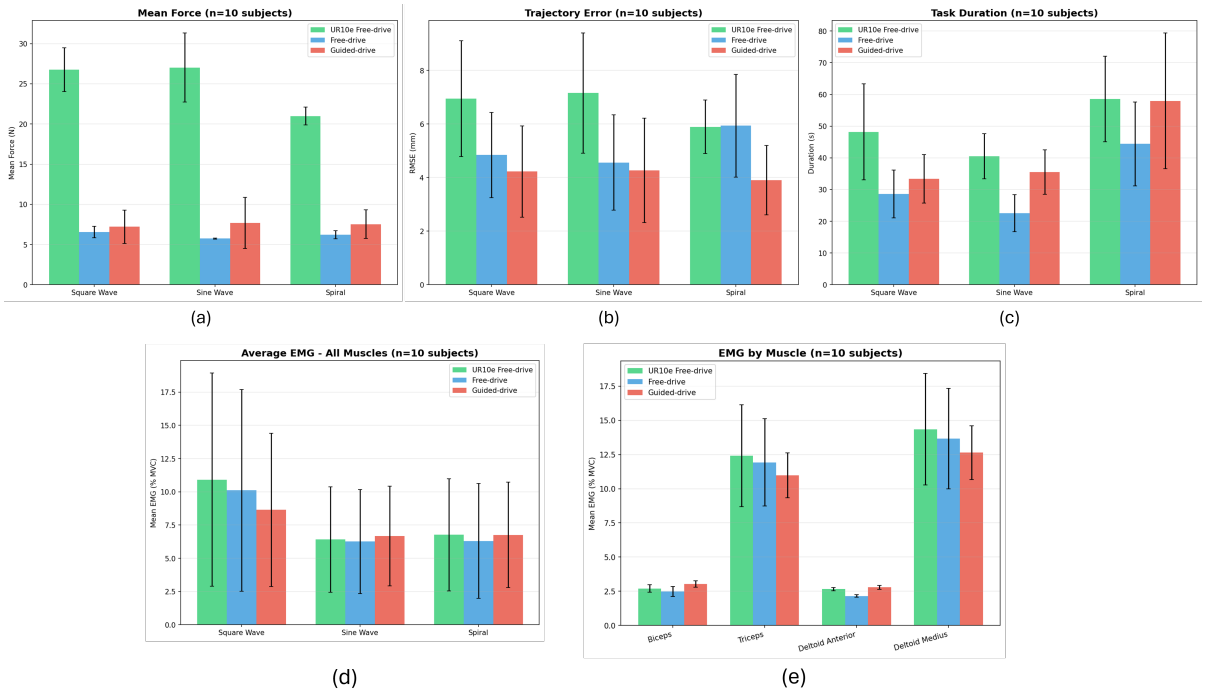


Figure 132: Performance metrics and muscle activation across control modes and trajectories ($n = 10$). (a) Mean interaction force (N). (b) Trajectory Root Mean Square Error (RMSE, mm).(c) Total task duration (s). (d) Average muscle activation across all monitored muscles (% MVC). (e) Individual muscle activation (% MVC) for the Biceps Brachii, Triceps Brachii, Deltoid Anterior, and Deltoid Medius. Error bars denote standard deviation ($\pm$ SD);

ries, differences in average monitored muscle activation across modes (Figure 132d) were smaller. The average monitored muscle activation was similar across modes for the sine-wave trajectory, with values of 6.41%, 6.26%, and 6.68% for the UR10e free-drive, implemented free-drive, and guided-drive modes, respectively. A similar pattern was observed for the spiral trajectory, with values of 6.77%, 6.31%, and 6.76%, respectively. the slight increase observed in guided-drive mode for these trajectories was primarily due to a higher activation of the Biceps Brachii and Deltoid Anterior (Figure 132e). Therefore, unlike the square-wave trajectory, the sine-wave and spiral trajectories did not show a consistent reduction in average monitored muscle activation with guidance. These results suggest that the effect of the control mode on muscle recruitment depended on the trajectory geometry.

Figure 133 illustrates the results of one representative participant performing the sine-wave trajectory in the three modes. The 2D reference trajectory and the executed trajectory are shown, together with the computed performance metrics. For this participant, average force, trajectory error, and execution time decreased progressively from the UR10e free-drive mode to the free-drive mode, and further to the guided-drive mode.

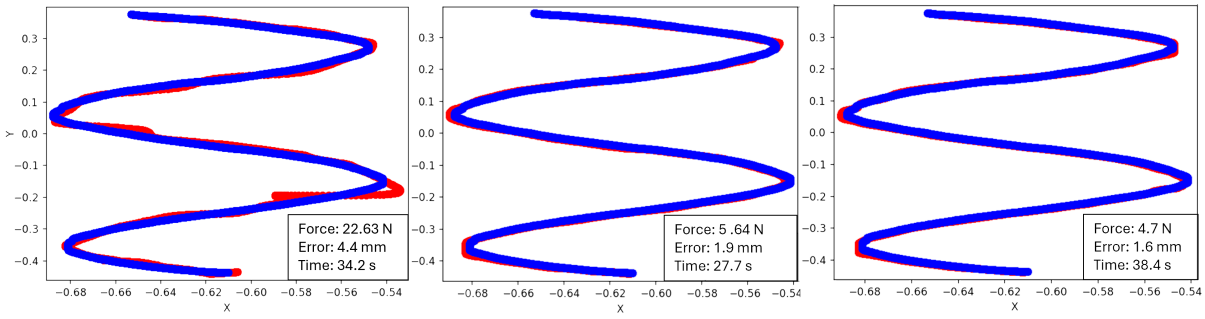


Figure 133: 2-D Cartesian position (in meters) of the sine wave reference trajectory (in blue) and the trajectory performed by a participant (in red). The corresponding mean exerted force, trajectory error, and execution time are reported for each control mode: (a) UR10e free-drive, (b) implemented free-drive, and (c) guided-drive.

8.5.2 Usability and User Experience

Table 49 summarizes the questionnaire results comparing the UR10e free-drive mode with the implemented free-drive mode. All participants indicated a preference for the implemented free-drive mode, reporting it as more comfortable, requiring less force, easier to follow the trajectory, and overall preferable for daily use.

Table 49: Participant preferences between the manufacturer's (UR10e) free-drive mode and the implemented free-drive mode.

Question	**UR10e Free-drive**	**Implemented Free-drive**
Q1. Which mode do you find more comfortable?	0	10
Q2. Which mode do you believe requires the least amount of force?	0	10
Q3. Which is the easiest mode to follow the trajectory?	0	10
Q4. Overall, which mode would you prefer to use on a daily basis?	0	10

Figure 134 presents the results comparing the guided-drive mode with the implemented free-drive mode. Participants rated the guided-drive mode positively for facilitating task execution (average score

4.5/5) and for making execution more intuitive (average score 4.2/5). Average ratings for perceived force (3.6/5) and time demand (3.4/5) were lower. Finally, the SUS questionnaire applied to the guided-drive mode yielded an average score of 91.25/100, which according to SUS guidelines corresponds to excellent usability.

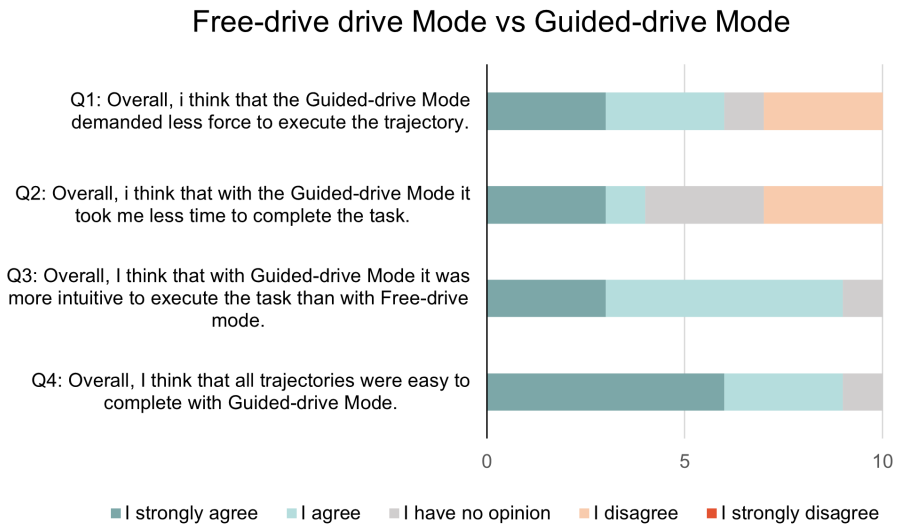


Figure 134: Participant preferences between the Free-drive Drive Mode and Guided-drive Mode

8.6 Discussion

8.6.1 Performance metrics

The reduction of exerted force from the UR10e free-drive to the free-drive mode, approximately 75% when averaged across trajectories, demonstrates the effectiveness of the proposed free-drive approach in reducing physical effort. The observed increase of 21%, when averaged across trajectories, in exerted force from the free-drive to the guided-drive mode was expected, since the guided-drive strategy increases damping and inertia when the end effector deviates from the reference trajectory, thereby resisting undesired motion. The larger inter-subject variability observed in the guided-drive mode further reflects the different ways participants reacted to these corrective forces. Participants' initial tendency to counteract the robot rather than immediately adapt likely contributed to both the higher exerted force and longer execution times. This suggests that a learning period may be required for participants to fully adapt to the guided-drive controller.

Although the guided-drive mode required more effort and generally longer execution times than the

implemented free-drive mode, it consistently reduced trajectory error compared to the other modes. This highlights the controller's potential for improving accuracy in collaborative tasks, even at the cost of speed. The single-subject example in Fig. 133 reinforces this trend, where guided assistance improved accuracy despite longer execution times.

Regarding muscle activation, the clearest reductions were observed during the square-wave trials, where both DELTM and TRIC activation decreased across modes. This may be related to the repeated changes between horizontal and vertical movement directions required by this trajectory, which likely demanded more active control of the arm than the smoother trajectories. For the sine-wave and spiral trajectories, changes in muscle activation were smaller and less consistent, probably because the smoother and more continuous movements distributed the effort more evenly across the monitored muscles. In these trajectories, the Biceps Brachii and Deltoid Anterior showed a slight increase in guided-drive mode, suggesting that participants may have used these muscles more while interacting with the corrective behavior of the controller. The discrepancy between the force measured at the end-effector and the activity of the monitored muscles also suggests that unmeasured muscles may have contributed significantly to task execution. This limitation highlights the need for broader muscle monitoring in future studies.

8.6.2 Usability and User Experience

The questionnaire results show a clear preference for the implemented free-drive mode over the manufacturer's free-drive mode, which participants consistently found more comfortable, less force-demanding, and easier to use. This confirms that introducing variable admittance significantly improves the hand-guiding experience.

For the guided-drive mode, the results suggest a trade-off between support and effort. Participants appreciated the corrective behavior, rating it as more intuitive and helpful in trajectory execution, which aligns with the observed reduction in trajectory error. However, ratings for perceived force and execution time were lower, consistent with the higher measured force and longer execution times reported earlier. This outcome reflects the expected effect of increased damping and stiffness when deviations occur, which improves accuracy but increases physical demand.

The high SUS score (91.25/100) reinforces that, despite these trade-offs, the guided-drive mode is well accepted and can be considered to have excellent usability. This indicates strong potential for training or assistance scenarios where accuracy and trajectory adherence are prioritized, even at the expense of slightly higher effort and time.

8.7 Conclusions

This chapter examined the use of a variable-admittance control strategy for task execution correction in physical human–robot interaction. The focus was on the guided-drive mode, which modulates compliance based on the relation between the user-applied force and a predefined reference trajectory, with the aim of supporting task quality while preserving intuitive interaction. Experimental results showed that this approach can reduce trajectory errors and improve perceived comfort and intuitiveness compared to less structured interaction modes, highlighting its potential as a corrective assistance mechanism rather than a strict tracking controller.

The main contribution of this chapter is the design and experimental validation of a guided-drive admittance controller with adaptive parameters for task correction. By shaping the interaction through variable damping and inertia, the controller provides corrective guidance that discourages off-trajectory motions while remaining compliant along the desired task direction. This contribution demonstrates how task quality can be influenced through interaction dynamics without enforcing rigid motion constraints.

The practical outcome of this work is a proof-of-concept task quality management module based on physical interaction. The guided-drive mode constitutes a reusable control component that can support user training and task consistency in collaborative settings, while remaining compatible with platforms that do not provide low-level force or torque control.

Future work will focus on extending this control strategy by embedding learning-based mechanisms for reference generation. In particular, Dynamic Movement Primitives will be integrated to learn and generalize reference trajectories from demonstrations, building upon the framework introduced in [280]. This extension will allow the guided-drive controller to adapt its reference trajectories to different tasks and users, strengthening its role within a broader task quality management framework. Additional future work includes refining the parameter adaptation rules and validating the approach in more complex and realistic collaborative scenarios.

Chapter 9

Conclusions

This chapter provides concluding remarks and highlights the main contributions by contextualizing the results of the previous chapters. It also outlines future directions to enhance the technological development of the Ergoaware HRC framework, as well as opportunities for further technical improvements.

9.1 Concluding Remarks

Motivated by the high impact of WRMSDs, this thesis aimed at their reduction through a broad, interdisciplinary approach that culminates in a human-aware HRC framework. The work spans wearable sensing, combining IMU-based posture monitoring with textile sEMG electrodes, perception algorithms for pose estimation and human activity recognition, ergonomic risk assessment and reporting, and the design of control strategies deployed on a UR10e collaborative robot. Together, these contributions integrate sensing, perception, and control into a cohesive system that supports safer, ergonomically informed human–robot collaboration.

Chapter 3 presented the development and technical validation of two wearable sensing technologies that can be customized and integrated into various research environments without the constraints that come with proprietary commercial solutions. The first is Ergowear, a smart garment instrumented with nine IMUs and six haptic actuators, designed to monitor upper-body motion and enable the implementation of biofeedback strategies. The second is an e-textile-based sEMG sensing system, for which textile electrodes were thoroughly characterized, and two electrode sizes and conductive fiber types were benchmarked. The results demonstrate that the proposed sensing platforms fulfill Objective 1 and KPIs 1. In particular, the Ergowear system achieved reliable wireless communication, with a measured packet loss rate of 0% during extended recording sessions, and exhibited stable mechanical integration, with no observed cable failures or sensor detachment. Similarly, the e-textile-based sEMG system attained signal quality comparable to a commercial reference system, achieving a root mean square error below 10% when benchmarked against

Delsys electrodes. These outcomes confirm the suitability of both systems as reliable hardware foundations for garment-integrated motion and muscle activity monitoring.

Chapter 4 focused on the development of computational tools for ergonomic assessment. It began with a systematic investigation of data-driven inertial pose estimation, examining how far learning-based pipelines can replace classical estimation strategies without relying on explicit biomechanical models. This allowed to fulfill objective 2 and KPI 2, as the hybrid models were able to achieve comparable performance to classical sensor fusion filters on the high quality data and inferences times below 2ms.

This was followed by a comprehensive benchmarking analysis of deep learning architectures for human activity and posture recognition, evaluating convolutional, recurrent, transformer-based, and hybrid models under a unified experimental protocol. Finally, the chapter introduced the design and implementation of a graphical user interface whose backend integrates posture recognition and ergonomic risk metrics into an expedited and holistic assessment pipeline, enabling real-time, interpretable evaluation of postural risk. These outcomes fulfilled Objective 3 and KPI 3, as the embedded HAR model met the defined performance targets, achieving 95.29% test accuracy and an inference time of 0.29 ms, thereby providing trustworthy accuracy and the necessary inference time for real-time postural assessment. The ErgoReport GUI also fulfilled the qualitative GUI component of the KPI by providing an expedited and interpretable postural risk assessment tool, supported by positive ergonomist feedback.

Chapter 5 introduced the ErgoAware HRC framework, describing its overall architecture and design principles. The chapter illustrated how the ROS-based framework integrates wearable sensing technologies, a simulation environment for testing and training algorithms, human-aware controllers, and ergonomic assessment of the human operator within a unified and extensible system.

Chapter 6 investigated the estimation and mitigation of muscle fatigue in dynamic human–robot collaboration using surface electromyography and learning-based models. The chapter focused on subject- and muscle-independent fatigue estimation and its integration into a fatigue-aware collaborative control strategy capable of adapting workstation configurations in response to predicted fatigue states. The best-performing attention-based recurrent model achieved 79.0% test accuracy, fulfilling the predefined fatigue classification KPI. However, because this model required full-trial input, the control used a reduced-context model, which achieved 62.9% accuracy. This lower performance was acceptable for proof-of-concept validation because the model showed a conservative error pattern, with fatigue-related states more often overestimated than missed, supporting a preventive control rule triggered at the Transition-to-Fatigue state. The resulting adaptation generally reduced average muscle activation, with reductions in four out of five participants, and promoted antagonist muscle alternation after fatigue detection. However, the predefined

fatigue exposure reduction threshold was not consistently reached, since only one participant exceeded the 30% reduction target. Questionnaire results further showed reduced perceived fatigue, appropriate perceived timing of intervention, and preference for the assisted condition. Overall, Objective 4 was achieved as a proof-of-concept: the fatigue classification KPI was fulfilled, whereas the fatigue exposure reduction KPI was partially fulfilled.

Chapter 7 introduced a RL-based control strategy that enabled a collaborative robot to personalize assistance according to individual ergonomics and physical constraints during a shared object transportation task. The controller was trained using a two phase approach combining simulation-based pre-training with real world fine-tuning, allowing efficient learning while safely closing the simulation-to-real gap. The results showed that the robot adapted its behavior to user-specific needs, maintained the final average ergonomic risk below the predefined threshold of 2.5, eliminated pain risk according to the adopted machine-state-based proxy, and achieved efficient task execution. Among the evaluated approaches, Deep Q-Networks provided the best real-world performance, completing the task with fewer steps and shorter execution times. These results fulfilled Objective 5 and KPI 5.

Chapter 8 presented and evaluated a variable-admittance control strategy for task quality management in physical human–robot interaction, with a focus on supporting novice users during task execution. The proposed guided-drive mode modulates damping and inertia based on the relationship between user-applied forces and a predefined reference trajectory, providing corrective assistance without enforcing rigid motion constraints. Validation results showed that trajectory errors remained below 5 mm and interaction forces below 8 N, while preserving intuitive and comfortable interaction (SUS score of 91.25/100), thus achieving Objective 6 and KPI 6. These results show that the strategy can improve task execution quality while preserving intuitive and comfortable interaction.

9.2 Research Questions

- **RQ 1: How effective are commercial knitting threads for sEMG signal acquisition and electrical transmission, and to what extent can a textile-based sEMG electrode reliably acquire electromyographic signals?**

  This research question was addressed in Section 3.3 through a systematic evaluation of textile-based conductive tracks and electrodes manufactured using commercial knitting threads from two suppliers, Shieldex and Imbut (Elitex SC), adressing objective 1.

  Electrical and mechanical characterization revealed marked differences between the fibers. Con-

ductive tracks produced with Shieldex fibers exhibited large resistance fluctuations during tensile and abrasion testing, with several samples showing near-complete loss of conductivity due to degradation of the metallic coating. In contrast, tracks produced with Elitex SC fibers demonstrated substantially smaller resistance variations under the same tests, indicating superior mechanical robustness and coating stability. Similar trends were observed in the electrode abrasion tests, where Shieldex-based electrodes showed resistance increases of several orders of magnitude, while Elitex SC electrodes remained below 5 Ω despite repeated abrasion cycles.

It is important to note that all mechanical tests were deliberately designed as extreme, accelerated degradation scenarios, exceeding the mechanical stresses expected during on-body use. In practical wearable conditions, textile electrodes are subjected to significantly lower abrasion forces and strain, as they are in continuous contact with skin rather than abrasive surfaces. As a result, the reported resistance increases reflect worst-case degradation rather than typical operating conditions.

Regarding signal acquisition, both textile electrode geometries evaluated (4×2 cm and 2×1 cm) were capable of acquiring sEMG signals with good fidelity. The larger 4×2 cm electrodes achieved an average RMSE of 9.344%, comparable to commercial Ag/AgCl electrodes (9.304%), while the smaller electrodes exhibited higher error (16.602%). These results confirm the influence of electrode surface area on signal quality, with larger contact areas yielding improved acquisition performance.

- **RQ 2: To what extent can data driven inertial sensing pipelines replace explicit biomechanical modeling for pose estimation, when compared across sensor quality and modeling strategies?**

  The results presented in this chapter indicate that data-driven inertial sensing pipelines have potential to partially replace explicit biomechanical modeling for pose estimation, with effectiveness strongly dependent on sensor quality and on whether model-based priors are retained within the learning architecture.

  Purely data-driven approaches, in which neural networks directly map raw MARG signals to segment orientations without biomechanical constraints, were able to achieve reasonable pose estimation QADs. However, these models did not consistently outperform classical sensor fusion methods, even when high-quality inertial data were available. Their performance was more competitive in low-cost sensing scenarios, suggesting that data-driven models can compensate, to some extent, for hardware limitations and imperfect sensor attachment. Nevertheless, error levels remained substantial high for joints with high kinematic redundancy, such as the upper limbs, indicating that

the absence of explicit biomechanical structure limits generalization and stability.

Hybrid approaches that combine neural networks with classical sensor fusion filters demonstrated the most robust performance across both low-cost and high-end sensor configurations. In particular, architectures that preserved the structured integration and correction mechanisms of classical filters, such as complementary and Madgwick filters, while learning data-driven residual corrections achieved orientation errors comparable to state-of-the-art classical methods. These results indicate that retaining algorithmic structure and kinematic consistency at the sensor fusion level, rather than relying on fully end-to-end learning, is essential for drift mitigation, temporal stability, and reliable full-body inertial pose estimation.

- **RQ 3: How can postural ergonomic risk be continuously and objectively quantified and attributed over time in a manner that is interpretable and usable for end users, and how do architectural design choices and spatio-temporal information from inertial sensors influence the performance and interpretability of the posture recognition models that enable this assessment?**

  The results show that continuous ergonomic risk quantification can be achieved by combining posture recognition with joint angle–based ergonomic scoring and cumulative exposure modeling within a unified assessment framework, adressing objective 3. The proposed joint kinematic wear index integrates instantaneous ergonomic scores over time and explicitly models both wear and recovery phases. Unlike snapshot-based ergonomic methods, this approach accounts for posture duration, repetition, and recovery dynamics. The results demonstrate that significant joint stress can accumulate even during short task executions, particularly for back and shoulder flexion, and that higher ergonomic risk levels lead to faster stress accumulation, confirming that cumulative exposure cannot be inferred from isolated posture evaluations. Attribution of ergonomic risk to specific working postures is enabled by synchronizing ergonomic scores with posture labels produced by the posture recognition module. This allows posture-resolved risk profiles to be computed and makes it possible to identify which postures contribute most to cumulative joint stress. Across tasks, stooping consistently emerged as the posture associated with the highest risk, while squatting and kneeling were associated with lower risk levels, supporting their consideration as safer alternatives depending on task constraints. Differences observed between ergonomic scoring methods further highlight the importance of posture-level attribution for meaningful interpretation.

  The reliability and interpretability of this ergonomic assessment depend on the performance of

the underlying posture recognition models. The results show that architectural design choices and the exploitation of spatio-temporal information from inertial sensors play a decisive role in enabling robust posture recognition. Hybrid architectures that explicitly separate spatial and temporal processing and fuse their outputs consistently outperformed standalone models. CNNs effectively captured discriminative spatial features from inertial signals, while temporal models such as LSTMs and Transformers proved most effective when operating on features extracted by CNNs, capturing posture dynamics and improving robustness to transitions. Learnable fusion mechanisms further improved recognition by allowing the models to adaptively weight spatial and temporal contributions on a per-posture basis, supporting both high recognition performance and stable real-time inference. Temporal resolution was also found to be a core design parameter, with balanced temporal windows providing the best trade-off between responsiveness and stability.

