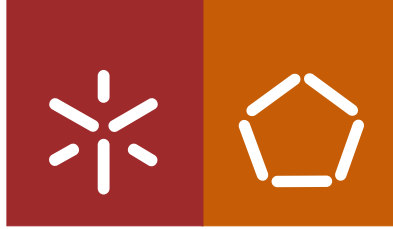


Universidade do Minho
Escola de Engenharia

Marco André Ferreira Rodrigues

**Novos Paradigmas de Interface de Utilizador
para Aplicações na Área da Saúde**



Universidade do Minho

Escola de Engenharia

Marco André Ferreira Rodrigues

Novos Paradigmas de Interface de Utilizador para Aplicações na Área da Saúde

Dissertação de Mestrado
Mestrado Integrado em Engenharia Biomédica
Informática Médica

Trabalho realizado sob a orientação do
Professor Doutor Vitor Manuel Rodrigues Alves

Outubro de 2012

É AUTORIZADA A REPRODUÇÃO PARCIAL DESTA DISSERTAÇÃO APENAS PARA EFEITOS DE INVESTIGAÇÃO, MEDIANTE DECLARAÇÃO ESCRITA DO INTERESSADO, QUE A TAL SE COMPROMETE;

Universidade do Minho, ___/___/_____

Assinatura: _____

Aos meus pais

Agradecimentos

Uma dissertação de mestrado envolve muito mais do que um estudo de determinada temática. Envolve objetivos, dedicação e pessoas. Desta forma, todo trabalho que aqui apresento é mais do que o resultado do meu esforço durante os últimos meses. É, acima de tudo, o esforço de todos os que me acompanharam nesta jornada.

Ao Professor Doutor Victor Rodrigues Alves queria fazer um agradecimento especial, não só pela orientação da tese, mas pelo tempo que guardou para me aconselhar, desde a escolha do tema, à redação da dissertação.

Aproveito, também, para agradecer ao Professor Doutor José Neves pela atenção e cooperação que demonstrou e agradecer particularmente o contributo que deu e dá ao ramo da Informática Médica.

Obrigada, também, à Sofia Gomes que colaborou comigo na correção ortográfica da presente dissertação.

De igual modo, agradeço à minha família e aos meus amigos por serem o suporte que eu precisei durante todo este processo.

O meu sincero obrigado a todos.

Resumo

As instituições de saúde vivem atualmente num ambiente de crescente densidade de informação, sendo esta uma área de intensa transferência eletrónica de dados. Como resultado, tem-se recorrido cada vez mais às interfaces de utilizador como forma de auxílio no processamento de dados. Tratando-se de uma questão inovadora, o surgimento de novas tecnologias de informação e comunicação tem sido um motor para o desenvolvimento de novos tipos de interfaces de utilizador.

Com o aumento do volume de informação, tem surgido problemas no que diz respeito ao seu processamento. Cada vez mais são exigidos que os serviços sejam prestados com maior eficiência, implicando maior rapidez na obtenção de dados. É neste contexto que surge a necessidade de direcionar os novos avanços no campo da saúde, tendo como linha orientadora a busca de um serviço mais eficiente e com maior qualidade. Neste sentido surge a necessidade de optimização dos processos de acesso e utilização dessa informação, alterando a forma como a informação sobre os utentes é obtida. É, essencialmente, nestes pontos que se foca esta dissertação.

Adicionalmente, e do ponto de vista funcional, pode dizer-se que estes desenvolvimentos apresentam características favoráveis à prevenção e ao controlo de infeções hospitalares, reduzindo a necessidade de contato direto entre os objetos, o que leva, à diminuição da propagação das ditas infeções.

Com a concretização deste estudo procura-se avaliar as potencialidades do reconhecimento gestual aplicado às interfaces de utilizador, na sua implementação na área específica da saúde.

Paralelamente desenvolveu-se um protótipo destinado aos utentes que frequentem as instituições de saúde. Este protótipo teve como objetivo a validação das interfaces de utilizador analisadas, considerando a utilização das tecnologias recentemente introduzidas no mercado. Foi dada especial atenção a interfaces de utilizador sem contato entre dispositivos e utilizadores (e.g. utilizando a tecnologia do *Kinect* da *Microsoft*).

No final é apresentado um estudo estatístico relativo à avaliação por parte dos utilizadores da interface do protótipo desenvolvido, onde se conclui a funcionalidade da utilização de gestos se revelou intuitiva e de fácil execução.

Abstract

Health institutions are currently living in an environment of increasing information density, being considered an area of intense electronic transfer of data. As a result, there has been an increasing use of the user interfaces in order to aid data processing. Since this is a matter innovative, the emergence of new information and communication technologies has been a catalyst for the development of new types of user interfaces.

With the increasing volume of information, problems have arisen with regard to their processing. Increasingly it is being required the services to be delivered with more effectiveness resulting in faster data retrieval. It's in this context that arises the need to drive new advances in healthcare, having as a guideline the search for a service with more efficient and higher quality. In this way occurs the need to optimize the process of access and use of this information, changing the way that the patient's information is obtained. This dissertation focuses essentially on these points.

Additionally it can be said that these developments have characteristics favourable to the prevention and control of hospital infections, reducing the need for direct contact with the objects, which leads to decreased spread of infections said.

This study aim to evaluate the potential of gesture recognition applied to the user interfaces implemented specifically in the healthcare area.

Alongside it was developed a prototype intended to the patients that usually attend to healthcare institutions. The goal of this prototype is to validate the user interfaces proposals, considering the use of technology recently introduced on the market. It was given special attention to user interfaces without contact between users and devices (e.g. using the technology of the Microsoft Kinect).

At the end it is presented a statistical study on the evaluation of the user interface prototype by users, in which it is proved that the functionality of using gestures is intuitive and easy to perform.

Índice

AGRADECIMENTOS	III
RESUMO	V
ABSTRACT	VII
ÍNDICE	IX
LISTA DE FIGURAS	XIII
LISTA DE TABELAS	XVII
NOTAÇÃO E TERMINOLOGIA	XIX
GLOSSÁRIO	XXI
CAPÍTULO 1	1
INTRODUÇÃO	1
1.1 – Enquadramento	2
1.2 – Interfaces de Utilizador em Meio Hospitalar	3
1.3 – Objetivos	4
1.4 – Organização do documento	5
CAPÍTULO 2	7
INTERFACES DE UTILIZADOR	7
2.1 – Desenvolvimento	8
2.1.1 – Análise de Domínio	8
2.1.2 – <i>Design</i> - Janelas Virtuais	14

2.1.3 – Desenvolvimento do Protótipo	15
2.1.4 – Programação e testes	15
2.2 – Tipos	16
2.2.1 – Interface de Linha de Comandos	16
2.2.2 – Interface Gráfica de Utilizador	17
2.2.3 – Interface Natural de Utilizador	17
2.3 – Meio Hospitalar	18
CAPÍTULO 3	23
RECONHECIMENTO GESTUAL	23
3.1 – Evolução tecnológica	24
3.2 – Linguagem Gestual	29
3.3 – Tecnologia Touchscreen	31
3.3.1 – Resistivo	33
3.3.2 – Capacitivo	34
3.3.3 – Monitor de Ondas Acústicas (SAW)	35
3.3.4 – Monitor Infravermelhos	36
3.3.5 – <i>Frustrated Total Internal Reflection</i>	37
3.4 – Free-Forms	38
3.4.1 – Características de Reconhecimento	40
3.4.2 – Modelos de Reconhecimento	40
3.4.3 – Análise gestos	41
3.4.4 – Reconhecimento de gestos	42
3.4.5 – Técnicas de Reconhecimento Gestual	43
3.5 – Interfaces de Utilizador	45
3.6 – Dispositivos	48
3.6.1 – Dispositivos Móveis	49
3.6.2 – Terminais Táteis - <i>Kiosks</i>	51
3.6.3 – Dispositivos de reconhecimento de profundidade	52
CAPÍTULO 4	57
TECNOLOGIA KINECT DA MICROSOFT	57

4.1 – Sensor	58
4.2 – Reconhecimento	62
4.2.1 – Detecção de Esqueleto	64
4.2.2 – Reconhecimento de Voz	67
4.2.3 – Reconhecimento Facial	68
4.3. – Testes das Funcionalidades do Dispositivo	68
4.3.1 – Sensor RGB	71
4.3.2 – Sensor de Profundidade	73
4.3.3 – Conjunto de Microfones	83
4.4. – Utilização em Ambientes Hospitalares	85
4.4.1 – Reconhecimento Gestual	86
4.4.2 – Reconhecimento de Voz	96
4.4.2 – Interfaces de Utilizador Multimodais	98
CAPÍTULO 5	101
INTERFACE HOSNUI	101
5.1 – Descrição das Funcionalidades	104
5.2 – Estudo Estatístico	109
CAPÍTULO 6	119
CONCLUSÕES	119
6.1 – Síntese	120
6.2 – Conclusões	121
6.3 – Novos Desafios	123
REFERÊNCIAS	125
ANEXO A	133
ANEXO B	137
ANEXO C	139

ANEXO D _____ **141**

ANEXO E _____ **143**

Lista de Figuras

<i>Figura 2.1 - Relação entre os diferentes modelos mentais. Imagem retirada de [15].</i>	12
<i>Figura 2.2 – A - Ambiente com implementação Interface ponto a ponto; B – Ambiente com implementação de Motor de Interface. Adaptado de [35].</i>	21
<i>Figura 3.1 - Dispositivo Cyberglove. Imagem retirada [40].</i>	26
<i>Figura 3.2 - Jogo MoCap Boxing desenvolvido pela Konami. Imagem retirada de [31].</i>	27
<i>Figura 3.3 - Demonstração da aplicação do método para manipular e navegar entre as imagens de radiologia sem necessitar contacto direto. Retirado de [47].</i>	28
<i>Figura 3.4 - Ilustração do funcionamento da UI Fusion4D. Retirado de [50].</i>	29
<i>Figura 3.5 – Classificação de Gestos e Movimentos segundo Quek [53].</i>	31
<i>Figura 3.6 - Superfície tátil com tecnologia capacitiva. Retirado de [56].</i>	34
<i>Figura 3.7 - Superfície tátil com tecnologia capacitiva. Retirado de [56].</i>	35
<i>Figura 3.8 - Superfície tátil com tecnologia de ondas acústicas. Retirado de [56].</i>	36
<i>Figura 3.9 - Superfície tátil com tecnologia de infravermelhos. Retirado de [56].</i>	37
<i>Figura 3.10 - Superfície tátil com tecnologia de FTIR. Retirado de [56].</i>	38
<i>Figura 3.11 - Esquema representacional da análise e reconhecimento de gestos. Adaptado de [44].</i>	42
<i>Figura 3.12 - Ilustração do modelo 3D(A) e do modelo baseado na aparência(B). Imagem retirada de [59].</i>	45
<i>Figura 3.13 - Principais tipos de Kiosks. A – Dispositivo freestanding; B – Dispositivo wall-mounted; C – Dispositivo móvel; Imagem adaptada de [8].</i>	52
<i>Figura 3.14 - Dispositivo PSeYE. Retirado de [71].</i>	53
<i>Figura 3.15 - Dispositivo Wii Motion. Retirado de [74].</i>	54
<i>Figura 3.16 - Chip da Wii Motion integrado com câmara infravermelhos. Retirado de [73].</i>	54
<i>Figura 3.17 - Dispositivo Leap. Retirado de [75].</i>	55
<i>Figura 4.1 - Dispositivo Kinect (A) e sua constituição (B). Retirado de [80].</i>	59
<i>Figura 4.2 - Representação dos intervalos de distancia possíveis em metros da Kinect.</i>	60
<i>Figura 4.3 - Processo de lightcoding através de uma Imagem obtida pelo sensor de profundidade com pontos de infravermelhos emitidos pelos emissor de infravermelhos.</i>	61
<i>Figura 4.4 - Processo de obtenção de imagens com noção de profundidade.</i>	62
<i>Figura 4.5 - Arquitetura da interação entre software e hardware numa UI. Retirada de [80].</i>	64

<i>Figura 4.6 - A –Espaço tridimensional captado pelo Kinect; B - Representação das articulações identificadas pelo Kinect. Adaptado de [80].</i>	65
<i>Figura 4.7 - Modos de rastreamento do esqueleto possíveis da Kinect (modo padrão e modo sentado). Adaptado de [82].</i>	66
<i>Figura 4.8 - Processo de reconhecimento facial obtido com auxílio da aplicação Face Tracking Basics disponível no navegador do SDK para a Windows Kinect.</i>	68
<i>Figura 4.9 - UI de controlo de inclinação do dispositivo Kinect.</i>	71
<i>Figura 4.10 - Exibição da imagem a cor obtida a uma taxa de 30 FPS e uma resolução de 640x480 através do sensor RGB da Kinect.</i>	72
<i>Figura 4.11 - Demonstração da capacidade da câmara na captação de movimentos bruscos.</i>	73
<i>Figura 4.12 - Efeito da exposição do sensor RGB a alta ou baixa luminosidade.</i>	73
<i>Figura 4.13 - Imagem com dados de profundidade do meio sem utilizador.</i>	74
<i>Figura 4.14 - Imagem com dados de profundidade do meio com utilizador reconhecido.</i>	75
<i>Figura 4.15 - Efeito da baixa luminosidade na obtenção dos dados de profundidade.</i>	76
<i>Figura 4.16 – Demonstração do funcionamento da aplicação 3D Viewer do Kinect Studio.</i>	77
<i>Figura 4.17 - Modelo do esqueleto do utilizador obtido através do dispositivo Kinect.</i>	77
<i>Figura 4.18 - Modelo do esqueleto do utilizador com as vinte articulações detetadas.</i>	79
<i>Figura 4.19 - Modelo do esqueleto do utilizador com as articulações do braço esquerdo e perna direita inferidas.</i>	79
<i>Figura 4.20 - Efeito da presença de objetos de grande volume durante o processo de rastreamento do esqueleto do utilizador.</i>	79
<i>Figura 4.21 - Demonstração da capacidade do dispositivo Kinect em reconhecer diferentes utilizadores realizando o rastreamento do esqueleto de apenas dois.</i>	80
<i>Figura 4.22 - Resultado obtido do modelo do esqueleto do utilizador a partir do modo sentado.</i>	81
<i>Figura 4.23 - Resultado obtido do modelo do esqueleto do utilizador a partir do modo sentado com o utilizador na posição de pé.</i>	81
<i>Figura 4.24 - Modelo do esqueleto do utilizador a partir do modo padrão com o utilizador na posição de sentado.</i>	82
<i>Figura 4.25 – Influência da presença de grande quantidade de iluminação no processo da obtenção de dados de profundidade.</i>	82
<i>Figura 4.26 – Demonstração do funcionamento do sensor de profundidade quando exposto diretamente sobre a luz do sol.</i>	83
<i>Figura 4.27 - Demonstração da propriedade de deteção da fonte de som do dispositivo Kinect. Esta propriedade baseia-se nos diferentes tempos de chegada do mesmo som aos microfones.</i>	84

<i>Figura 4.28 – Demonstração da captação dos microfones presentes no dispositivo (A) expostos a um reduzido ruído de fundo; (B) expostos a altos níveis de ruído de fundo onde se pode verificar a necessidade do utilizador em elevar o tom da voz para ser detetado.....</i>	<i>85</i>
<i>Figura 4.29 – Demonstração da simulação de gesto dinâmico reconhecido através do dispositivo Kinect. Definiu se o movimento de elevar o joelho direito a cima do joelho esquerdo para parar o emulador.....</i>	<i>87</i>
<i>Figura 4.30 - Janela de controlo da aplicação Kinect Mouse Cursor. Aplicação permite substituir o uso do rato para a manipulação do ambiente do SO.....</i>	<i>88</i>
<i>Figura 4.31 – Exemplo de gesto inapropriado de implementar numa UI de reconhecimento gestual.</i>	<i>89</i>
<i>Figura 4.32 – Método de trocar de imagem da aplicação Slide Gesture.....</i>	<i>90</i>
<i>Figura 4.33 – Mecanismo de pausa da aplicação Shape Game através do comando de voz “Freeze”.</i>	<i>91</i>
<i>Figura 4.34 – Influencia da posição de lado no processo do rastreamento. Os gestos representados não correspondem aos do utilizador.</i>	<i>92</i>
<i>Figura 4.35 –Influencia de movimentos rápidos no processo do rastreamento.....</i>	<i>93</i>
<i>Figura 4.36 – Demonstração da incapacidade de reconhecimento fora dos limites do dispositivo.</i>	<i>93</i>
<i>Figura 4.37 – Método de seleção presente na aplicação Basic Interactions.....</i>	<i>95</i>
<i>Figura 4.38 – Rastreamento das mãos obtido recorrendo aplicação desenvolvida por Cerezo et al. [84] no âmbito do estudo sobre o reconhecimento de dedos utilizando o dispositivo Kinect.</i>	<i>95</i>
<i>Figura 4.39 - Capacidade de reconhecimento do dispositivo. A partir do comando de voz “Red”, o dispositivo representa a cor através da barra vertical.....</i>	<i>97</i>
<i>Figura 4.40 - Capacidade de reconhecimento de voz do dispositivo Kinect através da aplicação Speech Basics disponível no navegador do SDK.</i>	<i>98</i>
<i>Figura 4.41 – Menu de seleção da aplicação Basic Interactions.....</i>	<i>99</i>
<i>Figura 4.42 – Ambiente de interação da aplicação Shape Game.....</i>	<i>100</i>
<i>Figura 5.1 – Janela exibida pelo HosNui quando o utilizador não se encontra a interagir com este.</i>	<i>104</i>
<i>Figura 5.2 – Janela de boas vindas do HosNui.....</i>	<i>105</i>
<i>Figura 5.3 – Mecanismo de clique do HosNui.....</i>	<i>106</i>
<i>Figura 5.4 – Feedback transmitido sobre o estado do rastreamento do utilizador durante a interação o HosNui.</i>	<i>106</i>
<i>Figura 5.5 – Janela de escolha de categorias do HosNui.....</i>	<i>107</i>
<i>Figura 5.6 - Janela de escolha de temas dentro de uma determinada categoria do HosNui.....</i>	<i>108</i>
<i>Figura 5.7 - Exemplo de uma janela do HosNui, onde é abordado o tema dos direitos e deveres do utente.....</i>	<i>109</i>

<i>Figura 5.8 – Natureza da interação (questão n°7).</i>	112
<i>Figura 5.9 - Adequação dos gestos (questão n°8).</i>	112
<i>Figura 5.10 – Precisão do reconhecimentos de gestos (questão n°9).</i>	113
<i>Figura 5.11 – Adequação para comandos de voz (questão n°10).</i>	114
<i>Figura 5.12 - Precisão do reconhecimentos de voz (questão n°11).</i>	114
<i>Figura 5.13 – Repartição do reconhecimento de voz e gestual (questão n°12).</i>	115
<i>Figura 5.14 – Nível de satisfação com a interação (questão n°14).</i>	116
<i>Figura 5.15 – Utilidade da interface (questão n°15).</i>	117
<i>Figura 5.16 – Satisfação com interface (questão n°16).</i>	117
<i>Figura A.1 – Janela de seleção dos templates disponíveis no Visual Studio.</i>	133
<i>Figura A.2 – Processo de adição de referência no Visual Studio.</i>	134
<i>Figura A.3 - Processo da seleção da referência da Kinect no Visual Studio</i>	135
<i>Figura A.4 - Processo de referenciação da Kinect no Visual Studio através da aba .Net.</i>	135
<i>Figura A.5 – Referencia citada no Solution Explorer.</i>	136
<i>Figura D.1 – Painel de seleção do ângulo de inclinação.</i>	141

Lista de Tabelas

<i>Tabela 3.1 – Características das diferentes tecnologias Touchscreen</i>	<i>33</i>
<i>Tabela 4.1 – Requisitos de Sistema para instalação Kinect</i>	<i>58</i>
<i>Tabela 4.2 – Especificações do dispositivo Kinect</i>	<i>60</i>
<i>Tabela 4.3 – Comparação entre modos sentado e pé</i>	<i>66</i>
<i>Tabela 5.1 – Resultados obtidos através dos dados recolhidos no inquérito.</i>	<i>110</i>

Notação e Terminologia

Notação Geral

A notação ao longo do documento segue a seguinte convenção:

- **Texto em itálico** – para palavras em língua estrangeira (e.g., Inglês). Também utilizado para dar ênfase a um determinado termo ou expressão.
- **Texto em negrito** – utilizado para realçar um conceito ou palavra no meio de um parágrafo.

Acrónimos

2D – Duas dimensões

3D – Três dimensões

ANSI - *American National Standards Institute* (Instituto Americano de Padrões)

ASR - *Automatic Speech Recognition* (Reconhecimento de Voz Automático)

CMOS - *complementary metal-oxide-semiconductor* (semicondutor de metal-óxido complementar)

CLI - *Command Line Interface* (Interface de Linha de Comandos)

EMR – *Electronic Medical Record* (Registo Médico Eletrónico)

FPS – *Frames per second* (Quadros por segundo)

FTIR - Frustrated Total Internal Reflection

GSM - *Global System for Mobile Communication* (Sistema Global para Comunicações Móveis)

GUI – *Graphical User Interface* (Interfaces de Utilizador Gráficas)

HIS – *Hospital Information System* (Sistema de Informação do Hospital)

HL7 – *Health Level 7*

HMM – Hidden Markov Model (Modelo oculto de Markov)

IDE - *Integrated Development Environment* (Ambiente de Desenvolvimento Integrado)

LIS – *Laboratory Information System* (Sistema de Informação dos Laboratórios)

NUI – *Natural User Interface* (Interface Natural de Utilizador)

PACS - *Picture Archiving and Communication System* (Sistema de Arquivo e Distribuição de Imagem)

PDA - *Personal digital assistant* (Assistente Pessoal Digital)

RIS – *Radiology Information System* (Sistema de Informação de Radiologia)

SAW - *Surface Acoustic Wave*

SDK – *Software Development Kit* (Kit de desenvolvimento de *software*)

SO – Sistema Operativo

UI – *User Interface* (Interface de Utilizador)

Glossário

Acelerómetro - Dispositivo responsável por medir a aceleração relacionada com o peso de uma massa de teste.

Affordance - Forma que o objeto apresenta de modo que seja intuitivo para o utilizador qual a sua forma de funcionamento.

Algoritmo *K-Nearest Neighbors* – Algoritmo de classificação de objetos baseado na proximidade entre os exemplos de treino pertencentes ao espaço das características. O objeto é classificado tendo em conta a maioria dos seus vizinhos, sendo atribuído à classe mais comum entre os vizinhos k mais próximos. O valor k é um valor inteiro. Caso este valor seja 1, o objeto é atribuído à classe do seu vizinho mais próximo.

Ambiente de Desenvolvimento Integrado - *Software* que reúne características e ferramentas de apoio ao desenvolvimento de software com o objetivo de agilizar este processo. Normalmente inclui editor e compilador de código para as linguagens suportadas, depurador para auxílio no processo de encontrar e corrigir erros de código, entre outros.

Amostra - Subconjunto retirado da população, que se supõe ser representativo de todas as características da mesma, sobre o qual será feito o estudo, com o objectivo de serem tiradas conclusões válidas sobre a população.

Bluetooth – Especificação para área de redes pessoais sem fios. Fornece uma forma de conexão e troca de informação entre dispositivos como telemóveis, computadores, entre outros, através de uma frequência de rádio de curto alcance, sendo globalmente licenciada e segura.

Classificador *Naive Bayes* – Classificador probabilístico que tem como base o teorema *Bayes*. Para uma amostra desconhecida, permite calcular a probabilidade de esta pertencer ou não a uma das classes possíveis. Este classificador considera que o efeito de um valor atribuído sobre determinada classe é independente dos valores dos outros atributos.

Código aberto – Metodologia que promove a livre distribuição e acesso ao código fonte do projeto e respectivos detalhes de implementação.

Extensómetro – Transdutor capaz de medir deformações de corpos através da alteração da sua resistência elétrica. A fração alterada na resistência é proporcional à fração de mudança no comprimento do material.

Firmware - Conjunto de instruções operacionais programadas diretamente no hardware de um equipamento electrónico. Está normalmente envolvido com operações de baixo nível das quais, sem um dispositivo, seria completamente não-funcional.

Histograma – Representação gráfica de uma distribuição de frequências de um conjunto de medições.

Infeção Nosocomial – qualquer tipo de infeção adquirida após entrada do utente num hospital ou após a sua alta, quando essa infeção estiver diretamente relacionada com o internamento ou procedimento hospitalar, como por exemplo uma cirurgia.

Interpolação - método que permite construir um novo conjunto de dados a partir de um conjunto discreto de dados pontuais previamente conhecidos.

LED – Díodo emissor de luz.

Machine learning - Ramo da inteligência artificial responsável pela concepção e desenvolvimento de algoritmos que, a partir de dados empíricos (e.g. sensores, bases de dados), estabelece padrões ou previsões.

MS-Dos (*Miscrosoft Disk Operating System*) – Sistema operativo da *Microsoft* no qual a interação é realizada através da introdução de linhas de comandos. Inicialmente foi comprado para ser usado na linha de computadores IBM PC.

Patógeno - Causador ou micro-organismo específico que provoca doenças.

População - somatório dos indivíduos ou elementos, com qualquer característica comum e que estão sujeitos a uma análise estatística, por terem interesse para o estudo. Quanto à sua origem pode ser: um conjunto de pessoas; um conjunto de objectos ou um conjunto de acontecimentos. Quanto à sua natureza pode ser: existente ou real; hipotética ou parcialmente existente. Pode ainda ser: um conjunto finito ou um conjunto infinito.

Radiação Infravermelha – Radiação não ionizante na porção invisível do espectro eletromagnético que está adjacente aos comprimentos de onda longos, encontrando-se no final do espectro visível de luz (vermelho).

RGB – corresponde ao sistema usado para representar as cores em dispositivos electrónicos como televisões, computadores, entre outros. Este sistema é composto pelas cores vermelho, verde e azul, que, combinados em diferentes proporções, podem formar qualquer cor do espectro visível. Para a formação da imagem, cada cor vai ter uma percentagem de intensidade que vai de 0 a 100. Cada cor tem associada uma partição de 8 bits, correspondendo a uma gama de números decimais que vai desde 0 a 255 (256 possíveis valores).

Sensor CMOS - componente electrónico em estado sólido que converte a luz num padrão de cargas eléctricas que se traduz em dados digitais. Usado normalmente em câmaras digitais.

Shell – software responsável por fornecer uma interface ao utilizador para aceder aos recursos internos do sistema operativo.

Sistema de Informação – Sistema que manipula dados e gera informação, usando ou não recursos de tecnologia da informação. É constituído por um conjunto de componentes inter relacionados que atuam com o intuito de recolher, recuperar, processar, armazenar e distribuir informações com finalidade de facilitar o planeamento, o controlo, a coordenação e a análise de processos em organizações.

UNIX – Sistema operativo multitarefa e multiutilizador desenvolvido em 1969 pelos laboratórios *Bell*. O sistema *Unix* foi desenvolvido na linguagem de programação C.

USB 2.0 – Lançado em 2000, permite taxas efetivas de transferência até 35 Mb/s.

Padrão YUV – Modelo de representação da cor dedicado ao vídeo analógico. Baseia-se num

modo de transmissão vídeo. O parâmetro Y representa a luminância (informação a preto e branco) e os parâmetros U e V representam a crominância (informação sobre a cor). Este modelo foi criado para permitir transmitir informações coloridas para as televisões a cores, garantindo que as televisões a preto e branco existentes continuavam a fixar uma imagem em tons de cinzentos.

CAPÍTULO 1

Introdução

Ideias chave:

- Aposta das instituições de saúde em novas tecnologias de informação.
- Importância das infeções hospitalares no modo de interação com as interfaces de Utilizador.
- Diversidade de paradigmas de interfaces de utilizador.

Nos dias de hoje as instituições de saúde têm vindo a deparar-se com um aumento significativo de dados e informação. Surge deste modo a necessidade de optimização dos processos de acesso e utilização dessa informação, alterando a forma como a informação sobre os utentes é obtida, auxiliando na tomada de decisões clinicas [1]. Com o surgimento de novas tecnologias de informação e comunicação criaram-se novas oportunidades no desenvolvimento e concepção de Interfaces de Utilizador (UI). As UI dizem respeito a ferramentas que têm como função facilitar o desempenho de tarefas de manipulação de informação [2]. Estas têm como intuito a criação de sistemas que são modelados tendo em conta as características e tarefas dos utilizadores, conseguindo desse modo uma maior aceitação por parte do utilizador e uma diminuição do número de erros [3]. Uma vez que as instituições de saúde têm vindo cada vez mais a revelarem-se como uma área de intensa transferência electrónica de dados, as UI surgem como uma ferramenta com grande potencial e de grande utilidade.

Por outro lado, desde sempre que a preocupação das instituições de saúde passa pela prevenção e controlo de infeções hospitalares. Devido a factores climatéricos e estruturais, assim como pelo facto de abrangerem um grande proporção da população, o ambiente hospitalar apresenta-se como um meio propício para desenvolvimento e propagação de infeções [4]. Um exemplo dos veículos destas infeções é precisamente o dos tradicionais dispositivos de interface (e.g. teclados, ratos, monitores tácteis) que poderão e deverão ser evitados.