Usability for end-users is addressed through the ErgoReport, which translates continuous, posture-resolved ergonomic data into interpretable visual summaries. Feedback from ergonomists indicated that posture-based aggregations and cumulative risk visualizations support rapid interpretation and decision making, demonstrating that complex spatio-temporal ergonomic information can be presented in a form suitable for practical use.

- **RQ 4: To what extent can a deep learning model accurately estimate muscle fatigue in a subject independent manner and inform a control strategy to effectively mitigate fatigue progression and enhance task endurance?**

  This RQ is related to objective 4. The results demonstrate that a deep learning model can estimate muscle fatigue in a subject-independent manner with moderate but meaningful accuracy, sufficient to inform a fatigue-aware control strategy, though not yet robust enough for precise real-time fatigue quantification. Under subject- and muscle-independent evaluation, attention-based recurrent architectures achieved weighted $F1$-scores below 0.80, reflecting the inherent difficulty of fatigue classification from a single sEMG channel in dynamic tasks. Misclassifications were predominantly confined to adjacent fatigue states, while severe confusions between non-fatigued and fatigued conditions were rare, indicating that the models captured the ordinal progression of fatigue rather than arbitrary class boundaries.

  Despite these limitations, the estimated fatigue states proved effective for control purposes. When integrated into a collaborative control strategy, fatigue predictions triggered adaptive workstation reconfigurations that redistributed muscular load between antagonist muscles. This resulted in

lower average muscle activation and extended task execution compared to a non-adaptive condition. Average muscle activation decreased in four out of five participants, while one participant showed increased activation with HRC. Subjective evaluations nevertheless confirmed reduced perceived fatigue, appropriate timing of intervention, and strong user preference for the fatigue-aware strategy.

- **RQ 5: Can collaborative robots learn individual ergonomics and physical constraints to provide personalized, effective assistance during shared tasks?**

  This RQ is related to objective 5. This research question was addressed by demonstrating that a collaborative robot can learn to adapt its assistance to individual ergonomic characteristics and physical constraints through a reinforcement learning based control strategy. The proposed controller was able to personalize its behavior during a shared object transportation task, leading to safe, ergonomic, and pain free collaboration.

  A key element enabling this capability was the adopted training methodology, which combined simulation based pre training with real world fine tuning. Pre training in simulation allowed the controller to acquire the core task structure and assistance behavior without exposing the human collaborator to unsafe or inefficient exploration. Real world fine tuning was then required to adapt the learned policies to individual users and to compensate for modeling simplifications, particularly those contributing to the simulation to real gap.

  Human posture response was approximated during simulation using inverse kinematics, enabling the computation of feasible human postures in response to object motion. This approximation supported pre training by providing consistent posture solutions, while real world fine tuning accounted for discrepancies between simulated and actual human behavior.

  Personalization was further supported by the design of the action space and reward function. Action space shaping was used to exclude actions that would place the object outside the human collaborator's reachable workspace or that did not contribute to task progression, ensuring safe and efficient interaction. The reward function explicitly penalized unsafe ergonomics, transport inefficiency, and violations of physical constraints. Given the critical role of pain avoidance, a weighted formulation was adopted to prioritize penalties associated with exceeding physical limitations.

  The experimental results showed that this approach enabled the controller to tailor its actions to human specific needs, maintaining low ergonomic risk, eliminating pain risk according to the adopted proxy, and achieving efficient task execution. These findings indicate that collaborative robots can learn individualized ergonomic and physical constraints and use this knowledge to provide effective

and personalized assistance during shared tasks.

- **RQ 6: How can an admittance-based control strategy be designed to support novice training and task quality management, and how do users interact with and experience this control approach?**

  This RQ is related to objective 6. An admittance-based control strategy can support novice training and task quality management by shaping the physical interaction between the user and the robot through adaptive compliance rather than strict trajectory enforcement. In the proposed approach, the robot does not impose motion but instead modulates damping and inertia based on the alignment between the user-applied force and a task reference trajectory, as well as the current deviation from that reference. This allows the robot to remain highly compliant when the user acts consistently with the task, while progressively increasing resistance when motions deviate from the desired execution, thereby providing continuous, physically grounded feedback.

  Task quality management is achieved by embedding corrective behavior directly into the interaction dynamics. Off-trajectory motions are made increasingly effortful, while motions that contribute to task completion are facilitated. Importantly, the controller distinguishes between deviations that should be corrected and recovery motions that help return to the reference, allowing higher compliance when the user is outside the desired path but acting in a corrective manner. This results in a guidance mechanism that supports learning by doing, where users are free to explore the task while being subtly steered toward correct execution.

  From the user perspective, this control strategy is experienced as intuitive and supportive rather than restrictive. Experimental results indicate that users interacting with the guided-drive mode achieve lower trajectory errors while perceiving the interaction as more comfortable and easier to understand compared to less structured modes. The robot is perceived as a “guiding partner” that assists task execution through physical cues, making the control approach particularly suitable for novice training scenarios where continuous feedback and error awareness are critical.

  Overall, the results show that adaptive admittance control provides an effective means of balancing guidance and autonomy in physical human–robot interaction, enabling task quality improvement while maintaining a positive user experience.

9.3 Prospect for future work

As a multidisciplinary research effort, this work naturally opens several complementary directions for future development, spanning sensing technologies, computational methods, control frameworks, and validation in real-world contexts.

From the perspective of sensing systems, future work should focus on the miniaturization of both IMU and sEMG hardware and on the design and development of an integrated wearable platform that combines these sensing modalities in a modular and efficient manner. In parallel, biofeedback strategies should be developed and embedded directly into the sensing system, enabling real-time feedback to the user based on physiological and kinematic measurements.

Regarding human pose estimation, other architectures should be explored, such as transformers and physics-based modeling approaches and improved sensor-to-segment calibration algorithms. These advances would enhance robustness and generalization across users, sensor placements, and movement conditions, particularly in unconstrained environments.

For human activity recognition and the ErgoReport framework, future work should prioritize field validation with a larger and more diverse population and gather more end-user feedback to iterate the tool. Additionally, biofeedback strategies derived from posture recognition and ergonomic assessment outputs should be developed and integrated with the wearable system, closing the loop between assessment and intervention.

At the system level, the underlying software framework should be migrated from ROS to ROS 2, given that ROS Noetic reached its end of life in May 2025. This transition is necessary to ensure long-term maintainability, real-time performance improvements, and compatibility with modern robotic platforms.

With respect to muscle fatigue estimation, future work should focus on improving robustness and temporal resolution by enriching the input representation beyond a single sEMG channel. The inclusion of synergistic muscles and the fusion of sEMG with kinematic information are expected to help disentangle fatigue-related changes from movement-induced variability and improve subject-independent generalization. In parallel, lightweight temporal models and teacher–student approaches should be explored to enable online fatigue estimation while retaining the benefits of long-term temporal context observed in offline settings. To address limited labeled data availability, generative approaches such as adversarial data synthesis and biomechanical surrogates based on OpenSim models could be leveraged to produce physiologically plausible fatigue trajectories and scale model training.

Concerning physical limitations and user-specific constraints, the development of a digital pain biomarker

was envisioned from the start and is currently being addressed by the research team. Future efforts should focus on integrating this biomarker into the control framework and validating the complete controller in the presence of physical limitations. Extending the experimental protocol to include workers with musculoskeletal disorders is a critical step toward assessing the inclusiveness and clinical relevance of the proposed approach.

Finally, with respect to task quality management, future work should explore the integration of DMPs with the admittance controller to leverage their temporal and spatial adaptability. In this context, DMPs could be used to learn and generalize task reference trajectories, enabling the control strategy to adapt across tasks, users, and execution styles while preserving the corrective and supportive behaviors demonstrated in this work. Also, the controller should be tested in a real industrial task and with the target population.

Bibliography

[1] Xsens, "Understanding strapdown integration," 2021, https://xsenstechnologies.force.com/knowledgebase/s/article/Understanding-Strapdown-Integration?language=en_US. Accessed: 2022-01-05.

[2] L. Peternel, C. Fang, N. Tsagarakis, and A. Ajoudani, "A selective muscle fatigue management approach to ergonomic human-robot co-manipulation," *Robotics and Computer-Integrated Manufacturing*, vol. 58, pp. 69–79, 2019.

[3] European Agency for Safety and Health at Work, "Return to work after msd-related sick leave in the context of psychosocial risks at work," European Agency for Safety and Health at Work, Tech. Rep., 2019. [Online]. Available: https://osha.europa.eu/en/publications/return-work-after-msd-related-sick-leave-context-psychosocial-risks-work

[4] EU-OSHA, "OSH in figures: Work-related musculoskeletal disorders in the EU: Facts and figures," European Agency for Safety and Health at Work, Bilbao, Spain, Technical Report, 2010, european Agency for Safety and Health at Work. Accessed: 2025-05-13. [Online]. Available: https://osha.europa.eu/sites/default/files/Chronic_MSDs_Report.pdf

[5] S. Bevan, "Economic impact of musculoskeletal disorders (msds) on work in europe," *Best Practice & Research Clinical Rheumatology*, vol. 29, no. 3, pp. 356–373, 2015.

[6] Health and Safety Executive, "Health and safety at work: Summary statistics for great britain 2018," Health and Safety Executive, United Kingdom, Tech. Rep., 2018, accessed: 2022-10-13. [Online]. Available: https://www.hse.gov.uk/statistics/overall/hssh1819.pdf

[7] European Agency for Safety and Health at Work, "Research on work-related low back disorders," European Agency for Safety and Health at Work, Belgium, Tech. Rep., Dec. 2000. [Online]. Available: https://osha.europa.eu/sites/default/files/TE3200273ENC_-_Work-related_Low_Back_Disorders.pdf

[8] ——, "Work-related musculoskeletal disorders: Prevalence, costs and demographics in the eu," European Agency for Safety and Health at Work, Luxembourg, Tech. Rep., 2019, accessed: 2025-05-13. [Online]. Available: https://osha.europa.eu/en/publications/msds-facts-and-figures-overview-prevalence-costs-and-demographics-msds-europe

[9] L. Lu, F. M. Megahed, and L. A. Cavuoto, "Interventions to mitigate fatigue induced by physical work: A systematic review of research quality and levels of evidence for intervention efficacy," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 63, no. 1, pp. 151–191, February 2021.

[10] European Commission, "The 2024 ageing report: Economic and budgetary projections for the eu member states (2022–2070)," 2024, accessed May 13, 2025. [Online]. Available: https://economy-finance.ec.europa.eu/document/download/971dd209-41c2-425d-94f8-e3c3c3459af9_en?filename=ip279_en.pdf

[11] L. Bernfort, J. Persson, C. Linderoth, and K. Ekberg, "Supervisor ratings of productivity loss associated with presenteeism and sick leave due to musculoskeletal disorders and common mental disorders in sweden," *Work*, vol. 68, pp. 1091–1100, 2021.

[12] V. Garrow, "Presenteeism: A review of current thinking," Institute for Employment Studies, Tech. Rep., 2016. [Online]. Available: https://www.employment-studies.co.uk/resource/presenteeism-review-current-thinking

[13] Q. Shi and J. C. MacDermid, "Disability and return to work following musculoskeletal injury," PhD Dissertation, Graduate Program in Health and Rehabilitation Sciences, Western University, 2016, accessed: 2024-12-26. [Online]. Available: https://ir.lib.uwo.ca/etd/4013/

[14] S. Nahavandi, "Industry 5.0—a human-centric solution," *Sustainability*, vol. 11, no. 16, p. 4371, 2019.

[15] S. Kana, S. Lakshminarayanan, D. M. Mohan, and D. Campolo, "Impedance controlled human–robot collaborative tooling for edge chamfering and polishing applications," *Robotics and Computer-Integrated Manufacturing*, vol. 72, p. 102199, 2021.

[16] L. F. C. Figueredo, R. C. Aguiar, L. Chen, S. Chakrabarty, M. R. Dogar, and A. G. Cohn, "Human comfortability: Integrating ergonomics and muscular-informed metrics for manipulability analysis

during human-robot collaboration," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 351–358, April 2021.

[17] L. Peternel, N. Tsagarakis, D. Caldwell, and A. Ajoudani, "Robot adaptation to human physical fatigue in human–robot co-manipulation," *Autonomous Robots*, vol. 42, no. 5, pp. 1011–1021, 2018.

[18] B. Busch, G. Maeda, Y. Mollard, M. Demangeat, and M. Lopes, "Postural optimization for an ergonomic human-robot interaction," in *IEEE International Conference on Intelligent Robots and Systems*, vol. 2017-Septe, 2017, pp. 2778–2785.

[19] W. Kim, J. Lee, L. Peternel, N. Tsagarakis, and A. Ajoudani, "Anticipatory robot assistance for the prevention of human static joint overloading in human–robot collaboration," *IEEE robotics and automation letters*, vol. 3, no. 1, pp. 68–75, 2017.

[20] N. Vignais, M. Miezal, G. Bleser, K. Mura, D. Gorecky, and F. Marin, "Innovative system for real-time ergonomic feedback in industrial manufacturing," *Applied ergonomics*, vol. 44, no. 4, pp. 566–574, 2013.

[21] N. Carbonaro, G. Mascherini, I. Bartolini, M. N. Ringressi, A. Taddei, A. Tognetti, and N. Vanello, "A wearable sensor-based platform for surgeon posture monitoring: a tool to prevent musculoskeletal disorders," *International Journal of Environmental Research and Public Health*, vol. 18, no. 7, p. 3734, 2021.

[22] L. van der Spaa, M. Gienger, T. Bates, and J. Kober, "Predicting and optimizing ergonomics in physical human-robot cooperation tasks," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 1799–1805.

[23] E. Merlo, E. Lamon, F. Fusaro, M. Lorenzini, A. Carfi, F. Mastrogiovanni, and A. Ajoudani, "An ergonomic role allocation framework for dynamic human–robot collaborative tasks," *Journal of Manufacturing Systems*, vol. 67, pp. 111–121, 2023.

[24] Xsens, "Mvn user manual," 2021, https://www.xsens.com/hubfs/Downloads/usermanual/MVN_User_Manual.pdf. Accessed: 2021-11-02.

[25] L. Ma, D. Chablat, F. Bennis, and W. Zhang, "A new simple dynamic muscle fatigue model and its validation," *International journal of industrial ergonomics*, vol. 39, no. 1, pp. 211–220, 2009.

[26] G. C. David, “Ergonomic methods for assessing exposure to risk factors for work-related musculoskeletal disorders,” *Occupational medicine*, vol. 55, no. 3, pp. 190–199, 2005.

[27] A. Ebied, A. M. Awadallah, M. A. Abbass, and Y. El-Sharkawy, “Upper limb muscle fatigue analysis using multi-channel surface emg,” in *2020 2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES)*. IEEE, 2020, pp. 423–427.

[28] P. Karthick, S. Ramakrishnan *et al.*, “Automated detection of muscle fatigue conditions from cyclostationary based geometric features of surface electromyography signals,” *Computer Methods in Biomechanics and Biomedical Engineering*, 2021.

[29] Y. Rong, D. Hao, X. Han, Y. Zhang, J. Zhang, and Y. Zeng, “Classification of surface emg using wavelet packet energy analysis and a genetic algorithm-based support vector machine,” 2013.

[30] N. Meziane, J. G. Webster, M. Attari, and A. J. Nimunkar, “Dry electrodes for electrocardiography,” *Physiological Measurement*, vol. 34, 2013.

[31] L. Peternel, N. Tsagarakis, D. Caldwell, and A. Ajoudani, “Adaptation of robot physical behaviour to human fatigue in human-robot co-manipulation,” in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2016, pp. 489–494.

[32] L. Peternel, C. Fang, N. Tsagarakis, and A. Ajoudani, “Online human muscle force estimation for fatigue management in human-robot co-manipulation,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1340–1346.

[33] Z. Feng, Y. Wang, and L. Liu, “Muscle fatigue detection method based on feature extraction and deep learning,” in *2022 7th International Conference on Intelligent Computing and Signal Processing (ICSP)*. IEEE, 2022, pp. 97–100.

[34] D. Terracina, A. Moniri, J. Rodriguez-Manzano, P. H. Strutton, and P. Georgiou, “Real-time forecasting and classification of trunk muscle fatigue using surface electromyography,” in *2019 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. IEEE, 2019, pp. 1–4.

[35] S. M. Cerqueira, A. F. Da Silva, and C. P. Santos, “Smart vest for real-time postural biofeedback and ergonomic risk assessment,” *IEEE Access*, vol. 8, pp. 107 583–107 592, 2020.

[36] E. Valero, A. Sivanathan, F. Bosché, and M. Abdel-Wahab, “Analysis of construction trade worker body motions using a wearable and wireless motion sensor network,” *Automation in Construction*, vol. 83, pp. 48–55, 2017.

[37] X. Yan, H. Li, A. R. Li, and H. Zhang, "Wearable imu-based real-time motion warning system for construction workers' musculoskeletal disorders prevention," *Automation in construction*, vol. 74, pp. 2–11, 2017.

[38] European Agency for Safety and Health at Work, "Return to Work After MSD-Related Sick Leave in the Context of Psychosocial Risks at Work," -, 2019, accessed: 2024-12-26. [Online]. Available: https://osha.europa.eu/en/publications/return-work-after-msd-related-sick-leave-context-psychosocial-risks-work

[39] A. Bestick, R. Pandya, R. Bajcsy, and A. D. Dragan, "Learning human ergonomic preferences for handovers," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 3257–3264.

[40] G. Dulac-Arnold, N. Levine, D. J. Mankowitz *et al.*, "Challenges of real-world reinforcement learning: definitions, benchmarks and analysis," *Machine Learning*, vol. 110, pp. 2419–2468, 2021.

[41] P. K. R. Maddikunta, Q.-V. Pham, B. Prabadevi, N. Deepa, K. Dev, T. R. Gadekallu, R. Ruby, and M. Liyanage, "Industry 5.0: A survey on enabling technologies and potential applications," *Journal of Industrial Information Integration*, vol. 26, p. 100257, 2022.

[42] W. Wang, R. Li, Y. Chen, Z. M. Diekel, and Y. Jia, "Facilitating human–robot collaborative tasks by teaching-learning-collaboration from human demonstrations," *IEEE Transactions on Automation Science and Engineering*, vol. 16, no. 2, pp. 640–653, 2018.

[43] S. El Zaatari, M. Marei, W. Li, and Z. Usman, "Cobot programming for collaborative industrial tasks: An overview," *Robotics and Autonomous Systems*, vol. 116, pp. 162–180, 2019.

[44] D. Martins, S. M. Cerqueira, T. P. Vieira, A. Pombeiro, and C. P. Santos, "Towards a user-specific ergonomics-based approach for an activity assessment tool," in *2023 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*. IEEE, 2023, pp. 220–225.

[45] C. Huang, W. Kim, Y. Zhang, and S. Xiong, "Development and validation of a wearable inertial sensors-based automated system for assessing work-related musculoskeletal disorders in the workspace," *International Journal of Environmental Research and Public Health*, vol. 17, no. 17, p. 6050, 2020.

[46] K. B. Martinez, M. Nazarahari, and H. Rouhani, "K-score: A novel scoring system to quantify fatigue-related ergonomic risk based on joint angle measurements via wearable inertial measurement units," *Applied Ergonomics*, vol. 102, p. 103757, 2022.

[47] J. Zhao, E. Obonyo, and S. G. Bilén, "Wearable inertial measurement unit sensing system for musculoskeletal disorders prevention in construction," *Sensors*, vol. 21, p. 1324, 2 2021.

[48] J. Zhao and E. Obonyo, "Applying incremental deep neural networks-based posture recognition model for ergonomics risk assessment in construction," *Advanced Engineering Informatics*, vol. 50, p. 101374, 2021.

[49] J. De Kok, P. Vroonhof, J. Snijders, G. Roullis, M. Clarke, K. Peereboom, P. van Dorst, and I. Isusi, "Work-related musculoskeletal disorders: prevalence, costs and demographics in the eu," *European Agency for Safety and Health at Work*, 2019.

[50] C. M. Lind, J. A. Diaz-Olivares, K. Lindecrantz, and J. Eklund, "A wearable sensor system for physical ergonomics interventions using haptic feedback," *Sensors*, vol. 20, no. 21, p. 6010, 2020.

[51] C. M. Lind, F. Abtahi, and M. Forsman, "Wearable motion capture devices for the prevention of work-related musculoskeletal disorders in ergonomics—an overview of current applications, challenges, and future opportunities," *Sensors*, vol. 23, no. 9, p. 4259, 2023.

[52] D. Wang, F. Dai, and X. Ning, "Risk assessment of work-related musculoskeletal disorders in construction: State-of-the-art review," *Journal of Construction Engineering and management*, vol. 141, no. 6, p. 04015008, 2015.

[53] C. M. Lind, B. De Clercq, M. Forsman, A. Grootaers, M. Verbrugghe, L. Van Dyck, and L. Yang, "Effectiveness and usability of real-time vibrotactile feedback training to reduce postural exposure in real manual sorting work," *Ergonomics*, vol. 66, no. 2, pp. 198–216, 2023.

[54] M. Sun and J. Hill, "A method for measuring mechanical work and work efficiency during human activities," *Journal of biomechanics*, vol. 26, no. 3, pp. 229–241, 1993.

[55] R. Bootsman, P. Markopoulos, Q. Qi, Q. Wang, and A. A. Timmermans, "Wearable technology for posture monitoring at the workplace," *International Journal of Human-Computer Studies*, vol. 132, pp. 99–111, 2019.

[56] R. Lee, C. James, S. Edwards, G. Skinner, J. L. Young, and S. J. Snodgrass, "Evidence for the effectiveness of feedback from wearable inertial sensors during work-related activities: A scoping review," *Sensors*, vol. 21, no. 19, p. 6377, 2021.

[57] W. Kim, V. R. Garate, J. M. Gandarias, M. Lorenzini, and A. Ajoudani, "A directional vibrotactile feedback interface for ergonomic postural adjustment," *IEEE Transactions on Haptics*, vol. 15, no. 1, pp. 200–211, 2021.

[58] M. Lorenzini, S. Ciotti, J. M. Gandarias, S. Fani, M. Bianchi, and A. Ajoudani, "Performance analysis of vibrotactile and slide-and-squeeze haptic feedback devices for limbs postural adjustment," in *2022 31st IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2022, pp. 707–713.

[59] S. Fani, S. Ciotti, and M. Bianchi, "Multi-cue haptic guidance through wearables for enhancing human ergonomics," *IEEE Transactions on Haptics*, vol. 15, no. 1, pp. 115–120, 2021.

[60] M. Aggravi, F. Pausé, P. R. Giordano, and C. Pacchierotti, "Design and evaluation of a wearable haptic device for skin stretch, pressure, and vibrotactile stimuli," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2166–2173, 2018.

[61] Y. Chen, M. Guo, and Z. Wang, "An improved algorithm for human activity recognition using wearable sensors," in *2016 Eighth International Conference on Advanced Computational Intelligence (ICACI)*. IEEE, 2016, pp. 248–252.

[62] F. Kuncan, Y. Kaya, Z. Yiner, and M. Kaya, "A new approach for physical human activity recognition from sensor signals based on motif patterns and long-short term memory," *Biomedical Signal Processing and Control*, vol. 78, p. 103963, 2022.

[63] I. Dirgová Luptáková, M. Kubovčík, and J. Pospíchal, "Wearable sensor-based human activity recognition with transformer model," *Sensors*, vol. 22, no. 5, p. 1911, 2022.

[64] Y.-W. Kim, W.-H. Cho, K.-S. Kim, and S. Lee, "Inertial-measurement-unit-based novel human activity recognition algorithm using conformer," *Sensors*, vol. 22, no. 10, p. 3932, 2022.

[65] K. Kim and Y. K. Cho, "Automatic recognition of workers' motions in highway construction by using motion sensors and long short-term memory networks," *Journal of construction engineering and management*, vol. 147, no. 3, 2021.

[66] J. Zhao and E. Obonyo, "Convolutional long short-term memory model for recognizing construction workers' postures from wearable inertial measurement units," *Advanced Engineering Informatics*, vol. 46, p. 101177, 2020.

[67] Y. Yan, H. Fan, Y. Li, E. Hoeglinger, A. Wiesinger, A. Barr, G. D. O'Connell, and C. Harris-Adamson, "Applying wearable technology and a deep learning model to predict occupational physical activities," *Applied Sciences*, vol. 11, no. 20, p. 9636, 2021.

[68] H. K. Jang, H. Han, and S. W. Yoon, "Comprehensive monitoring of bad head and shoulder postures by wearable magnetic sensors and deep learning," *IEEE Sensors Journal*, vol. 20, no. 22, pp. 13 768–13 775, 2020.

[69] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[70] K. Kim, J. Chen, and Y. K. Cho, "Evaluation of machine learning algorithms for worker's motion recognition using motion sensors," *Computing in Civil Engineering 2019*, 2019.

[71] R. Khusainov, D. Azzi, I. E. Achumba, and S. D. Bersch, "Real-time human ambulation, activity, and physiological monitoring: Taxonomy of issues, techniques, applications, challenges and limitations," *Sensors*, vol. 13, no. 10, pp. 12 852–12 902, 2013.

[72] C. J. De Luca, "Myoelectrical manifestations of localized muscular fatigue in humans," *Critical Reviews in Biomedical Engineering*, vol. 11, no. 4, pp. 251–279, 1984.

[73] R. M. Enoka and J. Duchateau, "Muscle fatigue: what, why and how it influences muscle function," *The Journal of Physiology*, vol. 586, no. 1, pp. 11–23, 2008.

[74] S. Rampichini, T. M. Vieira, P. Castiglioni, and G. Merati, "Complexity analysis of surface electromyography for assessing the myoelectric manifestation of muscle fatigue: A review," *Entropy*, vol. 22, no. 5, p. 529, 2020. [Online]. Available: https://doi.org/10.3390/e22050529

[75] M. Mugnosso, F. Marini, M. Holmes, P. Morasso, and J. Zenzeri, "Muscle fatigue assessment during robot-mediated movements," *Journal of NeuroEngineering and Rehabilitation*, vol. 15, no. 1, p. 119, 2018.

[76] J.-C. Yaacoub, G. Merlo, C. Castellini, A. Carfi, A. Parmiggiani, and A. D. Luca, "A probabilistic model for cobot decision making to mitigate human fatigue in repetitive co-manipulation tasks," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 6352–6358.

[77] M. Boyer, L. J. Bouyer, J. S. Roy, and A. Campeau-Lecours, "A real-time algorithm to estimate shoulder muscle fatigue based on surface emg signal for static and dynamic upper limb tasks," *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pp. 100–106, 2021.

[78] V. Kanal, M. Abujelala, J. Brady, G. Wylie, and F. Makedon, "Adaptive robotic rehabilitation using muscle fatigue as a trigger," in *Proceedings of the 16th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, 2019, pp. 135–142.

[79] M. Papakostas, V. Kanal, M. Abujelala, K. Tsiakas, and F. Makedon, "Physical fatigue detection through emg wearables and subjective user reports: a machine learning approach towards adaptive rehabilitation," in *Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments*, 2019, pp. 475–481.

[80] Y. Zhang, S. Chen, W. Cao, P. Guo, D. Gao, M. Wang, J. Zhou, and T. Wang, "Mffnet: Multi-dimensional feature fusion network based on attention mechanism for semg analysis to detect muscle fatigue," *Expert Systems with Applications*, vol. 185, p. 115639, 2021.

[81] J. Wang, Y. Sun, and S. Sun, "Recognition of muscle fatigue status based on improved wavelet threshold and cnn-svm," *IEEE Access*, vol. 8, pp. 207 914–207 922, 2020.

[82] J. Wang, S. Sun, and Y. Sun, "A muscle fatigue classification model based on lstm and improved wavelet packet threshold," *Sensors*, vol. 21, no. 19, p. 6369, 2021.

[83] X. Chen, M. Liu, and S. Zhang, "An lstm-attention-based method to muscle fatigue detection by integrating multi-source semg signals," in *2021 40th Chinese Control Conference (CCC)*. IEEE, 2021, pp. 8475–8480.

[84] Y. Jiang, P. Malliaras, B. Chen, and D. Kulić, "Real-time forecasting of exercise-induced fatigue from wearable sensors," *Computers in Biology and Medicine*, vol. 148, p. 105905, 2022.

[85] K. Mohanavelu, R. Lamshe, S. Poonguzhali, K. Adalarasu, and M. Jagannath, "Assessment of human fatigue during physical performance using physiological signals: a review," *Biomedical and Pharmacology Journal*, vol. 10, no. 4, pp. 1887–1896, 2017.

[86] H. Al-Libawy, A. Al-Ataby, W. Al-Nuaimy, and M. A. Al-Taee, "Hrv-based operator fatigue analysis and classification using wearable sensors," in *2016 13th International Multi-Conference on Systems, Signals & Devices (SSD)*. IEEE, 2016, pp. 268–273.

[87] R. Gao, H. Yan, and Z. Yang, "Evaluation of tractor driving vibration fatigue based on multiple physiological parameters," *PloS one*, vol. 16, no. 7, p. e0254636, 2021.

[88] A. Greco, A. Guidi, F. Felici, A. Leo, E. Ricciardi, M. Bianchi, A. Bicchi, L. Citi, G. Valenza, and E. P. Scilingo, "Muscle fatigue assessment through electrodermal activity analysis during isometric contraction," in *2017 39th annual international conference of the Ieee engineering in medicine and biology society (Embc)*. IEEE, 2017, pp. 398–401.

[89] Y. Jiang, P. Malliaras, B. Chen, and D. Kulić, "Model-based data augmentation for user-independent fatigue estimation," *Computers in Biology and Medicine*, vol. 137, p. 104839, 2021.

[90] H. Kim, S. Rho, S. Han, D. Lim, and W. Jeong, "Fabrication of textile-based dry electrode and analysis of its surface emg signal for applying smart wear," 2022. [Online]. Available: https://doi.org/10.3390/polym14173641

[91] A. Catarino, A. Rocha, and F. Ferreira, "Textile based electrodes for ecg and emg measurements," *ITMC2017-International Conference on Intelligent Textiles and Mass Customisation*.

[92] A. Shafti, R. B. R. Manero, A. M. Borg, K. Althoefer, and M. J. Howard, "Embroidered electromyography: A systematic design guide," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, pp. 1472–1480, 9 2017.

[93] A. Shafti, A. Ataka, B. U. Lazpita, A. Shiva, H. A. Wurdemann, and K. Althoefer, "Real-time robot-assisted ergonomics," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, May 2019, pp. 1975–1981.

[94] Z. Xie, L. Lu, H. Wang, B. Su, Y. Liu, and X. Xu, "Improving workers' musculoskeletal health during human-robot collaboration through reinforcement learning," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, p. 001872082311775, May 2023.

[95] M. Amani and R. Akhavian, "Intelligent ergonomic optimization in bimanual worker-robot interaction: A reinforcement learning approach," *Automation in Construction*, vol. 168, p. 105741, Dec. 2024.

[96] A. Zolotas, F. Flacco, and A. De Luca, "Imposing motion variability for ergonomic human-robot collaboration," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 10 322–10 328.

[97] E. Papadopoulos, M. Malatesta, G. Fontana, C. Mavroidis, and A. Ajoudani, "An ergo-interactive framework for human-robot collaboration via learning from demonstration," *IEEE Robotics and Automation Letters*, vol. 9, no. 6, pp. 5253–5260, 2024.

[98] L. Chen, L. F. C. Figueredo, and M. R. Dogar, "Planning for muscular and peripersonal-space comfort during human-robot forceful collaboration," in *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*. IEEE, Nov. 2018, pp. 1–8.

[99] A. G. Marin, M. S. Shourijeh, P. E. Galibarov, M. Damsgaard, L. Fritzsch, and F. Stulp, "Optimizing contextual ergonomics models in human-robot interaction," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, Oct. 2018, pp. 1–9.

[100] M. Meregalli Falerni, V. Pomponi, H. R. Karimi, M. Lavit Nicora, L. A. Dao, M. Malosio, and L. Roveda, "A framework for human–robot collaboration enhanced by preference learning and ergonomics," *Robotics and Computer-Integrated Manufacturing*, vol. 89, p. 102781, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S073658452400067X

[101] R. Rahal, G. Matarese, M. Gabiccini, A. Artoni, D. Prattichizzo, P. R. Giordano, and C. Pacchierotti, "Caring about the human operator: Haptic shared control for enhanced user comfort in robotic telemanipulation," *IEEE Transactions on Haptics*, vol. 13, no. 1, pp. 197–203, 2020.

[102] H. Modares, I. Ranatunga, F. L. Lewis, and D. O. Popa, "Optimized assistive human–robot interaction using reinforcement learning," *IEEE Transactions on Cybernetics*, vol. 46, no. 3, pp. 655–667, 2016.

[103] Y. Niu, F. Schulte, and R. R. Negenborn, "Human aspects in collaborative order picking – letting robotic agents learn about human discomfort," *Procedia Computer Science*, vol. 180, pp. 877–886, 2021, proceedings of the 2nd International Conference on Industry 4.0 and Smart Manufacturing (ISM 2020). [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050921003926

[104] A. Ghadirzadeh, X. Chen, W. Yin, Z. Yi, M. Björkman, and D. Kragic, "Human-centered collaborative robots with deep reinforcement learning," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 566–571, 2021.

[105] Mordor Intelligence, “Wearable sensors market – size, growth, share & trends,” https://www.mordorintelligence.com/industry-reports/global-wearable-sensors-market, 2025, accessed: 2025-07-22.

[106] Rokoko, “What is motion capture, and how does it work in 2022?” https://www.rokoko.com/insights/what-is-motion-capture-and-how-does-it-work-in-2022, accessed: 2023-01-08.

[107] A. Resende, S. Cerqueira, J. Barbosa, E. Damásio, A. Pombeiro, A. Silva, and C. Santos, “Ergowear: an ambulatory, non-intrusive, and interoperable system towards a human-aware human-robot collaborative framework,” in *2021 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2021, pp. 56–61.

[108] N. Choe, H. Zhao, S. Qiu, and Y. So, “A sensor-to-segment calibration method for motion capture system based on low cost mimu,” *Measurement*, vol. 131, pp. 490–500, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0263224118307176

[109] G. Ligorio, E. Bergamini, L. Truppa, M. Guaitolini, M. Raggi, A. Mannini, A. M. Sabatini, G. Vannozzi, and P. Garofalo, “A wearable magnetometer-free motion capture system: Innovative solutions for real-world applications,” *IEEE Sensors Journal*, vol. 20, no. 15, pp. 8844–8857, 2020.

[110] G. Škulj, R. Vrabič, and P. Podržaj, “A wearable imu system for flexible teleoperation of a collaborative industrial robot,” *Sensors*, vol. 21, no. 17, 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/17/5871

[111] Movella Technologies, “Sensor placement in xsens link system,” https://base.movella.com/s/article/Sensor-Placement-in-Xsens-Link-System?language=en_US, accessed: 22 July 2025.

[112] Rokoko. Smartsuit pro ii - quality body motion capture in one simple mobile mocap suit. Rokoko Electronics ApS. Accessed: 2025. [Online]. Available: https://www.rokoko.com/products/smartsuit-pro-ii

[113] Noitom Technology Ltd., “Perception neuron motion capture system,” https://www.noitom.com/productinfo.html?id=1, accessed: 22 July 2025.

[114] Nansense Inc., “Nansense motion capture suits,” https://www.nansense.com/suits/, accessed: 22 December 2025.

[115] Synertial Ltd., “Synertial motion capture products,” https://www.synertial.com/products, accessed: 22 July 2025.

[116] Xsens, “Sensor placement in xsens awinda system,” 2021, https://xsenstechnologies.force.com/knowledgebase/s/article/Sensor-Placement-in-Xsens-Awinda-System-1605791447430?language=en_US. Accessed: 2022-01-05.

[117] M. Palermo, S. M. Cerqueira, J. André, A. Pereira, and C. P. Santos, “From raw measurements to human pose - a dataset with low-cost and high-end inertial-magnetic sensor data,” *Scientific Data*, vol. 9, no. 1, p. 591, 2022. [Online]. Available: https://doi.org/10.1038/s41597-022-01690-y

[118] TDK Invensense, “MPU-9250 Product Specification,” TDK Invensense, Technical Report, 2016.

[119] *iNEMO inertial module: always-on 3D accelerometer and 3D gyroscope*, STMiicroelectronics, 2019. [Online]. Available: www.st.com

[120] STMicroelectronics N.V., “LIS3MDL Datasheet,” STMicroelectronics, Technical Report, May 2017. [Online]. Available: https://www.st.com

[121] Precision Microdrives, “10mm vibration motor, 3mm type,” Product datasheet, Part No. 310-122, accessed: 2 December 2021.

[122] S. M. B. A. Cerqueira, “Smart vest for postural monitoring and ergonomic risk assessment,” Master’s thesis, University of Minho, Guimarães, Portugal, 2019. [Online]. Available: https://hdl.handle.net/1822/77506

[123] ON Semiconductor, “AN-5083: FIS1100 AttitudeEngine™ Low Power Motion Co-Processor for High Accuracy Tracking Applications,” ON Semiconductor, Application Note AN-5083, 2015.

[124] J. K. Lee, E. J. Park, and S. N. Robinovitch, “Estimation of attitude and external acceleration using inertial sensor measurement during various dynamic conditions,” *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 8, pp. 2262–2273, Aug. 2012. [Online]. Available: https://ieeexplore.ieee.org/document/6172226

[125] *WINC1500-MR210PA Datasheet: IEEE 802.11 b/g/n IoT Module*, Atmel Corporation, Mar. 2015. [Online]. Available: https://ww1.microchip.com/downloads/en/devicedoc/atmel-42376-smartconnect-winc1500-mr210pa_datasheet.pdf

[126] F. Gemperle, C. Kasabach, J. Stivoric, M. Bauer, and R. Martin, “Design for wearability,” in *Digest of Papers. Second International Symposium on Wearable Computers (Cat. No.98EX215)*, 1998, pp. 116–122.

[127] A. Ancans, M. Greitans, R. Cacurs, B. Banga, and A. Rozentals, “Wearable sensor clothing for body movement measurement during physical activities in healthcare,” *Sensors*, vol. 21, no. 6, p. 2068, 2021, special Issue: Wearable/Wireless Body Sensor Networks for Healthcare Applications. [Online]. Available: https://www.mdpi.com/1424-8220/21/6/2068

[128] Analog Devices, “Introduction to spi interface,” *Analog Dialogue*, vol. 50, no. 3, 2016, accessed: 2025. [Online]. Available: https://www.analog.com/en/resources/analog-dialogue/articles/introduction-to-spi-interface.html

[129] E. A. for Safety and H. at Work, ““work-related musculoskeletal disorders: prevalence, costs and demographics in the eu”,” 2019.

[130] J. Martins, S. M. Cerqueira, A. W. Catarino, A. F. da Silva, A. M. Rocha, J. Vale, M. Ângelo, and C. P. Santos, “Integrating semg and imu sensors in an e-textile smart vest for forward posture monitoring: First steps,” *Sensors*, vol. 24, no. 14, 2024. [Online]. Available: https://www.mdpi.com/1424-8220/24/14/4717

[131] O. Ozturk and M. K. Yapici, “Surface electromyography with wearable graphene textiles,” *IEEE Sensors Journal*, vol. 21, pp. 14 397–14 406, 7 2021.

[132] K. A. Ohiri, C. O. Pyles, L. H. Hamilton, M. M. Baker, M. T. Mcguire, E. Q. Nguyen, L. E. Osborn, K. M. Rossick, E. G. Mcdowell, L. M. Strohsnitter, and L. J. Currano, “E-textile based modular semg suit for large area level of effort analysis,” 2022. [Online]. Available: www.nature.com/scientificreports

[133] A. Niijima, T. Yamada, T. Isezaki, R. Aoki, and T. Watanabe, “hitoecap: Wearable emg sensor for monitoring masticatory muscles with pedot-pss textile electrodes,” vol. 17. [Online]. Available: https://doi.org/10.1145/3123021.3123068

[134] Special threads - imbut: smart textiles solutions. [Online]. Available: https://www.imbut.de/en/special-threads/

[135] Swicofil - your global yarn fiber speciality competence. [Online]. Available: https://www.swicofil.com/

[136] Startseite – shieldex® – metallized technical textiles. [Online]. Available: https://www.shieldex.de/

[137] Conductive sewing embroidery thread: Silver-tech. [Online]. Available: https://www.amann.com/products/product/silver-tech/

[138] *Shieldex 117/17 HC+B*, Statex Produktions-und Vertiebs GmbH, 2016. [Online]. Available: https://www.shieldex.de/en/products/shieldex-117-17-hcb/

[139] *Shieldex 235/36 HC+B*, Statex Produktions-und Vertiebs GmbH. [Online]. Available: https://www.shieldex.de/en/products/shieldex-235-36-hcb-2/

[140] *Elitex Skin Contact*, imbut GmbH, 2017. [Online]. Available: https://www.imbut.de/images/pdf/ELITEX_SC_235_110_englisch.pdf

[141] D. J. Spencer, *Knitting technology : a comprehensive handbook and practical guide*. Technomic Pub, 2001.

[142] *Single-Lead, Heart Rate Monitor Front End*, Analog Devices, 2012. [Online]. Available: www.analog.com

[143] P. Konrad, *The ABC of EMG: A Practical Introduction to Kinesiological Electromyography*. Noraxon Inc., 2005, chapter 2: Signal processing.

[144] C. J. D. Luca, "Surface electromyography: Detection and recording," Delsys Inc., Tech. Rep., 2002, technical Note, available at https://www.delsys.com/knowledge-center/technical-notes/.

[145] Analog Devices. (2009) AD5241/AD5242 (Rev. E): Digital Potentiometers. Accessed: 16 June 2024. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/ad5241_5242.pdf

[146] G. J. Lehman and S. M. McGill, "The importance of normalization in the interpretation of surface electromyography: a proof of principle," *Journal of manipulative and physiological therapeutics*, vol. 22, pp. 444–446, 1999. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/10519560/

[147] S. F. del Toro, Y. Wei, E. Olmeda, L. Ren, W. Guowu, and V. Díaz, "Validation of a low-cost electromyography (emg) system via a commercial and accurate emg device: pilot study," *Sensors (Switzerland)*, vol. 19, 12 2019.

[148] *Agilent Technologies Agilent 34410A/11A 6 ½ Digit Multimeter Service Guide*, Agilent Technologies. [Online]. Available: https://www.testunlimited.com/pdf/an/34410-90010.pdf

[149] Testron Group, "Testron group – martindale abrasion tester," https://www.testrongroup.com/product/martindale-abrasion-tester/, 2023, accessed on 7 March 2023.

[150] SENIAM Project, "Seniam: Surface electromyography for the non-invasive assessment of muscles," http://seniam.org, 2023, accessed on 7 March 2023.

[151] Y. Jiang, Y. Qi, W. Wang, B. Bent, R. Avram, J. Olgin, and J. Dunn, "EventDTW: An Improved Dynamic Time Warping Algorithm for Aligning Biomedical Signals of Nonuniform Sampling Frequencies," *Sensors*, vol. 20, no. 9, p. 2700, 2020. [Online]. Available: https://doi.org/10.3390/s20092700

[152] C. S. Barrera, E. Piña-Martínez, R. Roberts, and E. Rodriguez-Leal, "Impact of size and shape for textile surface electromyography electrodes: A study of the biceps brachii muscle," *Textile Research Journal*, vol. 92, no. 21–22, pp. 3097–3110, 2022. [Online]. Available: https://doi.org/10.1177/00405175211048936

[153] D. Pani, A. Achilli, A. Spanu, A. Bonfiglio, M. Gazzoni, and A. Botter, "Validation of polymer-based screen-printed textile electrodes for surface emg detection," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 7, pp. 1370–1377, 2019. [Online]. Available: https://doi.org/10.1109/TNSRE.2019.2916397

[154] M. Palermo, S. Cerqueira, J. André, and C. C. Santos, "Complete inertial pose (cip) dataset," Jan. 2022. [Online]. Available: https://doi.org/10.5281/zenodo.5801927

[155] S. M. Cerqueira, M. Palermo, and C. P. Santos, "Deep inertial pose: A deep learning approach for human pose estimation," *arXiv preprint arXiv:2506.06850*, 2025. [Online]. Available: https://arxiv.org/abs/2506.06850

[156] M. Al-Amri, K. Nicholas, K. Button, V. Sparkes, L. Sheeran, and J. L. Davies, "Inertial measurement units for clinical movement analysis: Reliability and concurrent validity," *Sensors*, vol. 18, no. 3, 2018. [Online]. Available: https://www.mdpi.com/1424-8220/18/3/719

[157] T. Lisini, F. Farina, A. Garulli, A. Giannitrapani, and D. Prattichizzo, "Upper body pose estimation using wearable inertial sensors and multiplicative kalman filter," *IEEE Sensors Journal*, 09 2019.

[158] J. Figueiredo, S. Carvalho, J. P. Vilas-Boas, L. Goncalves, J. Moreno, and C. Santos, "Wearable inertial sensor system towards daily human kinematic gait analysis: Benchmarking analysis to mvn biomech," *Sensors*, vol. 20, p. 2185, 04 2020.

[159] B. Fan, Q. Li, and T. Liu, "How Magnetic Disturbance Influences the Attitude and Heading in Magnetic and Inertial Sensor-Based Orientation Estimation," *Sensors*, vol. 18, no. 2, p. 76, dec 2017. [Online]. Available: http://www.mdpi.com/1424-8220/18/1/76

[160] A. Atrsaei, H. Salarieh, A. Alasty, and M. Abediny, "Human Arm Motion Tracking by Inertial/Magnetic Sensors Using Unscented Kalman Filter and Relative Motion Constraint," *Journal of Intelligent & Robotic Systems*, vol. 90, no. 1-2, pp. 161–170, may 2018. [Online]. Available: http://link.springer.com/10.1007/s10846-017-0645-z

[161] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, "Estimation of imu and marg orientation using a gradient descent algorithm," in *2011 IEEE International Conference on Rehabilitation Robotics*, 2011, pp. 1–7.

[162] R. Mahony, T. Hamel, and J.-M. Pflimlin, "Nonlinear complementary filters on the special orthogonal group," *IEEE Transactions on Automatic Control*, vol. 53, no. 5, pp. 1203–1218, 2008.

[163] L. Pacher, C. Chatellier, R. Vauzelle, and L. Fradet, "Comparison of lower limb calibration methods for movement analysis with inertial measurement unit (imu)," *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 23, no. sup1, pp. S215–S217, 2020.

[164] M. Paulich, M. Schepers, N. Rudigkeit, and G. Bellusci, "Xsens mtw awinda: Miniature wireless inertial-magnetic motion tracker for highly accurate 3d kinematic applications," 05 2018.