1.1 – Enquadramento

A implementação da UI permite estabelecer uma interação direta com os objetos disponíveis no ambiente gráfico. Deste modo, são estabelecidas associações com os objetos do dia-a-dia, permitindo assim uma interação intuitiva. No entanto, apresentam como desvantagem o facto de que, devido a serem utilizadas por diferentes utilizadores, os dispositivos de entrada e saída de dados sujam-se com grande facilidade, o que pode levar à transmissão de doenças contagiosas [5].

Com o surgimento de novas tecnologias (e.g. novos sistemas operativos *Android*, *Windows 8* e *Mac OS X*, *Tablet PC*, dispositivos móveis, câmaras com noção de profundidade, *superfícies táteis*), verifica-se uma maior aposta por parte das instituições de saúde em novos dispositivos, como é o caso dos *Kiosks* e os *Tablets PC* [6].

A adoção dos dispositivos móveis, como por exemplo PDA (Assistente Pessoal Digital - *Personal digital assistant*), *SmartPhone*, *Tablet PC* tem promovido potencializar a forma como se faz o acesso à informação, permitindo aumentar o fluxo de trabalho e auxiliar a tomada de decisões nos cuidados de saúde. Os dispositivos móveis caracterizam-se por ser dispositivos de reduzidas dimensões que apresentam como principal vantagem o facto de oferecerem acesso ubíquo aos dados médicos. Normalmente a entrada e saída de dados nestes dispositivos é estabelecido com base em superfícies táteis, através da qual é possível interagir com o sistema simplesmente através do toque. Os dispositivos móveis inseridos em meios hospitalares revelam ser uma ferramenta funcional e útil na documentação, referências médicas e acesso a dados de utentes [7].

Outro tipo de interfaces que tem vindo a ser introduzido nas instituições de saúde é o dos denominados *Kiosks*. Estes permitem aos utilizadores realizar atividades de rotina como o

admissão, assim como atualizar os seus dados pessoais e fornecer informações relevantes sobre doenças reduzindo deste modo a necessidade de interagir com funcionários. Este tipo de dispositivos também permite proceder a pagamentos e à recolha de documentação. Deste modo, é possível obter uma redução da carga de trabalho, melhoria da qualidade dos dados, maior consistência do processo de registo e uma redução das listas de espera. A sua adequada implementação visa atender à crescente expectativa dos utentes, relativamente aos avanços tecnológicos existentes e, ao mesmo tempo, melhorar a precisão e aumentar o uso dos sistemas de informação. O utilizador interage normalmente com o sistema através de um ecrã *tátil* [8].

Entretanto, tendo em conta os avanços tecnológicos mais recentes, têm vindo a surgir novas formas de interagir. Uma dessas formas, que vai ser objecto de estudo neste trabalho, diz respeito à captação dos movimentos do corpo humano através de técnicas de visão computacional. A interpretação desses movimentos permite a concepção e desenvolvimento de novas UI sem contacto físico com nenhum dispositivo [9].

A *Microsoft* disponibilizou recentemente um dispositivo denominado de *Kinect* para o sistema operativo (SO) *Windows 7* com um sensor de movimento que se caracteriza por uma câmara periférica. A tecnologia *Kinect* assenta na combinação de varias tecnologias baseadas no uso de câmaras RGB, sensor de profundidade e microfones, usando o design de interação [10]. O uso deste tipo de tecnologia pode potenciar o desenvolvimento de interfaces onde se pretende evitar o contato físico com periféricos.

Com o trabalho que irá ser feito pretende-se estudar as potencialidades da tecnologia de reconhecimento gestual para o desenvolvimento e concepção de interfaces com especial aplicação na área da saúde. Pretende-se também criar protótipos das UI propostas de modo a poder avaliar o seu potencial.

1.2 – Interfaces de Utilizador em Meio Hospitalar

Com o aumento do uso de UI em instituições de saúde, tem-se levantado o problema da propagação de infeções através do contato com os dispositivos. As mãos têm um papel bem instituído na transmissão de infeções nosocomiais, também conhecidas como infeções hospitalares. Este tipo de situações estão intrinsecamente ligadas ao facto de, nos ambientes hospitalares, os doentes se encontrarem sujeitos a um maior risco de obtenção de infeções,

uma vez que estão expostos a microrganismos aos quais não se encontrariam expostos no seu dia-a-dia. Além disso, apresentam as suas defesas imunitárias mais debilitadas devido ao seu estado, o que pode também favorecer o surgimento de infeções [11][12][13].

A causa das infeções nosocomiais pode estar relacionada com diferentes factores, destacando-se o local de instalação. A este fator associa-se infeções do tipo urinárias, cirúrgicas, respiratórias, entre outras. Normalmente a sua origem deve-se a um agente causador. Este tipo de infeções pode ser cruzada, caso se verifique a transferência de um microrganismo de uma pessoa ou objeto para outra pessoa, ou então oportunista, provocada geralmente por germes não patogénicos de um organismo comprometido. O seu desenvolvimento toma vantagem das debilidades imunitárias do hospedeiro, do grau de agressividade dos agentes infecciosos, do modo de transmissão e do ambiente inserido. A sua transmissão pode dar-se por [4][14]:

- Via direta – contato com mãos, lábios ou objetos contaminados;
- Via indireta – por vetores (roedores, insetos), alimentos ou águas contaminadas.

Uma vez que a interação com as UI requerem o uso das mãos, estas podem-se tornar um veículo de transmissão de patógenos nosocomiais, sujeitando os seus utilizadores. Deste modo, é importante procurar-se alternativas viáveis para substituição do toque como meio de interação com as interfaces que se encontrem em ambientes críticos. Uma alternativa para contornar este problema seria a implementação de UI de reconhecimento de gestos e voz.

1.3 – Objetivos

Este trabalho tem como principal objectivo estudo das potencialidades de reconhecimento gestual e o desenvolvimento de concepção de novas UI para aplicações na área da saúde.

Pretende-se validar as UI propostas através da avaliação de protótipos desenvolvidos utilizando tecnologias recentemente introduzidas no mercado.

1.4 – Organização do documento

O trabalho que se segue encontra-se estruturado em seis capítulos que correspondem ao desenvolvimento da temática em questão e cujo objetivo é a explicitação das principais matérias envolvidas na investigação realizada. Assim sendo, adicionalmente ao presente capítulo introdutório, encontram-se os seguintes capítulos:

- **Capítulo 2** – Interfaces de Utilizador.

Neste capítulo é abordada a temática específica das UI. O objetivo principal desta secção é o processo do seu desenvolvimento. São ainda referidos os principais tipos de interfaces. Por fim, é abordado o seu processo de implementação em meios hospitalares, referindo o processo e dando a conhecer melhor a sua realidade.

- **Capítulo 3** – Reconhecimento Gestual.

Neste capítulo é abordada a temática específica do reconhecimento gestual. Neste sentido, inicialmente é apresentada uma descrição introdutória da evolução tecnológica deste tipo de funcionalidade. Posteriormente são expostos alguns conceitos de linguagem gestual para a correta implementação em UI. São, também, abordadas as duas principais categorias de reconhecimento gestual: *superfície tátil* e *Free-Forms*. Por fim são destacados os diferentes tipos de dispositivos de reconhecimento gestual existentes.

- **Capítulo 4** – Tecnologia *Kinect* da *Microsoft*.

Neste capítulo, procede-se à descrição das características técnicas do dispositivo *Kinect*, utilizado por circunstância da dissertação, que possibilitam realizar o reconhecimento gestual e voz. É descrita a forma como o dispositivo realiza a deteção e rastreamento do utilizador e consequente reconhecimento gestual, de voz e facial. São descritos os testes realizados com o intuito de avaliar as suas funcionalidades.

- **Capítulo 5** – Interface *HosNui*.

Neste capítulo do documento, expõe-se as etapas de desenvolvimento do prototipo de interface (*HosNui*), tendo em conta as potencialidades do dispositivo *Kinect*, bem como as ferramentas utilizadas e a forma como está organizada a informação presente na interface. É

também apresentada a avaliação realizada no âmbito da validação da interface e respetivos resultados.

- **Capítulo 6** – Conclusões.

No último capítulo são apresentadas as conclusões de todo o trabalho efectuado e indicadas algumas propostas de trabalho futuro.

CAPÍTULO 2

Interfaces de Utilizador

Ideias chave:

- Fases de desenvolvimento da interface de utilizador: análise, *design*, protótipo, programação, testes e operação.
- Importância da caracterização do utilizador para adaptar as funcionalidades da interface.
- Existência de diferentes tipos de interface de utilizador (e.g. CLI, GUI, NUI).
- Abordagens de Implementação de interfaces de utilizador em meios hospitalares: criação de interface ponto-a-ponto ou implementação de um motor de interface.

As UI dizem respeito aos meios disponibilizados ao utilizador para ver, ouvir e interagir com o sistema. Por outras palavras, é o ambiente que permite uma interação humano-computador (Human-Computer Interaction - HCI) com o sistema. Outras partes do sistema como as bases de dados não se podem classificar como interface, visto que não podem interagir diretamente com esta [15].

As UI são definidas pelo conjunto de características que possibilitam a interação do utilizador com os dispositivos, computadores ou programas. Estas devem fornecer ferramentas de entrada que permitam ao utilizador manipular o sistema e ferramentas de saída de modo a permitir que o sistema mostre os resultados da ação do utilizador [15].

O desenvolvimento de interfaces competentes e eficazes tem uma maior aceitação por parte dos utilizadores, proporcionando um bem estar durante a sua utilização e ajudando a

aumentar o fluxo de trabalho. Deste modo, o utilizador apresenta maior comodidade para realização destas tarefas, dispondo de uma menor esforço para as completar [16].

2.1 – Desenvolvimento

Para o desenvolvimento de UI é necessário recorrer primeiro à organização de dados num conjunto de janelas. De seguida são definidas as funções necessárias para o correto funcionamento da interface que possibilitem o controlo do sistema pelo utilizador. Por último, define-se a aparência gráfica das janelas e das funções. O sistema desenvolvido pode incluir atividades de controlo como testes de usabilidade. Com este modelo é pretendido o desenvolvimento de sistemas eficientes e de fácil aprendizagem [17].

O processo de desenvolvimento de interfaces é composto por algumas fases como:

- Análise;
- Design;
- Protótipo;
- Programação;
- Testes;
- Operação.

Na primeira fase tem-se em consideração as exigências do utilizador de modo a reter as possíveis soluções. Posteriormente, tendo em conta as possíveis soluções, é feito um planeamento do software a desenvolver, desenvolve-se o protótipo e testa-se. De seguida é programado o novo sistema. É então que se procedem a testes com o intuito de detetar e corrigir erros até que se encontrar um resultado ótimo. Por ultimo é posto o sistema em operação onde se poderá encontrar novos problemas, os quais, corrigidos permitirão ao sistema cobrir um maior numero de soluções [15].

2.1.1 – Análise de Domínio

Previamente ao desenvolvimento de uma interface, é necessário saber qual o utilizador alvo e o que este pretende obter com o uso do sistema. Deste modo, torna-se importante caracterizar o tipo de utilizador, conhecendo quais os seus objetivos ao usar a interface, o tipo de tarefa

que pretende ver realizada, a forma como caracterizam as suas tarefas e os seus conhecimentos em lidar com interfaces do mesmo tipo. Deste modo, é necessário, numa primeira fase, criar modelos do que se pretende construir, tendo em atenção o valor que este pode acrescentar nos objetivos do utilizador. São também identificados os tipos de dados que serão manipulados [15][16][18].

No entanto, para o desenvolvimento da interface não se pode ter em conta apenas os desejos dos utilizadores. Nem todos são designers de interfaces, sendo que muitos têm dificuldade em explicar qual o tipo de produto que pretendem. Em muitos casos, não são capazes de avaliar quais as suas necessidades em relação a uma certa interface. Assim, é imprescindível saber quais os problemas do utilizador para posteriormente encontrar a solução. Saber quais as tarefas que o utilizador tem vindo a desenvolver é uma mais valia, de modo a tornar essas tarefas mais fáceis e mais eficientes. Um factor importante seria saber também quais os desejos do utilizador sobre as tarefas que vem a desenvolver, de modo a poder torná-los possíveis. A análise é um passo muito importante, uma vez que interfaces que apresentem funcionalidades sem utilidade para o utilizador são facilmente rejeitadas [16][19].

Com análise de domínio pretende-se retirar uma descrição das tarefas a desenvolver com utilização da interface, o modelo de dados e um perfil de utilizador, incluindo os requerimentos de usabilidade.

Usabilidade

A usabilidade é um fator de qualidade que se deve ter em conta aquando do desenvolvimento de UI. Usabilidade pode ser interpretada como a facilidade com que o utilizador aprende e interage com uma interface. Sistemas mais fáceis de utilizar têm uma usabilidade mais alta e vice-versa. A usabilidade modela a efetividade e eficiência dos sistemas e, em consequência, a satisfação dos utilizadores. Um sistema com elevada usabilidade ganha, em termos de poupança de tempo ao utilizador, garantindo o acesso a um maior número de pessoas sem necessitar de se especializar num sistema específico [15][20].

A usabilidade é definida por um conjunto de fatores como [15][20]:

- Funcionalidade – Capacidade do sistema suportar as tarefas dadas pelo utilizador;
- Facilidade de aprendizagem – Facilidade que o utilizador tem para aprender a interagir com o sistema;
- Eficiência – Eficiência do sistema tendo em conta as necessidades do utilizador;

- Retenção – Facilidade com que o utilizador memoriza os processos do sistema;
- Satisfação – Até que ponto o sistema consegue atingir as expectativas do utilizador;
- Compreensibilidade – Facilidade com é perceptível entender a potencialidade do sistema.

Estes fatores devem estar presentes durante o desenvolvimento de uma interface. No entanto, é muito difícil criar interfaces que sejam boas em todos os fatores. Deste modo, tendo em conta os objetivos, deve-se tentar adequar de forma equilibrada cada um destes fatores sem comprometer os restantes [15].

Os benefícios da usabilidade para o utilizador incluem [15][16]:

- Rápida aprendizagem – Leva a que o utilizador gaste menos tempo para aprender a manusear de forma eficaz a interface;
- Métodos de trabalho mais rápidos – Maior fluxo de trabalho por parte do utilizador, fruto da elevada eficiência do sistema;
- Redução do número de erros – Sistemas mais simples expõem o utilizador a um menor número de erros;
- Utilizadores mais motivados – Sistemas *user friendly* originam maiores níveis de motivação por parte dos utilizadores o que leva a que façam um trabalho melhor e mais rápido;
- Maior número de utilizadores – Sistemas de fácil aprendizagem e motivantes influenciam a adoção do sistema por um maior número de utilizadores.

Denomina-se de *user friendly* uma UI que combine os fatores facilidade de aprendizagem e compreensibilidade [16].

Modelo Mental

O modelo mental é um modelo criado pelo utilizador de forma inconsciente sobre o sistema, refletindo a forma como acredita que este funciona. Corresponde à perceção que o utilizador tem quanto à sua funcionalidade e finalidade. Este modelo poderá não corresponder à real finalidade do sistema caso, por exemplo, tenha baixa usabilidade. Uma interface tem maior aceitação por parte do utilizador quando o sistema funciona da forma que o utilizador espera que funcione. Para se criar um modelo correto é necessário uma boa interface [15][19][20].

A interface ideal deve representar na perfeição a implementação planeada, garantindo uma completa perceção das funções do sistema ao utilizador. De modo a construir modelos mentais é necessário primeiro saber a forma como o utilizador alvo pensa. Tal pode ser conseguido através de simples entrevistas ao público alvo. Posteriormente, procede-se aos testes de usabilidade em protótipos, de modo a obter uma aproximação do modelo mental do utilizador e depois apresenta-se o protótipo ao utilizador em estudo. Deste modo, pretende-se entender como o utilizador alvo acha que o produto funciona. Este tipo de informação revela-se muito útil na obtenção de modelos mentais para construção de UI [19].

No entanto, nem sempre é possível construir UI com base no modelo mental do utilizador, uma vez que não é certo que funcione da forma que os utilizadores esperam. Neste caso, deve-se tentar fazer com que a interface seja o mais intuitiva possível, de modo a tornar simples a aprendizagem dos novos mecanismos. Assim, é possível alterar o modelo mental do utilizador [19].

Com vista auxiliar a formação do modelo mental dos utilizadores, a construção de UI deve ter em conta fatores como [19][20]:

- Simplicidade - Regras simples permitem que os utilizadores criem modelos mentais mais creíveis com o atual funcionamento do sistema, facilitando a sua aprendizagem;
- Familiaridade - Relação que os objetos apresentam com os do mundo real. Este fator visa familiarizar o utilizador com o funcionamento da interface, visto que já possui o modelo mental apropriado para estabelecer as relações necessárias;
- Disponibilidade - Auxílio que a UI fornece ao utilizador aquando da sua interação;
- Flexibilidade – Diferentes hipóteses que uma UI apresenta para realizar a mesma tarefa;
- *Feedback* - Exibição de ajuda completa e contínua por parte do sistema sobre as ações do utilizador. Auxiliam na correção de erros. Permite assim que os utilizadores adaptem e construam modelos mentais fiáveis e robustos;
- Segurança - Transmite maior liberdade ao utilizador para a manipulação da UI. Ajuda a evitar que certas ações involuntárias do utilizador prejudiquem as atividades do sistema;
- *Affordance* – Elementos da interface devem ser construídos de modo a que, através da sua aparência, seja implícito o seu modo de funcionamento.

O modelo mental pode ser dividido por partes (Figura 2.1). A percepção que o utilizador tem sobre o destino dos dados disponíveis na interface corresponde ao modelo mental de dados. O utilizador não considera que os dados simplesmente desaparecem, mas sim que estes são armazenados num local. Deste modo, tendo em conta a forma como os dados estão dispostos, o utilizador gera um modelo de como estes estão armazenados e como se relacionam com os outros dados armazenados. As funções do sistema são as ferramentas que o sistema dispõe para ser gerido (e.g. mostrar, alterar e apagar dados). Um modelo correto permite ao utilizador perceber a funcionalidade do sistema em termos de dados armazenados e a sua relação com o que está exposto na interface. Tendo em conta a terminologia HCI, a forma como o sistema funciona corresponde ao modelo estrutural e engloba o modelo de dados e modelo de funções do sistema. O modelo mental de domínio corresponde à percepção que o utilizador tem da relação entre o universo exterior com o sistema. De modo a fazer uma ligação entre o domínio e o modo como o sistema funciona, existe o modelo de mapeamento responsável por obter um ótimo funcionamento do sistema. O mapeamento de funções de tarefas permite ter uma ideia de qual a função do sistema que é a mais apropriada. Tendo em conta a terminologia HCI, o modelo de mapeamento está inserido no modelo funcional e diz respeito à forma como se usa o sistema. Outro termo utilizado para o modelo de mapeamento denomina-se modelo *Task-action*, onde o mapeamento é feito desde as tarefas até às ações [15][16][20][21].

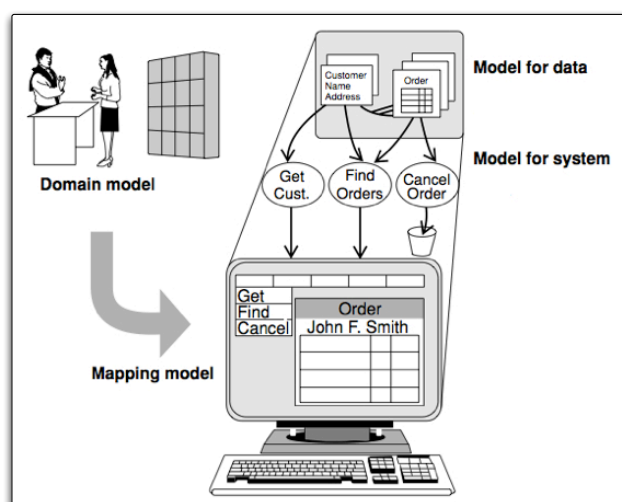


Figura 2.1 - Relação entre os diferentes modelos mentais. Imagem retirada de [15].

Níveis de Diálogo

A interação HCI é feita através de níveis de diálogo que vão desde aspetos de baixo nível, como uma simples interação do menu da UI, a aspetos de alto nível, como a gestão da base de dados [15].

Na conceção de um sistema, o primeiro passo é sempre a análise do domínio, onde é feito um levantamento de quais as necessidades do utilizador e das suas principais tarefas. A forma como as operações reais são interpretadas pelo sistema corresponde ao nível de domínio. É responsável pelas tarefas e dados da aplicação de domínio, tendo em conta o que se passa no universo exterior ao sistema. É também definido um modelo de dados, determinando quais os tipos de dados a armazenar pelo sistema. São especificados os dados que o utilizador irá utilizar no domínio da aplicação, definindo o nível de dados [15].

No nível de mapeamento é definido o mapeamento das tarefas para o sistema, de modo a corresponder aos objetivos do utilizador. Para cada tarefa do utilizador são planeadas todas as funções necessárias para que o sistema obtenha as funcionalidades para que foi concebido. Estas funções correspondem a uma janela virtual onde são definidos os objetos e respetivas operações. As funções que resultam do nível de planeamento designam-se de funções semânticas e funções de busca. As que têm como utilidade a forma como as janelas virtuais se dispõem no monitor denominam-se funções de navegação. O conjunto das funções e respetivas funcionalidades correspondem ao nível funcional. No nível de planeamento é definida a ordem de disposição das janelas virtuais no monitor uma vez que o espaço do monitor é limitado [15].

O nível de sintaxe é responsável pela forma como os dados são apresentados e pela forma como as funções são ativadas. Aquando da criação de janelas virtuais, é elaborada a sintaxe dos dados. A sintaxe das funções apenas é criada a partir do momento em que se define todas as funções necessárias [15].

O nível físico corresponde ao *hardware* utilizado para o normal funcionamento da interface, como por exemplo monitor, teclado, rato, entre outros [15].

2.1.2 – Design - Janelas Virtuais

As janelas virtuais correspondem a uma apresentação dos dados do sistema de uma forma orientada ao utilizador. Por outras palavras, corresponde às janelas que a interface irá dispor no monitor para interação do utilizador. Numa primeira fase, estas janelas apresentam apenas dados e, com o avançar da criação da interface, vão se acrescentando botões e funções essenciais para o seu funcionamento. Durante a conceção da UI deve ter-se em atenção alguns fatores como [15]:

- Criação de um número mínimo de janelas virtuais essenciais para o funcionamento das principais tarefas. Deve ter em conta a funcionalidade do sistema;
- Disponibilidade de todos os dados;
- Facilidade de interpretação da funcionalidade das janelas virtuais por parte dos utilizadores.

A introdução de funções deve ter em conta alguns passos [15]:

1. Identificar quais as funções semânticas a introduzir para o funcionamento do sistema;
Projetar a janela virtual no monitor;
2. Identificar quais as funções de navegação necessárias para fazer a transição entre as janelas;
3. Desenhar apresentação das funções, adicionando botões, menus, ícones, entre outros.

Tendo em conta a descrição de dados e de tarefas da análise de domínio, são projetadas as janelas virtuais. Estas irão conter detalhes gráficos do sistema, com base nos dados reais. Estas janelas representam a visão que o utilizador tem do sistema.

Posteriormente, são definidas quais as funções necessárias a serem implementadas no sistema, para que o utilizador desempenhe as suas tarefas. Estas funções são implementadas em forma de botões nas janelas virtuais. De seguida, é feita a ligação entre as várias janelas, recorrendo às funções de navegação. Nesta fase é também definida uma lista com a descrição das funcionalidades de todas as funções [15].

A fase seguinte passa pela criação de um protótipo que corresponde à combinação das várias janelas virtuais e respetivas funções.

2.1.3 – Desenvolvimento do Protótipo

Antes de proceder à “programação” da UI, é útil criar um protótipo do sistema, de modo a poder encontrar erros num primeiro esboço desta. Para isso, é necessário criar uma conceção geral do esboço, aprofundado até aos detalhes [19].

O desenvolvimento do protótipo deve respeitar:

- Aparência gráfica definida aquando do planeamento das janelas virtuais;
- Lista de funções de cada página e a sua aparência gráfica;
- Utilidade de cada função.

O processo de desenvolvimento incide inicialmente em decidir qual o tipo de protótipo a desenvolver, tendo em conta análises retiradas anteriormente. Posteriormente elabora-se a versão gráfica de cada página da aplicação. Para isso é necessário tomar em conta a versão gráfica das janelas virtuais, o planeamento das páginas, as apresentações das páginas e a lista de erros recolhida durante o design. O protótipo deve englobar todos os menus e caixas de texto para as mensagens de erros [15].

A introdução de realismo nos objetos presentes nas UI é uma mais valia no auxílio para os utilizadores entenderem o funcionamento do sistema. O estabelecimento de uma relação entre as leis que governam o mundo real e o funcionamento do sistema facilita a sua aprendizagem. Tal pode ser conseguido com a adição de símbolos que representem ações ou ideias do mundo real ou então detalhes reais, sendo uma boa hipótese para a implementação em UI [19][22].

Posteriormente, procede-se à avaliação do protótipo e aos testes de usabilidade. Com isso, pretende-se encontrar potenciais problemas e defeitos numa primeira fase de desenvolvimento, problemas que possam surgir posteriormente com a utilização do sistema. Quanto mais cedo os erros forem detetados, mais facilmente poderão ser corrigidos [15][16][19].

2.1.4 – Programação e testes

Posteriormente, procede-se à implementação técnica do protótipo. É nesta fase que se procede à programação e respetivos testes ao sistema. São programados os componentes

envolventes da UI. Os testes são normalmente feitos a diferentes seções de sistema, sendo posteriormente testado o sistema completo. Com isso pretende-se encontrar defeitos do programa e, se possível, repará-los. Nesta fase, deve-se realizar mais testes de usabilidade, de modo a encontrar erros de usabilidade caso ainda existam [15]

2.2 – Tipos

Nos últimos anos tem-se vindo a observar uma certa evolução nas UI. Inicialmente não passavam de terminais onde eram introduzidos os comandos apropriados para realizar as tarefas por interface de linha de comandos (Command Line Interface - CLI). Este tipo de interfaces caracterizavam-se por não serem *user friendly*, visto que exigiam que o utilizador conhecesse todos os comandos. Deste modo, estes sistemas apenas estavam ao alcance de um pequeno grupo de utilizadores. Posteriormente, surgiram as Interfaces de Utilizador Gráficas (*Graphical User Interface* - GUI), nas quais o utilizador interagia através de elementos como janelas, ícones e menus. Enquanto as CLI são manipuladas com base no teclado, as GUI são manipuladas tendo em conta também o rato. Com os mais recentes desenvolvimentos no campo das UI, têm vindo a surgir as Interfaces de Utilizadores Naturais (NUI). As NUI têm a potencialidade de aproximar cada vez mais os utilizadores às novas tecnologias, deixando-os com um certo à vontade em manusear estas novas interfaces [23][24].

2.2.1 – Interface de Linha de Comandos

As CLI são interfaces nas quais o utilizador interage através de linhas de comandos. Normalmente tem implementada um programa interpretador que aceita comandos de texto como entrada de dados. Esses comandos são então convertidos nas funções apropriadas de interação do sistema operativo [25].

As CLI eram inicialmente utilizadas para diferentes propósitos como administração de sistemas e acesso a recursos da rede. Os primeiros sistemas operativos como *MS-DOS* e *Unix* eram manipulados através de linhas de comandos. Apesar de grande parte das CLI terem sido substituídas pelas GUI, estas ainda continuam a ser utilizadas para desempenharem tarefas mais especializadas, mais eficientes por linhas de comandos. Normalmente são mais

utilizadas quando é necessária a introdução de comandos ou consultas mais específicas e com muitas opções. Neste caso, o processo é mais rápido através deste tipo de interfaces do que com as GUI [25][26].

2.2.2 – Interface Gráfica de Utilizador

As GUI surgiram mais tarde, na década de 80, através da *Apple Inc.* Esta desenvolveu uma interface que utilizava janelas, ícones, menus e apontadores. Estas viriam a ser conhecidas como GUI. Mais tarde, durante a década de 90, a *Microsoft* também viria a implementar as GUI nos seus sistemas operativos, impulsionando a consolidação destas interfaces no mercado [22][27].

As GUI são um tipo de interfaces que permite aos utilizadores interagirem com o sistema, usando objetos em vez de linhas de comandos. Estes objetos encontram-se sempre visíveis ao utilizador e são usados para desempenhar as tarefas necessárias. As ações são normalmente realizadas por manipulação direta dos elementos gráficos. Este tipo de interface permite ao utilizador executar funções apenas por identificação de ícones, sem ser necessário decorar linhas de comandos específicas. Deste modo, o utilizador não precisa de ser especializado no sistema para poder interagir com este [27][28][29][30].

As GUI têm como objetivo um mercado de trabalho mais profissional onde não faz sentido implementar interfaces com recurso a teclados *touchscreen* para elaborar relatórios extensos [23].

2.2.3 – Interface Natural de Utilizador

Nos últimos anos as NUI têm vindo a ganhar cada vez mais destaque no setor das UI. Um dos fatores responsáveis pelo seu crescimento deve-se à sua interação se basear em ações já do conhecimento do utilizador tem conhecimento. Deste modo, a sua aprendizagem é mais intuitiva, o que proporciona um maior conforto para a sua utilização. A sua interação baseia-se na manipulação natural e direta de objetos realistas, pondo de parte as convenções tradicionais das GUI como as janelas, botões e menus. Este tipo de interfaces caracterizam-se por serem de baixa complexidade, assegurando uma aprendizagem mais intuitiva e simples do

sistema. As NUI caracterizam-se por adotarem o uso de superfícies táteis, câmaras reconhecimento de gestos, entre outros, como forma de interação [19][23][26].

As NUI não têm intenção de substituir as GUI, mas sim descobrir novos potenciais nichos de mercado, introduzindo novas formas de interagir com os sistemas. Este tipo de interfaces tem como intuito atrair novos utilizadores, deixando-os numa posição confortável face aos recentes avanços tecnológicos [23].

Com o surgimento das NUI, o uso de rato, *joystick*, teclado, entre outros, já não é muito apropriado para o tipo de interações que estas interfaces requerem. Neste contexto, o mais indicado é o uso de movimentos gestuais através de sistemas de reconhecimento adequados. A interação através de gestos, tanto em superfícies táteis como em interfaces que utilizem câmaras de reconhecimento, devem poder identificar movimentos diretamente, produzindo efeito imediato no ambiente. Este efeito deve ser similar aos movimentos que o utilizador produz no seu dia-a-dia, de modo a facilitar a sua compreensão. O facto de este tipo de interação possibilitar ao utilizador manipular diretamente os objetos, ao invés de ter que conhecer os comandos próprios para tais operações, facilita o seu uso, permitindo focar-se apenas nas suas tarefas. Este tipo de interação denomina-se manipulação direta. Esta forma de manipulação de objetos é obtida através de recursos semelhantes aos usados no mundo físico. No entanto, o uso de gestos pode trazer alguns problemas como nos casos em que a interface exija um movimento específico para desempenhar uma determinada tarefa e não dê *feedback* enquanto não for feito na perfeição. Por esta razão, importa que os gestos permitidos sejam simples e intuitivos. O facto de a UI fornecer *feedback* imediato, ao confirmar a captação do movimento, facilita a tarefa do utilizador. É importante que não seja necessário que o utilizador aprenda gestos complexos e excessivamente precisos [19][22][23][24][26][31].

2.3 – Meio Hospitalar

Apesar do crescimento da implementação de UI em meios hospitalares, estas ainda não se encontram completamente implementadas. Este facto deve-se, em parte, a problemas financeiros e institucionais, assim como a problemas ao nível de construção de interfaces. Com vista ao desenvolvimento de interfaces especializadas para a prática médica, é necessário ter em conta os diferentes tipos de utilizadores (e.g. médicos, enfermeiros ou

utentes) e as respetivas tarefas. Assim, é importante proceder a uma análise das características dos utilizadores para, posteriormente, analisar as suas tarefas, de modo a identificar as funções que o sistema deve possuir. Deste modo, é possível desenvolver uma UI médica adequada ao tipo de serviço [32].

Estes princípios devem ser seguidos desde o início do seu desenvolvimento. Estas medidas têm o intuito de facilitar a sua aprendizagem e aumentar a satisfação, aceitação e consequentemente a produtividade do utilizador. O seu efeito traduz-se numa diminuição do número de erros e tempo necessário para aprendizagem do sistema [3][32][33].

O uso deste tipo de tecnologias permite dar uma melhor resposta ao crescente número de informação nas instituições de saúde, disponibilizando ferramentas para auxílio da sua gestão. Consequentemente, o uso deste tipo de tecnologias auxilia o processo de tomada de decisões dos profissionais de saúde, fornecendo informações adequadas ao diagnóstico dos utentes no momento certo [3].

Inicialmente, para o desenvolvimento de UI hospitalares é necessário identificar qual o tipo de utilizador alvo, de modo a começar a conceber o produto. Posteriormente, é feito um conjunto de análises como [3]:

- Utilizador - Examinar as características do utilizador;
- Ambiente - Examinar ambiente de trabalho do utilizador (social e cultural);
- Tarefas – Examinar tarefas e objetivos;
- Funcional – Examinar as estruturas de trabalho e as atividades cognitivas do utilizador;
- Representação – Examinar a forma de representação da informação ao utilizador.

Da análise de utilizador pretende-se recolher informação sobre o utilizador alvo da interface a ser desenvolvida. Este tipo de informação auxiliará a construir uma aplicação em torno das características identificadas [3][34].

Durante a análise de tarefas são analisados os objetivos dos utilizadores e a forma como é feita a sua interação com o sistema. A identificação das funções do sistema, formatos de entrada e saída de dados, categorias de informação, as necessidades de informação do utilizador e as suas limitações são parâmetros que se pretende identificar [3][34].

A forma como a interface apresenta a informação ao utilizador, tendo em conta as suas tarefas, é definida na análise representacional. Nesta fase são identificadas quais as formas mais simples para fornecer a informação necessária. Normalmente, este tipo de análises tem

lugar durante o processo de desenvolvimento inicial, tendo como intuito fornecer informações essenciais para o seu desenvolvimento, de modo a evitar erros o quanto antes [3].

Nas instituições de saúde é também importante ter um sistema eficiente de troca de informação entre os diferentes serviços. Para os profissionais de saúde é vital que a troca de informação seja organizada e precisa, dentro dos prazos estabelecidos. Atualmente existem duas abordagens essenciais para o devido efeito (Figura 2.2). Estas podem ser: 1) através da criação de UI ponto-a-ponto entre as aplicações internas com as aplicações dos diversos fornecedores ou, então, 2) através da implementação de um motor de UI com mecanismos que possibilitem a correta comunicação de dados entre as diversas aplicações internas e as aplicações dos fornecedores externos. Como exemplos de serviços que necessitam destes tipos de interfaces temos: o registo médico eletrónico (*Electronic Medical Record - EMR*), o sistema de informação do hospital (HIS), o sistema de informação de radiologia (*Radiology Information System - RIS*), o sistema de informação dos laboratórios (*Laboratory Information System - LIS*), o departamento de urgência, o sistema de arquivo e distribuição de imagem (*Picture Archiving and Communication System - PACS*), entre outros [35][36].

Nas interfaces ponto-a-ponto, cada serviço especifica o tipo de dados e o respetivo formato que suporta, estabelecendo uma relação de um para um. Deste modo, cada serviço que emita dados terá de possuir uma interface específica para trocar dados com os diferentes recetores. Tendo em conta a relação custo/eficiência, é uma boa solução para ambientes pequenos com um ou dois serviços. Para ambientes maiores não se justifica, visto que a ligação é feita ponto-a-ponto, o que iria proporcionar um aumento de custo, à medida que se iriam adicionando serviços. Além disso, não permite monitorizar o estado das ligações dos diferentes serviços [35][37].

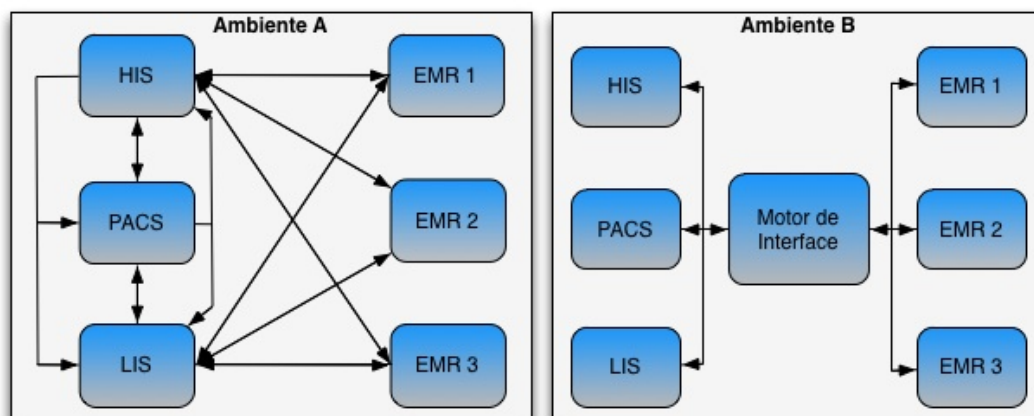


Figura 2.2 – A - Ambiente com implementação Interface ponto a ponto; B – Ambiente com implementação de Motor de Interface. Adaptado de [35].

Com implementação de um motor de interface, pretende-se fazer uma ligação dos diferentes serviços, utilizando um protocolo *standard* de mensagens de modo a obter um mapeamento do dados. Um dos protocolos mais utilizados é o *Health Level 7* (HL7) aprovado pelo Instituto Americano de Padrões (*American National Standards Institute - ANSI*). O padrão HL7 fornece uma estrutura que permite a troca, integração e recuperação de informação médica entre os diferentes serviços [35][36][38][39].

CAPÍTULO 3

Reconhecimento Gestual

Ideias chave:

- O reconhecimento de gestos consiste no reconhecimento de qualquer movimento físico a ser identificado e interpretado por um sistema.
- Necessidade de perceber o papel das formas gestuais na comunicação entre humanos.
- Classificação dos gestos por categorias.
- Os gestos são constituídos pelas componentes de movimento, atividade e ação.
- O modelo computacional dos gestos pode ser dividido em dois grupos: Modelos 3D e modelos baseados na aparência.
- Os dispositivos de reconhecimento gestual são compostos por três componentes: Sensor, comparador e actuador.

O capítulo anterior abordou a questão das interfaces de utilizador, seguindo-se agora um espaço para trabalhar a temática do reconhecimento gestual. Como o nome indica, reconhecimento de gestos consiste no reconhecimento de caracteres, símbolos e todo tipo de interações. Diz respeito à forma como a interação é estabelecida entre o utilizador e a interface. A forma como são definidos os gestos irá contribuir para a qualidade com que é feita a interação com a interface. Os gestos podem-se definir como qualquer tipo de movimento físico possível de ser identificado e interpretado por um sistema, de modo a fornecer uma resposta. O movimento físico pode ir desde um simples toque a um gesto com o braço [31][40].

Empresas como a *Nintendo*, *Apple*, *Nokia*, *Samsung* e *Microsoft* têm vindo a lançar novos dispositivos, os quais utilizam gestos como forma de interação. A crescente exposição do consumidor a este tipo de dispositivos tem vindo a influenciar este processo. Estima-se que nos próximos anos o desenvolvimento destes dispositivos continue a crescer, influenciando o modo de vida do consumidor. São exemplos destes tipos de dispositivos a consola *Wii* da *Nintendo*, capaz de reconhecer gestos a três dimensões, através de um acelerómetro, o dispositivo *Kinect* capaz de fazer o reconhecimento de gestos de forma livre, os *Kiosks* implementados nas caixas multibanco ou então o *iPhone* da *Apple* que permitem controlar objetos através de uma superfície tátil [23][31].

Atualmente, as UI de reconhecimento gestual podem ser divididas em duas categorias: *touchscreen* e *free-forms*. A interação com a interface através de superfícies táteis é feita através do toque direto no monitor. A interação com as interfaces *free-forms* é feita de forma livre através de gestos, não necessitando de toque direto no dispositivo.

A interação direta consiste na manipulação dos objetos presentes no ambiente do sistema sem recorrer a linhas de comandos para implementar instruções. Este tipo de interação estabelece uma maior aproximação entre o ambiente virtual computacional e o mundo real físico. Este tipo de interação é notável nas interfaces de superfícies táteis e interfaces gestuais, onde o utilizador utiliza o seu próprio corpo para manipular o ambiente. Quando comparada com os sistemas tradicionais, as UI gestuais oferecem ao utilizador, uma variedade mais ampla de possíveis ações para manipular o sistema [31].

3.1 – Evolução tecnológica

O primeiro monitor *tátil* foi desenvolvido em 1967. Este monitor utilizava fios elétricos incorporados no monitor que permitiam detetar perturbações no fluxo elétrico em cada uma das posições definidas do monitor. Durante a década de 70, continuaram-se a desenvolver estudos sobre a tecnologia *touchscreen*, sendo notória, já na altura, o potencial da técnica. A primeira UI gestual surgiu em 1971, através de Samuel C. Hurst e denominava-se *Elograph*. Em 1974, Samuel Hurst criou a companhia *Elographics*, que viria a ser responsável por desenvolver a tecnologia *five-wire resistive*, muito utilizada ainda nos dias de hoje. O primeiro dispositivo com superfície tátil surgiu em 1977, através da *Elographics* apoiada pela Siemens. Este dispositivo tinha como nome *Accutouch*, sendo a sua superfície de contato curva e em

vidro. Apesar disso, não se verificou uma grande aceitação devido à percepção de se tratar de uma tecnologia cara e pouco fiável [31][41][42].

No final da década de 70, Myron Krueger criou o primeiro sistema interativo de manipulação indireta denominado de *Videoplace*. Este sistema permitia ao utilizador interagir com o sistema através de gestos. O sistema era composto por um sistema de projetores, câmaras de vídeo e *hardware* específico [31].

Os primeiros dispositivos a fazerem uso das superfícies táteis começaram a ser comercializados durante a década de 80, sendo utilizados em restaurantes, postos de vendas, hospitais, entre outros. A impressora Xerox 5700 foi dos primeiros dispositivos a fazer uso de uma UI manipulada através de uma superfície tátil. Um dos primeiros computadores destinado a uso pessoal, a ser comercializado e que fazia uso da tecnologia *touchscreen* foi o *Hewlett-Packard* 150. Este computador permitia ao utilizador interagir com o sistema através do toque no monitor. Desde então tem surgido um grande número de dispositivos a fazer uso das superfícies táteis [31][41][43].

Em 1982, Nimish Mehta da Universidade de Toronto desenvolveu um dos primeiros sistemas multi-toque denominado *Flexible Machine*. Este projeto foi desenvolvido no âmbito de projeto da tese de mestrado. Este sistema consistia na criação de uma grelha ao longo do monitor, com atividade elétrica ao longo de eixos horizontais e verticais. Conforme o número de toques, eram afetadas determinadas colunas e linhas da grelha, sofrendo cada uma alterações de capacitâncias. Deste modo, obtinha-se as coordenadas dos diferentes locais. Este sistema permitia ao utilizador interagir com o sistema através de mais do que um ponto de contato em simultâneo com o monitor [31][41][43].

A primeira UI de gestos baseada numa luva surgiu em meados de 1982 através de Thomas G. Zimmerman. Este dispositivo incorporava um sensor ótico flexível desenvolvido por Young Harvill, incorporado numa luva que permitia medir a flexão do dedo. Posteriormente, juntamente com Jaron Lanier, incorporaram a luva com tecnologia de rastreamento magnético e ultrassónico da posição da mão. O seu funcionamento consiste em medir as perdas de luz que a luva liberta, através de ranhuras, durante a flexão. Este processo permite medir a posição da mão. Posteriormente, em 1987, a empresa de videojogos Nintendo comercializou o primeiro dispositivo de rastreamento de gestos que tinha como nome Power Glove. Este dispositivo consistia numa luva ligada à corrente com sensores de flexão de dedos e botões incorporados na parte de trás da mão. Em 1989, Kramer desenvolveu o *CyberGlove* (Figura

3.1), dispositivo esse que consistia numa luva incorporada com extensômetros entre os dedos de modo a medir deslocações destes. No entanto, devido ao seu elevado preço e devido ao incómodo de necessitar do uso da luva para o seu funcionamento, nunca conseguiu ter grande aproximação ao utilizador [40][44].



Figura 3.1 - Dispositivo *Cyberglove*. Imagem retirada [40].

No início da década de 90, Pierre Wellner desenvolveu a *Digital Desk*, sendo dos primeiros dispositivos a fazer uso de gestos. Este dispositivo consistia num conjunto de câmaras e um projetor que iria projetar uma superfície digital sobre uma mesa. A superfície digital permitia ao utilizador interagir com o ambiente da interface. A partir da década de 90, verificou-se um crescente interesse nestes tipos de dispositivos com a sua implementação em caixas multibanco, terminais de *check in* em aeroportos, entre outros [31].

Em 2001, a *Lionhead Studios* desenvolveu o jogo *Black & White*, que permitia usar gestos através de uma luva própria para controlar o personagem. No mesmo ano, a *Konami* lançou um jogo de boxe denominado de *MoCap Boxing* (Figura 3.2). Neste jogo, quando posicionado a uma certa distância e com umas luvas de boxe colocadas, o sistema reconhecia o utilizador através de sensores infravermelhos de movimento, permitindo a este lutar contra adversários virtuais através dos seus movimentos. Em 2003, a Sony lança para o mercado a câmara *EyeToy* para a consola *Playstation 2*. Este dispositivo permitia reconhecer os gestos do utilizador de forma a controlar o jogo. Posteriormente, em 2006, a *Nintendo* lançou a consola *Wii*. Em resposta, a Sony desenvolveu o *Playstation Move* para a consola *Playstation 3*, no qual os princípios de aquisição de dados gestuais são idênticos aos utilizados pela *Wii*. Só mais tarde, em 2010, a *Microsoft* lançou a câmara *Kinect* para a consola Xbox, tendo surgido posteriormente, a 1 de Fevereiro de 2012, uma versão para a plataforma *Windows*. Esta tinha

a particularidade de permitir controlar o jogo através de gestos livres e de comandos de voz captados por uma câmara [31][45][46][47].



Figura 3.2 - Jogo *MoCap Boxing* desenvolvido pela *Konami*. Imagem retirada de [31].

Em 2007, surgiu o *iPhone* e o *iPod Touch* pelas mãos da *Apple*. Influenciados pelo grande sucesso dos dispositivos da *Apple*, em 2008, empresas como a *LG*, *Sony Ericsson* e *Nokia* desenvolveram os seus próprios, *Smartphones* com tecnologia *touchscreen* incorporada. No mesmo ano, a *Microsoft* lançou o *Tablet MS Surface*. A *Apple* só viria a lançar o seu *Tablet PC* conhecido por *iPad*, em 2010. Com o surgimento destes dispositivos móveis, o interesse dos utilizadores nestes tipos de dispositivos disparou. Desde então o desenvolvimento de UI gestuais não tem parado, surgindo a cada ano que passa novos tipos de dispositivos [31][41]. Este tipo de tecnologia tem vindo a ser introduzido no ramo da saúde. Recentemente, *Wachs et al.* [48] desenvolveu um método estéril para manipular e navegar entre as imagens de radiologia sem necessitar de contato direto. Através de movimentos livres das mãos, o médico consegue controlar o *software* de imagem sem necessitar de estabelecer contato. Deste modo, o médico consegue controlar a informação necessária sem contaminar o utente. *Ruppert et al.* [49], desenvolveu um método idêntico para manipular e navegar entre as imagens sem necessitar de contato direto em cirurgia urológica (Figura 3.3). Este método utiliza a câmara *Kinect* desenvolvida pela *Microsoft*. O sistema caracteriza-se por ser de simples implementação, de baixo custo e de código aberto.



Figura 3.3 - Demonstração da aplicação do método para manipular e navegar entre as imagens de radiologia sem necessitar contacto direto. Retirado de [47].

O reconhecimento de gesto também tem sido utilizado em aplicações de manipulação de objetos em ambientes 3D. É o caso da aplicação *Fusion4D* (Figura 3.4) desenvolvida por Matsumura, Sonnino *et al.* [50] no âmbito do projeto Vida que utiliza o dispositivo *Kinect* como forma de interação com a UI através de gestos e voz. O *Fusion4D* permite ao utilizador manipular modelos anatómicos 3D com as próprias mãos através de um ambiente virtual. Permite realizar operações como mover, rodar, ampliar e particionar o objeto. Também permite navegar ao longo de uma linha temporal, obtendo uma noção da evolução do objeto ao longo do tempo. Para interagir com a interface é necessária a utilização de óculos 3D. A Figura 3.4 ilustra a forma de interação com a UI desenvolvida.

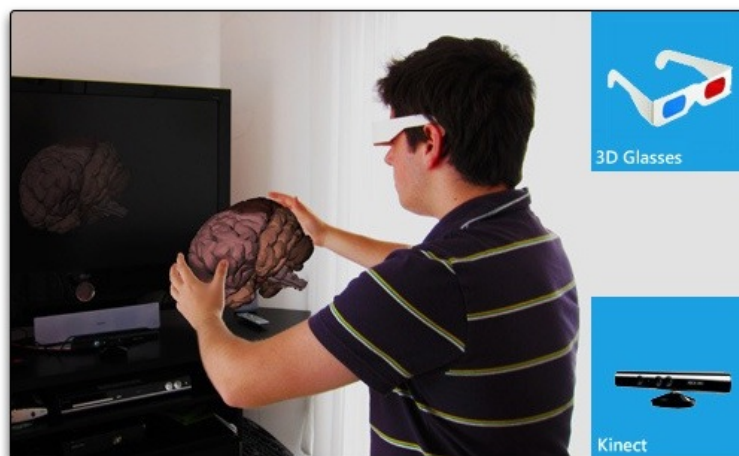


Figura 3.4 - Ilustração do funcionamento da UI Fusion4D. Retirado de [50].

Apesar de se encontrar ainda numa fase inicial, aplicação deste tipo de tecnologias na área da saúde tem mostrado ser uma mais valia no auxílio da obtenção do diagnóstico, revelando uma utilização mais eficiente dos recursos.

3.2 – Linguagem Gestual

A linguagem gestual diz respeito aos movimentos do corpo que podem ser interpretados como informação útil para a interação com o sistema. No entanto, numa primeira fase, é necessário entender qual o papel das diferentes formas gestuais na comunicação entre humanos. Neste contexto, a comunicação é realizada, em certa medida, através de gestos que podem variar de cultura para cultura [40].

No que toca aos objetos, movimentos como apontar, tocar, arrastar, entre outros, fazem parte de uma lista de gestos que é possível realizar. Assim é necessário classificar os gestos de acordo com a sua função [40][51].

Segundo Claude Cadoz [52], podem ser divididos em três grupos:

- *Semiotic* – Gestos usados para transmitir informação útil;
- *Ergotic* – Gestos usados para manipular objetos presentes no mundo real;
- *Epistemic* – Gestos usados para recolher informação do ambiente através do toque.

Dentro os gestos *semiotics*, estes podem ser classificados como:

- Simbólicos - Gestos os quais têm significados diferentes de cultura para cultura;

- Dêiticos - Gestos que indicam direções;
- Icônicos - Gestos que permitem estabelecer relação entre tamanhos, formas ou orientação de objetos.

Atualmente, grande parte da interação com os computadores é realizada através de manipulação direta. Deste modo, a aplicação de gestos para manipulação dos sistemas poderia ser melhor explorada, uma vez que os humanos se encontram plenamente familiarizados com este tipo de interação, ao contrário de ter que aprender uma nova forma de interação [40].

No domínio da HCI, os gestos são utilizados para simular o uso natural da mão como manipulador de gestos. Cada gesto pode ser interpretado como uma trajetória num determinado espaço 3D. Segundo Quek [53] (Figura 3.5), para o reconhecimento de gestos, os movimentos de mãos e membros podem ser classificados em dois grupos: gestos e movimentos involuntários. Dentro dos gestos estes podem ser ou manipulativos ou comunicativos. Os manipulativos são gestos que servem para manipular os objetos presentes no ambiente da UI, enquanto os comunicativos, podendo ser símbolos ou atos, servem essencialmente para comunicar. Os gestos simbólicos têm intencionalidade linguística. Os atos estão relacionados com o movimento da mão como por exemplo, apontar para um objeto. Dependendo do tipo de UI, os gestos e movimentos podem ser estáticos ou então movimentos mais sofisticados, utilizando modelos 3D. Estes podem consistir em posições simbólicas das mãos, movimentos de mão ou a combinação entre ambos [40][44].

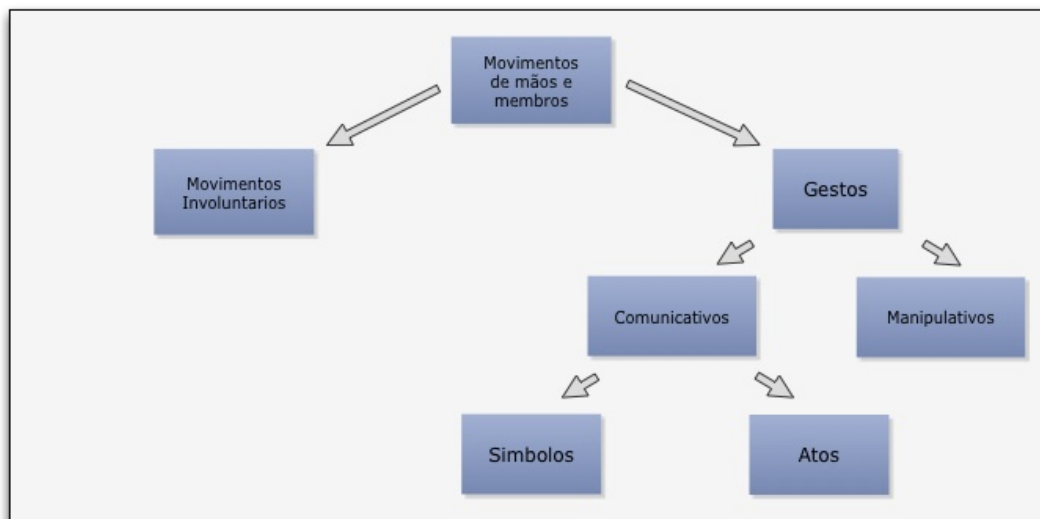


Figura 3.5 – Classificação de Gestos e Movimentos segundo Quek [53].

A linguagem gestual insere-se nos gestos comunicativos e caracteriza-se por ser muito estruturada. Os gestos de controlo, muito usados como gestos de navegação, servem de ponto de interação em ambientes virtuais. Os gestos manipulativos servem essencialmente como forma de interação com objetos virtuais. Gestos comunicativos são mais indicados para proceder à interação humana [24].

O reconhecimento de gestos é uma técnica que abrange a modelação e análise de movimentos, reconhecimento de padrões e *machine learning*. Os quatro aspetos que representam o reconhecimento gestual são a forma, posição, orientação e o movimento da mão [24].

3.3 – Tecnologia *Touchscreen*

As superfícies táteis permitem detetar a presença e a localização do toque na área de contato. A interação com este tipo de superfícies é possível de ser realizada através do toque do dedo humano ou através de uma caneta específica. A sua utilização é bastante simples para o utilizador, uma vez que o processo é bastante intuitivo e natural, assemelhando-se a operações do mundo físico. A tecnologia *touchscreen* apresenta características ideais para uso em UI devido à forma eficiente e simples com que permite que seja realizada a interação por parte do utilizador [26][43][54][41].

Este tipo de tecnologia surgiu em 1960, mas apenas nos últimos tempos se tem observado uma maior inserção no mercado como forma de interação com os sistemas. A implementação da tecnologia *touchscreen* veio despertar um novo nicho tecnológico, permitindo explorar potencialidades das novas UI. Tem-se registado um aumento significativo de dispositivos incorporados com este tipo de tecnologia. Esta expansão também se tem verificado no sector da saúde, sendo considerado uma mais valia para a melhoria contínua dos serviços prestados [26][43][41].

Um grande número de dispositivos e sistemas disponíveis oferecem a possibilidade de interação através de gestos, disponibilizando mecanismos apropriados para captar, rastrear e traduzir os movimentos em dados. Os fatores que caracterizam as UI táteis são tempo de resposta, qualidade e precisão do estímulo do toque, consistência e robustez [55].

A tecnologia *touchscreen* permite ao utilizador interagir diretamente com os objetos presentes no ambiente da interface. Este tipo de tecnologia tem vindo a beneficiar de novas formas de contato como é o caso do múltiplo toque, fornecendo uma melhor interação entre o utilizador e o computador. Este tipo de interação permite que seja feita através de vários toques no ecrã, reconhecendo múltiplos pontos de contato por um ou vários utilizadores em simultâneo. As superfícies táteis beneficiam o utilizador em relação ao uso do rato, uma vez que este apenas permite um gesto de cada vez como entrada[43].

De modo a tirar máximo proveito do potencial da tecnologia *touchscreen*, é necessário criar interfaces específicas para tal. Esse tipo de interfaces denominam-se de NUI. Estas permitem a utilização de múltiplos pontos de contato e que a introdução de dados seja realizada tendo em conta gestos simples e intuitivos como rotação ou expansão. A tecnologia *touchscreen* implementada numa interface apropriada, oferece um nível de interação superior ao do uso do rato devido ao facto da sua interação se aproximar da forma como o utilizador interage com objetos reais. No entanto, o mesmo já não se pode dizer quanto ao teclado, uma vez que os caracteres de texto não detêm carácter físico para serem manipulados por toque. Uma hipótese seria a introdução de teclados na própria superfície de contato mas, como já foi anteriormente dito, devido à forma do dedo humano, este processo seria impreciso e lento. Uma outra hipótese seria a introdução de sistemas de reconhecimento de voz, apesar de estes sistemas exigirem algum tempo de aprendizagem de reconhecimento da voz do utilizador e exigir um ambiente sem muitas interferências de sons de fundo [43].