[165] D. Roetenberg, "Inertial and magnetic sensing of human motion," Ph.D. dissertation, University of Twente, Netherlands, May 2006.

[166] M. Schepers, M. Giuberti, G. Bellusci *et al.*, "Xsens mvn: Consistent tracking of human motion using inertial sensing," *Xsens Technol*, pp. 1–8, 2018.

[167] J. Li, Z. Wang, B. Qi, J. Zhang, and H. Yang, "Meme: A mutually enhanced modeling method for efficient and effective human pose estimation," *Sensors*, vol. 22, no. 2, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/2/632

[168] M. Gamra and M. Akhloufi, "A review of deep learning techniques for 2d and 3d human pose estimation," *Image and Vision Computing*, vol. 114, p. 104282, 2021.

[169] Y. Huang, M. Kaufmann, E. Aksan, M. J. Black, O. Hilliges, and G. Pons-Moll, "Deep inertial poser: Learning to reconstruct human pose from sparse inertial measurements in real time," *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, vol. 37, pp. 185:1–185:15, Nov. 2018, first two authors contributed equally.

[170] X. Yi, Y. Zhou, and F. Xu, "Transpose: Real-time 3d human translation and pose estimation with six inertial sensors," *ACM Transactions on Graphics*, vol. 40, no. 4, 08 2021.

[171] X. Liao, J. Dong, K. Song, and J. Xiao, "Three-dimensional human pose estimation from sparse imus through temporal encoder and regression decoder," *Sensors*, vol. 23, no. 7, 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/7/3547

[172] J. C. Alcaraz, S. Moghaddamnia, and J. Peissig, "Efficiency of deep neural networks for joint angle modeling in digital gait assessment," *EURASIP J. Adv. Signal Process.*, vol. 2021, no. 1, p. 10, 2021. [Online]. Available: https://doi.org/10.1186/s13634-020-00715-1

[173] Y. Kumano, S. Kanoga, M. Yamamoto, H. Takemura, and M. Tada, "Estimating whole-body walking motion from inertial measurement units at wrist and heels using deep learning," *International Journal of Automation Technology*, vol. 17, no. 3, pp. 217–225, 2023.

[174] Y. Kobayashi, N. Hida, K. Nakajima, M. Fujimoto, and M. Mochimaru, "AIST Gait Database 2019," 2019, accessed: 2022. [Online]. Available: https://unit.aist.go.jp/harc/ExPART/GDB2019.html

[175] H. Coskun, F. Achilles, R. DiPietro, N. Navab, and F. Tombari, "Long short-term memory kalman filters: Recurrent neural estimators for pose regularization," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 5525–5533.

[176] Xsens, "Getting started with the awinda in mvn tutorial," 2021, https://xsenstechnologies.force.com/knowledgebase/s/article/Getting-Started-with-the-Awinda-in-MVN-Tutorial?language=en_US. Accessed: 2022-01-05.

[177] I. H. Lopez-Nava and A. Munoz-Melendez, "Wearable inertial sensors for human motion analysis: A review," *IEEE Sensors Journal*, vol. 16, no. 22, pp. 7821–7834, 2016.

[178] P. Picerno, "25 years of lower limb joint kinematics by using inertial and magnetic sensors: A review of methodological approaches," *Gait & Posture*, vol. 51, pp. 239–246, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0966636216306373

[179] D. Roetenberg, H. Luinge, and P. Slycke, "Xsens mvn: Full 6dof human motion tracking using miniature inertial sensors," *Xsens Motion Technologies BV, Tech. Rep*, vol. 1, 2009.

[180] F. L. Markley and J. L. Crassidis, *Fundamentals of Spacecraft Attitude Determination and Control*, ser. Space Technology Library. Springer New York, NY, 2014. [Online]. Available: https://doi.org/10.1007/978-1-4939-0802-8

[181] A. M. Sabatini, "Kalman-filter-based orientation determination using inertial/magnetic sensors: Observability analysis and performance evaluation," *Sensors*, vol. 11, no. 10, pp. 9182–9206, 2011. [Online]. Available: https://www.mdpi.com/1424-8220/11/10/9182

[182] S. O. H. Madgwick, S. Wilson, R. Turk, J. Burridge, C. Kapatos, and R. Vaidyanathan, "An extended complementary filter for full-body marg orientation estimation," *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 4, pp. 2054–2064, 2020.

[183] S. Rong, "Quantifications and characteristics of dynamic soft tissue artifacts captured by wearable inertial measurement unit sensors," Ph.D. dissertation, University of British Columbia, 2022. [Online]. Available: https://open.library.ubc.ca/collections/ubctheses/24/items/1.0421332

[184] A. Behrad and N. Roodsarabi, "3D Human Motion Tracking and Reconstruction Using DCT Matrix Descriptor," *ISRN Machine Vision*, vol. 2012, pp. 1–11, may 2012. [Online]. Available: https://www.hindawi.com/journals/isrn/2012/235396/

[185] H. R. Gonçalves, A. Rodrigues, and C. P. Santos, "Gait monitoring system for patients with parkinson's disease," *Expert Systems with Applications*, vol. 185, p. 115653, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417421010447

[186] M. Nomura, S. Watanabe, Y. Akimoto, Y. Ozaki, and M. Onishi, "Warm starting cma-es for hyperparameter optimization," *arXiv preprint arXiv:2012.06932*, 2020.

[187] C. Bouten, K. Koekkoek, M. Verduin, R. Kodde, and J. Janssen, "A triaxial accelerometer and portable data processing unit for the assessment of daily physical activity," *IEEE transactions on bio-medical engineering*, vol. 44, pp. 136–47, 04 1997.

[188] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Advances in Neural Information Processing Systems*, 2019, pp. 8024–8035. [Online]. Available: https://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

[189] F. William and The PyTorch Lightning team, "PyTorch Lightning," 3 2019. [Online]. Available: https://github.com/PyTorchLightning/pytorch-lightning

[190] D. Weber, C. Gühmann, and T. Seel, "Neural networks versus conventional filters for inertial-sensor-based attitude estimation," in *2020 IEEE 23rd International Conference on Information Fusion (FUSION)*. IEEE, 2020, pp. 1–8.

[191] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[192] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, F. Bach and D. Blei, Eds., vol. 37. Lille, France: PMLR, 2015, pp. 448–456.

[193] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, G. Gordon, D. Dunson, and M. Dudík, Eds., vol. 15. Fort Lauderdale, FL, USA: PMLR, 11–13 Apr 2011, pp. 315–323. [Online]. Available: https://proceedings.mlr.press/v15/glorot11a.html

[194] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

[195] L. Zhao, Y. Song, C. Zhang, Y. Liu, P. Wang, T. Lin, M. Deng, and H. Li, "T-gcn: A temporal graph convolutional network for traffic prediction," *IEEE Transactions on Intelligent Transportation Systems*, 2019.

[196] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[197] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[198] L. N. Smith and N. Topin, "Super-convergence: Very fast training of neural networks using large learning rates," in *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, vol. 11006. International Society for Optics and Photonics, 2019, p. 1100612.

[199] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *International conference on machine learning*, 2013, pp. 1310–1318.

[200] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929–1958, 2014.

[201] A. Krogh and J. A. Hertz, "A simple weight decay can improve generalization," in *Advances in Neural Information Processing Systems 4*, J. E. Moody, S. J. Hanson, and R. P. Lippmann, Eds. Morgan-Kaufmann, 1992, pp. 950–957.

[202] D. Q. Huynh, "Metrics for 3d rotations: Comparison and analysis," *Journal of Mathematical Imaging and Vision*, vol. 35, no. 2, pp. 155–164, 2009.

[203] J. Romano, J. D. Kromrey, J. Coraggio, and J. Skowronek, "Appropriate statistics for ordinal level data: Should we really be using t-test and Cohen's d for evaluating group differences on the NSSE and other surveys?" in *Annual Meeting of the Florida Association of Institutional Research*, 2006.

[204] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li, "On the continuity of rotation representations in neural networks," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 5738–5746.

[205] A. Pepe, J. Lasenby, and P. Chacón, "Learning rotations," *Mathematical Methods in the Applied Sciences*, 2022. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/mma.8698

[206] D. R. Martins, S. M. Cerqueira, and C. P. Santos, "Seeking optimal and explainable deep learning models for inertial-based posture recognition," *Knowledge-Based Systems*, vol. 306, p. 112700, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705124013340

[207] S.-M. Mäkela, A. Lämsä, J. S. Keränen, J. Liikka, J. Ronkainen, J. Peltola, J. Häikiö, S. Järvinen, and M. Bordallo López, "Introducing vtt-coniot: A realistic dataset for activity recognition of construction workers using imu devices," *Sustainability*, vol. 14, no. 1, p. 220, 2021.

[208] D. R. Martins, S. Cerqueira, and C. P. Santos, "Inertial Dataset for Posture Recognition in Agriculture and Construction Tasks," Jan. 2024. [Online]. Available: https://doi.org/10.5281/zenodo.10455976

[209] S. W. Smith, "Chapter 15 - moving average filters," in *Digital Signal Processing*, S. W. Smith, Ed. Boston: Newnes, 2003, pp. 277–284. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780750674447500522

[210] K. L. Sainani, "Dealing with non-normal data," *Pm&r*, vol. 4, no. 12, pp. 1001–1005, 2012.

[211] S. Vora and H. Yang, "A comprehensive study of eleven feature selection algorithms and their impact on text classification," in *2017 Computing Conference*. IEEE, 2017, pp. 440–449.

[212] A. Fujita, J. R. Sato, M. A. A. Demasi, M. C. Sogayar, C. E. Ferreira, and S. Miyano, "Comparing pearson, spearman and hoeffding's d measure for gene expression association analysis," *Journal of bioinformatics and computational biology*, vol. 7, no. 04, pp. 663–684, 2009.

[213] G. Aquino, M. G. Costa, and C. F. Costa Filho, "Explaining one-dimensional convolutional models in human activity recognition and biometric identification tasks," *Sensors*, vol. 22, no. 15, p. 5644, 2022.

[214] K. Fauvel, T. Lin, V. Masson, É. Fromont, and A. Termier, "Xcm: An explainable convolutional neural network for multivariate time series classification," *Mathematics*, vol. 9, no. 23, p. 3137, 2021.

[215] H. A. Afan, A. Yafouz, A. H. Birima, A. N. Ahmed, O. Kisi, B. Chaplot, and A. El-Shafie, "Linear and stratified sampling-based deep learning models for improving the river streamflow forecasting to mitigate flooding disaster," *Natural Hazards*, vol. 112, no. 2, pp. 1527–1545, 2022.

[216] R. Assaf, I. Giurgiu, F. Bagehorn, and A. Schumann, "Mtex-cnn: Multivariate time series explanations for predictions with convolutional neural networks," in *2019 ieee international conference on data mining (icdm)*. IEEE, 2019, pp. 952–957.

[217] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*. pmlr, 2015, pp. 448–456.

[218] X. Li, S. Chen, X. Hu, and J. Yang, "Understanding the disharmony between dropout and batch normalization by variance shift," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 2682–2690.

[219] R. L. Kumar, J. Kakarla, B. V. Isunuri, and M. Singh, "Multi-class brain tumor classification using residual network and global average pooling," *Multimedia Tools and Applications*, vol. 80, pp. 13 429–13 438, 2021.

[220] T.-Y. Hsiao, Y.-C. Chang, H.-H. Chou, and C.-T. Chiu, "Filter-based deep-compression with global average pooling for convolutional networks," *Journal of Systems Architecture*, vol. 95, pp. 9–18, 2019.

[221] L. Alawneh, B. Mohsen, M. Al-Zinati, A. Shatnawi, and M. Al-Ayyoub, "A comparison of unidirectional and bidirectional lstm networks for human activity recognition," in *2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. IEEE, 2020, pp. 1–6.

[222] R. Zhao, R. Yan, J. Wang, and K. Mao, "Learning to monitor machine health with convolutional bi-directional lstm networks," *Sensors*, vol. 17, no. 2, p. 273, 2017.

[223] Y. Shavit and I. Klein, "Boosting inertial-based human activity recognition with transformers," *IEEE Access*, vol. 9, pp. 53 540–53 547, 2021.

[224] I. Kandel and M. Castelli, "The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset," *ICT express*, vol. 6, no. 4, pp. 312–315, 2020.

[225] P. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*, ser. Always learning. Pearson, 2014. [Online]. Available: https://books.google.pt/books?id=BExxngEACAAJ

[226] W. S. Ahmed *et al.*, "The impact of filter size and number of filters on classification accuracy in cnn," in *2020 International conference on computer science and software engineering (CSASE)*. IEEE, 2020, pp. 88–93.

[227] M. Inoue, S. Inoue, and T. Nishida, "Deep recurrent neural network for mobile human activity recognition with high throughput," *Artificial Life and Robotics*, vol. 23, pp. 173–185, 2018.

[228] F. Li, S. Li, X. Fan, X. Li, and H. Chang, "Structural attention enhanced continual meta-learning for graph edge labeling based few-shot remote sensing scene classification," *Remote Sensing*, vol. 14, no. 3, p. 485, 2022.

[229] A. C. Leon, "3.12 - descriptive and inferential statistics," in *Comprehensive Clinical Psychology*, A. S. Bellack and M. Hersen, Eds. Oxford: Pergamon, 1998, pp. 243–285. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B0080427073002649

[230] G. B. Health and S. Executive, *Upper Limb Disorders in the Workplace*, ser. HS(G) series / HSE. HSE Books, 2002. [Online]. Available: https://books.google.pt/books?id=AyZ8AAAACAAJ

[231] D. R. Martins, S. M. Cerqueira, A. Pombeiro, A. F. da Silva, A. M. A. C. Rocha, and C. P. Santos, "Ergoreport: A holistic posture assessment framework based on inertial data and deep learning," *Sensors*, vol. 25, no. 7, p. 2282, Apr. 2025.

[232] Y.-K. Kong, S.-J. Lee, K.-S. Lee, G.-R. Kim, and D.-M. Kim, "Development of an ergonomics checklist for investigation of work-related whole-body disorders in farming–awba: Agricultural whole-body assessment," *Journal of Agricultural Safety and Health*, vol. 21, no. 4, pp. 207–215, 2015.

[233] Y.-K. Kong, S.-J. Lee, K.-S. Lee, J.-G. Han, and D.-M. Kim, "Development of an ergonomic checklist for the investigation of work-related upper limb disorders in farming-aula: Agricultural upper-limb assessment," *Journal of the Ergonomics Society of Korea*, vol. 30, no. 4, pp. 481–489, 2011.

[234] Y.-K. Kong, J.-G. Han, and D.-M. Kim, "Development of an ergonomic checklist for the investigation of work-related lower limb disorders in farming-alla: Agricultural lower-limb assessment," *Journal of the Ergonomics Society of Korea*, vol. 29, no. 6, pp. 933–942, 2010.

[235] K.-H. Choi, D.-M. Kim, M.-U. Cho, C.-W. Park, S.-Y. Kim, M.-J. Kim, and Y.-K. Kong, "Application of aula risk assessment tool by comparison with other ergonomic risk assessment tools," *International Journal of Environmental Research and Public Health*, vol. 17, no. 18, p. 6479, 2020.

[236] Y.-K. Kong, S.-y. Lee, K.-S. Lee, and D.-M. Kim, "Comparisons of ergonomic evaluation tools (alla, rula, reba and owas) for farm work," *International journal of occupational safety and ergonomics*, vol. 24, no. 2, pp. 218–223, 2018.

[237] D. Kee and W. Karwowski, "Luba: an assessment technique for postural loading on the upper body based on joint motion discomfort and maximum holding time," *Applied ergonomics*, vol. 32, no. 4, pp. 357–366, 2001.

[238] L. McAtamney and E. N. Corlett, "Rula: a survey method for the investigation of work-related upper limb disorders," *Applied ergonomics*, vol. 24, no. 2, pp. 91–99, 1993.

[239] F. D. Davis, “Perceived usefulness, perceived ease of use, and user acceptance of information technology,” *MIS quarterly*, pp. 319–340, 1989.

[240] M. Speicher, A. Both, and M. Gaedke, “Inuit: the interface usability instrument,” in *Design, User Experience, and Usability: Design Discourse: 4th International Conference, DUXU 2015, Held as Part of HCI International 2015, Los Angeles, CA, USA, August 2–7, 2015, Proceedings, Part I*. Springer, 2015, pp. 256–268.

[241] N. Coleman, “Sumi (software usability measurement inventory) as a knowledge elicitation tool for improving usability,” *Ireland: Department of Applied Psychology, University College Cork.*, 1993. [Online]. Available: https://sumi.uxp.ie/en/

[242] Delsys Incorporated, *TRIGNO Wireless System SDK User’s Guide*, 2019, pp. 1–29. [Online]. Available: https://www.delsys.com/downloads/USERSGUIDE/trigno/sdk.pdf

[243] ROCHU Soft Robotics, “Rochu soft gripper,” https://asenc.co.th/gripper/soft_gripper.pdf, 2024, powerPoint presentation in PDF format.

[244] Universal Robots A/S, “Universal_robots_ros_driver: Universal robots ros driver supporting cb3 and e-series,” https://github.com/UniversalRobots/Universal_Robots_ROS_Driver, 2021, version 2.0.0, accessed in 2021.

[245] “Universal Robots A/S, Offline Simulator: URSim for non-Linux 5.12.6 LTS,” https://www.universal-robots.com/download/software-e-series/simulator-non-linux/offline-simulator-e-series-ur-sim-for-non-linux-5126-lts/, 2018, uRSim for non-Linux systems, accessed in 2025.

[246] D. Gorbani, C. Sartore, and collaborators, “human-gazebo: Urdf models of humans created to perform human-robot interaction experiments,” https://github.com/robotology/human-gazebo, 2020, version 1.0, accessed in 2021.

[247] J. F. Tornero-Aguilera, J. Jimenez-Morcillo, A. Rubio-Zarapuz, and V. J. Clemente-Suárez, “Central and peripheral fatigue in physical exercise explained: A narrative review,” *International Journal of Environmental Research and Public Health*, vol. 19, no. 7, p. 3909, 2022.

[248] J.-j. Wan, Z. Qin, P.-y. Wang, Y. Sun, and X. Liu, “Muscle fatigue: general understanding and treatment,” *Experimental & Molecular Medicine*, vol. 49, no. 10, p. e384, 2017. [Online]. Available: https://doi.org/10.1038/emm.2017.194

[249] X. Pan, G. Zhang, N. Yu, C. Cai, H. Ma, and B. Yan, "Low-frequency human motion energy scavenging with wearable tumbler-inspired electromagnetic energy harvesters," *International Journal of Mechanical Sciences*, vol. 268, p. 109029, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020740324000729

[250] N. Li, R. Zhou, B. Krishna, A. Pradhan, H. Lee, J. He, and N. Jiang, "Non-invasive techniques for muscle fatigue monitoring: A comprehensive survey," *ACM Comput. Surv.*, vol. 56, no. 9, Apr. 2024. [Online]. Available: https://doi.org/10.1145/3648679

[251] A. A. Didi and F. Bereksi Reguig, "Modeling local muscle fatigue phases from emg: A comparative study of classification and regression approaches," *Edelweiss Applied Science and Technology*, vol. 9, no. 6, pp. 925–938, 2025.

[252] D. Farina, "Interpretation of the surface electromyogram in dynamic contractions," *Exercise and Sport Sciences Reviews*, vol. 34, no. 3, pp. 121–127, Jul. 2006.

[253] P. A. Ortega-Auriol, T. F. Besier, W. D. Byblow, and A. J. C. McMorland, "Fatigue influences the recruitment, but not structure, of muscle synergies," *Frontiers in Human Neuroscience*, vol. 12, p. 217, 2018. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnhum.2018.00217/full

[254] S. Das, G. Camporese, S. Cheng, and L. Ballan, "Distilling knowledge for short-to-long term trajectory prediction," *arXiv preprint arXiv:2305.08553*, 2024.

[255] V. Martins, S. M. Cerqueira, M. Balcells, E. R. Edelman, and C. P. Santos, "Towards a human-sensitive controller: Learning human specificities in ergonomics and physical constraints," in *2025 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2025, pp. 158–163.

[256] ——, "A Human-Sensitive Controller: Adapting to Human Ergonomics and Physical Constraints via Reinforcement Learning," arXiv preprint arXiv:2504.10102, 2025.

[257] M. Lagomarsino, M. Lorenzini, M. D. Constable, E. De Momi, C. Becchio, and A. Ajoudani, "Maximising coefficiency of human-robot handovers through reinforcement learning," *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 4378–4385, 2023.

[258] Y. Long, W. Wei, T. Huang, Y. Wang, and Q. Dou, "Human-in-the-loop embodied intelligence with interactive simulation environment for surgical robot learning," *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 4441–4448, 2023.

[259] M. Mainampati and B. Chandrasekaran, "Implementation of human in the loop on the turtlebot using reinforced learning methods and robot operating system," in *Proceedings of the IEEE Annual Information Technology, Electronics and Mobile Communication Conference*, 2021, pp. 448–452.

[260] S. Singh and J. Heard, "Human-aware reinforcement learning for adaptive human robot teaming," in *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction*, 2022, pp. 1049–1052.

[261] K. Puthuveetil, C. C. Kemp, and Z. Erickson, "Bodies uncovered: Learning to manipulate real blankets around people via physics simulations," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1984–1991, 2022.

[262] B. Chandrasekaran and M. Mainampati, "A human in the loop based robotic system using soft actor critic with discrete actions," in *Proceedings of the International Conference on Mechatronics, Robotics and Automation*, 2021, pp. 19–25.

[263] I. Iturrate, L. Montesano, and J. Mínguez, "Robot reinforcement learning using eeg-based reward signals," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2010, pp. 4822–4829.

[264] D. Xu, M. Agarwal, F. Fekri, and R. Sivakumar, "Playing games with implicit human feedback," Online, 2020, available at Semantic Scholar: CorpusID 210832309.

[265] A. L. Thomaz and C. Breazeal, "Reinforcement learning with human teachers: Evidence of feedback and guidance with implications for learning performance," in *Proceedings of the National Conference on Artificial Intelligence*, 2006, pp. 1000–1005.

[266] G. Masci, G. Cazzato, G. Milano, G. Ciolli, G. Malerba, C. Perisano, T. Greco, O. Palmacci, G. Maccauro, and F. Liuzza, "The stiff elbow: Current concepts," *Orthopedic Reviews*, vol. 12, p. 8661, 2020.

[267] M. Soori, B. Arezoo, and R. Dastres, "Artificial intelligence, machine learning and deep learning in advanced robotics: A review," *Cognitive Robotics*, vol. 3, pp. 54–70, Jan. 2023.

[268] V. Mnih *et al.*, “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[269] E. L. Zwerus, N. W. Willigenburg, V. A. Scholtes, M. P. Somford, D. Eygendaal, and M. P. V. D. Bekerom, “Normative values and affecting factors for the elbow range of motion,” *Sage Journals*, 2017.

[270] “Range of motion normative values - physiopedia.” [Online]. Available: https://www.physio-pedia.com/Range_of_Motion_Normative_Values

[271] V. Martins, S. M. Cerqueira, L. Coelho, and C. P. Santos, “An admittance-based control strategy for human-robot collaboration task quality management,” in *2024 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2024, pp. 207–212.

[272] Z. Jin, D. Qin, A. Liu, W.-a. Zhang, and L. Yu, “Model predictive variable impedance control of manipulators for adaptive precision-compliance tradeoff,” *IEEE/ASME Transactions on Mechatronics*, vol. 28, no. 2, pp. 1174–1186, 2023.

[273] G. Kang, H. S. Oh, J. K. Seo, U. Kim, and H. R. Choi, “Variable admittance control of robot manipulators based on human intention,” *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 3, pp. 1023–1032, 2019.

[274] L. Han, L. Zhao, Y. Huang, and W. Xu, “Variable admittance control for safe physical human–robot interaction considering intuitive human intention,” *Mechatronics*, vol. 97, p. 103098, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S095741582300154X

[275] F. Ferraguti, C. T. Landi, L. Sabattini, M. Bonfè, C. Fantuzzi, and C. Secchi, “A variable admittance control strategy for stable physical human–robot interaction,” *The International Journal of Robotics Research*, vol. 38, no. 6, pp. 747–765, 2019. [Online]. Available: https://doi.org/10.1177/0278364919840415

[276] Z. Li, H. Wei, H. Zhang, and C. Liu, “A variable admittance control strategy for stable and compliant human-robot physical interaction,” *IEEE Robotics and Automation Letters*, vol. 10, no. 2, pp. 1138–1145, 2025.

[277] G. Kang, H. S. Oh, J. K. Seo, U. Kim, and H. R. Choi, “Variable admittance control of robot manipulators based on human intention,” *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 3, pp. 1023–1032, 2019.

[278] SENIAM Group, “Seniam,” seniam.org, Accessed: February 24, 2024. [Online]. Available: http://www.seniam.org/

[279] U.S. General Services Administration, “System Usability Scale (SUS): Improving Products Since 1986,” digital.gov, Accessed: February 24, 2024. [Online]. Available: https://digital.gov/2014/08/29/system-usability-scale-improving-products-since-1986/

[280] A. V. Boas, J. André, S. M. Cerqueira, and C. P. Santos, “A dmps-based approach for human-robot collaboration task quality management,” in *2023 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2023, pp. 226–231.

[281] D. R. Martins, S. M. Cerqueira, and C. P. Santos, “Combining inertial-based ergonomic assessment with biofeedback for posture correction: A narrative review,” *Computers & Industrial Engineering*, vol. 190, p. 110037, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S036083522400158X

[282] STMicroelectronics N.V., “Stm32h7 nucleo 144 boards (mb1364),” STMicroelectronics, Tech. Rep., Jun. 2020, consulted in 2021.

[283] “Stm32h723/733, stm32h725/735 and stm32h730 value line advanced arm®-based 32-bit mcus,” STMicroelectronics N.V., Tech. Rep., December 2021. [Online]. Available: https://www.st.com/resource/en/product_presentation/microcontrollers_stm32h72x-3x_line_product-overview.pdf

Part III
Appendices

Appendix A
Literature Review

A.1 Wearables for ergonomic risk assessment

An electronic search was conducted in the IEEE Xplore and Scopus databases, following the guidelines of the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA). Studies published between 2013 and 2023 were analysed using the following combination of keywords:

```
(("ergonomic*" AND ("assessment" OR "risk")) OR "posture")
AND ("wearable" OR "inertial-based")
AND ("feedback" OR "biofeedback")
```

The search on the IEEE database included all metadata, whereas, on the Scopus database, it was restricted to the article title, abstract and keywords. The selected articles (i) were all written in English and (ii) involved wearable inertial technologies. The exclusion criteria were the following: (i) articles mentioning optical MoCap systems, (ii) studies addressing balance/gait training/rehabilitation, (iii) studies focused on sitting postures, and (iv) studies presenting only a simple/binary assessment or with lack of information. Review articles were also excluded from the search results. Yet, reviews' references were examined, as well as the references of other selected articles, and the relevant ones were included.

After a preliminary search, completed on 7 June 2023 and revised on 28 August 2023, 62 articles were identified in IEEE Xplore, 257 in Scopus, and 42 through other sources, as reported in Figure 135. After duplicate removal, the records were screened by title and abstract. This process resulted in 29 articles selected for full-text assessment to determine whether they met the inclusion criteria and were suitable for the scope of this section. A total of 20 articles were included in the original paper [281]. However, these articles addressed two complementary themes: inertial-based postural ergonomic assessment and biofeedback strategies. Since the present thesis focuses specifically on postural ergonomic assessment, the detailed synthesis is restricted to the 11 studies directly addressing this topic.

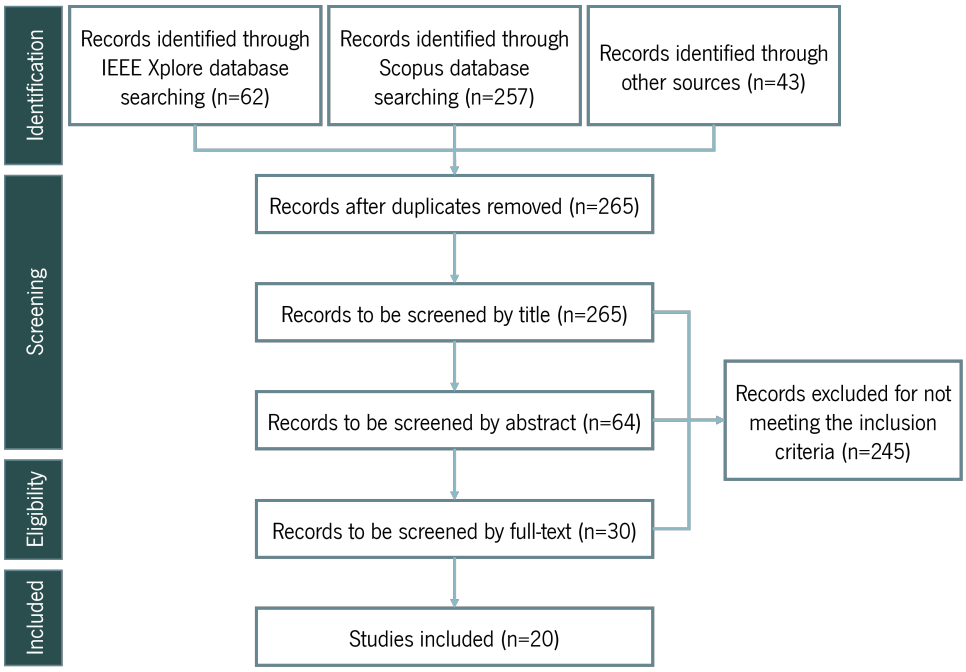


Figure 135: PRISMA flow diagram for the search strategy.

A.2 AI-based activity recognition models

A systematic literature search on AI-based activity recognition models was conducted in the Scopus database in January 2023, following PRISMA guidelines. The search combined the following keywords:

```
("activity recognition" OR "posture recognition" OR "motion recognition")
AND "wearable" AND "deep learning" AND "model"
```

Studies were included if written in English and based on wearable technologies. Those relying on optical MoCap, single-subject datasets, unreported dataset splits, or consisting of review papers were excluded. Additional relevant works were identified through reference screening of the retrieved articles. In total, 416 records were retrieved from Scopus and 67 from other sources. After duplicate removal, records were screened in two stages, first by title and then by abstract. This process resulted in 22 articles selected for full-text assessment, of which 9 met the inclusion criteria. The full selection process is illustrated in Figure 136.

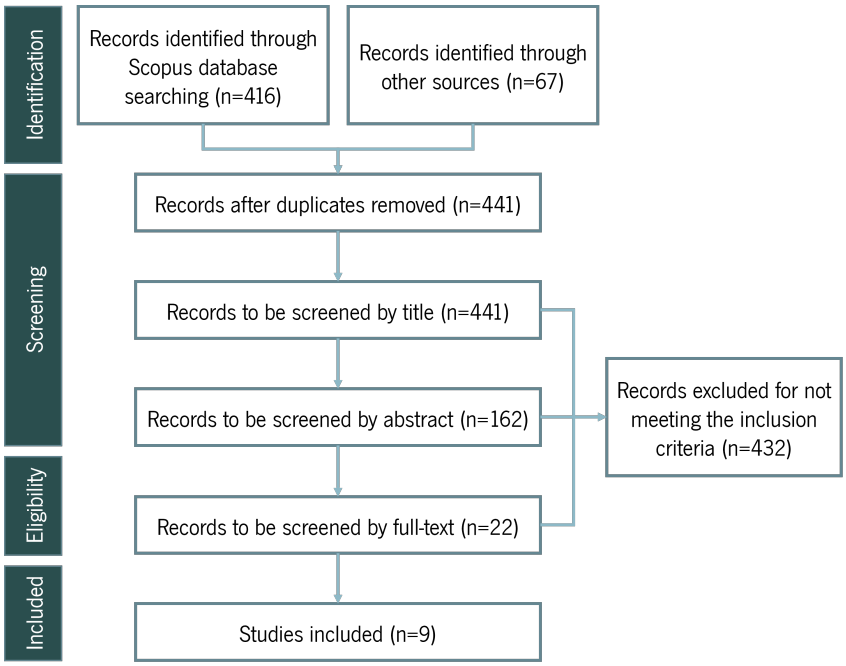


Figure 136: PRISMA flow diagram for the research on activity recognition models.

A.3 Muscle Fatigue Estimation

A structured literature search on fatigue estimation methods and cobot control strategies was conducted in the Scopus database in June 2023 and updated in August 2025. The search combined the following keywords:

```
("muscle" OR "fatigue" OR "upper body" OR "sEMG" OR "muscle fatigue")
AND (("machine learning" OR "deep learning")
OR ("Human-Robot Collaboration" OR "control strategy"))
```

Studies were considered relevant if they addressed fatigue estimation models, control strategies for human–robot collaboration, or both, and were published in English. Additional relevant works were identified through reference screening of the selected articles.

A.4 Control Strategies for Human-aware Collaborative Robots

The literature review in this chapter follows a targeted screening approach supported by reference checking, instead of a formal PRISMA-based systematic review. Some of the studies discussed in this section were identified through the structured literature search reported in Section 2.3. This set was complemented by targeted searches using keywords such as ”human–robot collaboration”, ”ergonomics”, ”kinematics”, ”physical restrictions”, ”physical human–robot collaboration”, ”learning”, and ”reinforcement learning”. These terms were combined using `AND` and `OR` Boolean operators. Relevant references

cited in the selected articles were also screened manually.

Studies were considered relevant if they addressed human–robot collaboration control strategies that optimized or accounted for human-related factors, such as ergonomics, posture, comfort, workload, physical effort, physical restrictions, or fatigue, and were published in English.

Appendix B
Wearable Technology for Human Awareness

B.1 Ergowear

B.1.1 Hardware Implementation Details

This appendix provides detailed information on the hardware implementation of the wearable system, including sensor specifications, communication setup, cabling, biofeedback circuitry, power analysis, and mechanical integration.

B.1.2 Sensor Specifications

The acquisition system is based on the LSM6DSOX (accelerometer and gyroscope) and LIS3MDL (magnetometer) sensors. Compared to the MPU-9250 used in previous prototypes, this configuration provides improved measurement stability.

The gyroscope zero-rate offset is specified as ±1 °/s, compared to ±5 °/s in the MPU-9250. The accelerometer zero-g offset is ±20 mg, whereas the MPU-9250 can reach up to ±80 mg [118–120]. These improvements reduce bias and noise, which directly affect drift in orientation estimation.

Each sensor module includes wire-to-board connectors that provide power and I^2C communication through four pins: voltage, ground, serial data (SDA), and serial clock (SCL).

B.1.3 Communication Setup

Communication between sensors and the processing unit is implemented using the I^2C protocol. Each sensor board supports only two static addresses, which limits the number of devices per bus.

To interface all nine sensors, the system uses five independent I^2C buses operating in fast mode (400 kHz) [119]. The communication structure is divided into three branches: one for the trunk and head, and one for each arm. Each branch connects three sensors.

Each branch requires six conductors: four for I^2C communication and two for power supply.

B.1.4 Wiring and Connectors

Due to the connector layout on the sensor boards, only four wires can be directly connected at each interface. To maintain continuity across sensors, a pass-through wiring scheme was implemented inside each module, allowing the communication lines to continue along the cable.

Circular push-pull connectors were selected for inter-module connections due to their mechanical robustness and ease of use. These connectors support repeated connection cycles and provide reliable electrical contact during movement.

B.1.5 Cabling

Shielded circular cables were used throughout the system to reduce the effect of electromagnetic interference, particularly in industrial environments. Each cable includes at least six conductors to support both communication and power lines.

The cable outer diameter was limited to 5 mm to ensure compatibility with the connectors and to maintain sufficient flexibility. Excessively stiff cables were avoided, as they can introduce sensor misalignment or restrict user movement.

A Y-splitter was introduced at the upper arm to separate connections to the forearm and hand. This avoids placing connectors near the wrist joint or creating excess cable loops, both of which can interfere with motion. Circular connectors were used at this junction to improve mechanical reliability.

B.1.6 Biofeedback Circuit

The biofeedback system uses six coin-style ERM vibration motors (model 310-122, Precision Microdrives Ltd, London, UK), each with a diameter of 10 mm, thickness of 3.4 mm, and mass of 1.2 g [121].

Each motor draws approximately 60 mA at 3.3 V, exceeding the current capacity of the microcontroller output pins. To address this, a PWM-controlled switching circuit was implemented using an n-channel MOSFET as a low-side switch.

When the control signal is LOW, the MOSFET remains in cutoff and the motor is inactive. When the signal is HIGH, the MOSFET allows current to flow through the motor. The use of PWM enables control of vibration intensity through duty cycle modulation.

A flyback diode is connected in parallel with each motor to dissipate inductive energy during switching. A gate resistor is included to limit the current during switching, and a pull-down resistor ensures the MOSFET remains off during system startup.

B.1.7 Power Analysis

The system is powered by a 5 V supply through the embedded processing board. All subsystems operate at 3.3 V.

The estimated current consumption of the system is summarized in Table 50. The total current draw is approximately 883 mA when all components operate simultaneously.

Table 50: Estimated current consumption of the wearable system components.

Component	**Current (mA)**
Feedback system ($60\,\text{mA} \times 6$)	360
Acquisition system ($0.82\,\text{mA} \times 9$)	4.92
WiFi module	268
STM32 board	250
Total	**882.92**

To support continuous operation for approximately 8 hours, a minimum battery capacity of 7063 mAh is required. The selected power bank exceeds this requirement.

The onboard 3.3 V regulator of the NUCLEO-H723ZG is limited to 500 mA [282]. To supply the full system load, an external 3.3 V regulator was added to power the WiFi module, using the board's 5 V output. The onboard regulator continues to supply the acquisition and feedback systems.

B.1.8 Mechanical Integration

The system was integrated into the garment using custom-designed enclosures and distributed wiring. All wiring was manually connected between the master PCB (Figure 137) and the sensor boards. Following the assembly of each connector, continuity tests were performed to verify robust connections between the sensor pins and the processing unit.

Ten enclosures were designed and fabricated using 3D printing. These include one master module housing the processing unit and nine sensor modules. The enclosures were designed with rounded edges to avoid damaging the textile and to improve comfort.Figure 138 depicted the interior view of the master, intermediary, and extremity modules,

Two types of sensor modules were developed. Intermediary modules include both input and output connections, allowing continuous wiring between sensors. Extremity modules include only a single connection and are used at terminal positions.

The total weight of the integrated wearable system, including garment and electronics, is approximately 1130 g.

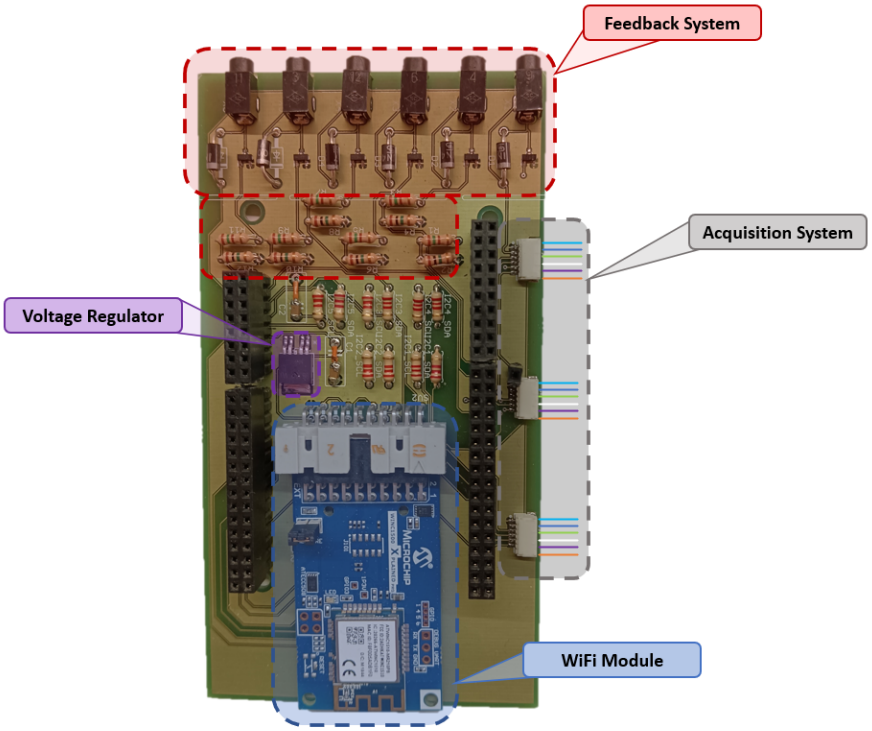


Figure 137: Ergowear's main board PCB.

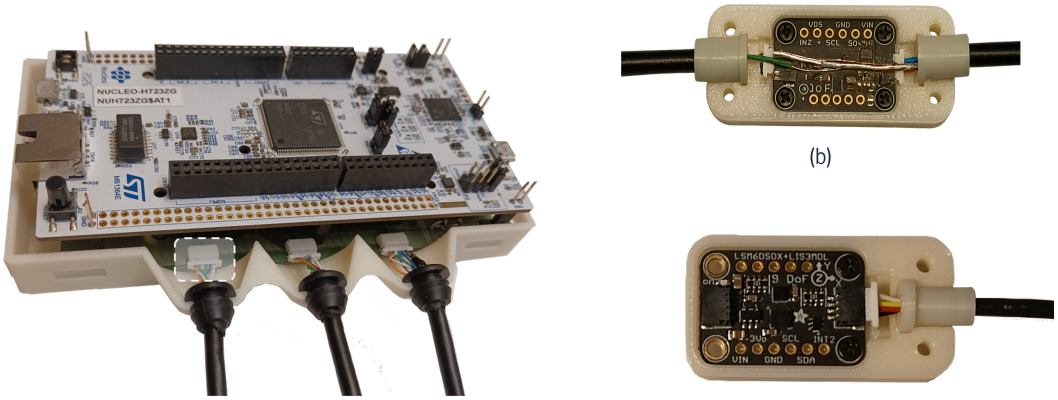


Figure 138: Interior view of ergowear modules: (a) master module; (b) intermediary module; (c) extremity module.

B.1.9 Embedded Software Development

The embedded software on the low-level processor is structured into four main components: the system setup, main loop, sensory acquisition (timer interruption) and WiFi callback routine. A figure depicting the low level controller operability is presented in Figure 139

During initialization, the microcontroller configures the GPIOs, timers, SPI and I^2C interfaces, followed by the setup of the WiFi module, which operates as an Access Point (AP). Once the main station connects

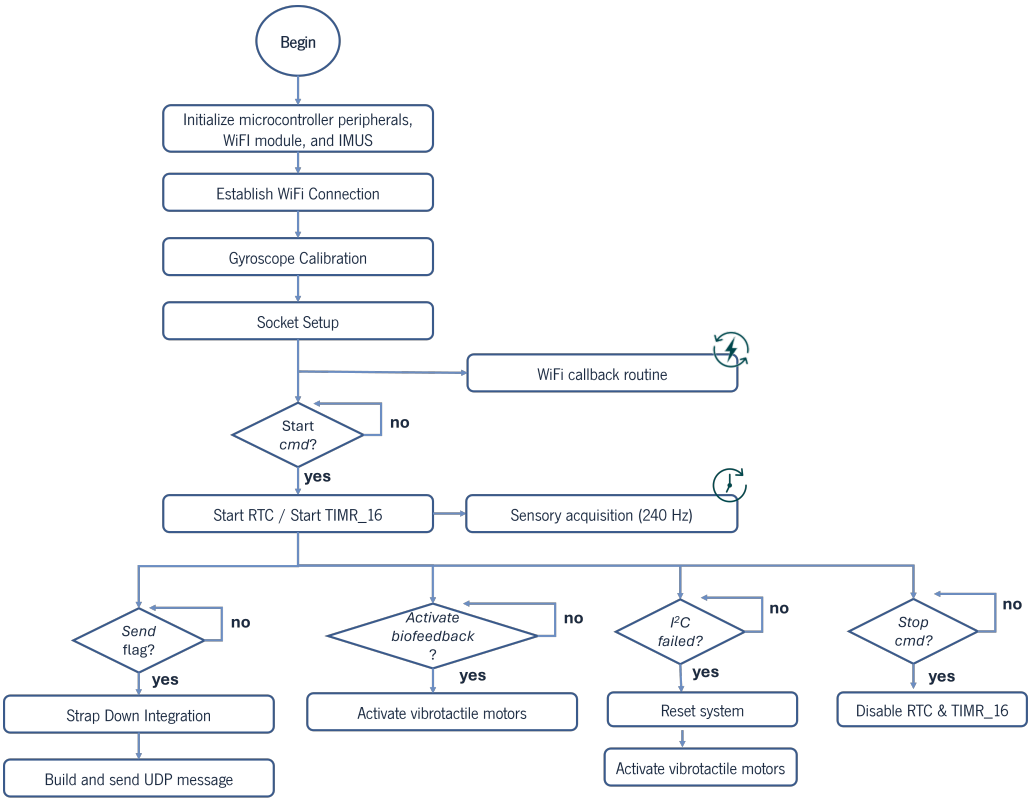


Figure 139: Low-level controller software routine. The circular arrow with a clock icon represents the periodic timer interrupt, while the circular arrow with a lightning bolt indicates an event-driven WiFi callback. *start cmd* is sent by the main station. *send flag* is set by the sensory acquisition routine.

to the network, communication is established with all nine IMUs (or a selected subset). A five-second static calibration routine is then executed, during which gyroscope data is sampled while the user maintains the n-pose. After calibration, a User Datagram Protocol (UDP) socket is initialized and bound to the IP address and designated port of the main station, establishing the communication channel.

After setup, the system enters the main program loop. This loop continuously monitors communication via the WiFi handler, which manages callbacks associated with specific WiFi events. These include connection establishment, disconnection, socket binding, message reception, and message transmission. The callback routine is also responsible for executing commands from the main station, such as activating or deactivating selected motors.

Upon reception of the `start_cmd` from the main station, the Real-Time Clock (Real Time Clock (RTC)) and Timer_16 are activated. This initiates a time-based interrupt routine that operates at 240 Hz and triggers a deterministic acquisition cycle where gyroscope data are read and bias-corrected. To maximize sampling frequency IMU acquisition is time-multiplexed across interrupts: each cycle reads three of the nine IMUs and applies accelerometer calibration, completing the full sensor array over five interrupt cycles. On every fifth interrupt, the RTC is read to provide a global timestamp. Once all sensors are updated and

time-aligned, a send flag indicates that a complete frame is ready. The interrupt routine remains bounded and non-blocking, avoiding communication and complex processing to preserve timing guarantees. When the main loop detects the send flag, it executes data processing and transmission. SDI is applied to five buffered gyroscope samples, reducing the output rate to 48 Hz while improving signal stability. The processor then assembles and transmits a UDP packet containing the message identifier, RTC timestamp, and complete sensor data (angular velocity, linear acceleration, magnetometer readings, and temperature) to the main station.

If a biofeedback activation command is received from the main station, it is handled by the WiFi callback routine, which sets a flag that triggers the activation of the corresponding vibrotactile motors.