No entanto este tipo de tecnologia também apresenta algumas desvantagens como o facto de não ser precisa, causar fadiga ao utilizador, estar mais sujeita a sujidade, entre outros. A baixa precisão deve-se, em parte, ao facto de a interação ser feita através do dedo que apresenta elevada dimensão. Também devido ao facto de necessitar de um maior movimento dos membros do utilizador, pode causar fadiga, caso utilizado por um longo período de tempo. Como a interação é feita através do dedo humano, o dispositivo fica sujo muito mais facilmente [43][41].

Atualmente a tecnologia *touchscreen* apresenta vários sistemas de reconhecimento de toque. Estas soluções apresentam vantagens e desvantagens conforme o tipo de necessidades do sistema. O tipo de tecnologia mais utilizada são as superfícies resistivas e capacitivas, existindo, no entanto, outras soluções como superfície por ondas acústicas, monitor de infravermelhos, *frustrated total internal reflection* (FTIR), entre outros, menos utilizados [43][54].

Tabela 3.1 – Características das diferentes tecnologias Touchscreen. Adaptado de [56].

Características	Resistivo	Capacitivo	SAW	Infravermelhos	FTIR
Durabilidade	Boa	Muito boa	Excelente	Excelente	Excelente
Multi-toque	Não	Sim	Não	Sim	Sim
Uso luva	Sim	Não	Sim	Sim	Sim
Custo Produção	Baixo	Elevado	Baixo	Elevado	Muito baixo

3.3.1 – Resistivo

As superfícies *táteis* com tecnologia resistiva (Figura 3.6) consistem em duas camadas, uma metálica condutora e outra resistiva, separadas por um espaço pequeno. Durante o funcionamento do monitor, as duas camadas são percorridas por uma corrente elétrica. Quando se realiza o toque no monitor, as duas camadas entram em contato e o campo elétrico é alterado. Posteriormente esta informação é captada por um controlador, através de sensores, que interpreta como uma corrente bloqueada e calcula as coordenadas do local onde ocorreu contato [41][43][54][56].

Este tipo de tecnologia é utilizada por exemplo em *Kioks* e na consola *Nintendo DS*. Caracteriza-se por oferecer tempos de resposta bastante rápidos, não ser afetada por

elementos como sujidade e pó e os custos de produção serem baixos, tendo em conta as outras tecnologias existentes. No entanto, não reconhece múltiplos toques [41][43][56][57].

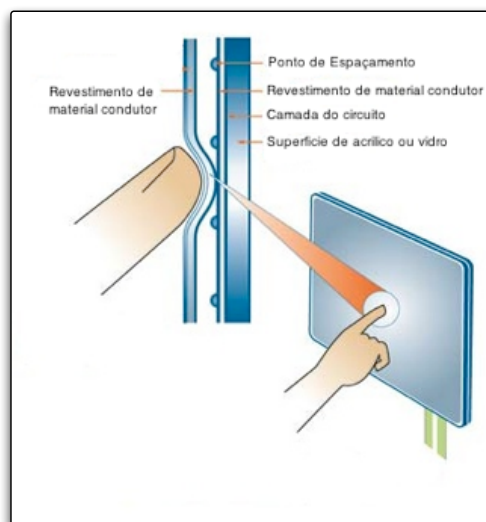


Figura 3.6 - Superfície tátil com tecnologia capacitiva. Retirado de [56].

3.3.2 – Capacitivo

As superfícies táteis com tecnologia capacitiva (Figura 3.7) caracterizam-se por permitirem captar o toque através de pequenas correntes elétricas emitidas por estas ações. Posteriormente, estas correntes são captadas por um controlador através da medição do fluxo em cada canto, o que permite calcular o ponto de contato. A superfície é composta por um painel transparente revestido por um material condutor que é estimulado aquando do contato com um objeto capacitivo como o dedo. Deste modo, não é possível manusear este tipo de superfícies com dispositivos que sejam compostos por materiais que não sejam condutores elétricos. Pode-se afirmar que as superfícies capacitivas correspondem ao oposto dos resistivos, no que refere ao tipo de tecnologia utilizada [41][43][56].

Esta tecnologia apresenta elevados custos de produção, em parte devido aos custos dos mecanismos para procederem ao processamento de correntes elétricas. Apesar dos elevados preços, este tipo de monitores caracteriza-se por possuir um alto tempo de duração. As superfícies capacitivas caracterizam-se por fornecerem máxima durabilidade. Este tipo de tecnologia é utilizado em dispositivos móveis e *Kiosks*, entre outros [41][43][56][57].

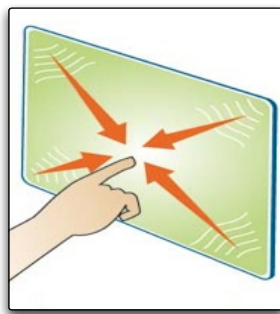


Figura 3.7 - Superfície tátil com tecnologia capacitiva. Retirado de [56].

3.3.3 – Monitor de Ondas Acústicas (SAW)

Os monitores de ondas acústicas (SAW) (Figura 3.8) consistem em superfícies que transmitem ondas de som de alta frequência (impercetível para o ouvido humano) ao longo de uma das arestas verticais e horizontais do monitor. As ondas de som são redirecionadas ao longo do monitor para um conjunto de dois sensores que se encontram nas arestas verticais e horizontais opostas. O seu funcionamento assenta no facto de que, uma vez que a velocidade do som é constante, quando sujeito ao toque, parte da energia é absorvida, causando uma baixa na amplitude da onda de som. Esta diferença é captada pelos sensores, permitindo reconhecer a posição ao longo dos eixos, fornecendo as coordenadas X e Y de posição de contato [41][43][56].

Os SAW apresentam alta durabilidade sendo muito usados em ambientes onde afluem muitos utilizadores. No entanto, este tipo de monitores são muito vulneráveis à presença de poeiras e água devido ao facto de não serem completamente selados. Muito utilizados em terminais multibanco e instituições de saúde [41][43][56].

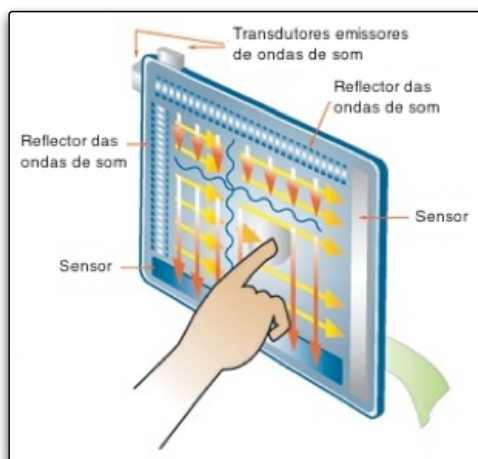


Figura 3.8 - Superfície tátil com tecnologia de ondas acústicas. Retirado de [56].

3.3.4 – Monitor Infravermelhos

A tecnologia de monitores de infravermelhos (Figura 3.9) consiste na criação de uma grelha vertical e horizontal de LED's infravermelhos e na introdução de pares de sensores de luz nos cantos do monitor. Estes sensores têm a função de detetar interrupções nos feixes de infravermelhos dos LED's, traduzindo estas interrupções em localizações espaciais X e Y [41][43][56].

Este tipo de tecnologia permite reconhecer múltiplos toques em simultâneo. No entanto, são influenciados pela presença de poeiras e sujidade, devido a estes poderem influenciar os feixes infravermelhos. Outro fator negativo deste tipo de tecnologia é o facto de ter baixa precisão, não sendo por isso muito utilizado [41][43][56][57].

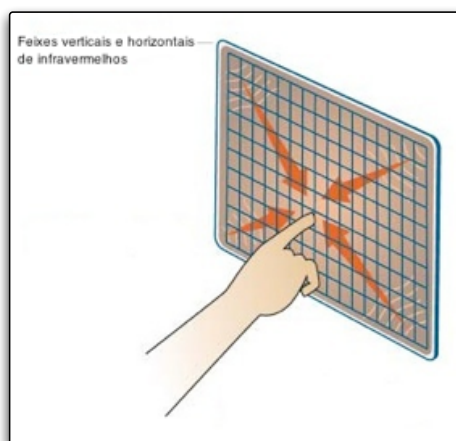


Figura 3.9 - Superfície tátil com tecnologia de infravermelhos. Retirado de [56].

3.3.5 – *Frustrated Total Internal Reflection*

Outra tecnologia é a *Frustrated Total Internal Reflection* (FTIR) (Figura 3.10). Esta técnica consiste em emitir luz infravermelha para o meio do monitor, através de LED's localizados nas bordas. Normalmente o meio para o qual a luz infravermelha é emitida é em vidro. No entanto, como apresentado no estudo em [58], pode ser usada uma superfície em acrílico em vez de uma superfície de vidro, uma vez que o acrílico apresenta propriedades de propagação de luz superiores às do vidro. Os extremos encontram-se polidos de forma a gerarem um meio ótimo de propagação de ondas de luz. A superfície é irradiada por LED's infravermelhos, que se encontram localizados nos extremos, de modo a maximizar a reflexão interna total. A luz é mantida no interior da superfície de vidro até que esta esteja sujeita a contato através de toque, causando dispersão da luz para fora da superfície. Esta luz é posteriormente captada por uma câmara que permite literalmente ver a localização do toque, invés de proceder a cálculos para determinar a sua posição. A câmara identifica o toque como ponto branco num ecrã de fundo escuro. Através de técnicas de processamento, permite determinar a localização e o tamanho do ponto [41][43][58].

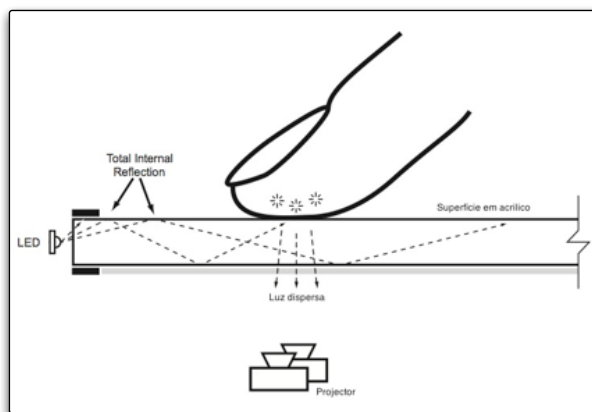


Figura 3.10 - Superfície tátil com tecnologia de FTIR. Retirado de [56].

O FTIR apresenta reduzidos custos de produção sendo, atualmente, utilizada no *Tablet PC Surface* da *Microsoft*. Nestes dispositivos, as câmaras encontram-se localizadas na base do dispositivo. Uma das desvantagens desta tecnologia é o facto de necessitar de um certo espaço entre a câmara e a superfície de contato [41][43].

3.4 – Free-Forms

Os gestos são constituídos pela componente de movimento, atividade e ação. O movimento diz respeito à forma primitiva de movimento que é interpretada; atividade é a sequência de movimentos e configurações estáticas; ação diz respeito às entidades usadas para descrever o acontecimento. Para proceder ao reconhecimento de gestos, é necessário fornecer configurações dinâmicas ou estáticas dos membros, de modo a permitir ao sistema interpretar a posição e os movimentos [24][44].

Os primeiros dispositivos a possibilitarem captar movimentos eram dispositivos com base numa luva que permitiam medir diretamente posições espaciais das mãos. Posteriormente, surgiram técnicas de captação de gestos através de métodos baseados em técnicas de visão computacional. Para isso contribuiu o surgimento de computadores de alto processamento e os avanços no campo da visão computacional, permitindo realizar processamentos de imagem em tempo real. Inicialmente as técnicas de reconhecimento apenas permitiam captar gestos de mãos estáticos e posturas, surgindo apenas posteriormente o reconhecimento de gestos dinâmicos [44].

Este tipo de tecnologia tem vindo a ser introduzida inclusive na área da saúde. Apesar das inúmeras vantagens do uso de ecrãs táteis, estes tem revelado ser um condutor de inúmeras infeções. A implementação de dispositivos de reconhecimento gestual com base em métodos de visão computacional pode revelar-se uma possível alternativa ao ecrã tátil, uma vez que não necessita de contacto direto [59].

É possível classificar as diferentes técnicas de reconhecimento de gestos em dois grupos: técnicas baseadas no uso de luva e técnicas baseadas em visão computacional. Os métodos baseados em luvas utilizam sensores incorporados na luva para digitalizar os movimentos da mão e dos dedos em dados multiparamétricos. Este tipo de tecnologia peca pelo facto de requerer uma utilização constante da luva por parte do utilizador de modo ao sistema poder desempenhar as funções de captação dos movimentos. Os métodos baseados em visão computacional oferecem uma experiência mais natural, fazendo o reconhecimento gestual através de técnicas de visão computacional que recorrem a câmaras. A captação de movimentos é conseguida através de uma câmara com reconhecimento de profundidade. No entanto esta tecnologia ainda apresenta algumas limitações. O facto da resolução das câmaras existentes ainda ser muito baixa, não permite proceder ao reconhecimento correto da mão e ao mesmo tempo cobrir o campo de visão estabelecido pelos movimentos dos membros. Além disso, a tecnologia de vídeo de 30 ou 60 *fps* não permite capturar movimentos rápidos [40][60][61].

Com o surgimento de novas formas de interação com as UI tem-se verificado que os dispositivos existentes como rato e teclado têm vindo a tornar-se pouco úteis, uma vez que não permitem realizar uma rápida e natural interação com a interface. O reconhecimento de gestos abrange um conjunto de técnicas concebidas de modo a reconhecer e interpretar o movimento das membros do utilizador. Os gestos podem ir desde apontar o dedo ou manipular objetos a formas de comunicação por gestos [40][44].

O reconhecimento computacional apresenta duas possibilidades para o reconhecimento de gestos. Técnicas baseadas em modelos, que fazem o reconhecimento da mão através da criação de um modelo tridimensional e técnicas baseadas em imagem nas quais, através da imagem da mão, conseguem calcular e reconhecer aspetos diretamente da imagem [40].

Normalmente, os sistemas de reconhecimento de gestos são compostos por três módulos. O primeiro é responsável por realizar a segmentação dos dados e rastreamento, permitindo adquirir e localizar membros e mãos ao longo da captação. O segundo módulo realiza a

extração das características essenciais, de modo a preparar os dados para posteriormente serem classificados pelo módulo de reconhecimento [44][61].

3.4.1 – Características de Reconhecimento

Para realizar o reconhecimento de gestos é necessário selecionar as características ideais dos membros a captar. A mão, por exemplo, tem várias formas, movimentos e texturas. No caso das posturas da mão, apesar de ser possível a identificação através das características geométricas como os contornos, nem sempre é possível devido a condições de luminosidade ou então por esta se encontrar parcialmente oculta. Deste modo, é necessário especificar o que a UI irá reconhecer para realizar a interpretação. É usada a imagem como dado de entrada de modo a permitir ao reconhecedor selecionar as características ideais [44][61].

3.4.2 – Modelos de Reconhecimento

Os principais modelos utilizados para o modelamento espacial dos gestos podem ser divididos em dois grupos [44]:

- Modelos 3D;
- Modelos baseados na aparência.

Para obter modelos de gestos de mão recorre-se a modelos 3D dos membros e mãos. Estes encontram-se divididos em dois grupos: modelos volumétricos e modelos esqueléticos. A diferença é que enquanto os modelos volumétricos descrevem aparência visual 3D dos membros, os modelos esqueléticos utilizam um reduzido conjunto de parâmetros, gerando apenas ângulos das articulações e o tamanho dos segmentos. Os modelos volumétricos são utilizados para proceder ao rastreamento e reconhecimento da postura do corpo através da abordagem análise por síntese. Esta abordagem sintetiza o modelo do corpo a três dimensões, variando certos parâmetros de modo a analisar a postura do corpo. O processo termina quando o modelo se assemelhar ao modelo real. De modo a representar certas partes da mão e dos membros, podem ser utilizadas estruturas semelhantes como cilindros, esferas e retângulos. Um dos problemas dos modelos volumétricos prende-se com o facto de que

recolher parâmetros através de técnicas de visão computacional necessários para construir um modelo não é simples. Nos modelos esqueléticos, as articulações apresentam diferentes graus de rotação assim como diferentes ângulos que as articulações da mão podem assumir [44].

Nos modelos baseados na aparência, os gestos são modelados tendo em conta a sua semelhança com um conjunto de gestos predefinidos. Estes podem depender de um conjunto de modelos 2D de mão ou membros, onde são definidos os contornos do objeto através de um conjunto de pontos definido através de nós de interpolação. A forma é definida por uma média do conjunto de pontos dentro de um certo grupo de formas. Por outro lado, existem modelos baseados em aparência que utilizam modelos de gestos através de sequências de imagens predefinidas. Os gestos validados são modelados por uma sequência ordenada de imagens onde cada elemento corresponde a uma perspetiva da mão. O reconhecimento é efetuado, recorrendo a sistemas de treino. O processo envolve um conjunto de comparações tendo em conta dados que representam o membro [44][61].

3.4.3 – Análise gestos

A análise de gestos compreende duas tarefas sequenciais: deteção/extração de características e cálculo de parâmetros. A primeira tarefa é responsável por detetar características relevantes da imagem ou sequência de imagens para cálculo dos parâmetros do modelo gestual escolhido. A partir das características recolhidas são estabelecidos os parâmetros do modelo [44].

Para proceder à deteção/extração das características é necessário, em primeiro lugar, proceder à localização do gesto no qual se procede à extração do gesto da imagem. O processo de localização pode ser realizado através de certos padrões como cores ou sinais de movimento. As técnicas de cor baseiam-se em técnicas de segmentação através de histograma correspondente. A desvantagem do uso de cores prende-se com o facto de existirem diferentes padrões de cor de pele em diferentes ambientes luminosos [44][62].

Posteriormente procede-se ao cálculo dos parâmetros do modelo e se possível, obtém-se uma possível previsão do modelo. No cálculo dos parâmetros, estes podem ser através de dados de modelos 3D ou então dados dos modelos baseados na aparência. Para os dados de modelos 3D, o processo envolve dois passos (Figura 3.11):

- Cálculo dos parâmetros iniciais;
- Atualização dos parâmetros ao longo do tempo (envolve movimento).

No cálculo dos parâmetros, através de dados baseados na aparência, os parâmetros são seleccionados como um conjunto de características chave dos *frames* da imagem [44].

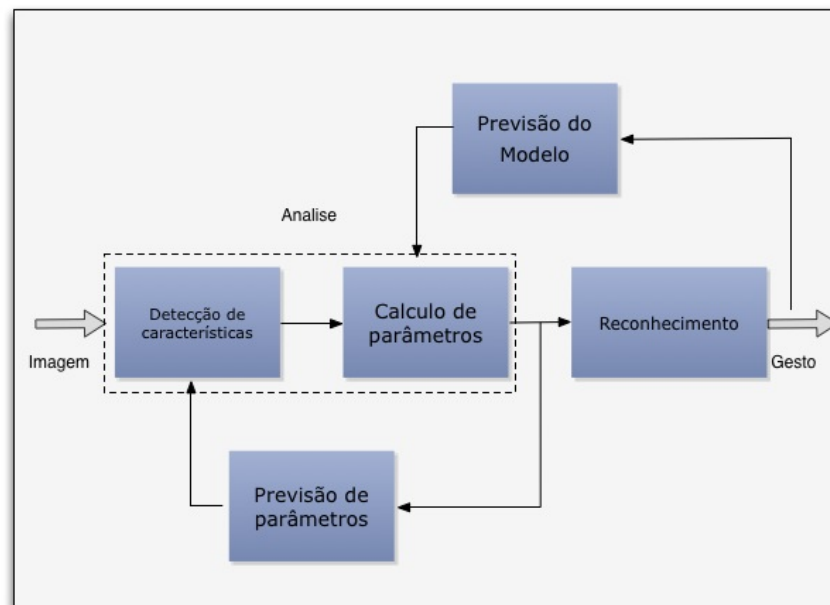


Figura 3.11 - Esquema representacional da análise e reconhecimento de gestos. Adaptado de [44].

3.4.4 – Reconhecimento de gestos

Posteriormente, na fase de reconhecimento dos gestos, são analisados os dados a partir das imagens dos gestos de modo a proceder ao reconhecimento de um gesto específico. Normalmente esta fase compreende duas tarefas: partição ótima do espaço de parâmetros e implementação do procedimento de reconhecimento. Durante a partição ótima do espaço de parâmetros é realizada a escolha do modelo de gestos a seguir e respetivos parâmetros. Usualmente a primeira tarefa faz uso de sistemas inteligentes de aprendizagem para reconhecer certos padrões dos gestos. Na tarefa de implementação do procedimento de reconhecimento é fundamental a eficiência computacional, uma vez que é um processo complexo [44][62].

3.4.5 – Técnicas de Reconhecimento Gestual

O reconhecimento de gestos é possível com recurso à componente espacial e temporal dos movimentos e gestos. É possível extrair os trajetos dos movimentos 2D a partir de sequências de imagens, sendo que em alguns casos é suficiente para o reconhecimento. Na captação da componente temporal dos gestos é necessário que a mão se encontre localizada na sequência de imagens e que se encontre segmentada em relação ao fundo da imagem [24][60].

A captação de movimento através de sequências de imagens apenas permite extrair a posição. No entanto, para extrair dados como orientação e forma da mão é necessário recorrer a captação 3D. O reconhecimento 3D faz o reconhecimento da mão no espaço, recolhendo dados de posição e orientação da mão. Este tipo de reconhecimento pode ser obtido através de câmaras de reconhecimento de profundidade. Realizar o reconhecimento da mão é muito difícil, uma vez que é muito articulada e a sua forma depende do ponto de vista, o que requer que sejam realizados mais estudos nesta matéria [24][60][60].

Modelo Oculto de Markov

O modelo oculto de Markov (HMM – *Hidden Markov Model*) lida com aspetos dinâmicos dos gestos. Tem como objetivo extrair gestos dêiticos e simbólicos. Este modelo realiza a extração dos gestos a partir de uma sequência de imagens de um vídeo. Este processo é realizado através do rastreamento da cor da pele. Durante o processo é utilizada uma tabela de indexação de cores da pele no espaço de cores YUV para proceder à filtração da imagem. Deste modo, os pixels são organizados em aglomerados de cores que definem objetos estatísticos com base na posição e na região da cor no espaço YUV. A organização em aglomerados permite determinar as áreas homogêneas da pele e fazer a extração do gesto [60][62]. Em [63], o modelo HMM é usado inicialmente como sistema de treino onde é construído um HMM para cada gesto. Posteriormente, na fase de reconhecimento, tendo em conta a base de treino adquirida pelo sistema, todas as entradas do sistema serão testadas com todos HMM, variando os parâmetros. O modelo que obtenha uma maior probabilidade de semelhança com o gesto é identificado.

Classificador *Naive Bayes*

O classificador *Naive Bayes* permite classificar os gestos tendo em conta as invariantes geométricas obtidas a partir da segmentação dos dados da imagem. A extração dos gestos é feita tendo em conta cada *frame* vídeo. Não depende da cor da pele para desempenhar a segmentação. Através da extração dos pixels de fundo, é realizada a segmentação do gesto, tendo em conta o histograma da imagem. Posteriormente procede-se à classificação dos gestos, recorrendo ao algoritmo *K-Nearest Neighbors*. Combinado com o algoritmo *Naive Bayes*, é possível obter a probabilidade de cada tipo de gesto. Este é um método eficiente e rápido no reconhecimento de gestos estáticos [60].

Métodos baseados em Técnicas de Visão Computacional

O reconhecimento de gestos através de técnicas de visão computacional recorre necessariamente à câmara, de modo a captar os movimentos dos membros. Tendo em conta estes dados, são extraídas as características necessárias dos membros para proceder ao reconhecimento dos mesmos. Considerando uma biblioteca de possíveis gestos, o reconhecimento é feito graças a uma comparação entre os gestos adquiridos e os gestos padrão. Deste modo, é possível manipular a UI a partir dos gestos reconhecidos. O reconhecimento de gestos baseados em técnicas de visão computacional podem ser o obtido através de dois métodos [60][59]:

- Métodos baseados em modelos 3D;
- Métodos baseados em aparência.

Nos métodos baseados em modelos 3D, os parâmetros do membro são calculados por comparações entre a imagem de entrada e a possível aparência 2D projetada. Este método assenta na cinemática 3D do membro. Esta abordagem é apropriada para interações em ambientes virtuais. Apesar dos modelos 3D disponibilizarem uma descrição pormenorizada do membro que permite uma maior variedade de gestos (Figura 3.12 A), necessita de uma base de dados de imagens muito extensa de modo a cobrir todas as formas [60][59].

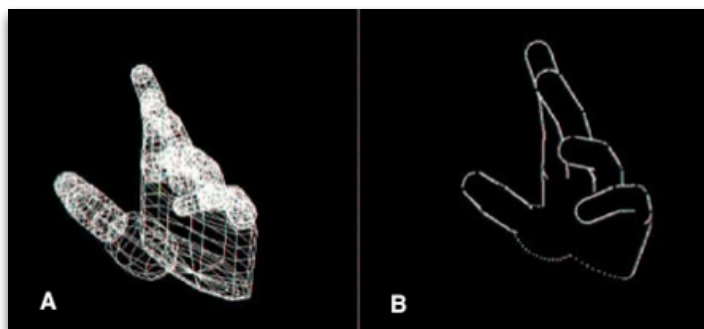


Figura 3.12 - Ilustração do modelo 3D(A) e do modelo baseado na aparência(B). Imagem retirada de [59].

Nos métodos baseados em aparência, são utilizadas as características das imagens para modelar a aparência visual do membro (Figura 3.12 B). Estes parâmetros são, então, comparados com as características da imagem extraídas do vídeo da câmara. Devido à reduzida complexidade do processo de extração de características de imagens 2D, estes métodos permitem efetuar o reconhecimento em tempo real [60][59].

3.5 – Interfaces de Utilizador

Os gestos representam uma forma fundamental e universal de comunicação não verbal. Com o desenvolvimento de novos tipos de UI, os gestos têm sido utilizados como forma de manipulação dos sistemas. O reconhecimento de gestos permite conceber interfaces que permitam utilizar gestos específicos como forma de obter informação e controlar o sistema. Para o desenvolvimento de UI gestuais é necessário ter em conta qual o tipo de gestos apropriados para a interação com o sistema. Os gestos devem estar em sintonia com o tipo de tarefa a desenvolver. A maioria das UI de gestos baseia-se na manipulação de objetos presentes num ambiente virtual, onde é possível desempenhar tarefas como rodar, arrastar, seleccionar, entre outros. Uma UI ideal tem associada gestos de simples execução a todo tipo de tarefas, inclusive as mais complexas. Deste modo, todas as suas funcionalidades encontram-se facilmente acessíveis ao utilizador [31][44][60][64].

Os gestos podem ser utilizados para desempenharem tarefas manipulativas e de comunicação. No entanto, apesar de existirem aplicações que fazem uso de gestos comunicativos, os gestos manipulativos prevalecem em sistemas HCI [44].

O uso de métodos de reconhecimento gestual em UI pode revelar-se uma mais valia na interação dos utilizadores com o computador, prestando uma experiência mais intuitiva e natural. A interpretação de gestos pode dar-se via um modelo 3D ou utilizando uma sequência de imagens [44].

Entre as vantagens do desenvolvimento de UI gestuais, destacam-se:

- Forma de interação natural com os objetos semelhante à forma como o utilizador manipula objetos reais;
- Não necessitam de dispositivos periféricos como rato e teclado para a interação com o sistema;
- Maior flexibilidade, uma vez que o ambiente gráfico da interface admite qualquer tipo de forma gráfica para a sua manipulação;
- Não necessitam de controlo físico, uma vez que os sensores conseguem detetar o tipo de sinal de entrada necessário;
- Maior interatividade.

No entanto, também apresenta algumas desvantagens como [31]:

- Para introdução extensa de caracteres, o teclado revela ser mais eficaz e fácil para a tarefa;
- Baseia-se muito no aspeto visual para confirmar determinadas ações, o que pode ser um inconveniente para utilizadores com problemas de visão.

As novas UI gestuais utilizam sensores óticos de profundidade, de modo a detetar gestos sem recorrer ao toque. Estes sensores consistem em sensores infravermelhos e LEDs infravermelhos. A captação de gestos pode ser conseguida utilizando métodos baseados na posição do objeto, o qual envolve três passos [64]:

- Converter dados recolhidos em informação de distância;
- Com essa informação, o sistema determinar a posição do objeto;
- Por último, o sistema compara a variação temporal dos dados posicionais para determinar se ocorreu algum gesto.

Interfaces Multimodais

Como se sabe a fala é a principal forma de comunicação entre os humanos. Vem daí a curiosidade da criação de sistemas de reconhecimento de voz que possibilitem uma interação idêntica. A introdução de interfaces com reconhecimento de voz e gestos (UI multimodais) tem em vista aumentar a eficácia do reconhecimento ao mesmo tempo que diminui o tempo de resposta das tarefas. A implementação de reconhecimento de voz e gestos visa aumentar o fluxo de trabalho, uma vez que o cérebro humano consegue dividir o tratamento de diferentes tipos de informação, trabalhando em paralelo. Esta capacidade deve-se ao facto de na memória, a informação espacial e verbal encontrarem-se armazenadas em sítios diferentes. Deste modo, o tratamento da informação é realizado em diferentes canais, sendo replicados em diferentes canais. Assim, é possível ao utilizador interagir com a interface através de gestos ao mesmo tempo que introduz comandos de voz sem produzir quebra de rendimento cognitivo [40].

As UI de reconhecimento de voz e gestos são bastante robustas, em parte devido ao facto dos comandos de voz e gestos se complementarem. O uso de gestos é mais apropriado para manipulação de comandos gráficos, enquanto o uso de voz é mais apropriado na utilização de tarefas de controlo e outros tipos de comandos que não gráficos. As interfaces não necessitam de reconhecer um grande número de expressões. O reconhecimento é tanto mais eficaz quanto menor for a lista de vocabulário. Combinar o uso de gestos com reconhecimento de voz pode trazer algumas vantagens às UI, dependendo da utilidade que se pretende oferecer. O reconhecimento de voz permite que o utilizador se expresse mais facilmente com o sistema, aumentando assim a eficiência. A interação gestual permite uma interação direta com os objetos, o reconhecimento de voz permite realizar tarefas mais descritivas. Visto que a implementação destes dois modos de interação em simultâneo não apresenta incompatibilidades, obtém-se um sistema mais robusto, aumentando o número de tarefas ao alcance do utilizador. Apesar de bastante intuitivo, o uso de reconhecimento de voz em UI coíbe o utilizador a um maior número de erros, devido à existência de imprecisões ou uso de termos incompletos [40].

O reconhecimento de voz, também conhecido por reconhecimento de voz automático (*Automatic Speech Recognition - ASR*) é o processo de converter o sinal de voz numa sequência de palavras, através de algoritmos específicos implementados em sistemas. Os

diferentes tipos de sistemas de reconhecimento de voz podem ser divididos em classes dependendo dos fragmentos que permitem reconhecer. Os diferentes tipos de ASR são [65]:

- Palavras isoladas – Faz o reconhecimento de palavras isoladas. Usado normalmente como comandos de controlo;
- Palavras ligadas – Idêntico ao reconhecimento de palavras isoladas;
- Diálogo contínuo – Permite ao utilizador ter um diálogo normal, enquanto o sistema determina o conteúdo;
- Diálogo espontâneo – Permite lidar com uma variedade de recursos naturais como palavras executadas em conjunto, gaguejo e diferenças de pronúncia.

As principais abordagens para realizar o reconhecimento de voz são fonética acústica, reconhecimento de padrões e inteligência artificial. Abordagem através de fonética acústica baseia-se na análise das características físicas dos sons da fala, de modo a encontrar sons específicos e atribuir-lhes etiquetas apropriadas. Fonética acústica baseia-se no facto de existirem finitas unidades fonéticas (fone) na linguagem vocal. Estas caracterizam-se por um conjunto de propriedades acústicas que se manifestam através do som vocal ao longo do tempo. A abordagem através do reconhecimento de padrões envolve o treino do sistema com os padrões válidos e posterior comparação. Pode ser aplicado para reconhecer uma palavra ou frase. É utilizada uma estrutura matemática bem formulada, de modo a estabelecer uma representação de padrões de fala consistente e obter uma comparação fiável. A comparação é feita tendo em conta um conjunto de amostras recorrendo a algoritmos de treino. A representação de padrões de fala pode ser sob a forma de modelos de fala ou então através de modelos estatísticos como HMM. Abordagem através de inteligência artificial explora os conceitos da fonética acústica e do reconhecimento de padrões [65].

3.6 – Dispositivos

Normalmente os dispositivos de reconhecimento gestual são compostos por três componentes: sensor, comparador e atuador. O sensor corresponde a um componente eletrónico que permite detetar alterações no meio. Estes sensores podem ser de pressão, luz, proximidade, acústico, movimento, entre outros. O tipo de sensor tem influência no tipo de interações gestuais possíveis. A calibração do sensor terá influência direta na velocidade de resposta do

sensor ao estímulo. O tamanho do sensor influencia o tipo de gestos possíveis para interagir com o sistema. Em sistemas mais complexos, como captação de gestos 3D, é necessário implementar múltiplos sensores de modo a permitir captar combinações de gestos mais complexos e ter uma noção de profundidade. O comando da consola Wii, por exemplo, tem implementado um acelerómetro. Este sensor sozinho não consegue detetar a posição de um utilizador. Apenas consegue detetar os movimentos efetuados pelo acelerómetro [31].

Posteriormente, o comparador recolhe a informação do sensor e confere um propósito aos dados, definindo qual o seu destino. Normalmente, o comparador é um microprocessador que executa *software* específico para a sua função. A partir da informação proveniente do comparador, o atuador vai reagir conforme o tipo de dado recebido [31].

3.6.1 – Dispositivos Móveis

Recentemente tem-se verificado que os dispositivos móveis como *smartphones*, PDA, Tablet PC, câmaras digitais, leitores de música, entre outros, têm vindo a ganhar cada vez mais destaque no dia-a-dia dos utilizadores. Uma das formas de interação com estes dispositivos é através de superfícies táteis. A partir deste tipo de tecnologia é possível interagir com o ambiente gráfico da UI de modo a completar as diversas tarefas. As suas propriedades de toque fazem com que o tipo de interação que esta tecnologia estabelece com as interfaces seja mais rápido e de fácil perceção, obtendo assim uma boa receção por parte do utilizador. Este tipo de tecnologia permite ao utilizador manipular objetos virtuais através de movimentos dos seus dedos. Deste modo, tem como intuito tornar as UI mais realistas, intuitivas e fáceis de usar para melhorar a experiência do utilizador [66].

Os dispositivos móveis têm vindo a ganhar cada vez mais importância no auxílio das funções dos profissionais de saúde. Ambientes clínicos têm vindo a beneficiar destes tipos de sistemas de informação e comunicação móvel. Estes têm possibilitado melhorar as comunicações entre serviços, facilitar o acesso à informação e melhorar o nível de qualidade do serviço prestado aos utentes [67][68].

Estes dispositivos permitem aos médicos ter acesso a informações atualizadas no devido momento, assim como aceder a funções de calendário e estabelecer um meio de comunicação com o hospital. Pretende, assim, auxiliar os médicos na prestação de cuidados mais eficientes [68].

Assistente Pessoal Digital

Para muitos profissionais de saúde, o uso de PDA é considerado uma ferramenta de trabalho indispensável, caracterizando-se por ser funcional e útil no acesso a documentação, referências médicas e dados de utentes. Permite, assim, aceder à informação clínica onde e quando esta é necessária, permitindo deste modo reduzir os erros médicos. No entanto, devido a certos fatores como segurança dos dados ou falta de apoio técnico e organizacional, há ainda uma barreira para a adoção destes dispositivos [7].

Smartphone

Nos dias de hoje, os *smartphones* apresentam características que permitem substituir o PDA, com vantagem de introduzir as funcionalidades do telemóvel. Caracterizam-se por integrar conexão sem fios, utilizando o sistema global para comunicações móveis (*Global System for Mobile Communication* - GSM). Estes dispositivos disponibilizam acesso e gestão de informação pessoal bem como a gestão de conta correio eletrónico. Apresentam alta resolução e um monitor pequeno o suficiente para ser implementado num telemóvel compacto. Existem diversos SOs disponíveis para os *smartphones*, cada um com características únicas. Destacam-se o *Windows Mobile*, iOS, Android e Blackberry [7][68].

O uso dos *smartphones* por profissionais de saúde proporciona algumas vantagens como o facto destes disponibilizarem diferentes tipos de aplicações médicas para várias necessidades. Estas aplicações podem ir desde bases de dados de medicamentos a aplicações que fornecem diagnósticos e gestão de síndromes de doenças, entre outros [68].

Tablet PC

Os Tablet PC's caracterizam-se por serem um dispositivo móvel de reduzidas dimensões com as potencialidades de um computador. O uso de Tablet PC's vem auxiliar os profissionais de saúde na medida em que oferecem acesso aos dados do utente e registos médicos em tempo real, podendo partilhar este tipo de informação com os utentes ou outro pessoal médico. Por outro lado, também garantem a privacidade dos dados dos utentes. Um dos fatores que auxiliou a implementação destes dispositivos foi o aumento do uso de redes sem fios nas instituições de saúde, vindo a comprovar o seu potencial. Deste modo, possibilita aos

profissionais de saúde o acesso a dados de utentes em qualquer parte da instituição em tempo real [69].

Estes dispositivos caracterizam-se pela sua facilidade de manuseio, disponibilizando a mais recente tecnologia de ponta. O facto destes dispositivos estarem integrados com sistemas operativos como Microsoft Windows 7 ou Mac iOS permite uma contínua atualização e evolução do sistema. Estes dispositivos disponibilizam uma superfície tátil que consegue capturar gestos dos utilizadores de modo a manipular a UI. Entre as várias vantagens do uso destes dispositivos no campo da saúde, destacam-se [69]:

- Acesso e partilha de dados em tempo real através da rede sem fios;
- Elevada portabilidade e peso reduzido;
- Privacidade dos dados dos utentes;
- Auxílio no processo de tomada de decisões;
- Disponibilização de aplicações práticas e simples.

3.6.2 – Terminais Táteis - *Kiosks*

Os *Kiosks* baseiam-se em terminais táteis, implementados com interfaces específicas, de modo a atender certas necessidades dos utilizadores. São utilizados maioritariamente para consulta de informação, proceder a pagamentos e entretenimento. Normalmente encontram-se dispostos em aeroportos, lojas ou instituições de saúde, ficando localizados em zonas de maior concentração de utilizadores. Os *Kiosks* podem ser usados como ferramentas para melhor entender o comportamento de obtenção de informação dos utilizadores [8][22][70].

Em instituições de saúde, este tipo de dispositivos tem a finalidade de realizar o processo de entrada de novos utentes, realizar a emissão e receção de informação assim como fornecer informações relevantes sobre doenças. Apesar da sua implementação nas instituições de saúde ainda ser baixa, os *Kiosks* têm demonstrado ser ferramentas eficazes para atender às cada vez mais altas expectativas dos consumidores e contribuir para melhor funcionamento da instituição [8][70].

Este tipo de interface é definida por um sistema interativo desenhado para tarefas *self-service*. O utente interage com o sistema através de um ecrã tátil, teclado ou ambos. Adicionalmente podem ser implementados acessórios opcionais como leitores de impressões digitais ou indentificadores biométricos. Os *Kiosks* são configurados para atualizarem automaticamente o

sistema principal, de modo a enviar notificações com a informação de entrada do utente diretamente à entidade responsável por gerir o processo. Os *Kiosks* devem encontrar-se instalados em áreas com uma grande densidade de utentes e em localizações óbvias [8].

Nas instituições de saúde, os *Kiosks* encontram-se implementados normalmente no serviço ambulatorio e no serviço de urgências por serem áreas de grande afluência de utentes. No serviço ambulatorio, os *Kiosks* auxiliam na realização da admissão de utentes, realização de pagamentos, atualização dos dados demográficos de utentes, realização de inquéritos, entre outros. No serviço de urgências, visam essencialmente o registo dos utentes bem como obtenção de dados para realização da triagem. As recentes melhorias destas interfaces como implementação de superfícies táteis e monitores de alta resolução têm tornado os *Kiosks* numa solução atrativa a implementar nos referidos serviços [8].

A figura 3.13 ilustra os três principais tipos de *Kiosks*: *freestanding*, *wall-mounted* e móveis. Apesar de compartilharem os mesmos componentes e características podem servir diferentes necessidades. O *freestanding* apesar de mais caro, pode ser instalado em qualquer lugar ao contrário do *Wall mounted*. Os *Kiosks* móveis destinam-se essencialmente ao registo de novos utentes com dificuldades de mobilidade no serviço ambulatorio e no serviços de urgência. Assemelha-se a um *Tablet PC* e, ao contrário dos outros tipos, este é usualmente usado pelos profissionais de saúde [8].

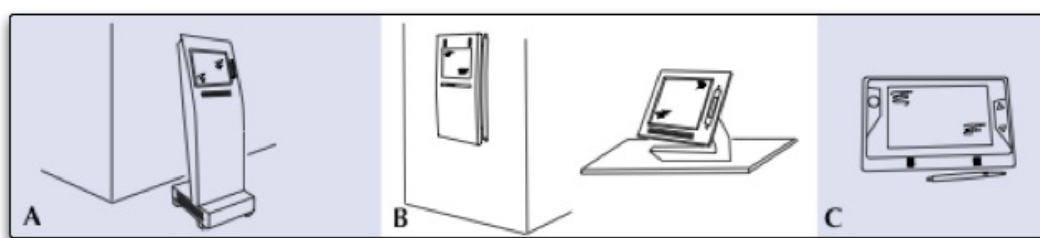


Figura 3.13 - Principais tipos de *Kiosks*. A – Dispositivo freestanding; B – Dispositivo wall-mounted; C – Dispositivo móvel; Imagem adaptada de [8].

3.6.3 – Dispositivos de reconhecimento de profundidade

Dispositivos como o *PSeye* da *Sony*, *Kinect* da *Microsoft*, ou mais recentemente o *leap*, desenvolvido pela *leap motion*, permitem captar gestos do utilizador sem recorrer ao toque com uma superfície. Por outro lado, a consola *Wii* utiliza um acelerómetro no comando, de

modo a detetar os movimentos do utilizador. Este tipo de introdução de dados é útil em circunstâncias onde não é conveniente implementar um dispositivo de reconhecimento gestual através do toque [23][31].

PSeye

O dispositivo *PSeye* (Figura 3.14) é uma câmara digital periférica desenvolvida para a consola *Playstation 3* da *Sony*. As imagens obtidas pela câmara são processadas através de técnicas de visão computacional para proceder ao reconhecimento de gestos. A conexão é feita através da porta USB. O dispositivo inclui um conjunto de 4 microfones que permite obter a localização e reconhecimento de voz. Dependendo da resolução ser de 640x480 ou 320x240, a captação de vídeo ocorre a uma taxa de 60 FPS ou 120 FPS, respetivamente [45][71].



Figura 3.14 - Dispositivo *PSeye*. Retirado de [71].

Wii

A consola *Wii*, desenvolvida pela Nintendo, utiliza uma câmara de infravermelhos de modo a captar os gestos do utilizador, executados com auxílio de um comando incorporado com acelerómetros. O comando *Wiimote* (Figura 3.15) é composto por uma câmara de infravermelhos de alta resolução, microfone e uma conexão sem fios por *Bluetooth*. A câmara de infravermelhos (Figura 3.16) é composta por um *chip* integrado com um sistema de rastreamento de múltiplos objetos. Este sistema permite obter altas resoluções de aquisição e um rastreamento mais rápido. A câmara disponibiliza uma resolução de 640x480 a uma taxa

de 30 FPS. Atualmente já existem controladores que permitem obter uma ligação com o computador pessoal [23][72][73].



Figura 3.15 - Dispositivo *Wii Motion*. Retirado de [74].

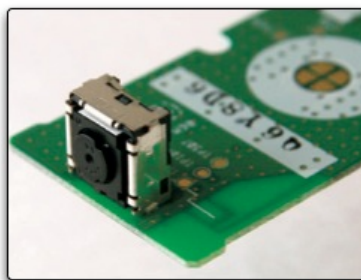


Figura 3.16 - Chip da *Wii Motion* integrado com câmara infravermelhos. Retirado de [73].

Leap

O *Leap* (Figura 3.17) diz respeito a um dispositivo de reduzidas dimensões que permite realizar o reconhecimento de gestos sem recorrer a qualquer contato direto com algum objeto. Apresentado como sendo até 200 vezes mais preciso que as soluções semelhantes existentes, a interação é feita através de movimentos naturais do utilizador. O dispositivo permite distinguir cada dedo da mão individualmente e rastrear os seus movimentos com uma precisão de 0.01 milímetros. Permite também criar gestos específicos de modo a associar a diferentes tarefas [75][76].

Apesar da sua forma de funcionamento não ter sido completamente revelada, sabe-se que se baseia no uso de câmaras e LED's infravermelhos. A sua conceção tem como foco a interação com computadores pessoais, de forma a substituir o rato e o teclado. É mais sensível que um

monitor tátil e mais preciso que o rato. Pode ser facilmente integrado num computador, recorrendo à porta USB. Permite associar vários dispositivos de modo a obter uma área de interação superior. De momento, apenas suporta *Windows*, apesar de estar em desenvolvimento um suporte para a plataforma *Linux* [75][76][77].



Figura 3.17 - Dispositivo *Leap*. Retirado de [75].

CAPÍTULO 4

Tecnologia *Kinect* da *Microsoft*

Ideias chave:

- O *Kinect* é um dispositivo periférico com capacidade de realizar reconhecimento gestual e de voz num ambiente 3D.
- O *Kinect* introduziu um novo modo de reconhecimento de gestos em tempo real através de dados de profundidade.
- O rastreamento é realizado através do modelo do esqueleto do utilizador.

Inicialmente desenvolvida como dispositivo periférico para a consola *Xbox*, o *Kinect* é composto de uma câmara RGB periférica com capacidade de realizar reconhecimento gestual e de voz num ambiente 3D. O desenvolvimento do dispositivo *Kinect* resulta da parceria entre a *Microsoft Research*¹ através de tecnologia de *software* e a *PrimeSense*², através de tecnologia de rastreamento e reconhecimento 3D. O dispositivo, com aproximadamente 23 cm de comprimento horizontal, permite detetar movimentos, presença de utilizadores através do seu arranjo esquelético, reconhecimento de faces e reconhecimento de voz com um conjunto incorporado de microfones. Atualmente já se encontra disponível o *Kinect* para plataforma *Windows* e respetivo kit de desenvolvimento de *software* (*software development kit - SDK*)³, assim como o *Windows Developer Toolkit* para desenvolvimento de *software*. O SDK é

¹ <http://research.microsoft.com/>

² <http://www.primesense.com/>

³ Disponível em <http://www.microsoft.com/en-us/kinectforwindows/develop/overview.aspx>

composto por um navegador que inclui diversas aplicações que permitem testar as funcionalidades do dispositivo. Também inclui aplicação *Kinect Studio* que permite gravar e reproduzir ensaios de profundidade produzidos através da *Kinect*, possibilitando analisar os dados de diferentes perspetivas [47][78].

A versão para a plataforma *Windows* é em tudo idêntica à versão para a *Xbox* com diferença do *firmware*. Também veio introduzir um modo novo de proximidade mais indicado para utilização em ambientes dos computadores pessoais, que apresenta limites ótimos de captação entre 0.8 a 2.5 metros. O modo padrão apresenta limites ótimos de captação entre 1.2 a 3.5 metros. Conforme o tipo de ambiente, a UI pode alterar entre o modo padrão ou o modo próximo [79].

Apesar da sua forma inovadora de interação com o meio, aponta-se como desvantagem o seu excessivo preço, tendo em conta outras soluções existentes, além disso apresenta grandes dimensões para aplicações em meios fechados [78].

O objeto de estudo desta dissertação incidirá sobre a versão para a plataforma *Windows* (Tabela 4.1).

Tabela 4.1 – Requisitos de Sistema para instalação *Kinect*.

<i>Kinect</i>
<i>Windows 7 ou 8, 32 ou 64-bit</i>
<i>Visual Studio 2010</i>
<i>Kinect SDK</i>
Processador de 1GHz ou mais rápido
1 GB (32 bit) ou 2 GB (64 bit) RAM
3.5 GB de espaço de disco duro
Placa gráfica com resolução igual ou superior a 1024 x 768

4.1 – Sensor

O sensor *Kinect* (Figura 4.1) é composto por duas câmaras, um emissor de infravermelhos e 4 microfones disposto em linha. A câmara permite armazenar 3 canais de dados. Um canal RGB com uma resolução de 1280x960 a 12 FPS, um canal RGB com resolução de 640x480 a

30 FPS e um canal YUV com resolução de 640x480 a 15 FPS . A câmara de infravermelhos fornece uma resolução de 80x60, 320x240 ou 640x480 a uma taxa de 30 FPS. Os dados provenientes destes sensores são tratados por *hardware* específico, incorporado no dispositivo responsável por realizar o processamento de sinal. Posteriormente este dados são transferidos para o computador através de uma ligação USB 2.0. No entanto, uma vez que o funcionamento do dispositivo requer uma corrente de 1.5 amperes e uma porta USB padrão, apenas consegue fornecer 0.5 amperes. A ligação tem adaptada uma fonte de alimentação extra capaz de fornecer o implemento de corrente necessário. Deste modo, a ligação ao dispositivo é estabelecida com auxílio de um adaptador que recebe corrente da rede pública e da porta USB do computador [79][80].

O conjunto de dados provenientes do sensor, permite proceder à localização e interpretação dos movimentos e sons que anteriormente não eram possíveis através de uma câmara normal. Estas apenas apresentavam dados como uma representação bidimensional de um universo tridimensional o que torna o seu processamento muito difícil [79].

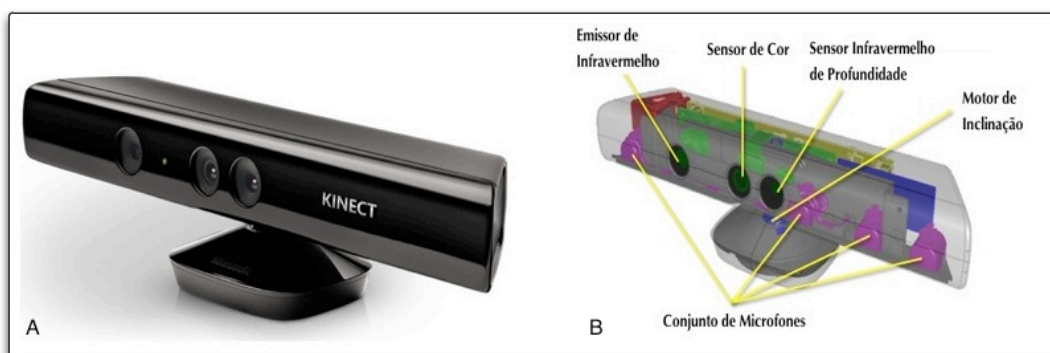


Figura 4.1 - Dispositivo *Kinect* (A) e sua constituição (B). Retirado de [80].

A área na qual os sensores de cor e infravermelho têm um campo de visão desbloqueado denomina-se de espaço de interação (Figura 4.2). No modo padrão, o sensor consegue identificar o utilizador entre uma distância de 0.8 a 4 metros. No entanto, para uma ótima interação com o utilizador (mover membros) a distância ideal situa-se entre 1.2 e 3.5 metros. No modo próximo, o sensor consegue identificar o utilizador entre uma distância de 0.4 e 3 metros e apresenta uma distância prática entre 0.8 e 2.5 metros [80].

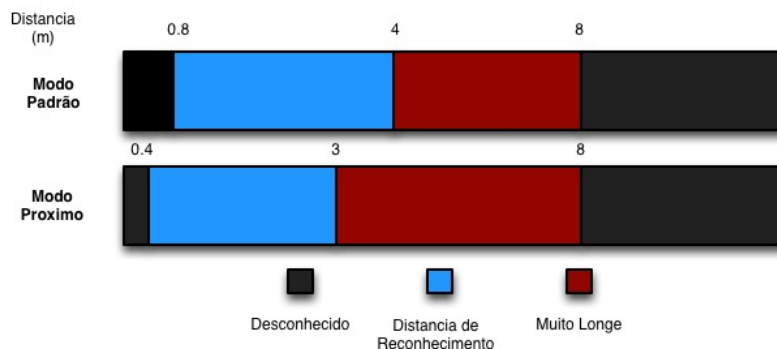


Figura 4.2 - Representação dos intervalos de distancia possíveis em metros da Kinect.

É possível ligar vários dispositivos num único computador graças às ferramentas existentes no SDK. Deste modo, também fornece funções que permitem enumerar o número de dispositivos e atribuir diferentes nomes e características aos dispositivos conforme as necessidades [80]. O *Kinect* dispõe também de um mecanismo denominado de motor de inclinação responsável por atribuir movimento à câmara. Este contém um acelerómetro que tem como função auxiliar o motor a ter mais precisão de movimento durante os mesmos. Deste modo é possível saber em qualquer instante a orientação da *Kinect*. A tabela 4.2 referencia as especificações do dispositivo *Kinect* [80].

Tabela 4.2 – Especificações do dispositivo *Kinect*.

Propriedade	Especificação
Ângulo de campo de visão	57° horizontal por 43° vertical
Ângulo de inclinação	± 27°
Taxa de <i>frames</i>	30 FPS (RGB e profundidade)
Formato de áudio	16 kHz
Características de entrada áudio	4 microfones com conversor ADC de 24 bit, supressor de ruído e cancelamento de eco

Sensor de profundidade

O dispositivo permite construir um mapeamento de profundidade da zona no qual está a ser usado. Este mapeamento é produzido no sensor e transmitido ao computador pela porta USB. Ao contrário de uma câmara normal, que transmite dados de cor para cada pixel, a câmara

Kinect transmite dados na forma de valores de distância para cada pixel. Estes valores são adquiridos através da emissão de infravermelhos do emissor. O processo de projeção contínua de infravermelhos denomina-se de *LightCoding* (Figura 4.3).



Figura 4.3 - Processo de *lightcoding* através de uma imagem obtida pelo sensor de profundidade com pontos de infravermelhos emitidos pelos emissor de infravermelhos.

Posteriormente o sensor de infravermelhos faz a captação dos pontos infravermelhos produzidos. Este consiste num sensor monocromático CMOS (semicondutor de metal-óxido complementar) que permite capturar infravermelhos sob qualquer condição de iluminação. Através do índice de reflexão de cada ponto, o sensor consegue identificar a sua distância e obter os dados de profundidade. O funcionamento do sensor de profundidade é influenciado pela presença de luminosidade no espaço uma vez que tem influência na reflexão dos pontos infravermelhos dos objetos. Deve ser evitada a sua utilização em espaços com muita iluminação ou em espaços exteriores, que sujeitem o sensor à exposição direta da luz do sol [79][80].

A Figura 4.4 representa os passos do processo utilizado para obtenção de dados de profundidade pelo dispositivo *Kinect*. O reconhecimento é realizado posteriormente por *software* especializado no computador.

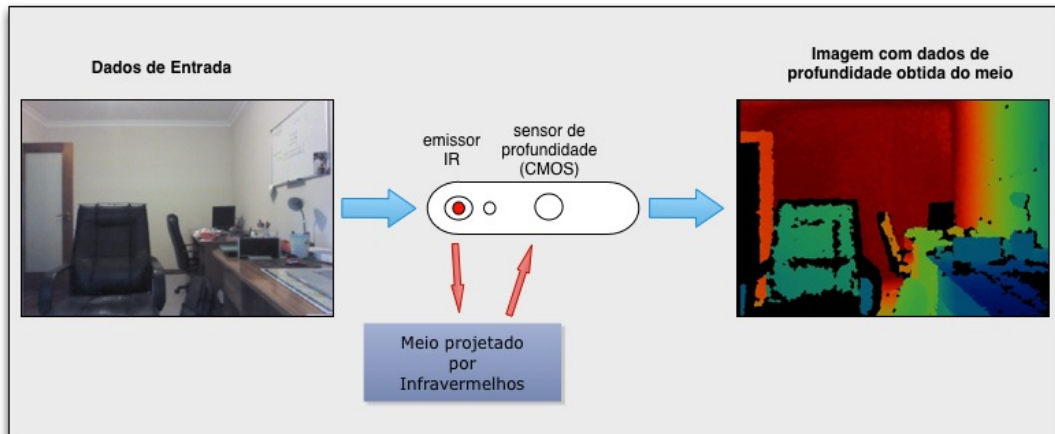


Figura 4.4 - Processo de obtenção de imagens com noção de profundidade.

A utilização de sensores de profundidade oferece certas vantagens como [81]:

- Permite trabalhar com baixos níveis de luz;
- Textura e cor dos objetos não são influenciados pela iluminação;
- Facilitar o processo de segmentação e tratamento de imagem para posteriormente treinar sistemas de reconhecimento gestual.

Microfones

Os microfones têm como função detetar a localização da fonte do som. Através da velocidade de propagação do som, os sensores conseguem localizar a fonte tendo em conta os diferentes tempos de receção dos 4 microfones. Permitem detetar a presença de vários utilizadores, calculando a direção de onde vem a voz, direcionando os microfones para esse local e removendo o ruído de fundo de forma a facilitar a sua interpretação. Os microfones permitem obter uma redução de ruído de fundo até 20dB (corresponde a um sussurro), melhorando a fiabilidade áudio. O cancelamento de eco permite aos microfones ignorarem o som proveniente do computador [79].

4.2 – Reconhecimento

O reconhecimento do utilizador, respetivos gestos e voz não é realizado pelo sensor. Este apenas é responsável pela recolha dos dados de profundidade. A câmara permite comparar as

informações retiradas das imagens com partes conhecidas do corpo humano. As partes identificadas são utilizadas para avaliar a posição do utilizador, através do modelo do esqueleto. O reconhecimento fica ao cargo de *software* específico que utiliza o alinhamento e a forma de movimento de várias partes do corpo para proceder à identificação. O sensor envia os dados para o computador responsável por processar os dados e proceder ao reconhecimento de posição, movimento e identidade do utilizador, caso seja necessário. O SDK trata-se de *software* que permite realizar o reconhecimento de gestos e voz através dos dados do sensor. Contem os controladores de dispositivo necessários para estabelecer ligação entre o dispositivo e o computador. Estes controladores suportam [81][82]:

- Acesso aos microfones através da aplicação padrão de áudio do *Windows*;
- Controladores de áudio e vídeo;
- Funções de enumeração de dispositivos, de modo a permitir utilizar vários dispositivos em simultâneo.