To maintain robust communication between the processor and the IMUs, an I^2C monitoring subroutine continuously checks the communication status. If a failure is detected, the system attempts recovery. If the subroutine exceeds a predefined timeout, a system reset is triggered and all vibrotactile motors are briefly activated to warn the user of the system's reset. In turn, if a stop command is received from the main station, the RTC and timer interrupt routine are deactivated.

UDP Packet As previously described, the UDP packet transmitted from the low-level processor to the main station includes the packet index number, the gyroscope, accelerometer, magnetometer, and thermometer data from each IMU, as well as the corresponding message timestamp. Each value is encoded as a fixed-length byte sequence, rather than being converted to a string format, which would result in variable message sizes and increased data overhead.

By maintaining a fixed-length message, the system avoids the use of physical separators and simplifies the parsing process on the receiving end. This strategy also reduces the time required for packet construction and transmission, thereby preserving the system's performance in terms of sampling frequency. The resulting UDP packet has a fixed size of 249 bytes.

RTC Setup The Nucleo-144 STM32H723ZG board features an embedded RTC, driven by a 32.768 kHz external crystal oscillator. This RTC operates in a calendar format, supporting timekeeping down to hours, minutes, and seconds, and also enables subsecond resolution through configurable prescalers [283].

The timestamps transmitted alongside the sensor data must provide millisecond resolution to enable accurate synchronization with the main station, which uses an equivalent timestamp format.

The RTC configuration is based on two prescalers: an asynchronous (7-bit) and a synchronous (13-bit) prescaler. These prescalers divide the oscillator frequency to produce a 1 Hz clock for the calendar

timekeeping. According to the reference manual [283], the calendar clock frequency is given by:

$$f_{\text{CK_SPRE}} = \frac{f_{\text{RTCCLK}}}{(PREDIV_S + 1)(PREDIV_A + 1)} \tag{B.1}$$

where $f_{\text{CK_SPRE}}$ is the resulting calendar clock frequency, f_{RTCCLK} is the frequency of the low-speed external (LSE) oscillator (32.768 kHz), and $PREDIV_S$ and $PREDIV_A$ represent the synchronous and asynchronous prescaler values, respectively.

Subsecond resolution is derived from a separate subdomain controlled exclusively by the asynchronous prescaler. Its output frequency is given by:

$$f_{\text{CK_APRE}} = \frac{f_{\text{RTCCLK}}}{PREDIV_A + 1} \tag{B.2}$$

where f_{CK_APRE} is the subsecond domain frequency.

To achieve a subsecond resolution of 1 millisecond, $PREDIV_A$ was set to 1023. To configure the calendar clock frequency to 1 Hz, $PREDIV_S$ was initialized with a value of 31.

B.2 sEMG e-textile based electrodes

B.2.1 sEMG Acquisition Module: Android Application

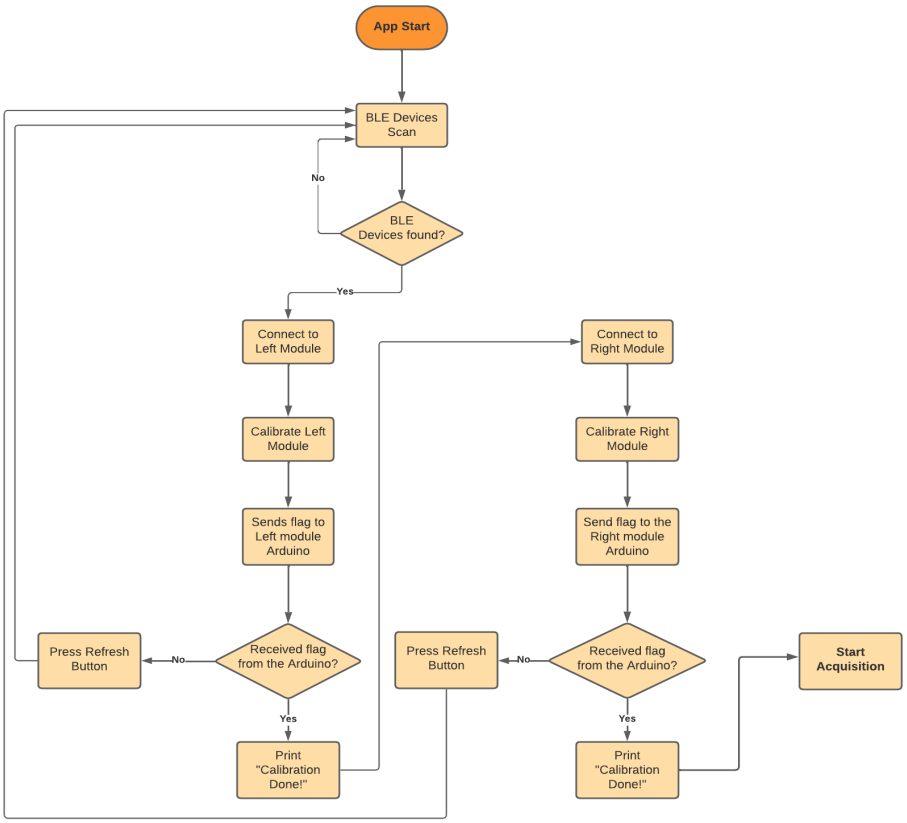


Figure 140: Android application flowchart.

Appendix C
Deep-learning based Pose Estimation

C.1 Dataset acquisition

This appendix reports supplementary information related to the experimental datasets used in Chapter 4.2.3. Tables 51 and 52 present the main anthropometric characteristics of the participants involved in the two data acquisition campaigns, corresponding to the Ergowear dataset (D1) and the MTw Awinda dataset (D2), respectively.

Table 51: Ergowear data acquisition (D1), participants' main anthropometric data.

Participant_ID	Gender (M/F)	Age (years)	Body mass (kg)	Body height (cm)
00	M	24	67	180
01	M	22	64	174
02	F	24	50	151
03	F	28	68	159
04	F	27	52	157
05	M	30	70	174
06	F	28	53	162
07	M	24	75	170
08	M	22	70	176
09	M	26	80	177
10	M	23	72	176
11	F	22	59	160
12	M	24	65	171
13	M	25	65	175
14	M	26	70	181
15	M	25	64	175
16	M	27	74	181
17	M	22	64	176
18	M	22	73	168
19	M	26	71	181
20	F	24	61	171
	15M, 6F	**25.0 (**± 2.3**)**	**66.0 (**± 7.8**)**	**171 (**± 8.6**)**

Table 52: MTwAwinda data acquisition (D2), participants' main anthropometric data. Some of the participants were also present in the first data collection.

Participant_ID	**Gender (M/F)**	**Age (years)**	**Body mass (kg)**	**Body height (cm)**
00	M	24	76	170
01	F	24	61	171
02	M	24	65	171
03	M	22	70	176
04	M	24	67	180
05	M	30	70	174
06	F	23	63	168
07	M	26	71	181
08	F	21	51	165
09	F	24	50	153
	6M, 4F	**24.2 (± 2.6)**	**64.4 (± 8.5)**	**171 (± 8.1)**

C.2 Pre-processing

Table 53: Descriptive analysis of MTwAwinda and Ergoaware datasets

Dataset	Sensor	Mean	Standard Deviation	Min	Max	Kurtosis	Skewness
Ergowear	Acc	-0.224	0.625	-21.878	17.614	4.303	0.134
	Gyr	-0.009	1.181	-37.392	34.139	16.09	-0.087
	Mag	-0.093	0.611	-37.651	50.417	9.839	0.173
MTwAwinda	Acc	-0.261	0.63	-21.145	17.211	5.798	-0.014
	Gyr	-0.008	1.49	-41.237	36.478	23.033	-0.287
	Mag	-0.236	0.449	-2.08	2.348	-0.871	0.509

C.3 Results

The following tables report the complete ablation study results for the Ergowear and MTw Awinda datasets. They provide a detailed quantitative comparison of model variants in terms of average QAD and inference time, complementing the analyses discussed in the main chapters

Table 54: Average QAD (°) and inference time (ms) obtained from the ablation studies on the Ergowear dataset. Performance improvements with respect to the baseline are shown in bold and underlined, slight performance decreases (less than 1°) are underlined, and larger performance decreases are reported in normal text.

Ablation study	LSTM		COMP LSTM		TGCN		TGCN COMP		LSTM MG detach	
	QAD (°)	Inference time (ms)	QAD (°)	Inference time (ms)	QAD (°)	Inference time (ms)	QAD (°)	Inference time (ms)	QAD (°)	Inference time (ms)
Baseline	**21.57 ± 14.97**	0.32 ± 0.01	**21.58 ± 18.89**	0.46 ± 0.01	**29.20 ± 21.43**	0.50 ± 0.01	**32.59 ± 23.60**	0.63 ± 0.01	**20.63 ± 16.45**	1.64 ± 0.02
No augmentation	22.20 ± 15.39	0.32 ± 0.01	21.70 ± 18.17	0.32 ± 0.01	29.73 ± 22.25	0.54 ± 0.01	34.85 ± 23.23	0.66 ± 0.01	**20.54 ± 16.38**	1.60 ± 0.02
Output (6D)	23.04 ± 16.82	0.72 ± 0.01	23.30 ± 21.44	0.84 ± 0.01	**23.06 ± 17.17**	0.72 ± 0.01	**25.18 ± 23.25**	0.85 ± 0.01	21.21 ± 16.35	1.98 ± 0.02
No magnetometer	36.97 ± 29.07	0.32 ± 0.01	33.95 ± 27.79	0.48 ± 0.01	41.15 ± 31.45	0.52 ± 0.01	42.41 ± 30.65	0.66 ± 0.01	–	–
Window (50)	**17.48 ± 11.01**	0.33 ± 0.01	27.44 ± 29.47	0.33 ± 0.01	30.08 ± 24.76	0.70 ± 0.11	46.56 ± 34.47	0.81 ± 0.06	22.70 ± 16.16	1.82 ± 0.06
Window (5)	46.08 ± 36.07	0.33 ± 0.01	84.60 ± 43.00	0.47 ± 0.01	–	–	–	–	23.46 ± 17.42	1.59 ± 0.02
Window (2)	69.76 ± 31.47	0.32 ± 0.01	84.75 ± 47.76	0.45 ± 0.01	–	–	–	–	23.11 ± 16.16	1.58 ± 0.02
Window (1)	109.41 ± 41.29	0.33 ± 0.01	95.80 ± 43.13	0.46 ± 0.01	–	–	–	–	23.17 ± 17.64	1.61 ± 0.01
MSE	30.80 ± 25.67	0.32 ± 0.01	33.46 ± 31.66	0.32 ± 0.01	–	–	–	–	23.53 ± 19.68	1.59 ± 0.03
qdist	22.12 ± 15.57	0.32 ± 0.01	**21.08 ± 17.64**	0.46 ± 0.01	–	–	–	–	21.23 ± 16.97	1.61 ± 0.01
RelQAD	35.28 ± 19.77	0.32 ± 0.04	**21.37 ± 17.72**	0.47 ± 0.01	–	–	–	–	20.78 ± 16.17	1.63 ± 0.02

Table 55: Average QAD° and inference time (ms) results of the ablation studies on MTwAwinda data performance improvements are in bold and underlined, slight performance decreases ($< 1°$) are underlined, and performance decreases are in normal text, all in relation to the baseline results.

Ablation study	LSTM		COMP-LSTM		TGCN		TGCN-COMP		LSTM-MG-detach	
	QAD (°)	Inference Time (ms)	QAD (°)	Inference Time (ms)	QAD (°)	Inference Time (ms)	QAD (°)	Inference Time (ms)	QAD (°)	Inference Time (ms)
Baseline	**27.97 ± 24.05**	0.34 ± 0.01	**24.30 ± 22.63**	0.49 ± 0.01	**37.21 ± 34.13**	0.60 ± 0.01	**38.68 ± 36.13**	0.74 ± 0.02	**8.08 ± 6.67**	1.63 ± 0.03
no augmentation	**27.65 ± 24.05**	0.57 ± 0.01	25.08 ± 23.95	0.72 ± 0.01	**35.20 ± 32.54**	0.33 ± 0.01	**36.00 ± 32.91**	0.48 ± 0.01	**8.05 ± 6.85**	1.64 ± 0.02
output (6D)	29.53 ± 26. 90	0.84 ± 0.01	31.4 ± 30.58	0.89 ± 0.21	30.23 ± 26.64	0.79 ± 0.03	**34.73 ± 31.42**	0.89 ± 0.01	**7.82 ± 6.76**	0.32 ± 0.01
no magnetometer	62.92 ± 39.69	0.36 ± 0.04	52.72 ± 37.88	0.50 ± 0.01	66.87 ± 40.04	0.59 ± 0.01	65.87 ± 41.60	0.73 ± 0.01	-	-
window (50)	**14.29 ± 12.17**	0.53 ± 0.04	98.18 ± 56.96	0.37 ± 0.04	**19.56 ± 17.20**	0.56 ± 0.01	**30.24 ± 25.83**	0.69 ± 0.01	**7.58 ± 6.72**	1.89 ± 0.05
window (5)	28.33 ± 30.83	0.34 ± 0.03	63.00 ± 48.63	0.47 ± 0.01	-	-	-	-	**7.80 ± 6.76**	1.84 ± 0.05
window (2)	36.26 ± 28.74	0.33 ± 0.01	47.77 ± 36.35	0.47 ± 0.01	-	-	-	-	**7.87 ± 6.76**	1.62 ± 0.01
window (1)	69.82 ± 40.09	0.34 ± 0.02	64.64 ± 40.19	0.49 ± 0.01	-	-	-	-	**7.93 ± 7.92**	1.66 ± 0.01
MSE	46.53 ± 36.94	0.36 ± 0.04	41.36 ± 35.59	0.49 ± 0.01	-	-	-	-	8.37 ± 7.00	1.66 ± 0.02
qdist	28.55 ± 24.24	0.35 ± 0.02	**23.86 ± 22.61**	0.48 ± 0.01	-	-	-	-	8.15 ± 6.85	1.68 ± 0.02
RelQAD	77.33 ± 45.35	0.35 ± 0.05	**24.11 ± 22.40**	0.49 ± 0.01	-	-	-	-	**7.97 ± 6.76**	1.67 ± 0.06

Appendix D
Human Activity Recognition

D.1 Feature Selection

This appendix displays part of the statistical analysis conducted to perform the feature selection of Section 4.3.3. Figure 142 presents the Kruskal-Wallis test results, and Figure 141 the heatmap of Spearman's correlation between the acquired signals.

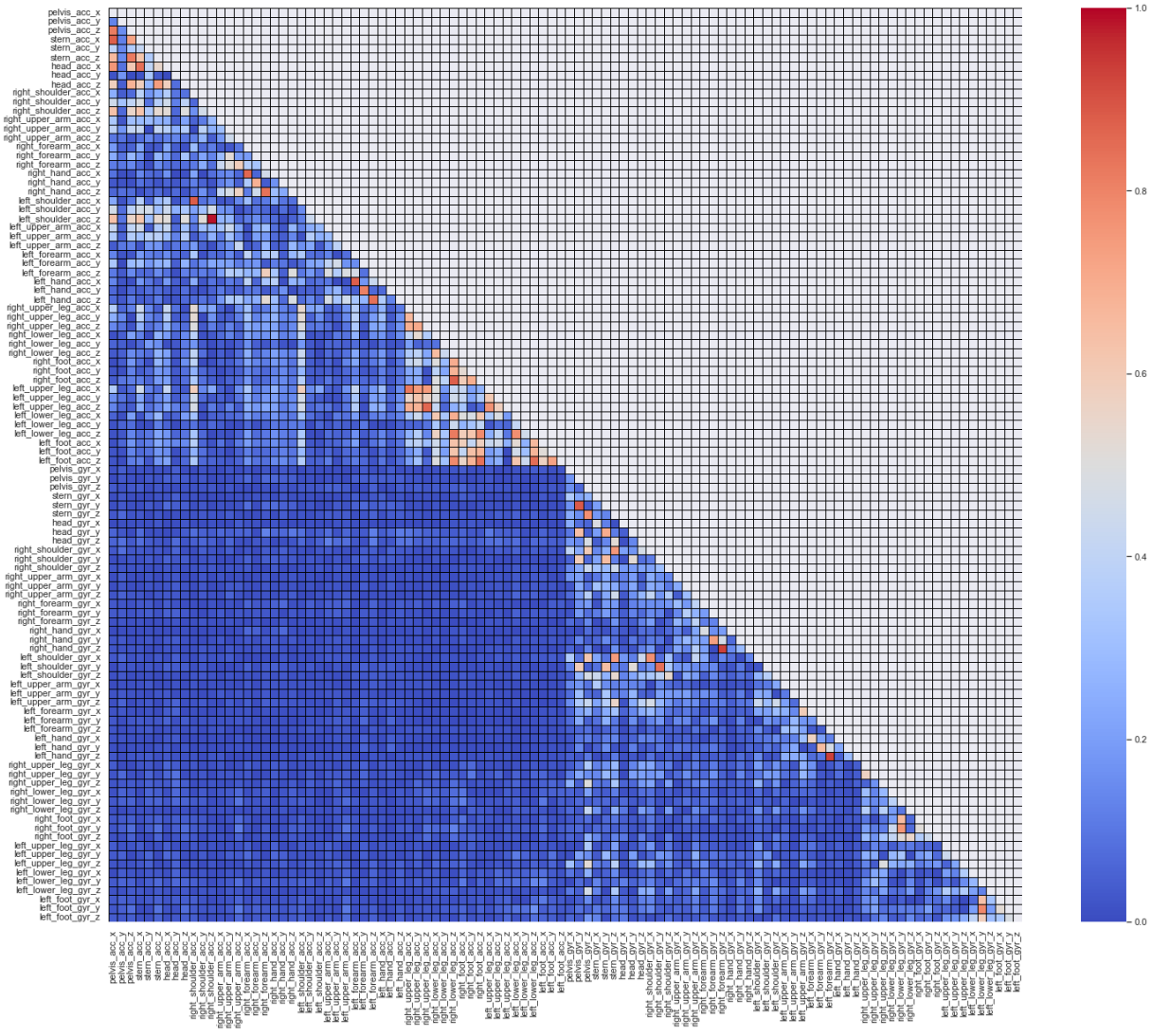


Figure 141: Spearman's correlation between features.

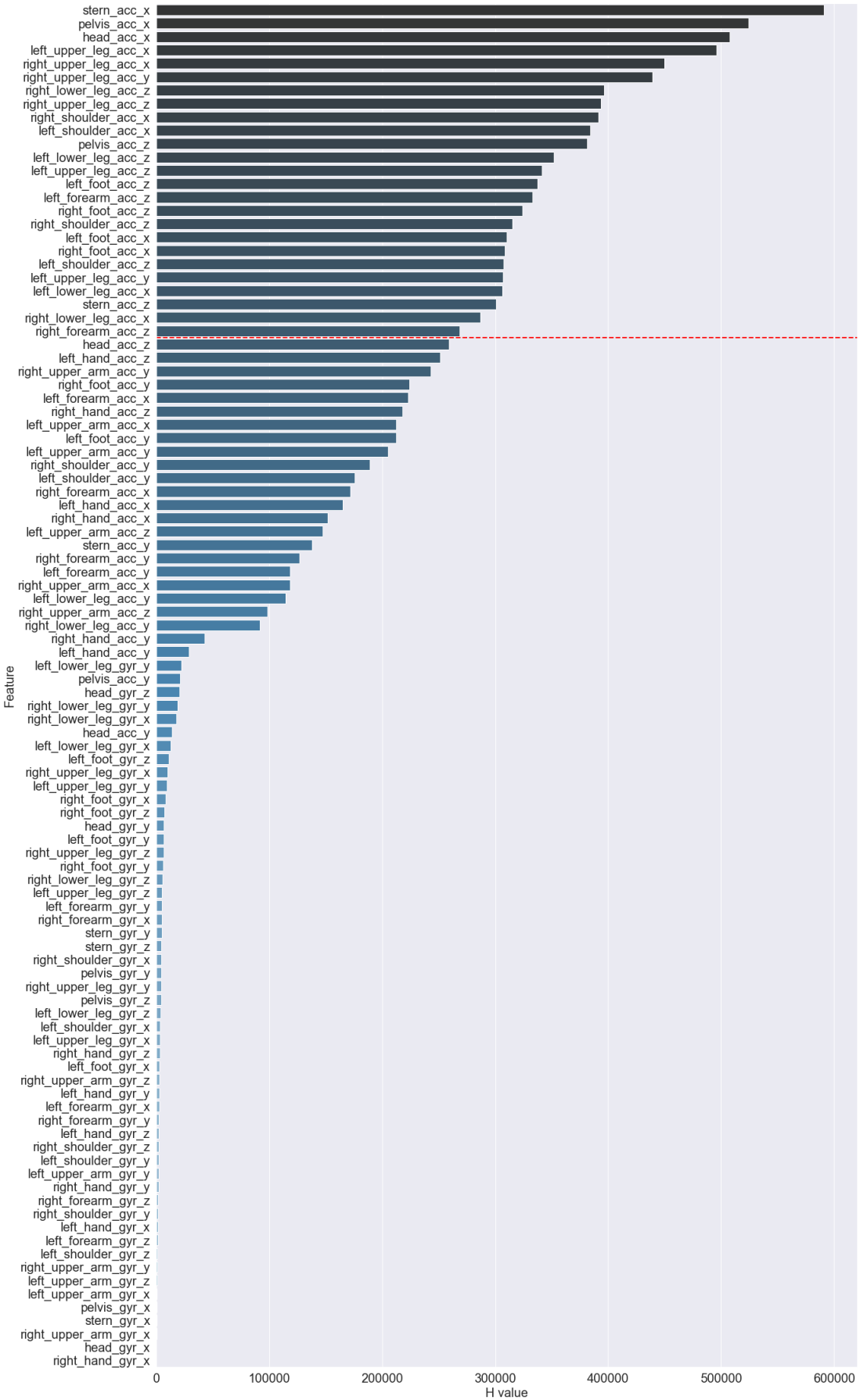


Figure 142: Kruskal-Wallis scores of the kinematic signals, with the red dashed line delimiting the top 25%.

D.2 Results

D.2.1 Hyperparameter Optimization Results

This appendix reports the top ten best performing models for each of the five architectures evaluated in this study, namely the CNN, LSTM, Transformer, hybrid CNN–LSTM, and hybrid CNN–Transformer models. For each architecture, the table summarises the hyperparameter configurations yielding the highest performance, providing a more complete view of the model ranking than the subset presented in the main text. These results complement the main analysis by offering detailed performance information for all top ranked configurations.

CNN

Table 56: Top 10 CNN models considering F1-score (their hyperparameters and validation metrics in the epoch with the lowest loss)

Hyperparameters subject to tuning							Epoch	Metrics				
LR	Batch Size	WS (s)	Overlap (%)	Layers	Filters[a]	Kernel Size		Loss	Accuracy (%)	F1-Score (%)	Precision (%)	Recall (%)
10^{-3}	64	1.5	0	4	128	3	47	0.3603	96.56	**94.72**	95.52	95.18
10^{-3}	64	1	50	5	64	3	46	0.2706	96.90	94.66	95.24	95.07
10^{-3}	32	1	50	5	64	5	46	0.2486	96.86	94.40	95.00	94.72
10^{-4}	32	1	50	5	64	5	37	0.4835	96.63	94.35	94.94	94.74
10^{-4}	32	1	50	5	128	5	43	0.4997	96.93	94.33	94.76	94.80
10^{-3}	32	1	50	4	128	3	37	0.2584	96.80	94.20	94.89	94.55
10^{-3}	64	1	15	4	64	3	30	0.2933	96.71	94.20	95.09	94.41
10^{-3}	32	1	50	4	64	3	31	0.2602	96.64	94.11	94.95	94.45
10^{-4}	32	1	30	5	64	3	46	0.5464	96.68	94.07	94.54	94.71
10^{-4}	64	1	50	5	128	5	43	0.6441	96.78	94.06	94.61	94.43

[a]Number of filters in the first convolutional layer

LSTM

Table 57: Top 10 LSTM models considering F1-score (their hyperparameters and validation metrics in the epoch with the lowest loss)

Hyperparameters subject to tuning					Epoch	Metrics				
LR	Batch Size	WS (s)	Overlap (%)	Layers		Loss	Accuracy (%)	F1-Score (%)	Precision (%)	Recall (%)
10^{-3}	64	1	50	2	33	0.3157	95.73	**92.45**	93.15	93.15
10^{-4}	32	1	50	3	39	0.5478	95.59	92.41	93.19	92.97
10^{-3}	32	1	50	3	32	0.3637	95.78	92.35	93.12	92.93
10^{-4}	32	1	50	2	37	0.5021	95.54	92.34	93.05	93.02
10^{-3}	64	1	0	2	25	0.3601	95.62	92.29	93.27	92.86
10^{-3}	64	1	50	3	37	0.3542	95.90	92.25	93.10	92.87
10^{-3}	32	1	50	2	22	0.3445	95.37	92.24	93.06	92.92
10^{-3}	32	1	30	2	27	0.3147	95.52	92.14	92.87	92.95
10^{-3}	32	1.5	50	3	37	0.3405	95.34	92.06	93.04	92.74
10^{-3}	32	1	30	3	34	0.3624	95.31	91.98	92.71	92.77

Transformer

Table 58: Top 10 transformer models considering F1-score (their hyperparameters and validation metrics in the epoch with the lowest loss)

Hyperparameters subject to tuning							Epoch	Metrics				
LR	Batch Size	WS (s)	Overlap (%)	Layers	Heads	Embedding Size		Loss	Accuracy (%)	F1-Score (%)	Precision (%)	Recall (%)
10^{-4}	64	1	50	8	8	256	49	1.5999	96.19	**93.66**	93.95	94.43
10^{-3}	64	1	50	2	16	256	39	0.2699	96.28	93.61	93.74	94.31
10^{-4}	64	1	50	5	4	256	48	1.2780	96.24	93.59	93.88	94.19
10^{-4}	64	1	50	8	16	256	49	1.6033	96.19	93.58	93.68	94.43
10^{-4}	64	1	0	8	16	256	49	2.7820	95.98	93.57	94.29	94.05
10^{-4}	32	1	50	5	16	256	49	0.7277	96.38	93.51	93.94	94.16
10^{-3}	64	1	50	8	8	256	49	0.3564	96.33	93.51	93.88	94.02
10^{-4}	64	1	50	2	16	256	49	0.7740	96.13	93.29	93.56	93.90
10^{-4}	32	0.5	50	8	16	256	40	0.5181	96.37	93.29	93.85	93.72
10^{-4}	64	1	50	8	4	256	49	1.6419	96.14	93.24	93.71	93.84

Hybrid CNN-LSTM

Table 59: Top 10 hybrid CNN-LSTM models considering F1-score (their hyperparameters and validation metrics in the epoch with the lowest loss)

Hyperparameters subject to tuning								Epoch	Metrics				
LR	Batch Size	WS (s)	Overlap (%)	Convolutional Layers	Filters[a]	Kernel Size	LSTM Layers		Loss	Accuracy (%)	F1-Score (%)	Precision (%)	Recall (%)
10^{-4}	64	1	0	5	128	5	2	46	0.9359	96.76	**95.06**	95.84	95.32
10^{-3}	64	1	15	5	64	5	2	27	0.3310	97.07	94.68	95.36	94.94
10^{-4}	64	1	15	5	128	5	2	46	0.8312	96.68	94.59	95.29	94.90
10^{-4}	64	1	50	5	64	5	3	37	0.7023	96.77	94.56	95.12	94.98
10^{-3}	64	1	50	5	64	5	2	31	0.2996	96.83	94.51	95.06	94.99
10^{-3}	64	1.5	0	5	64	3	3	36	0.3777	96.28	94.47	94.98	95.46
10^{-4}	32	1.5	50	4	128	5	2	48	0.6181	96.88	94.46	94.87	95.13
10^{-3}	64	1	15	4	64	3	2	49	0.3239	96.86	94.44	95.34	94.47
10^{-3}	64	1	50	5	64	3	2	31	0.2971	96.84	94.42	95.12	94.79
10^{-4}	32	1	50	5	128	5	3	37	0.5259	96.93	94.35	94.76	94.84

[a]Number of filters in the first convolutional layer

Hybrid CNN-Transformer

Table 60: Top 10 CNN–Transformer models considering F1-score (their hyperparameters and validation metrics in the epoch with the lowest loss)

Hyperparameters subject to tuning										Epoch	Metrics				
LR	Batch Size	Window Size	Overlap (%)	Conv. Layers	Filters[a]	Kernel Size	Encoder Layers	Heads	Embedding Size		Loss	Accuracy (%)	F1-Score (%)	Precision (%)	Recall (%)
10^{-4}	64	20	0	5	128	5	8	16	128	48	2.0413	96.82	**95.39**	96.04	95.75
10^{-4}	64	20	0	5	128	5	2	8	256	48	1.3407	96.90	95.34	95.97	95.73
10^{-4}	64	20	0	5	128	5	8	4	128	48	2.0471	96.90	95.29	96.05	95.60
10^{-4}	64	20	15	5	128	5	8	4	256	49	1.6532	96.90	95.16	95.63	95.38
10^{-3}	64	20	50	5	128	5	2	4	128	49	0.2424	97.03	95.10	95.74	95.34
10^{-4}	64	20	0	5	128	5	4	8	128	48	1.6543	96.82	95.08	95.60	95.51
10^{-4}	64	20	15	5	128	5	2	8	128	49	1.1761	96.92	95.04	95.68	95.19
10^{-4}	64	20	0	5	128	5	2	4	128	48	1.2236	96.76	94.99	95.61	95.46
10^{-4}	64	20	0	5	128	5	2	8	128	48	1.2227	96.76	94.95	95.61	95.40
10^{-4}	64	20	0	5	128	5	2	4	256	48	1.3425	96.84	94.94	95.59	95.28

[a]Number of filters in the first convolutional layer

Appendix E
Ergoreport

E.1 Implementations Details

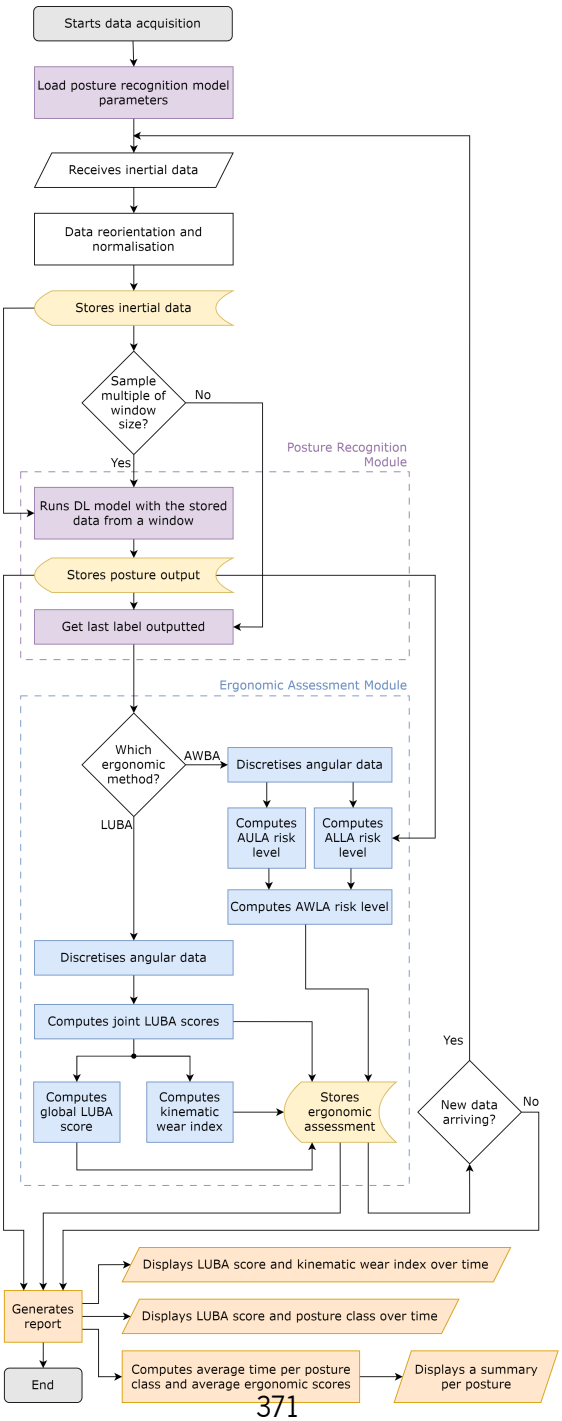
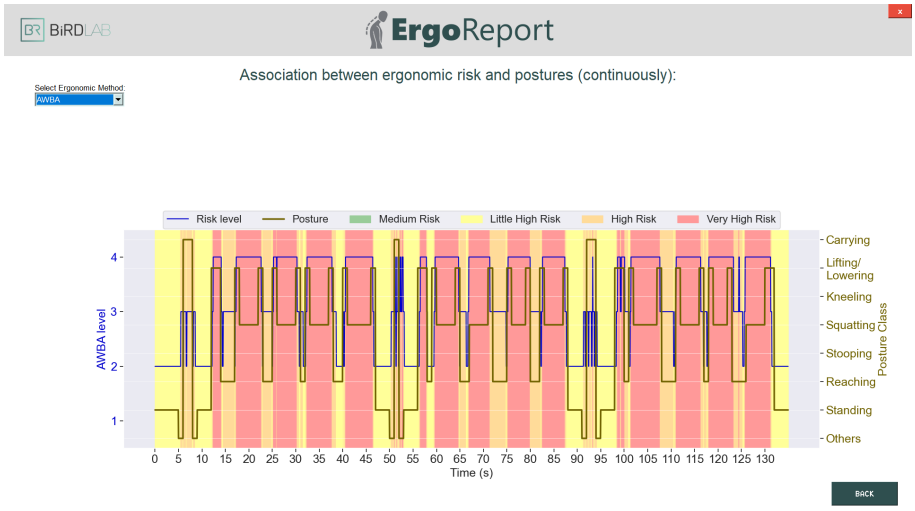


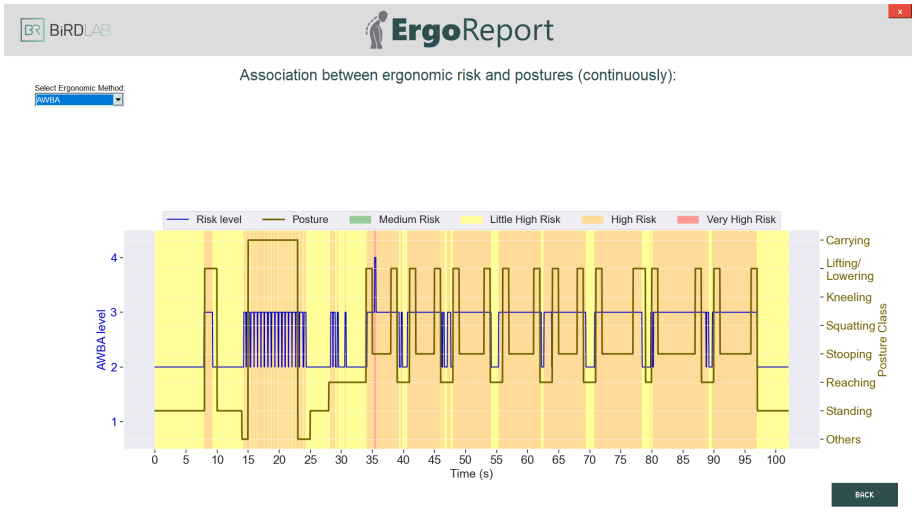
Figure 143: Flowchart of the proposed ergonomic assessment framework.

E.2 Association between Ergonomic Risk and Postures

This appendix provides the visual examples of the AWBA-based ergonomic assessment generated by the ErgoReport. Figure 144 illustrates the temporal evolution of AWBA risk levels and the corresponding postures adopted during representative agriculture and construction trials.



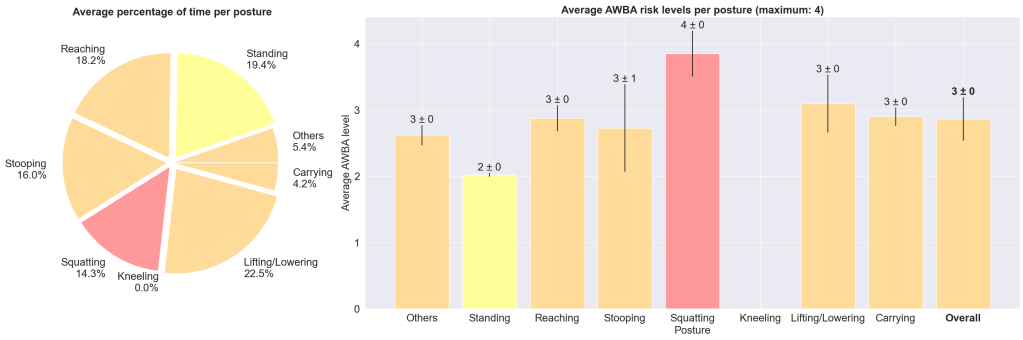
(a)



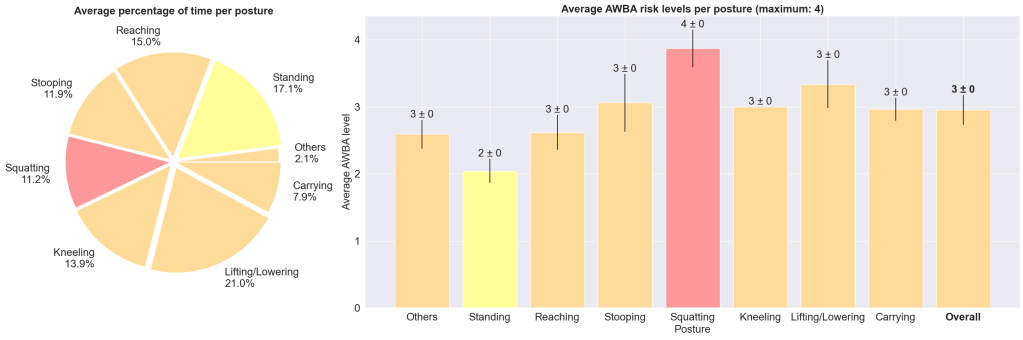
(b)

Figure 144: AWBA risk levels (on a scale from 1 to 4) and postures adopted during (**a**) an agriculture and (**b**) a construction trial.

Figure 145 presents aggregated summary reports, showing the average proportion of time spent in each posture and the corresponding mean AWBA risk levels across all trials for both task types.



(a)



(b)

Figure 145: Summary report with the average percentage of time spent at each posture and averages of the mean AWBA risk levels (on a scale from 1 to 4) for each posture, considering all (**a**) agriculture and (**b**) construction trials. Note that the time percentages do not have to sum to 100%, as these are the averages of all trials. The colours refer to the risk levels, as defined in Table 29.

Appendix F
Fatigue-based Control Strategy

F.1 Methods

F.1.1 Participants

Table 61: Participants' anthropometrics, exercise habits, and caffeine intake for Dataset 1.

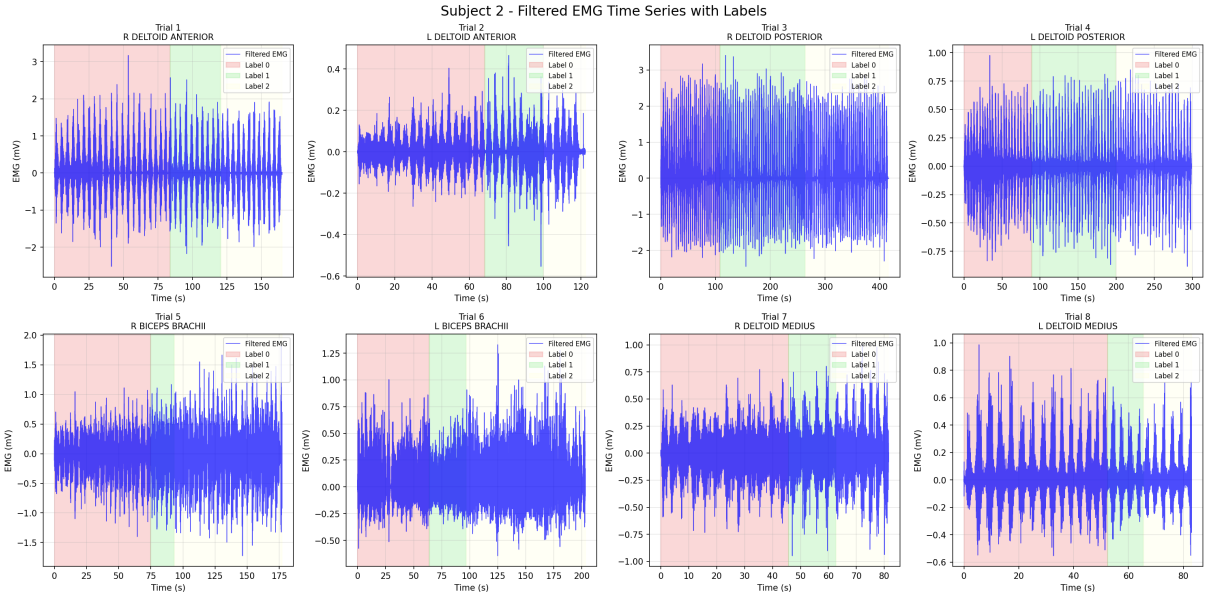
ID	Gender	Age (years)	Height (cm)	Weight (kg)	Workouts (per week)	Coffee (today)
S01	F	27	170	63.5	3	Yes
S02	F	23	153.5	46.8	2	Yes
S03	M	22	184	83	4	Yes
S04	F	23	167	57	6	Yes
S05	M	33	174	68	2	Yes
S06	M	24	177	77	5	Yes
S07	M	20	188	84	2	No
S08	M	27	174	58	2	Yes
S09	F	23	162	65	4	Yes
S10	F	25	163	56	0	Yes
S11	M	22	165	60	6	No
S12	M	20	181	72	0	No
S13	M	22	188	69	0	No

Table 62: Anthropometric characteristics of the participants included in Dataset 2.

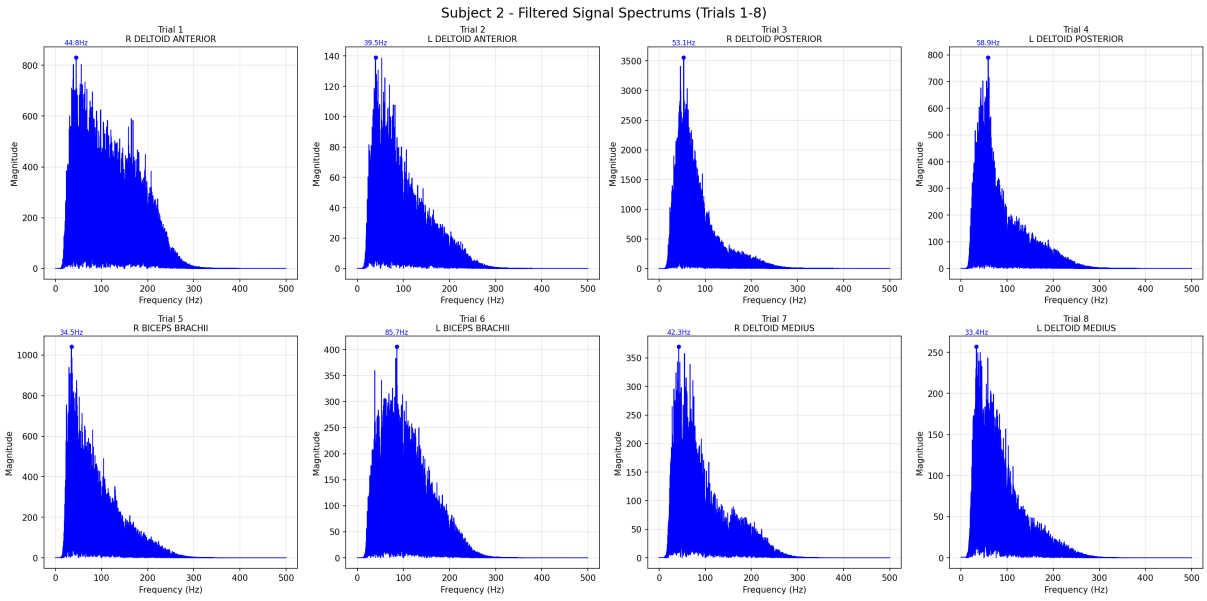
ID	Weight (kg)	Height (cm)	Age (years)
14	64.00	171	28
15	60.00	174	26
16	71.00	182	19
17	65.00	180	26
18	45.00	165	22
19	66.00	173	29
20	66.00	173	31
21	67.44	163	29
22	72.00	180	28
23	70.00	180	19

F.1.2 Data Preprocessing and Feature Selection

Filtered sEMG data for all 8 trials corresponding to the uni-articular movements of subject 2: (a) the time-domain signals and (b) the corresponding frequency spectra.



(a) Filtered sEMG signals in the time domain. Red, green, and white backgrounds indicate self-perceived fatigue levels (high, medium, low).



(b) Filtered sEMG signals in the frequency domain.

Figure 146: Filtered sEMG signals for Subject 2 across eight uni-articular movements: (a) time-domain with fatigue labels, and (b) frequency-domain spectra.

Table 63 summarizes the extracted sEMG features and reports the corresponding mathematical definitions used for their computation.

Table 63: sEMG feature extraction methods.

Feature	**Abbreviation**	**Formula**
Time-domain features		
Root Mean Square	RMS	$\sqrt{\frac{1}{N}\sum_{n=1}^{N} x[n]^2}$
Mean Absolute Value	MAV	$\frac{1}{N}\sum_{n=1}^{N} \lvert x[n]\rvert$
Integrated EMG	IEMG	$\sum_{n=1}^{N} \lvert x[n]\rvert$
Variance		$\frac{1}{N-1}\sum_{n=1}^{N}(x[n]-\bar{x})^2$
Maximum Amplitude		$\max_{1\le n\le N} \lvert x[n]\rvert$
Signal Energy		$\sum_{n=1}^{N} x[n]^2$
Zero Crossing		$\sum_{n=1}^{N-1} \mathbf{1}\{x[n]x[n+1]<0\}$
Slope Sign Change	SSC	$\sum_{n=2}^{N-1} \mathbf{1}\{(x[n]-x[n-1])(x[n+1]-x[n])<0\}$
Interquartile Range	IQR	$Q_{75}-Q_{25}$
MAV Slope		$\frac{\text{MAV}_{\text{end}}-\text{MAV}_{\text{start}}}{\Delta t}$
Frequency-domain features		
Mean Frequency	MNF	$\frac{\sum_{j=1}^{M} f_j P_j}{\sum_{j=1}^{M} P_j}$
Median Frequency	MDF	f_k such that $\sum_{j=1}^{k} P_j = \frac{1}{2}\sum_{j=1}^{M} P_j$
Peak Frequency		f_k such that $P_k = \max(P_1,\ldots,P_M)$
High-to-low Frequency Ratio		$\frac{\sum_{f_j\in[70,150]\text{ Hz}} P_j}{\sum_{f_j\in[20,60]\text{ Hz}} P_j}$
Entropy features		
Shannon Entropy		$-\sum_{i=1}^{B} p_i \log_2(p_i)$

Note. For time-domain features, $x[n]$ denotes the EMG signal amplitude at sample n, N is the number of samples in the analysis window, and $\bar{x}$ is the signal mean. For frequency-domain features, P_j denotes the power spectral density at frequency f_j, computed using the Fast Fourier Transform (FFT), and M is the number of frequency bins. For Shannon entropy, p_i is the probability of signal values falling in histogram bin i, with B bins determined by the Freedman–Diaconis rule.