O SDK também disponibiliza as bibliotecas que visam facilitar o tratamento dos dados de vídeos e de profundidade do dispositivo e assim criar UI mais robustas. A forma como a interação entre o dispositivo e as bibliotecas é feita com a UI esta retratada na Figura 4.5. Estas ferramentas têm como intuito auxiliar no desenvolvimento de UI com base na forma natural de introdução de dados. Este suporta C++,C# ou *Visual Basic.Net*. As funcionalidades do SDK oferecem a possibilidade de criar aplicações para a plataforma *Windows* com capacidade de [79][80]:

- Reconhecer e rastrear movimentos do utilizador, através do rastreamento esquelético;
- Determinar distâncias entre objetos e sensor através de dados de profundidade;
- Reconhecer e localizar voz e cancelar o ruído de fundo;
- Manipular aplicações através de comandos de voz com auxílio de um motor de reconhecimento com base numa gramática previamente definida;
- Realizar o reconhecimento facial em tempo real.

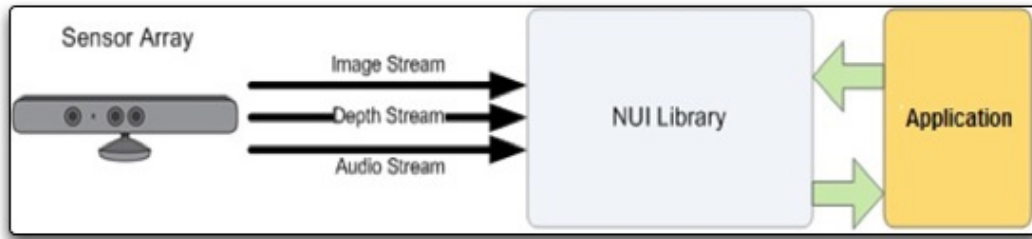


Figura 4.5 - Arquitetura da interação entre *software* e *hardware* numa UI. Retirada de [80].

4.2.1 – Detecção de Esqueleto

O dispositivo *Kinect* fornece uma nova forma de rastreamento de posição do utilizador em tempo real através da deteção do esqueleto que permite detetar utilizadores e rastrear os seus movimentos. O sensor permite identificar o utilizador através do modelo de um esqueleto. Este encontram-se representado num espaço tridimensional com eixos x,y,z (Figura 4.6 A). Os eixos x e y correspondem à largura e altura do utilizador e o eixo z à distância a que se encontra do sensor. O modelo do esqueleto (Figura 4.6 B) é composto por um conjunto de até vinte articulações, sendo que cada uma está associada a uma articulação do corpo. Para cada articulação o dispositivo atribui um estado associado, dependendo se consegue rastrear ou não. Existem três estados possíveis [74][76]:

- *Tracked* – Articulação visível para o sensor;
- *Not Tracked* – Articulação fora do alcance do sensor;
- *Inferred* – Apesar de não se encontrar visível, a articulação está inferida por meio de interpolação entre dados de articulações adjacentes.

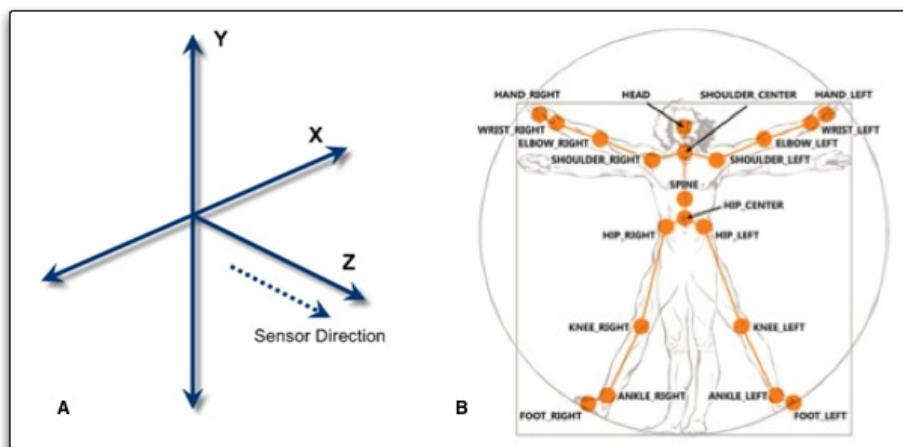


Figura 4.6 - A –Espaço tridimensional captado pelo Kinect; B - Representação das articulações identificadas pelo Kinect. Adaptado de [80]

Dentro do seu campo de visão, o sensor consegue detetar até 6 utilizadores. Desses 6 utilizadores, consegue rastrear em detalhe 2 utilizadores através da deteção do esqueleto. Normalmente o *Kinect* seleciona os utilizadores para rastreamento por ordem de identificação no campo de visão. Quer isto dizer que, dentro de um universo de 6 utilizadores, os dois primeiros a serem identificados serão os utilizadores a serem rastreados pelo dispositivo. No entanto, é possível alterar a ordem de seleção, atribuindo meios à UI para percorrer os esqueletos sugeridos e escolher o que mais se adequa às exigências. Cada esqueleto tem um identificador único. Caso o utilizador saia e volte a entrar no campo de visão do sensor, adquire um identificador novo [80].

Posteriormente o *Kinect* utiliza um método que permite localizar as articulações e rastrear os seus movimentos ao longo do tempo. Este novo método [81] utiliza imagens com dados de profundidade onde para cada parte do corpo é introduzida uma distribuição de cores por pixel. Para cada cor está associada a provável articulação correspondente àquele pixel. Este método permite calcular as posições das articulações do corpo humano de uma forma rápida e precisa.

Para o reconhecimento de gestos, é utilizada uma base de dados com diversas sequências de movimentos possíveis de serem realizados para diversas operações de controlo e manipulação de UI. Cada uma dessas sequências é composta por aproximadamente 500 mil *frames* [81].

No caso de ser necessária a utilização de mais do que um dispositivo *Kinect* é importante ter em conta que este processo pode resultar numa redução da precisão na deteção do

esqueleto, caso exista cruzamento dos campos de visão. Isto deve-se à interferência entre os infravermelhos dos diferentes sensores [80].

A deteção do esqueleto pode ser feita de duas formas: em pé ou sentado (Figura 4.7). Na posição de sentado é obtido na mesma um esqueleto do utilizador. Estes dois modos não podem ser utilizados em simultâneo. Entretanto se estiverem a ser utilizados mais do que um dispositivo, podem usar modos diferentes se estiverem a usar diferentes processos. Numa primeira análise, a grande diferença destes dois modos reside no número de articulações que são identificadas, uma vez que a parte inferior do corpo não é identificada pelo sensor. O modo sentado é mais adequado para realizar o reconhecimento do esqueleto quando o sensor de profundidade se encontra no modo próximo. Na tabela 4.3 é possível reparar nas principais diferenças entre os dois modos. Dependendo do tipo de linguagem de programação utilizado, existem linhas de código próprias para ativar os diferentes modos.

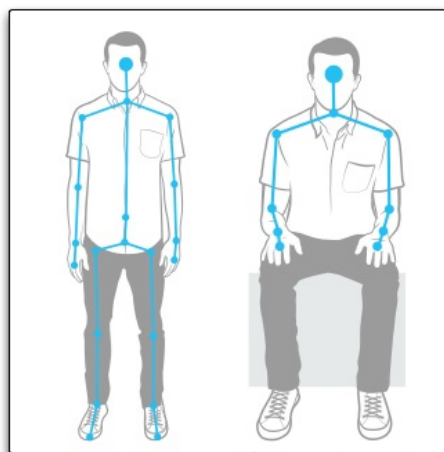


Figura 4.7 - Modos de rastreamento do esqueleto possíveis da *Kinect* (modo padrão e modo sentado).
Adaptado de [82].

Tabela 4.3 – Comparação entre modos sentado e pé.

	Modo Pé	Modo Sentado
Nº articulações	20	10 (parte cima do corpo)
Nº de Utilizadores	2 de 6	2 de 6
Deteção	Baseia-se na distância do utilizador com o fundo para iniciar a deteção.	Necessita de movimento por parte do utilizador para iniciar deteção.

Utilização	Não é aconselhável para rastrear utilizadores sentados	Permite fazer identificação de utilizadores em pé. Apenas necessita de movimento para iniciar a identificação.
-------------------	--	--

4.2.2 – Reconhecimento de Voz

O reconhecimento de voz é uma das potencialidades da *Kinect* que visa controlar aplicações através de comandos de voz de uma forma natural. Pretende, deste modo, oferecer uma alternativa à forma como o utilizador pode manipular a UI. O reconhecimento de voz permite ao utilizador usar certas palavras ou frases específicas para iniciar as funcionalidades pretendidas. A sua implementação tem a intencionalidade de aumentar a velocidade de interação com a UI. Para tal, o dispositivo dispõe de quatro microfones dispostos linearmente que permitem recolher os dados necessários para o reconhecimento de voz. Por definição, o dispositivo faz o rastreamento do som mais barulhento. Estas características visam essencialmente a obtenção de um processamento de som mais avançado permitindo adquirir amostras com maior qualidade e desempenhar funções como cancelamento do eco e remoção do ruído de fundo [79].

O reconhecimento é feito com base em ferramentas do SDK que contêm gramáticas necessárias para a comparação fonética dos dados obtidos através dos microfones. Estas ferramentas dispõem de um modelo acústico personalizado que se encontra otimizado para tal, permitindo localizar o utilizador através do som, iniciar o reconhecimento de voz e lidar com os eventos associados. O sistema de reconhecimento de voz dispõe de mecanismos de adaptação contínuo ao som que o utilizador produz. Esta opção pode ser alterada conforme o meio em que se encontre, uma vez que meios com muito ruído de fundo podem influenciar no posterior reconhecimento. Os modelos acústicos fornecidos pelo SDK permitem realizar o reconhecimento para diferentes idiomas⁴ como inglês, alemão, francês, espanhol, italiano e japonês [80].

⁴ Disponível em: <http://www.microsoft.com/en-us/download/details.aspx?id=34809>

4.2.3 – Reconhecimento Facial

Recorrendo às ferramentas disponíveis no *Windows Developer Toolkit* é possível rastrear pontos específicos que possibilitam o reconhecimento e rastreamento facial. A ferramenta permite, assim, reconhecer mais de uma centena de pontos na face do utilizador possibilitando atribuir-lhe uma identidade. Esses pontos também podem ser utilizados para criar um modelo 3D da face do utilizador. Podem ser usadas tanto imagens a cores como imagens com dados de profundidade, variando apenas a qualidade de rastreamento. O SDK utiliza o sistema de coordenadas da *Kinect* para realizar o reconhecimento facial [80][83].

O reconhecimento facial (Figura 4.8) pode ser utilizado em UI como forma de garantir acesso ao utilizador. Através dos pontos característicos da face, é possível identificar o utilizador e atribuir acesso ao sistema, substituindo assim a necessidade de introdução de senha de acesso.

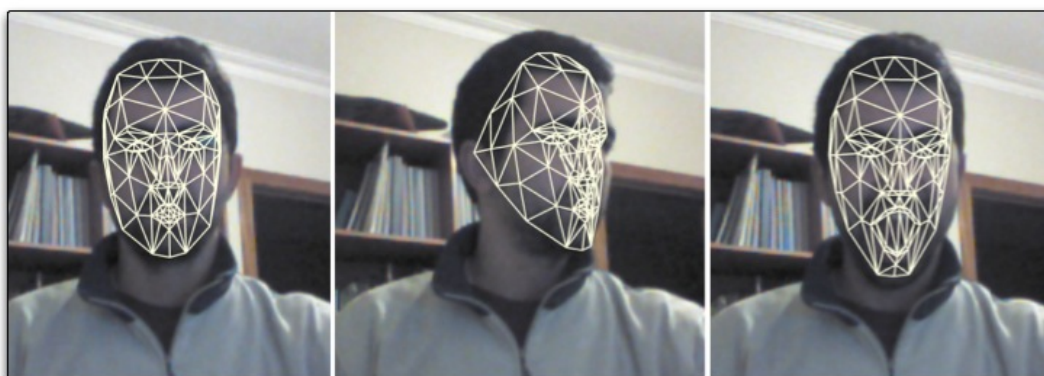


Figura 4.8 - Processo de reconhecimento facial obtido com auxílio da aplicação *Face Tracking Basics* disponível no navegador do SDK para a *Windows Kinect*.

4.3. – Testes das Funcionalidades do Dispositivo

Para a realização da presente dissertação, o Departamento de Informática da Universidade do Minho disponibilizou um dispositivo *Windows Kinect* para a sua concretização. O computador utilizado para se realizarem os testes do dispositivo foi um *MacBook Pro 13"* com processador 2.4 GHz *Intel Core 2 Duo*, 4 GB, placa gráfica *NVIDIA GeForce 320M 256 MB*. Foi utilizada a plataforma *Windows 7 32bits* instalada de raiz numa partição do disco. A versão mais recente

do SDK para a *Kinect* também permite utilizar o sensor através da plataforma *Windows* executada numa máquina virtual. No entanto, apenas foram feitos testes do seu funcionamento, uma vez que a versão do SDK com essa opção foi lançada a 8 de outubro de 2012. Apesar disso, foi possível comprovar o seu funcionamento recorrendo a uma máquina virtual. O *software* testado para o efeito foi *VMware Fusion Professional 5.0.1*.

De modo a possibilitar a interação com o dispositivo foi necessário recorrer à instalação do SDK para o *Windows Kinect* e respetivo *Windows Developer Toolkit*. Estes *kits* visam auxiliar o desenvolvimento de UI, sendo compostos por controladores necessários para o funcionamento do dispositivo, bibliotecas, ferramentas e exemplos de código fonte atualizados. O SDK inclui um navegador composto por diversas aplicações que possibilitam testar as diferentes funcionalidades da *Kinect*.

Numa fase inicial optou-se por realizar alguns testes às funcionalidades do sensor de modo a verificar quais as condições ideais para a sua empregabilidade, entre outras. Estes testes incidiram na análise das diferentes características do dispositivo em diferentes meios. Pretende-se com isso obter uma ideia das funcionalidades e potencialidades do dispositivo *Kinect* para poder aplicar a UI de reconhecimento de gestos em meios hospitalares. As funcionalidades testadas foram:

- Sensor RGB
 - Qualidade de imagem;
 - Tempo de resposta;
 - Influência de diferentes condições de luminosidade.
- Sensor de Profundidade
 - Imagem sensor infravermelho;
 - Influência de diferentes condições de luminosidade;
 - Tempo de resposta;
 - Captação a diferentes distâncias;
 - Calibração;
 - Fiabilidade de diferentes modos;
 - Número de utilizadores;
 - Influência de objetos presentes no ambiente.
- Conjunto de Microfones
 - Capacidade de deteção da fonte de som;

- Precisão de captação;
- Influência da distância;
- Adaptabilidade ao utilizador;
- Influência do ruído de fundo.

Para os testes foram utilizadas as aplicações fornecidas pelo navegador do SDK e aplicações criadas para testes específicos. Para o desenvolvimento de aplicações para *Kinect* foi utilizado o *Visual Studio 2010 Ultimate*, ambiente de desenvolvimento integrado (*Integrated Development Environment* - IDE) desenvolvido pela *Microsoft*. Para criar aplicações que façam uso das potencialidades do sensor, é necessário adicionar as bibliotecas fornecidas pelo SDK responsáveis por estabelecer a ligação com o computador através dos controladores do USB. O primeiro passo passa por criar o projecto WPF, adicionando posteriormente a referencia:

```
Using Microsoft.Kinect;
```

A utilização destas bibliotecas visa facilitar o modo como são invocadas as classes para o programa, evitando invocar as classes da *Kinect* pelo seu nome completo de todas as vezes que forem necessárias. As aplicações criadas no âmbito dos testes foram desenvolvidas na linguagem C# recorrendo às ferramentas disponibilizadas pelo SDK da *Kinect* (Anexo A).

O posicionamento do dispositivo teve em conta um local que permitisse ao sensor captar a orientação do piso de modo a obter dados enquadrados com o espaço envolvente. Não foram utilizados ângulos de inclinação extremos uma vez que tornava o rastreamento menos fiável. Para o controlo do ângulo de inclinação do dispositivo optou-se por criar uma aplicação para o efeito (Figura 4.9). A aplicação indica o ângulo de inclinação a que o dispositivo se encontra e permite definir outro ângulo. Deste modo, pretende-se controlar a inclinação do dispositivo ao longo dos testes (Anexo D).

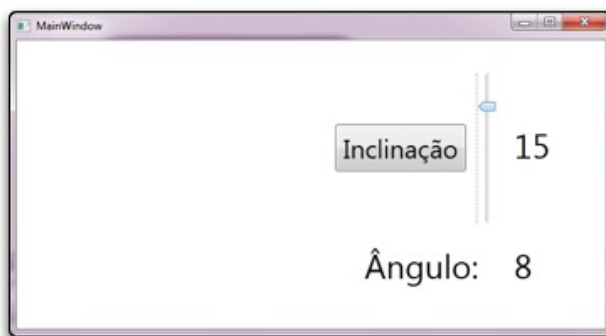


Figura 4.9 - UI de controlo de inclinação do dispositivo *Kinect*.

Achou-se importante testar o dispositivo em meios diferentes, de modo a poder observar o tipo de resposta do dispositivo quando exposto a diferentes condições. Os fatores que influenciaram a escolha do local de instalação do dispositivo foram essencialmente a iluminação, dimensões e presença de ruído de fundo. Foi possível observar que os diferentes parâmetros dos meios tiveram influência nos resultados obtidos.

4.3.1 – Sensor RGB

Os testes ao sensor RGB passaram por analisar a qualidade de imagem obtida, tempo de resposta e influência a diferentes condições de luminosidade. Numa primeira fase, o objetivo passou por simplesmente transmitir o vídeo do sensor RGB da *Kinect* da mesma forma que uma *webcam* normal. Para isso, utilizou-se uma aplicação criada para o efeito (Figura 4.10) (Anexo C). A funcionalidade principal do sensor RGB é captar, representar e exibir imagens a cores.

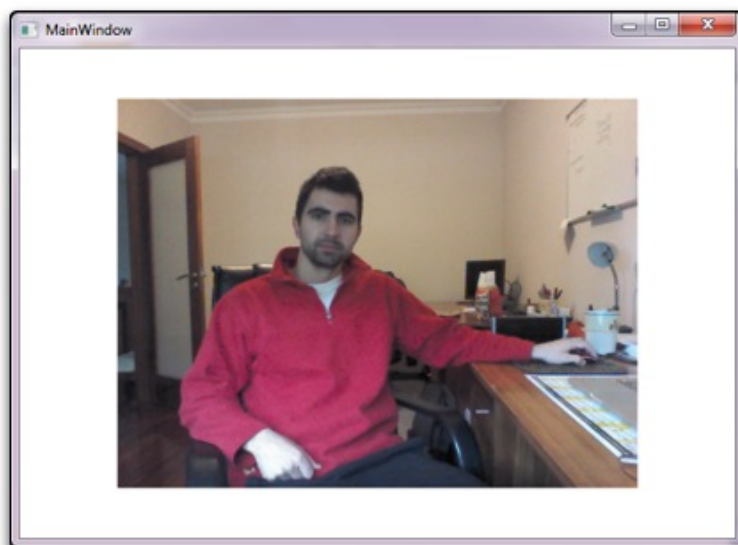


Figura 4.10 - Exibição da imagem a cor obtida a uma taxa de 30 FPS e uma resolução de 640x480 através do sensor RGB da *Kinect*.

Posteriormente realizou-se os ensaios de captura de vídeo a diferentes taxas de captura de frames. Para tal, utilizou-se a aplicação *Kinect Explorer* fornecida pelo SDK. As imagens registadas podem ser captadas através a uma taxa de 30 FPS a uma resolução de 640x480 ou então com uma resolução de 1280x960, mas a uma taxa de 12 FPS. A baixa resolução utilizada poderá ser um obstáculo para tarefas de deteção de movimentos mais subtis com mais pormenor como é o caso da deteção dos dedos. No entanto, para uma maior resolução da câmara, a taxa de recolha de *frames* por segundo desce para 12. À taxa de 30 FPS, o sensor permite rastrear movimentos suaves com baixa taxa de erro. Conforme os movimentos se tornam mais bruscos, o processo de rastrear movimentos fica mais sujeito a erros (Figura 4.11). Tal facto pode dever-se à otimização do *software* de processamento de imagem e à taxa de captação ser de apenas 30 FPS.

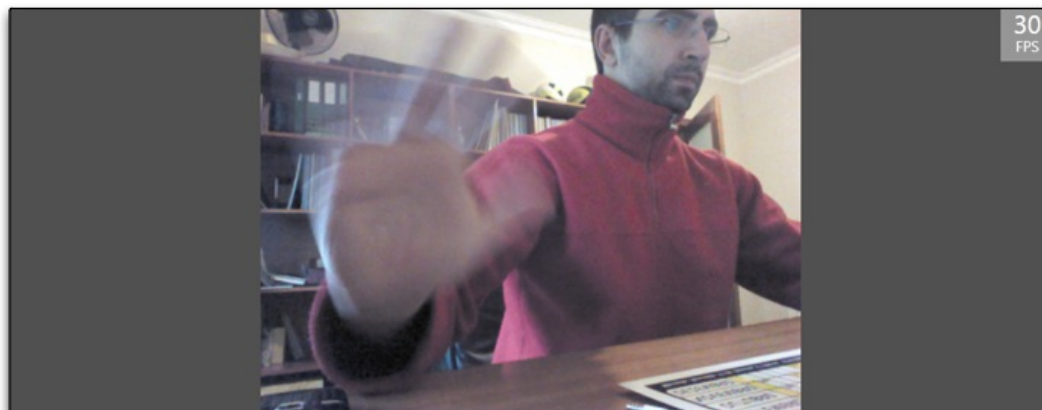


Figura 4.11 - Demonstração da capacidade da câmara na captação de movimentos bruscos.

Uma vez que o sensor RGB permite exibir vídeos através da recolha de *frames*, também permite tirar fotos a cores para onde quer que esteja apontada. Por último, foi possível verificar que tanto em ambientes com pouca iluminação como em ambientes com excessiva iluminação, como espaços exteriores, põe em a causa a qualidade da imagem RGB (Figura 4.12). Aquando da sua instalação deve ser evitado colocar o sensor à exposição da luz do sol.

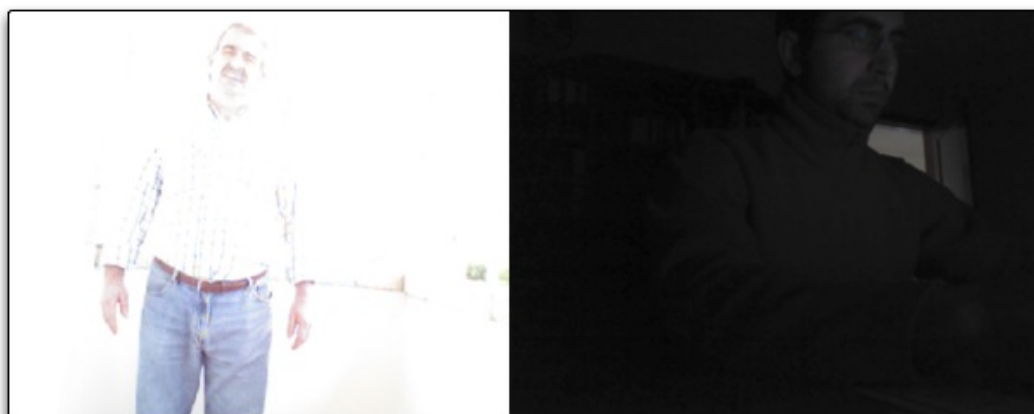


Figura 4.12 - Efeito da exposição do sensor RGB a alta ou baixa luminosidade.

4.3.2 – Sensor de Profundidade

Os testes do sensor de profundidade têm como objetivo avaliar o modo como é realizada a obtenção de dados de profundidade, analisando a influência dos fatores como dimensões do

meio, luminosidade, número de utilizadores e presença de objetos. Os testes foram realizados a uma resolução de 649x480, a uma taxa de 30 FPS.

Inicialmente, pretendeu-se observar a forma como o sensor cria o mapeamento dos dados de profundidade do meio em tempo real. Para tal, recorreu-se à aplicação *Kinect Explorer* que permite visualizar o mapa das diferentes intensidades de profundidade do meio. Como é possível observar pela figura 4.13, tendo em conta uma escala de diferentes tons de cinzento, é atribuída uma intensidade de cor a cada objeto dependendo da profundidade a que se encontra. Deste modo é possível obter uma noção de profundidade do meio.



Figura 4.13 - Imagem com dados de profundidade do meio sem utilizador.

A identificação do utilizador é feita a partir do momento em que este entra no campo de interação do sensor, atribuindo-lhe uma cor (Figura 4.14). A utilização de cores específicas para a identificação do utilizador em nada altera o processo de deteção e rastreamento do utilizador.

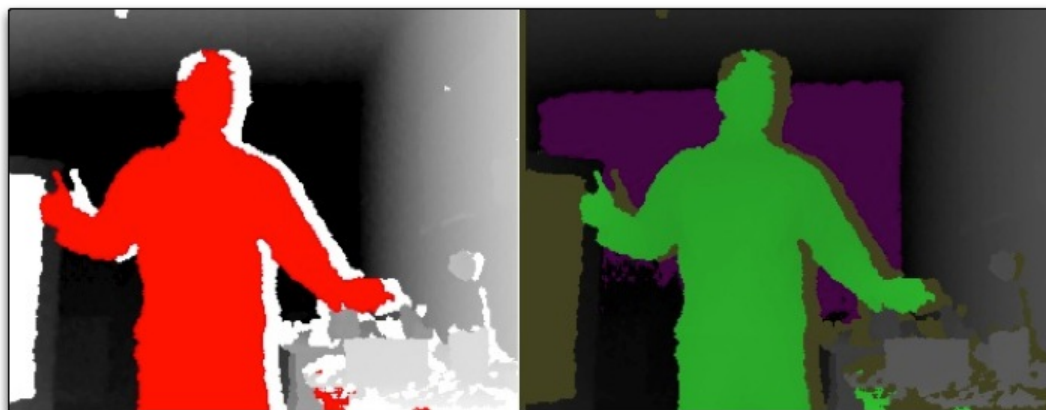


Figura 4.14 - Imagem com dados de profundidade do meio com utilizador reconhecido.

Posteriormente, pretendeu-se perceber o funcionamento do processo de obtenção de dados de profundidade. Para tal foi utilizada a aplicação *Infrared Basics WPF C#* em simultâneo com o visualizador de profundidade da aplicação *Kinect Studio*. Os valores de distância obtidos pelo dispositivo são adquiridos através da emissão de infravermelhos por parte do emissor e respetiva captação desses mesmos, por parte do sensor de infravermelhos. O sensor baseia-se no índice de reflexão de cada ponto. A utilização de infravermelhos permite ao sensor capturar dados de profundidade em ambientes de baixa luminosidade como é possível observar pela figura 4.15. A figura ilustra uma imagem com o mapeamento de infravermelhos do meio, nas condições referidas, e a imagem resultante, obtida com dados de profundidade. Como é possível verificar, a obtenção da imagem com dados de profundidade não é afetada pela baixa luminosidade. No entanto, o mesmo não se verifica nos casos em que o sensor está inserido em meios com altos níveis de luminosidade, como se irá comprovar posteriormente.

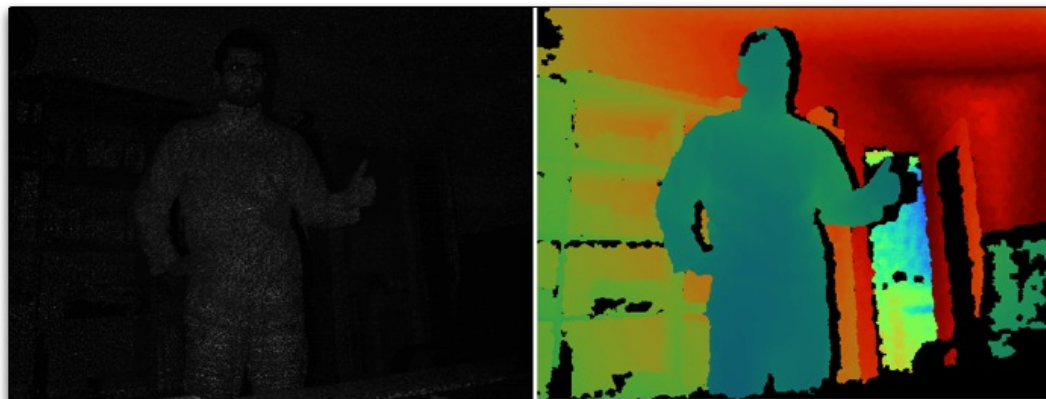


Figura 4.15 - Efeito da baixa luminosidade na obtenção dos dados de profundidade.