F.2 Results

F.2.1 EDA

Figure 147 presents the trial ranking based on the significance rate across all feature combinations.

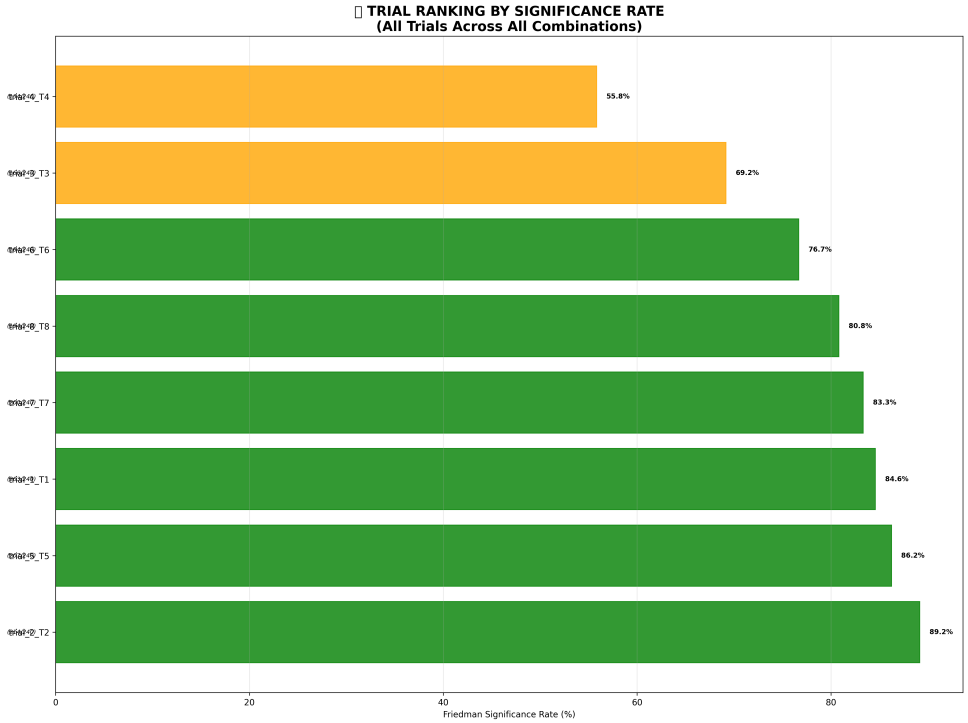
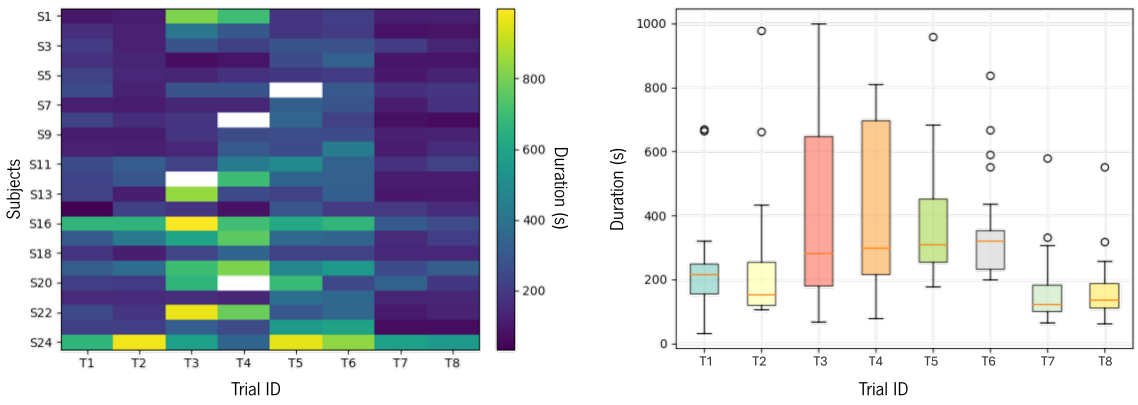


Figure 147: Trial ranking based on the significance rate computed across all window lengths and overlap combinations.

Figure 148 presents the duration of each trial in seconds, across all subjects.



(a) Trial duration heatmap across subjects

(b) Duration distribution across trials

Figure 148: Trial durations per trial.

F.2.2 Hyperaparameter Optimization Results

BiLSTM with Time Attention

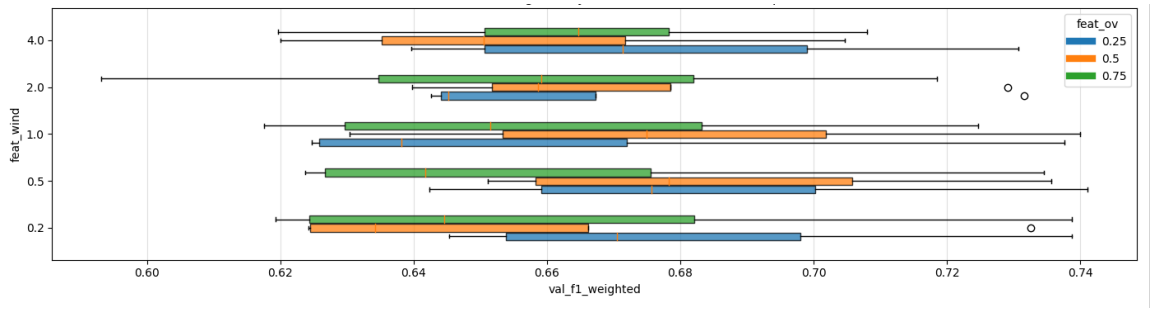
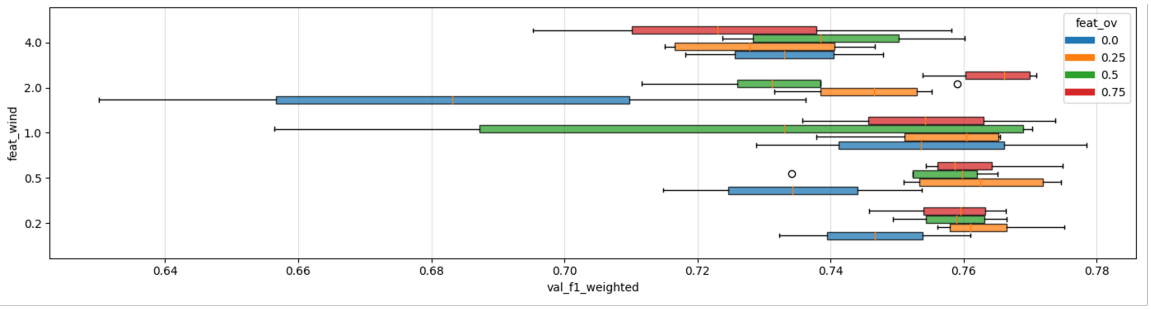
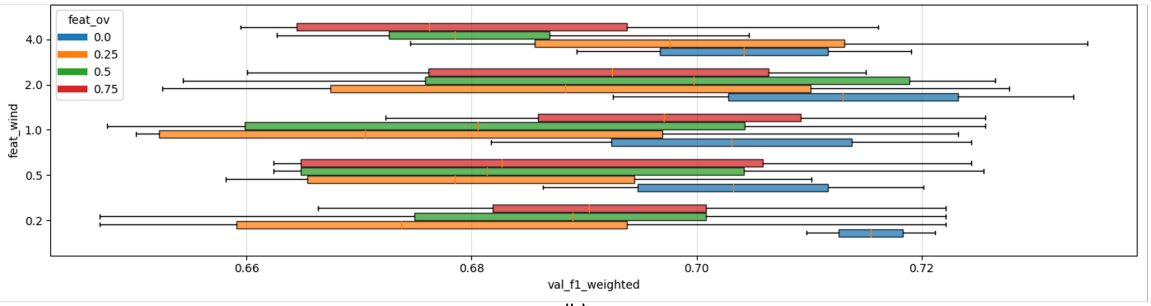


Figure 149: Distribution of validation weighted F1-scores for the BiLSTM with time attention (batch size = 16) across feature window sizes and overlaps.

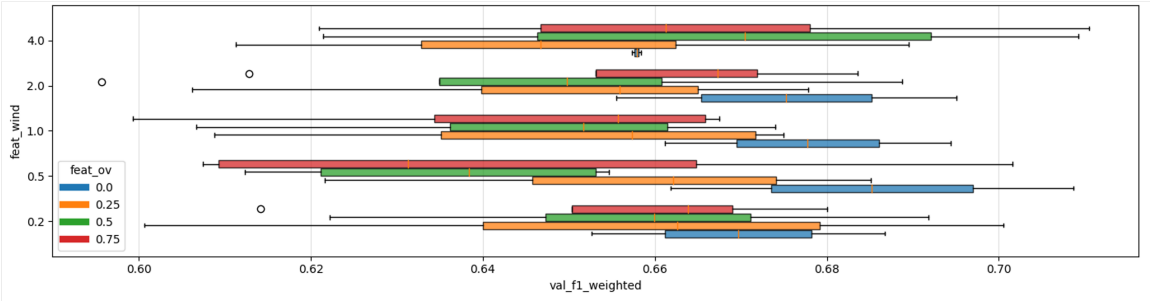
Frequency-Time BiLSTM with Attention



(a)



(b)



(c)

Figure 150: Validation weighted F1-score distributions of the Frequency–Time BiLSTM with attention as a function of feature window size and overlap for (a) batch size 1, (b) batch size 16, and (c) batch size 32.

F.3 Usability Questionnaire

The questionnaire administered to participants after completing the validation protocol is shown in Figure 151.

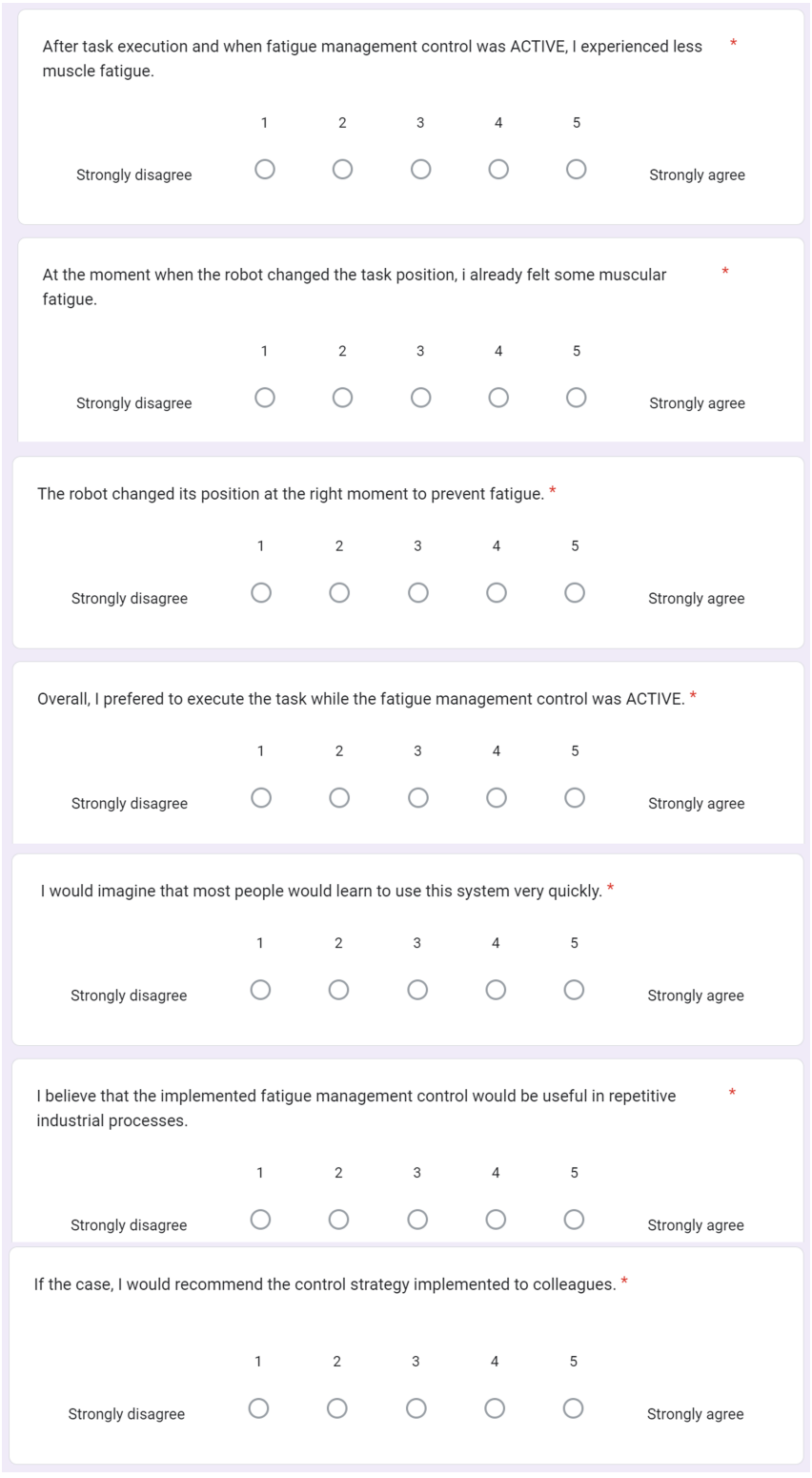


Figure 151: Questionnaire applied to participants after the validation protocol.

Appendix G
Learning Human Specificities based on Ergonomics and Physical restrictions

G.1 Results

G.1.1 Real-World Deployment of the Pre-Trained Model

Q-learning

Table 64 reports the per-trial performance metrics obtained from the execution of the pre-trained Q-Learning models in the real-world environment for both discretization levels. For each trial, the reward, average ergonomic risk, and average pain risk are presented, along with the corresponding simulation baseline. The table also reports the average and standard deviation across the ten real-world trials, as well as summary statistics for the number of steps, traveled distance, and transport time.

Table 64: Metrics obtained from the Q-Learning (both discretization levels) pre-trained models' execution in the real-world environment for participant 1.

Trial	**Discretization = 10 cm**			**Discretization = 6.25 cm**		
	Reward	Erg.[1]	Pain[2]	Reward	Erg.[1]	Pain[2]
simulation	***-70.37***	***2.00***	***0***	***-115.40***	***2.039***	***0***
1	-174.83	2.307	0.162	-257.00	2.282	0.107
2	-111.31	2.308	0.380	-242.79	2.418	0.072
3	-112.65	2.331	0.000	-235.17	2.399	0.000
4	-100.77	2.253	0.000	-239.00	2.376	0.000
5	-115.31	2.310	0.000	-237.15	2.413	0.000
6	-127.10	2.306	0.000	-235.49	2.392	0.000
7	-117.18	2.341	0.290	-239.02	2.419	0.000
8	-203.21	2.353	0.462	-239.10	2.417	0.178
9	-189.56	2.310	0.298	-236.82	2.432	0.178
10	-190.66	2.301	0.448	-254.68	2.404	0.022
Average	-144 ± 40.23	2.312 ± 0.027	0.137 ± 0.2	-249.26 ± 26.51	2.404 ± 0.049	0.029 ± 0.062

Avg. Steps / Dist. 10 cm: 9 steps / 1.066 m 6.25 cm: 15 steps / 1.155 m
Avg. Transport Time 10 cm: 24 s 6.25 cm: 33 s

[1] Average ergonomic risk levels obtained from the model's execution;
[2] Average pain risk levels obtained from the model's execution.

DQN

Table 65 reports the performance metrics obtained from the execution of the pre-trained DQN model in the real-world environment. For each trial, the reward, average ergonomic risk, average pain risk, number of steps, and traveled distance are presented, together with the corresponding simulation baseline. The table also includes the mean and standard deviation computed over the ten real-world trials, as well as the average transport time.

Table 65: Metrics obtained from the DQN pre-trained model's execution in the real-world environment for participant 1.

Trial	**Reward**	**Erg. Risk**[1]	**Pain Risk**[2]	**Steps**[3]	**Distance**[4]
Simulation	***66.98***	***2.124***	***0***	***5***	***0.921***
1	-63.520	2.374	0.551	5	0.895
2	-55.560	2.152	0.582	5	0.910
3	-55.780	2.059	0.728	6	0.938
4	-56.690	2.149	0.025	5	0.894
5	-56.390	2.106	0.438	5	0.920
6	-58.640	2.222	0.478	5	0.876
7	-59.670	2.213	0.721	6	0.919
8	-80.270	2.044	0.183	4	0.833
9	-55.890	2.213	0.739	5	0.916
10	-78.630	2.092	0.100	5	0.877
Average	-62.104	2.167 ± 0.094	0.57 ± 0.259	5.3 ± 0.483	0.90 ± 0.019

Avg. Transport Time 16.5 ± 1.080 seconds

[1] Average ergonomic risk levels obtained from the model's execution;
[2] Average pain risk levels obtained from the model's execution;
[3] Steps taken; [4] Object's traveled distance.

Figure 152 presents the metrics' evolution resulting from the fine-tuning of DQN's pre-trained model is presented in Figure 62. The figure reports the progression of the reward, average ergonomic risk, average pain risk, number of steps per episode, episode at which reward convergence begins, and the average Huber loss, all evaluated at the end of each episode.on.

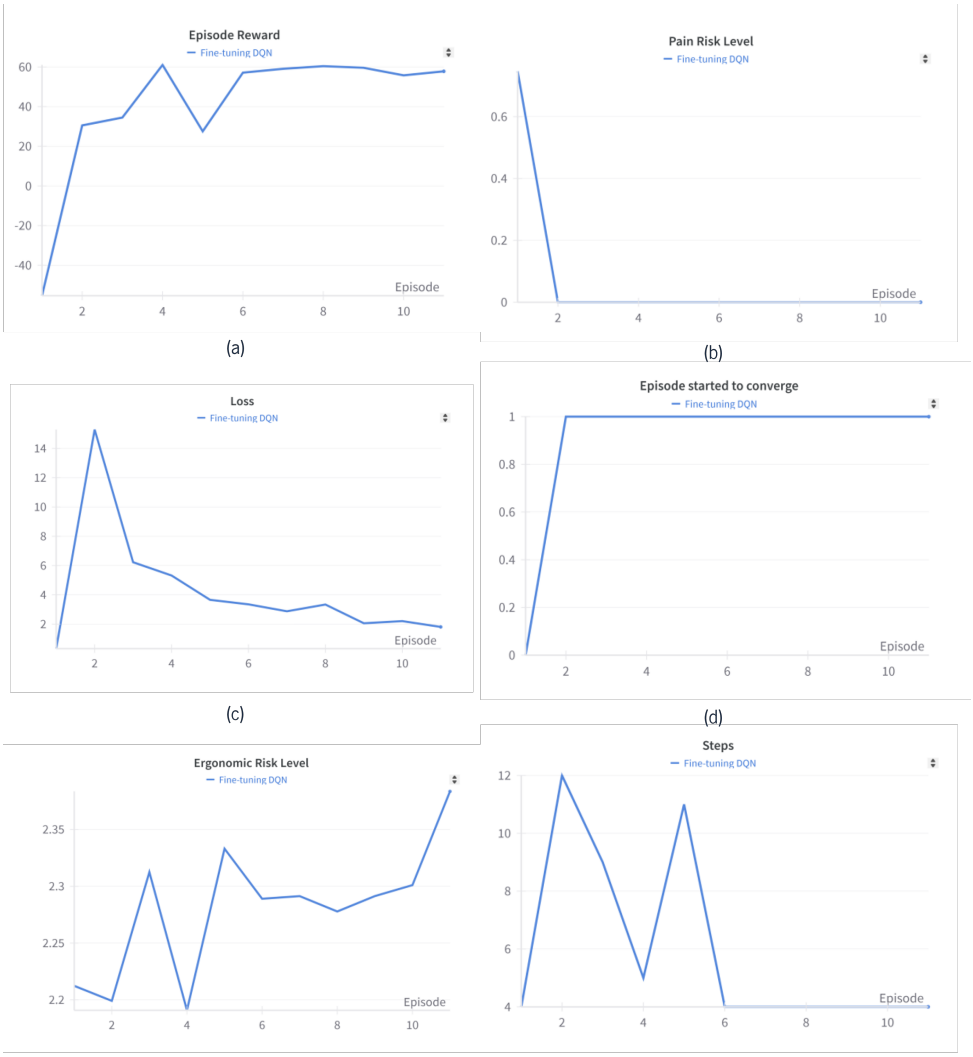


Figure 152: Evolution of key metrics during the fine-tuning of the DQN pre-trained model in real-world environment. The metrics include a) reward obtained per episode, b) average pain and e) ergonomic risk levels, c) average Hubber loss, d) episode in which the reward metric started to converge and f) steps taken per episode. All metrics were monitored at the end of each episode.

Appendix H
Task Quality Based Control Strategy

H.1 Results

H.1.1 Performance Metrics

Table 66: Average of the performance metrics for the 10 participants during square wave, sine wave and spiral line trajectories

Trajectory	Metric	UR10e Free-drive	Free-drive	Guided-drive
Square Wave	Force (N)	26.77 ± 2.87	6.56 ± 0.75	7.22 ± 2.18
	Error (mm)	7 ± 2.3	5.3 ± 2.3	4.2 ± 1.8
	Time (s)	48.07 ± 15.68	28.63 ± 7.92	33.4 ± 8.08
	DELTA (%)	2.78 ± 0.93	2.23 ± 0.72	2.69 ± 0.92
	DELTM (%)	20.08 ± 9.3	18.85 ± 8.02	15.39 ± 8.29
	BIC (%)	3.08 ± 1.72	2.92 ± 2.02	3.17 ± 1.23
	TRIC (%)	17.69 ± 11.42	16.44 ± 8.93	13.3 ± 8.58
Sine Wave	Force (N)	27.03 ± 4.52	5.74 ± 0.07	7.69 ± 3.35
	Error (mm)	7.2 ± 2.4	5.3 ± 3	4.3 ± 2.1
	Time (s)	40.54 ± 7.5	22.54 ± 6.14	35.55 ± 7.41
	DELTA (%)	2.51 ± 0.93	2.21 ± 0.87	2.67 ± 0.82
	DELTM (%)	10.92 ± 6.13	10.62 ± 5.8	10.89 ± 5.68
	BIC (%)	2.42 ± 1.06	2.49 ± 1.29	3.22 ± 1.74
	TRIC (%)	9.79 ± 5.74	9.7 ± 6.35	9.93 ± 6.6
Spiral Line	Force (N)	20.99 ± 1.16	6.22 ± 0.52	7.54 ± 1.89
	Error (mm)	5.9 ± 1	5.8 ± 2	3.9 ± 1.4
	Time (s)	58.47 ± 14.12	44.33 ± 13.72	57.72 ± 22.14
	DELTA (%)	2.66 ± 1.04	2.03 ± 0.93	3 ± 1.04
	DELTM (%)	12.04 ± 6.47	11.51 ± 6.02	11.63 ± 5.93
	BIC (%)	2.59 ± 1.45	2.05 ± 1.11	2.7 ± 1.32
	TRIC (%)	9.78 ± 5.61	9.63 ± 5.88	9.72 ± 6.5