Através dos dados de profundidade é possível recriar um modelo 3D do ambiente em que o utilizador está inserido. Neste sentido, foi utilizado o programa *Kinect Studio* disponibilizado com o SDK, de modo a visualizar os ambientes 3D através dos dados de profundidade. A partir do visualizador 3D é possível realizar determinadas operações sobre a maquete do meio como rodar e ampliar de modo a obter uma perspetiva diferente a partir de diferentes ângulos. Através da figura 4.16 é possível notar que a *Kinect* não atribui ao utilizador e aos objetos presentes uma forma plana. Pelo contrário os dados encontram-se representados com os seus contornos característicos. Assim, não é possível obter um modelo 3D completo uma vez que não se encontram visíveis para o sensor. Deste modo como não é possível para o sensor rastrear esses dados, a sombra do utilizador visível na figura 4.16 é uma consequência disso. Só é possível representar os dados visíveis ao sensor. Para obter um modelo 3D completo do utilizador e do meio envolvente seria necessária a implementação de mais do que um dispositivo, montados em sítios diferentes para obterem diferentes perspetivas do utilizador.

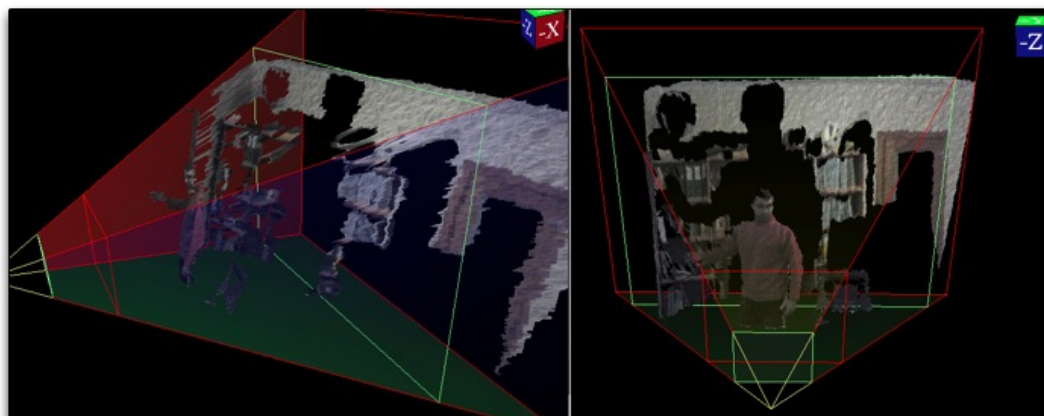


Figura 4.16 – Demonstração do funcionamento da aplicação 3D Viewer do Kinect Studio.

Um dos grandes avanços que a *Kinect* veio introduzir foi a forma de detetar a presença do utilizador em tempo real através de um novo método. Através de uma imagem de profundidade, o sensor consegue detetar a presença do utilizador, atribuindo com precisão um esqueleto à sua forma. Este esqueleto é constituído pelas diferentes articulações do corpo, variando o número, dependendo da sua visibilidade e do modo seleccionado (sentado ou em pé). Como é possível ver pela figura 4.17, o dispositivo *Kinect* reconhece o utilizador e atribui-lhe um esqueleto composto por 20 articulações.

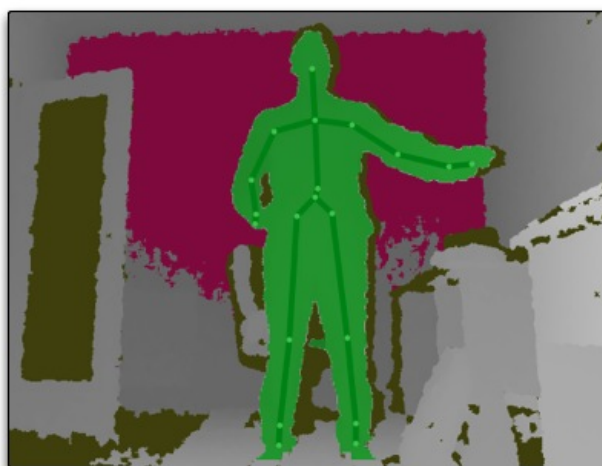


Figura 4.17 - Modelo do esqueleto do utilizador obtido através do dispositivo Kinect.

Para realizar o rastreamento, o utilizador apenas necessita de se colocar em frente ao sensor e certificar-se que o sensor consegue visualizar pelo menos a cabeça e a parte superior do

tronco. Não necessita de uma pose específica para iniciar o reconhecimento. Conforme a posição do utilizador as articulações podem ser identificadas, não identificadas ou inferidas. Para tal, foi utilizada aplicação *Skeletal Viewer* disponível no navegador do SDK. A aplicação permite identificar e rastrear em tempo real a disposição do esqueleto do utilizador. Como é possível observar na figura 4.18, no esqueleto do utilizador disposto de frente para a câmara, é possível observar todas as vinte articulações uma vez que estão todas dentro do campo de visão do sensor. No entanto, quando o utilizador se coloca na posição de lado (Figura 4.19), como o braço e a perna estão tapadas pelo tronco e pela perna direita, a articulação aparece inferida. O sensor infere a articulação por meio de interpolação entre dados das articulações adjacentes. Uma vez que certos gestos podem tapar algumas zonas do corpo, as articulações inferidas são essenciais para o reconhecimento de gestos.

Objetos que alterem por completo a forma do corpo podem alterar o modelo do esqueleto. Como é possível verificar pela figura 4.20, o esqueleto encontra-se distorcido uma vez que o sensor não consegue visualizar a região. Deste modo, o sensor tenta fazer uma previsão do modelo que pode não coincidir com a realidade. Por este motivo, aquando da utilização do sensor, é aconselhável desimpedir a área de interação de objetos volumosos de modo a não prejudicar a formação do esqueleto do utilizador. A utilização de vestuário ou objetos refletivos também podem alterar o rastreamento do esqueleto, uma vez que interfere com a reflexão dos pontos infravermelhos.

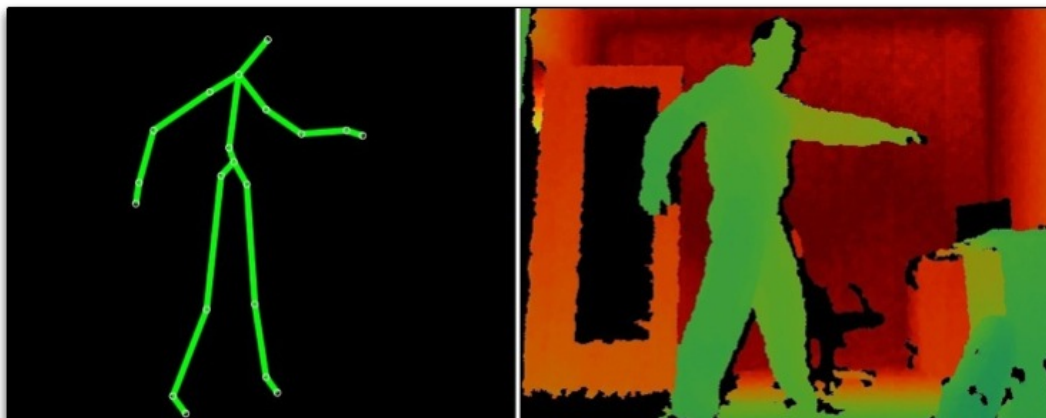


Figura 4.18 - Modelo do esqueleto do utilizador com as vinte articulações detetadas.

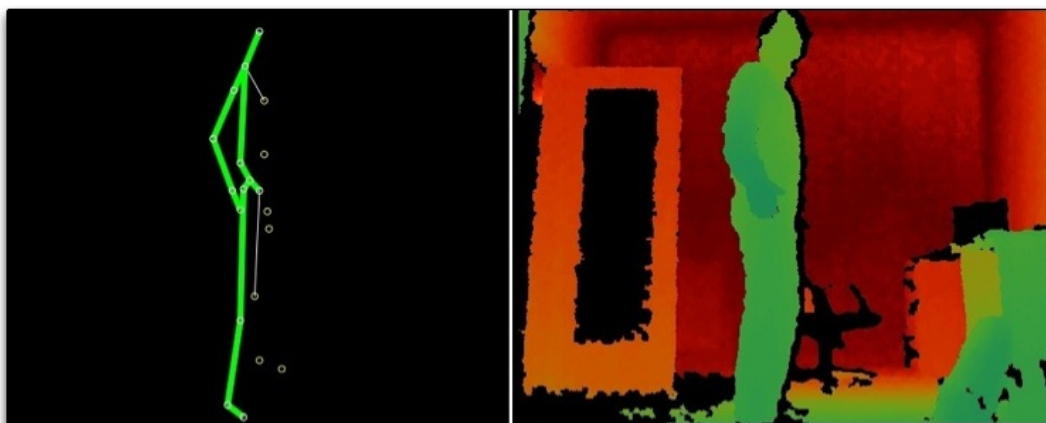


Figura 4.19 - Modelo do esqueleto do utilizador com as articulações do braço esquerdo e perna direita inferidas.

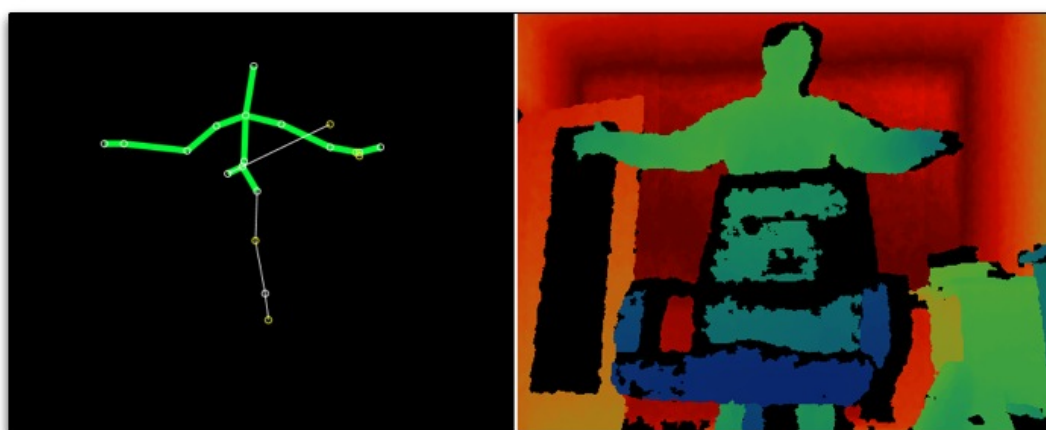


Figura 4.20 - Efeito da presença de objetos de grande volume durante o processo de rastreamento do esqueleto do utilizador.

Verificou-se também que o dispositivo consegue reconhecer até seis utilizadores dentro do seu campo de visão, permitindo fazer o rastreamento do esqueleto de apenas dois utilizadores. Para poder comprovar esse princípio recorreu-se à aplicação *Skeletal Viewer* disponível no navegador do SDK. Através da figura 4.21, é possível observar a identificação dos cinco utilizadores presentes e o rastreamento do esqueleto de apenas dois. Normalmente o *Kinect* seleciona os dois utilizadores para rastreamento por ordem de identificação.

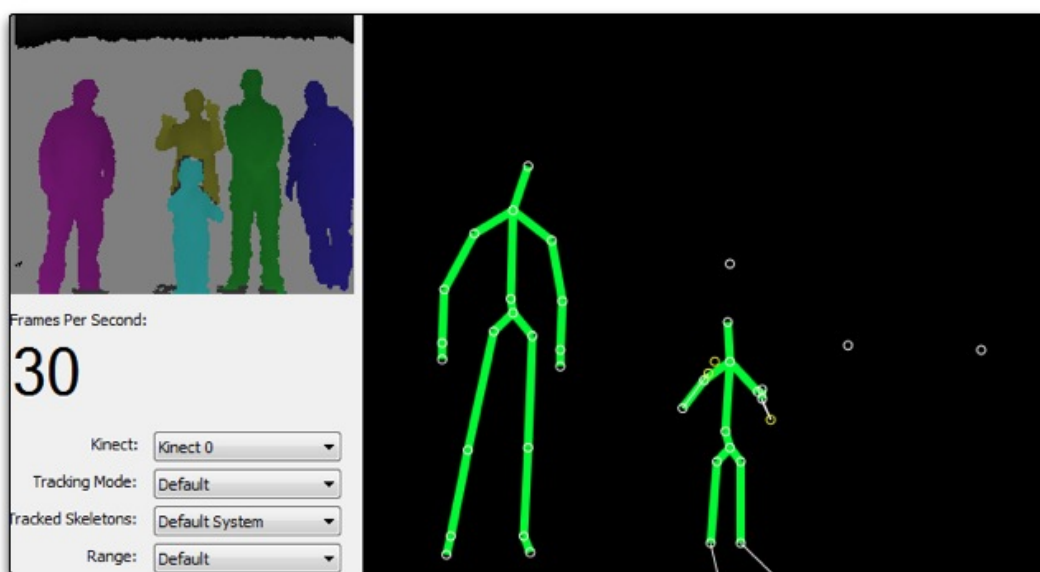


Figura 4.21 - Demonstração da capacidade do dispositivo *Kinect* em reconhecer diferentes utilizadores realizando o rastreamento do esqueleto de apenas dois.

O rastreamento do esqueleto encontra-se otimizado para reconhecer utilizadores que se encontrem em pé ou sentados. Para isso dispõe de dois modos: sentado e padrão. Através da aplicação *Skeletal Viewer*, disponível no navegador do SDK, é possível variar este parâmetro de modo a poder observar as diferenças. Na figura 4.22, é possível observar a representação do esqueleto através do modo sentado. O esqueleto é apenas constituído por dez articulações que compõem a parte superior do tronco. Apesar de ser mais indicado para usar na posição de sentado, também permite rastrear utilizadores em pé, necessitando apenas de movimento para ativar o esqueleto (Figura 4.23). No entanto, o modelo de esqueleto obtido será constituído por apenas 10 articulações, limitando o número de movimentos possíveis para o reconhecimento de gestos. O modo padrão também permite rastrear utilizadores sentados como é possível verificar pela figura 4.24. No entanto, não é aconselhável, uma vez que, como o modelo de esqueleto obtido é constituído por vinte articulações, no caso das pernas se

encontrarem fora do campo de interação do dispositivo, irá criar conflitos entre as ligações das articulações. A escolha do modo mais adequado deve ter em conta se o meio de interação é ou não muito movimentado uma vez que pode ter influência na interação com o dispositivo.

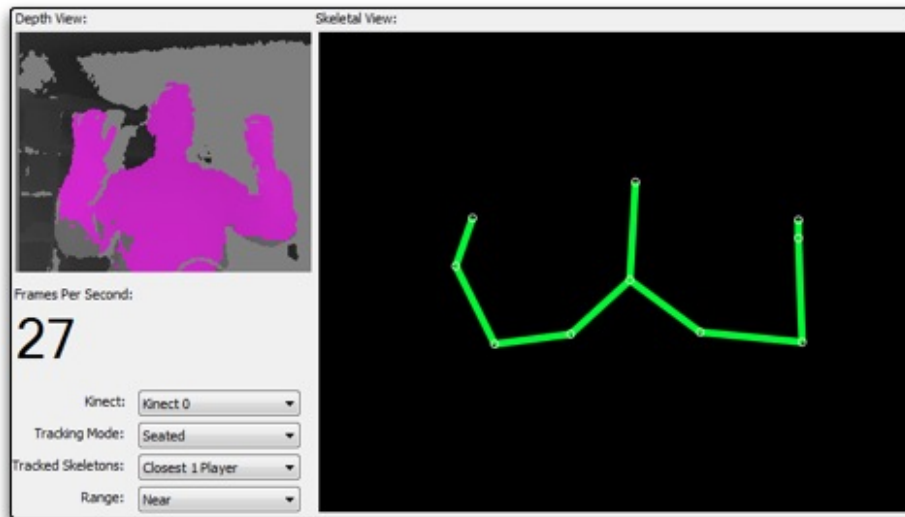


Figura 4.22 - Resultado obtido do modelo do esqueleto do utilizador a partir do modo sentado.

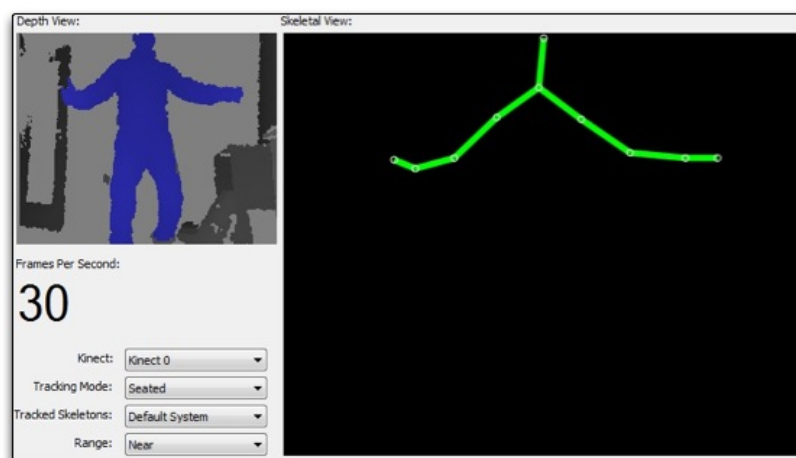


Figura 4.23 - Resultado obtido do modelo do esqueleto do utilizador a partir do modo sentado com o utilizador na posição de pé.

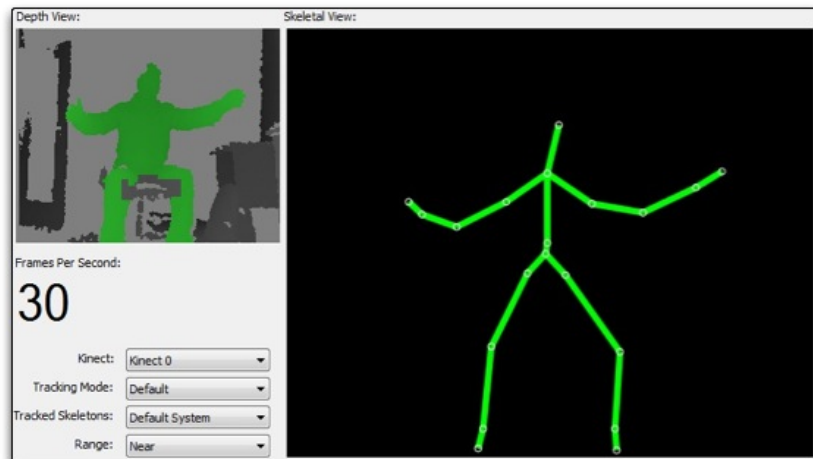


Figura 4.24 - Modelo do esqueleto do utilizador a partir do modo padrão com o utilizador na posição de sentado.

Foi possível constatar que a presença de luz natural do meio influencia o funcionamento do sensor de profundidade, afetando a captura de dados de profundidade e a formação do esqueleto do utilizador. Tendo em conta a figura 4.25, é possível reparar que a presença de grandes quantidades de luz torna o rastreamento de dados de profundidade menos confiável.

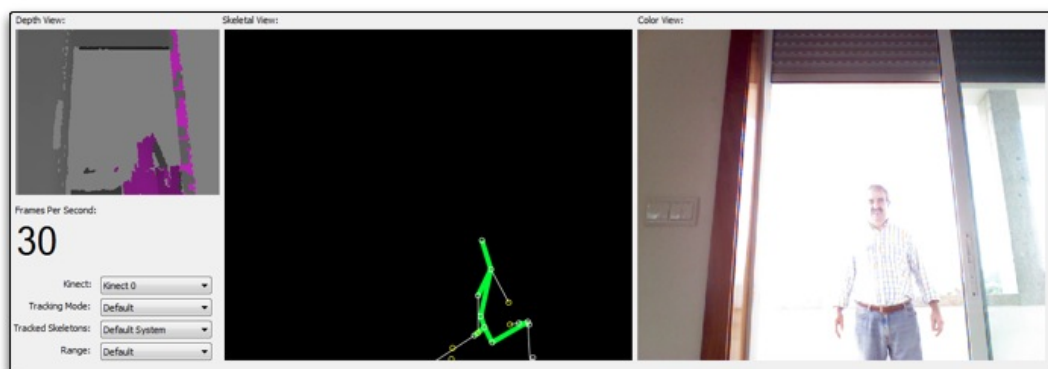


Figura 4.25 – Influência da presença de grande quantidade de iluminação no processo da obtenção de dados de profundidade.

A figura 4.26 permite concluir que em meios exteriores expostos diretamente a luz do sol, não é possível para o sensor de profundidade recolher qualquer tipo de informação de profundidade.



Figura 4.26 – Demonstração do funcionamento do sensor de profundidade quando exposto diretamente sobre a luz do sol.

4.3.3 – Conjunto de Microfones

Para realizar os testes dos microfones do *Kinect* optou-se por instalar o dispositivo numa área com reduzido ruído de fundo de modo à captação de sons não estar sujeita a influências externas numa fase inicial de testes. Numa fase posterior foi testada a resposta do dispositivo a este tipo de ruído e quais as suas consequências na captação e reconhecimento de voz.

Inicialmente optou-se por testar a capacidade em captar e detetar som e a sua fonte de emissão de som. Para tal utilizou-se uma aplicação a *Audio Explorer* e *Kinect Explorer* disponibilizada pelo SDK. Esta aplicação permite visualizar a onda de som e a direção do seu emissor no ambiente em que a *Kinect* está inserida, fornecendo o seu ângulo. O *Sound Source Angle* corresponde ao ângulo da fonte do som. O *Beam angle* está relacionado com o direcionamento dos microfones e corresponde ao ângulo usado para gravar dados de áudio. Ao mínimo som, o dispositivo consegue identificar a sua origem, atribuindo o ângulo da sua localização (Figura 4.27).

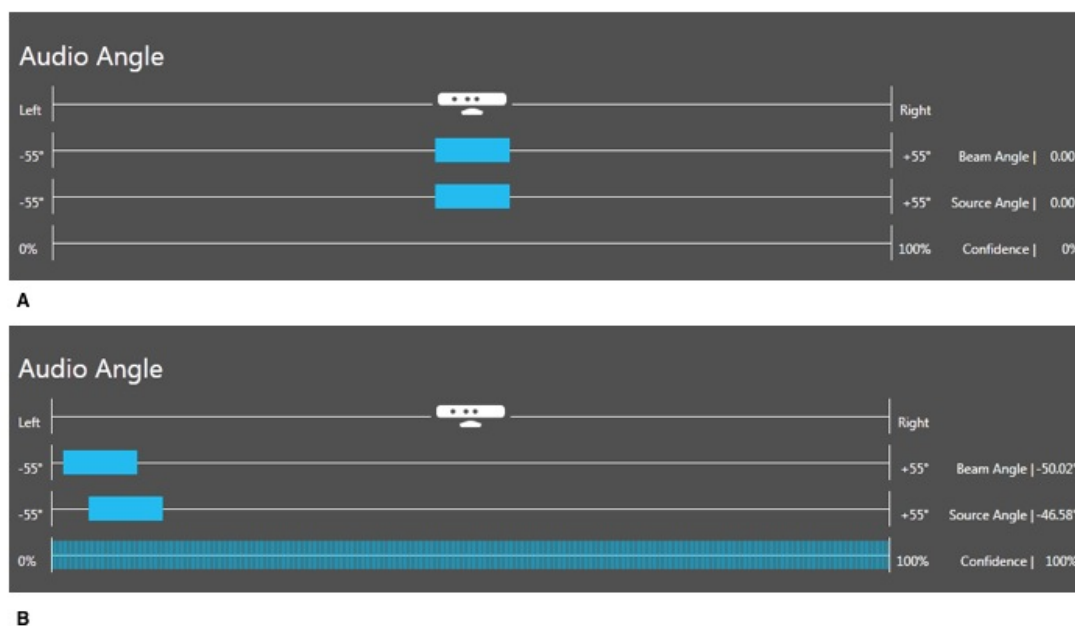


Figura 4.27 - Demonstração da propriedade de deteção da fonte de som do dispositivo *Kinect*. Esta propriedade baseia-se nos diferentes tempos de chegada do mesmo som aos microfones.

Como é possível constatar pela figura 4.28, a um ruído de fundo constante, o osciloscópio apresenta baixas variações da onda de som. Neste caso o sensor consegue identificar facilmente a localização do utilizador através da voz, visto que o ruído de fundo não tem grande influência na sua captação. À medida que se vai aumentando o ruído de fundo, a amplitude da onda de som vai aumentado gradualmente. Nestas condições, verifica-se uma maior dificuldade em interagir com o microfones, sendo necessário sobrepor o som dos comandos de voz ao ruído de fundo e aproximar o utilizador do dispositivo. Conclui-se que, para ambientes com muito ruído de fundo, a aplicabilidade do dispositivo não é a mais adequada. Deve ser evitada a colocação do dispositivo em ambientes ruidosos, direcionados para altifalantes e em superfícies que produzam ruído.

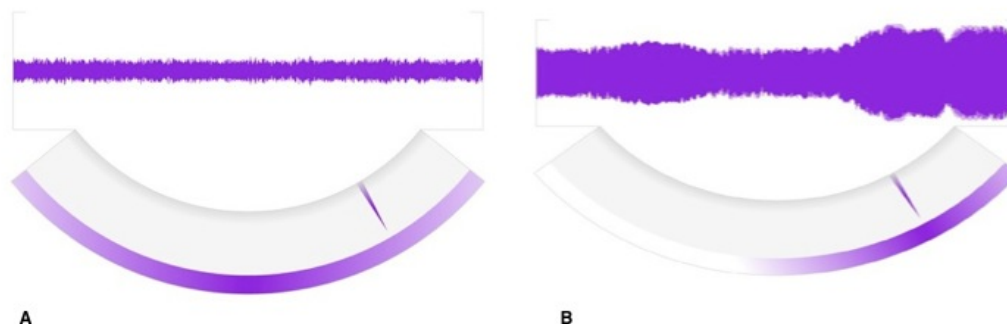


Figura 4.28 – Demonstração da captação dos microfones presentes no dispositivo (A) expostos a um reduzido ruído de fundo; (B) expostos a altos níveis de ruído de fundo onde se pode verificar a necessidade do utilizador em elevar o tom da voz para ser detetado.

4.4. – Utilização em Ambientes Hospitalares

As duas principais funcionalidades que foram introduzidas pelo dispositivo *Kinect* foi o reconhecimento de gestos e de voz. Estas funcionalidades têm características que visam revolucionar a forma de interação com as UI. O processo de desenvolvimento deste tipo de UI requer que a sua interação seja o mais natural e intuitiva possível, deixando os utilizadores numa posição confortável para as manusear.

Para o desenvolvimento de UI para implementação em ambientes hospitalares, é necessário ter em conta qual o tipo de utilizador a que se destina a UI. O método de interação destas UI deve basear-se em coisas que o utilizador tem conhecimento, de modo à interação ser o mais natural possível. Uma vez que em ambientes hospitalares, nem todos os utilizadores têm as mesmas tarefas, é necessário ter em conta qual o nicho a que a UI se destina, para atender às suas necessidades de informação. Dependendo do utilizador ser médico, enfermeiro ou utente, é necessário desenvolver UI que se adequem às tarefas que pretende ver atendidas. Por essas razões é necessário realizar uma análise prévia para estudar qual o impacto que terá no desempenho das tarefas do utilizador.

Para os médicos, a UI deve permitir realizar tarefas como por exemplo:

- Ver e analisar exames auxiliando no diagnóstico;
- Agendar tarefas e planejar eventos;
- Prescrever tratamentos ao utente;
- Preparar o doente para a alta;

Para os enfermeiros, a UI deve permitir realizar tarefas como por exemplo:

- Gerir tratamentos e medicação de doentes;
- Gerir material médico;
- Planejar, organizar, coordenar, executar e avaliar os serviços de assistência de enfermagem;

Para os utentes, a UI deve permitir realizar tarefas como por exemplo:

- Realizar processo de entrada;
- Marcar consultas;
- Proceder a pagamentos;
- Obter informações.

Através das potencialidades do dispositivo *Kinect*, pretende-se avaliar quais as melhores características a explorar em UI, de modo a serem implementadas em ambientes hospitalares.

4.4.1 – Reconhecimento Gestual

O dispositivo *Kinect* permite reconhecer vários tipos de gestos básicos, entre eles gestos estáticos, contínuos e dinâmicos. Os gestos estáticos são aqueles nos quais o utilizador necessita de se manter numa determinada posição durante algum tempo para ser reconhecido. Estes são usados normalmente em UI para seleccionar certos objetos, deixando o cursor em cima deste durante algum tempo. Os gestos contínuos dizem respeito ao rastreamento temporal de determinado movimento, como por exemplo o uso do gesto na substituição do rato. Gestos dinâmicos são gestos definidos previamente para executarem determinada tarefa aquando da sua execução. Na figura 4.29 é possível observar o efeito pretendido através de um gesto previamente definido. Foi utilizado o simulador *FAAST*⁵.

⁵ Disponível em <http://projects.ict.usc.edu/mxr/faast/>

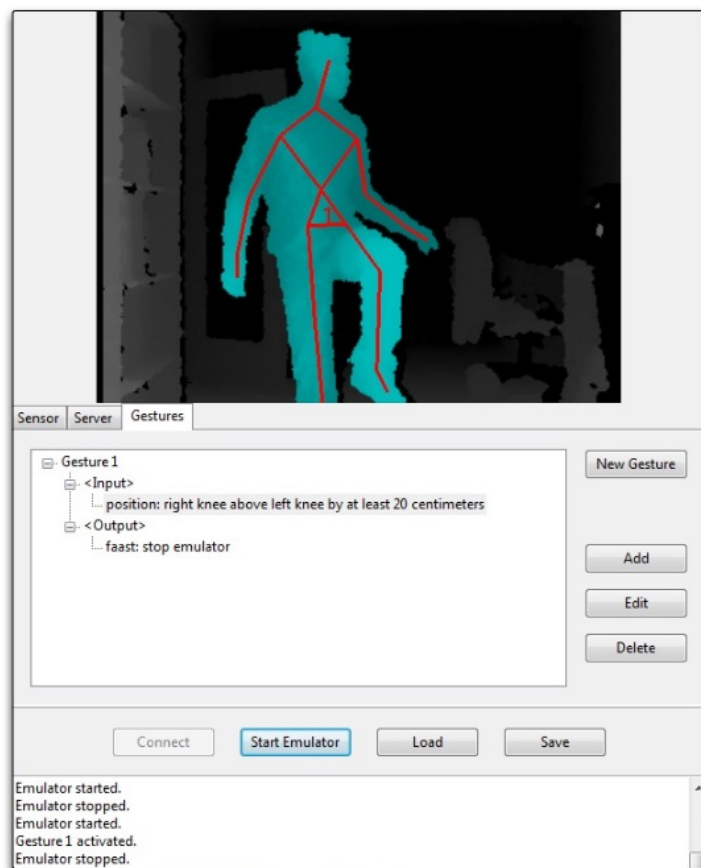


Figura 4.29 – Demonstração da simulação de gesto dinâmico reconhecido através do dispositivo *Kinect*. Definiu-se o movimento de elevar o joelho direito a cima do joelho esquerdo para parar o emulador.

Dentro dos gestos reconhecíveis pela UI, estes devem ser compostos maioritariamente por gestos que o utilizador conheça intuitivamente e façam sentido com base na sua compreensão do funcionamento do sistema. Exemplo disso é o movimento da mão como forma de substituição do uso do rato. Através das aplicações de simulação do rato, foi possível observar que o processo de controlar o cursor através de gestos é intuitivo. No entanto o mesmo já não se pode dizer para realizar a operação de clique. No caso da aplicação *Kinect Mouse Cursor*⁶ (Figura 4.30), o clique é realizado através da elevação vertical da mão assim como poderia ser de outra forma. Como não é um gesto intuitivo para o utilizador, a UI deve fornecer mecanismos de ajuda.

⁶ Disponível em <http://kinectmouse.codeplex.com/releases/view/81497>

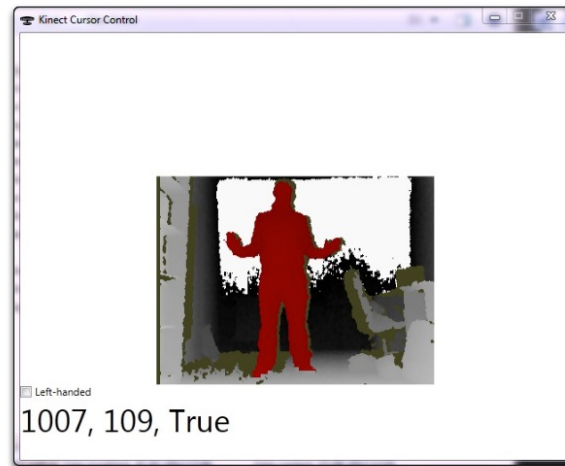


Figura 4.30 - Janela de controlo da aplicação *Kinect Mouse Cursor*. Aplicação permite substituir o uso do rato para a manipulação do ambiente do SO.

Os gestos devem ter uma razão lógica consoante a sua função, de modo à sua execução ser óbvia para o utilizador. Um exemplo disso é o gesto para ampliar. Deste modo não se deve empregar uma função diferente para um gesto que tenha um significado lógico característico. Exemplo disso seria atribuir a função de *scroll down* para um gesto horizontal. Os gestos devem também conter diferenças óbvias, de modo a diferenciá-los e evitar falsos reconhecimentos. As diferenças nos gestos podem passar pelo tempo de execução, percurso, direção e ponto de início e fim.

Os gestos complexos devem ser evitados de forma a não desmotivar o utilizador, assim como gestos estranhos e não profissionais, de modo a não colocar o utilizador numa posição constrangedora (Figura 4.31). A UI deve conter gestos de execução fácil, precisa e rápida. Gestos que requeiram muito esforço por parte do utilizador devem ser evitados. Caso contrário o utilizador pode ficar saturado, desistindo de utilizar.

A utilização de gestos naturais do corpo para o reconhecimento não são muito indicados uma vez que o utilizador pode ativar funções que não pretende ver realizadas inconscientemente. Por estes motivos não se devem implementar gestos como por exemplo levar a mão à cabeça ou à cara de modo a não se assemelharem com o gesto de tocar no cabelo ou bocejar.

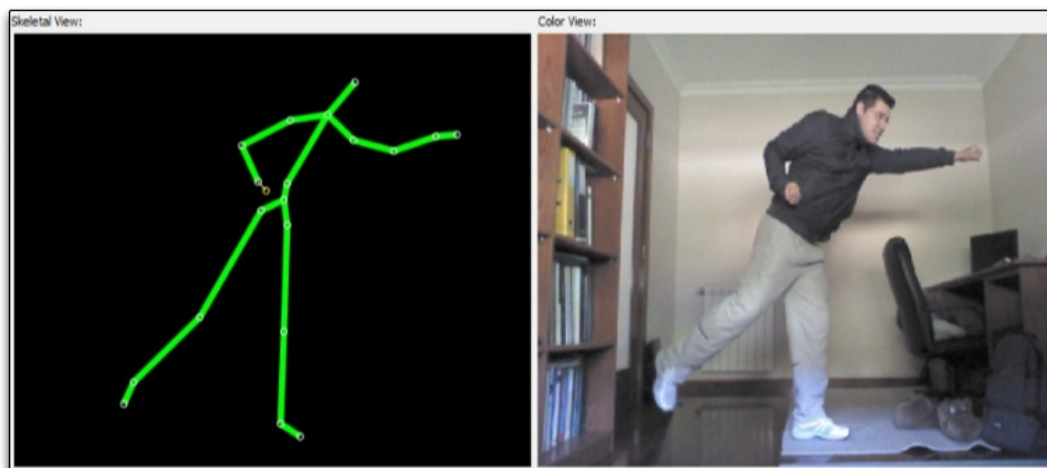


Figura 4.31 – Exemplo de gesto inapropriado de implementar numa UI de reconhecimento gestual.

Se possível, deve-se atribuir o mesmo gesto para tarefas que contenham operações idênticas. A atribuição de um gesto específico para realizar concretamente uma tarefa deve ser bem ponderada de modo a não sobrecarregar o utilizador. Maior número de gestos pode criar certas restrições à utilização da UI uma vez que o utilizador não conhece todos os movimentos possíveis. Esta medida tem em conta que as UI devem empregar um número reduzido de possíveis gestos. Maior número de gestos implica um maior esforço da parte do utilizador em aprender e lembrar da função de cada um. Os gestos devem pois ser distintos entre si de modo a evitar erros de execução e formação de falsos positivos.

A maior parte das interações possíveis na UI deve poder ser realizada com apenas uma mão. A utilização das duas mãos não é tão precisa, podendo confundir o utilizador. O uso das duas mãos deve ser reservado apenas para a execução de funções que não sejam críticas, não colocando em causa a execução das tarefas.

A utilização da UI visa diferentes utilizadores, cada um com diferentes formas de interagir. Por estes motivos, o reconhecimento de cada gesto deverá ser o mais flexível possível, de forma a facilitar o seu reconhecimento. Quanto mais rígida for a execução de determinado gesto, mais difícil será obter o seu reconhecimento. No entanto, deve-se ter em conta que se o gesto for muito flexível, a sua execução será mais fácil, o que poderá originar conflitos com outros gestos. Exemplo disso é aplicação *Slide Show*. Esta consiste numa UI na qual estão dispostas várias imagens ordenadas numericamente. Através do reconhecimento de gestos, o utilizador tem a possibilidade de mudar de imagem, passando para a seguinte ou anterior. A transição é feita através do movimento horizontal dos braços. A UI fornece *feedback* sobre o estado do

movimento através de um modelo do esqueleto disposto no canto inferior direito da janela. Como é possível verificar pela figura 4.32, aplicação permite para o mesmo gesto diferentes formas de execução.



Figura 4.32 – Método de trocar de imagem da aplicação *Slide Gesture*.

Entre as várias aplicações de reconhecimento de gestos testadas verificou-se que, desde que o utilizador se encontre dentro da zona de interação, o rastreamento de gestos está sempre ativo. Isto implica que a UI está constantemente a receber informação do utilizador, não sendo possível separar do modo de interação. A aplicação *Kinect Paint* é um exemplo. Deste modo caso não se pretenda que o reconhecimento esteja sempre ativo, deve-se implementar mecanismos que possibilitem separar o utilizador da interação com a UI para evitar a formação de falsos positivos. A solução pode passar por implementar mecanismos de ativação como por exemplo a execução de um gesto específico como acenar ou então através de comandos de voz. Da mesma forma também podem ser implementados mecanismos de desativação da interação, caso não se detete movimento durante um intervalo de tempo. As aplicações *Shape Game* e *Basic Interactions*, disponíveis no navegador do SDK, têm mecanismos de ativação e desativação do processo de rastreamento de utilizador. A aplicação *Shape Game* permite ativar e desativar a interação a partir de comandos de voz (Figura 4.33). A aplicação *Basic Interactions* desativa o processo de interação caso ao fim de algum tempo deixe de detetar utilização.

¹ Disponível em <http://paint.codeplex.com/releases/view/81498>

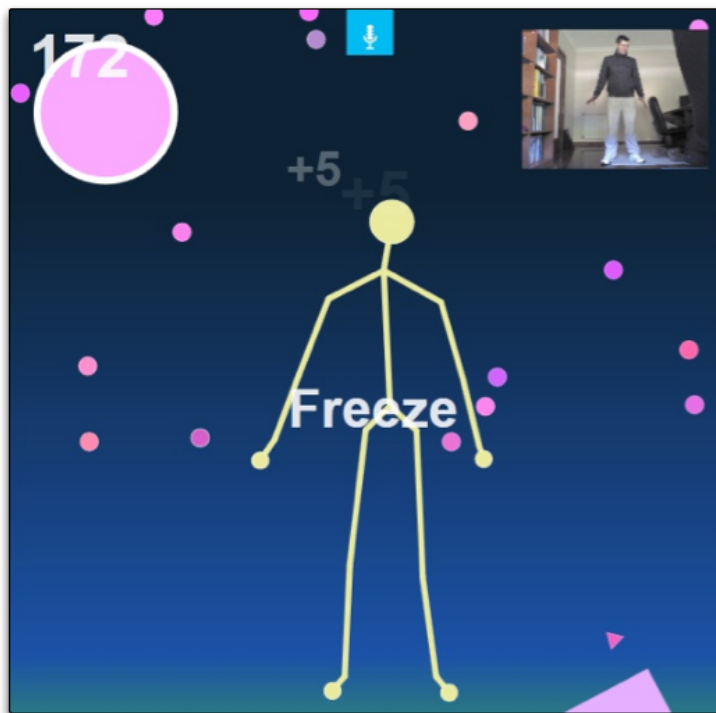


Figura 4.33 – Mecanismo de pausa da aplicação *Shape Game* através do comando de voz “Freeze”.

Para obter um rastreamento fiável através do modelo do esqueleto é necessário ter em conta certas limitações técnicas do dispositivo. O rastreamento é tanto mais estável se o utilizador estiver disposto completamente de frente para o dispositivo (Figura 4.34). Posições de lado implicam distâncias entre articulações diferentes, podendo influenciar a formação dos gestos.

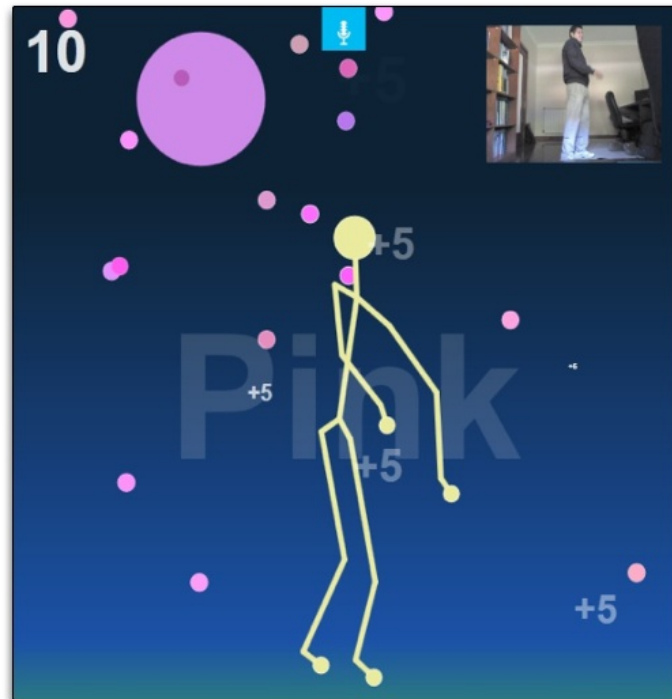


Figura 4.34 – Influencia da posição de lado no processo do rastreamento. Os gestos representados não correspondem aos do utilizador.

A posição dos braços durante a interação deve ser mantida maioritariamente de lado do tronco e não à sua frente. Caso contrário, a sobreposição das articulações podem criar conflito entre estas, pondo em causa a execução do gesto.

Devido ao sensor de profundidade ter uma taxa de 30 FPS, deve-se evitar a execução de gestos de uma forma muito rápida (Figura 4.35). No caso disto não acontecer, pode-se por em causa o reconhecimento do gesto.

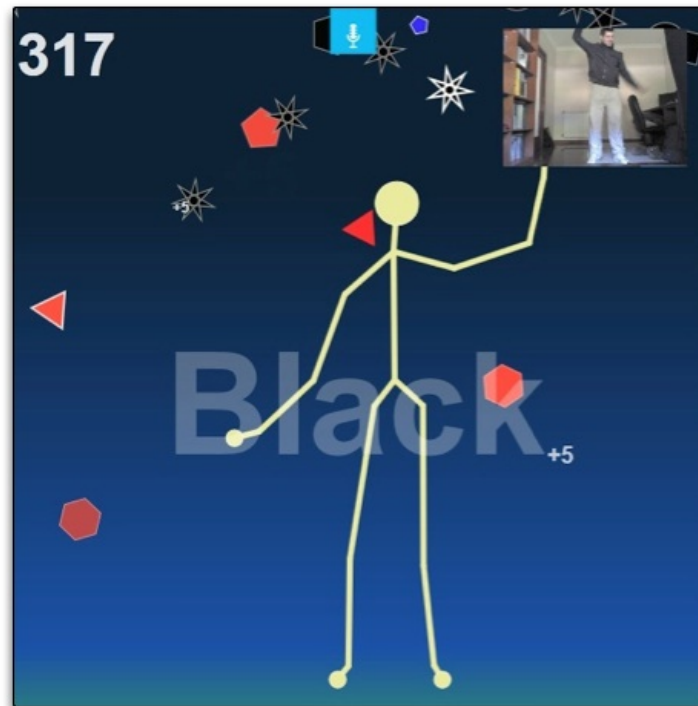


Figura 4.35 –Influencia de movimentos rápidos no processo do rastreamento.

Deve ser evitada a implementação de gestos no qual a sua execução seja realizada acima da zona da cabeça, visto estarem mais sujeitos a ultrapassar a zona de interação do dispositivo. Na figura 4.36, é possível notar que o dispositivo não consegue identificar a articulação da mão uma vez que esta se encontra fora do campo de interação.

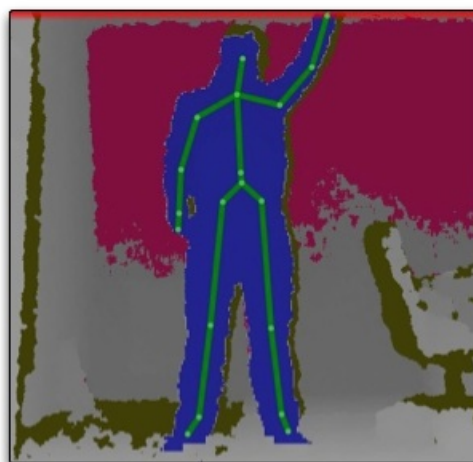


Figura 4.36 – Demonstração da incapacidade de reconhecimento fora dos limites do dispositivo.

A instalação do dispositivo deve ter em conta uma distância que permita que o campo de interação cubra o intervalo de alturas da média dos utilizadores. Caso a UI seja direcionada para os utentes acederem às informações da instituição de saúde, esta deve permitir que tanto uma criança como um adulto possam interagir.

Dependendo da distância a que se encontra o dispositivo, deve-se ter em conta o tamanho do monitor, assim como o tamanho dos textos e objetos presentes na UI. Em todas as aplicações testadas foi possível verificar que a utilização de um monitor de 13" condicionou a perceção dos objetos presentes no ambiente. Este fator é relevante para a UI obter aceitação do utilizador. O tamanho do monitor da UI deve estar relacionado com a distância a que o utilizador se encontra, para o dispositivo poder realizar um rastreamento ótimo. Quanto maior for a distância do utilizador do dispositivo, maiores terão que ser os objetos presentes na UI de modo a não colocar dificuldades na visualização.

A substituição do rato através de gestos é uma das possíveis aplicações da *Kinect* em UI. No entanto, como este tipo de interação exige movimentos contínuos dos membros, é necessário ter alguns fatores em conta de modo a não desgastar e saturar o utilizador. Com base na experiência obtida através das aplicações *Kinect Mouse Cursor* e *Kinect Magic Cursor*⁸, foi possível notar que apesar de a sua interação ser agradável, o processo sofre de algumas imprecisões e com o passar do tempo torna-se num processo desgastante. Assim a sua implementação em UI deve ter em conta alguns fatores. A sua manipulação não deve exigir movimentos perlongados de modo à utilização ser o quanto tanto confortável. Se possível intercalar a sua utilização com outras tarefas que não exijam movimentos. Os movimentos devem também representar o melhor possível a intenção do utilizador. O clique pode ser feito ou com auxílio da mão esquerda ou então deixando o cursor em cima do objeto durante algum tempo. Para este último, a UI deve fornecer uma forma de *feedback* para que o utilizador tenha perceção do progresso (Figura 4.37). O tempo não deve ser muito curto de forma a não originar falsos positivos, mas também não deve ser longo de modo a não exaustar o utilizador.

⁸ Disponível em <http://drenton72.wordpress.com/2012/09/06/kinect-magic-cursor-version-1-2-with-source-code/>

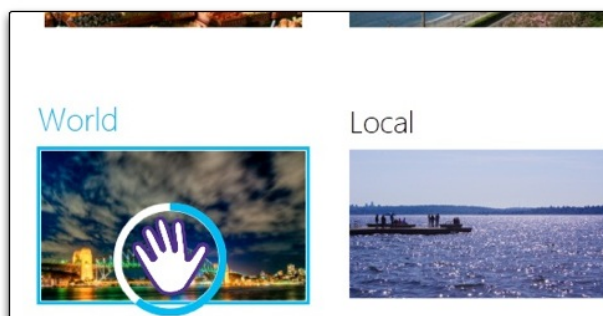


Figura 4.37 – Método de seleção presente na aplicação *Basic Interactions*.

Um dos pontos fracos da interação de UI através de gestos é a introdução de caracteres de texto e a *Kinect* não é exceção disso. Recorrendo ao teclado virtual e à aplicação *Kinect Mouse Cursor*, foi possível verificar que a introdução de caracteres de texto através de gesto é um processo moroso e cansativo. Deste modo, a necessidade de introdução de caracteres só deve ser aplicada em casos específicos e em última instância. O processo deve também disponibilizar mecanismos de ajuda como por exemplo introdução de caracteres através do reconhecimento de voz.

Por último, pode-se optar por dar mais controlo à UI através de rastreamento dos dedos (Figura 4.38). Estes métodos justificam-se em UI que exijam um controlo mais preciso sobre os objetos. Visa essencialmente desempenhar tarefas mais sensíveis como por exemplo agarrar e arrastar objetos ou então controlar dispositivos com as mãos.

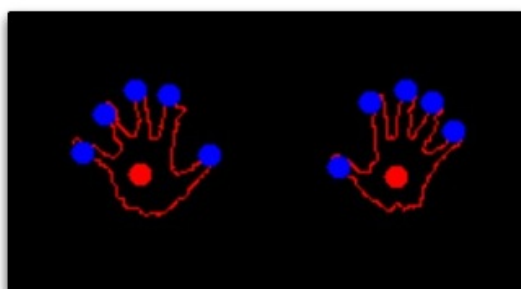


Figura 4.38 – Rastreamento das mãos obtido recorrendo aplicação desenvolvida por Cerezo *et al.* [84] no âmbito do estudo sobre o reconhecimento de dedos utilizando o dispositivo *Kinect*.

4.4.2 – Reconhecimento de Voz

O reconhecimento de voz é outra forma possível de manipular UI através do dispositivo *Kinect*. O reconhecimento de voz permite atribuir comandos a palavras ou frases que, uma vez proferidas, executam determinada função. O dispositivo permite usar dois modelos para o reconhecimento de voz:

- Escuta ativa – Dispositivo está constantemente a captar dados áudio, analisando todas as palavras e frases.
- Palavra-chave – Disposição para captar apenas determinada palavra ou frase, disparando o mecanismo específico associado a partir do momento em que a captar.

A escuta ativa é tanto mais eficaz quanto menor for o número de palavras ou frases reconhecíveis. A utilização de um extenso vocabulário forma mais facilmente falsos positivos devido à maior probabilidade de existência de palavras semelhantes.

A escolha das palavras deve ter em conta que sejam simples e comuns de modo a serem de fácil memorização, enquanto para as frases deve-se evitar que sejam muito extensas. Para evitar a formação de falsos positivos devem ser descartadas palavras que rimem e que tenham sons vogais idênticos. As frases não devem conter palavras iguais a outras frases. Caso seja necessário introduzir uma palavra ou frase extensa ou desconhecida, a UI deve disponibilizar a palavra de modo a que esteja implícito ao utilizador. A UI deve transmitir uma forma de *feedback* quando estiver a proceder ao reconhecimento de voz como forma de transmitir a sensação de controlo ao utilizador. A figura 4.39 ilustra um exemplo de uma aplicação, que utiliza o modelo de palavra-chave para o reconhecimento de voz . A aplicação permite reconhecer certos comandos de voz como “*red*”, “*green*” ou “*blue*”, retribuindo no monitor a respetiva cor, reconhecendo também a direção da localização do utilizador.

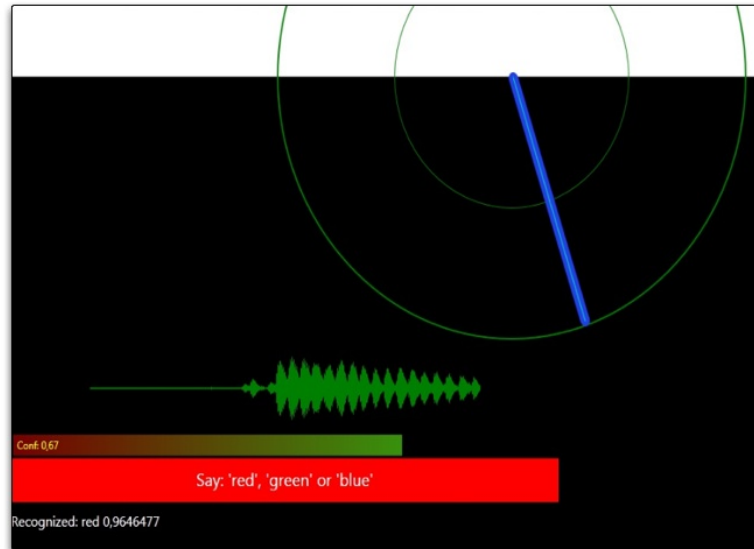


Figura 4.39 - Capacidade de reconhecimento do dispositivo. A partir do comando de voz “Red”, o dispositivo representa a cor através da barra vertical.

A aplicação *Speech Basics* disponibilizada pelo navegador do SDK utiliza o modelo de palavras chave. Esta aplicação permite dar direções a um avatar através de comandos de voz simples (*forward, back, turn left, turn right*) (Figura 4.40). A UI disponibiliza os comandos de voz ao utilizador como forma de auxiliar na sua interação. No entanto, a aplicação peca por alguma falta de precisão do reconhecimento que pode ser ainda mais agravada com a existência de muito ruído de fundo. Nem sempre reconhece o comando de voz à primeira tentativa, o que se pode tornar maçador ao fim de longos períodos de interação. Por esta razão, é importante proceder a testes ao local antes da implementação da UI em qualquer que seja o serviço, de modo a verificar qual a influência do ruído de fundo na interação.

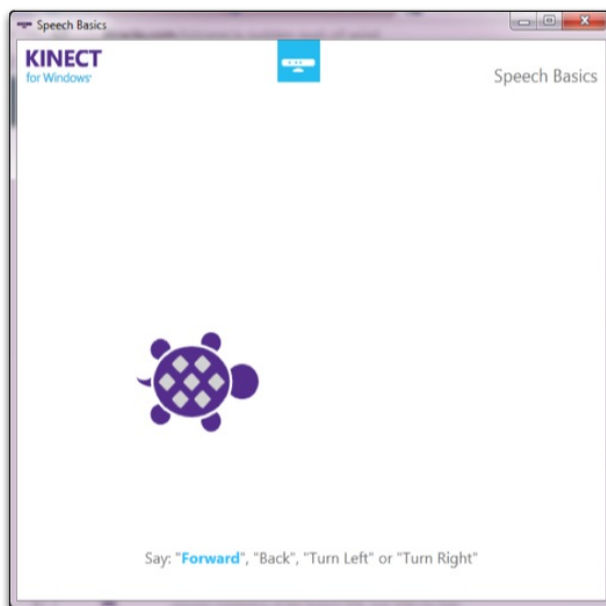


Figura 4.40 - Capacidade de reconhecimento de voz do dispositivo *Kinect* através da aplicação *Speech Basics* disponível no navegador do SDK.

4.4.2 – Interfaces de Utilizador Multimodais

As UI multimodais consistem em UI nas quais a sua interação é realizada através da combinação do reconhecimento de gestos com reconhecimento de voz. A adição do modo de reconhecimento de voz numa UI de reconhecimento de gestos visa essencialmente reduzir o número de gestos necessários, tornando as tarefas menos fatigantes. O facto de fornecer outra forma de apoio, ajuda a aumentar a segurança do utilizador na utilização da UI. Um exemplo de UI multimodal é a aplicação *Basic Interactions* disponível no navegador do SDK. Esta aplicação fornece uma plataforma composta por várias categorias de informação, estando ao alcance do utilizador através de gestos e comandos de voz. Como é possível observar pela Figura 4.41, cada objeto presente na UI é possível de ser selecionado através de um dos modos. Os objetos podem ser selecionados pelo comando de voz correspondente ao seu número ou então através de gestos. Esta opção permite repartir o esforço entre os dois modos.

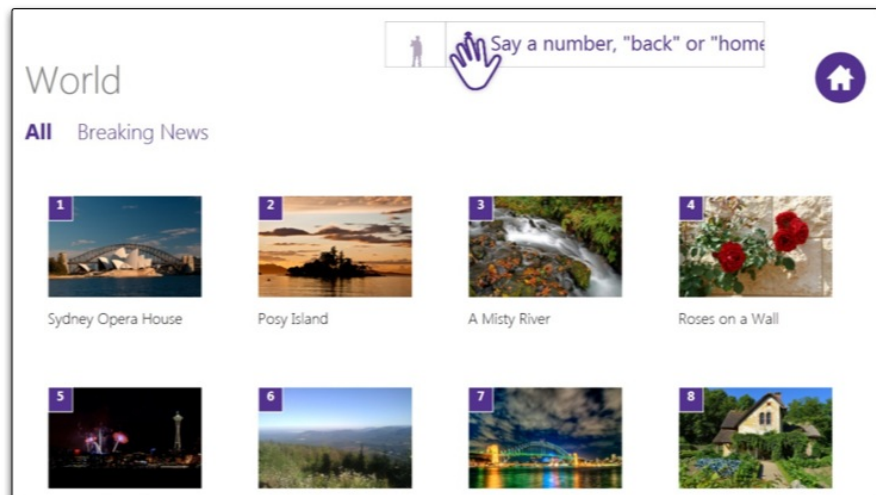


Figura 4.41 – Menu de seleção da aplicação *Basic Interactions*.

Um outro exemplo de UI multimodal é a aplicação *Shape Game* disponível no navegador do SDK (Figura 4.42). Esta consiste num jogo no qual o utilizador tem que derrubar figuras geométricas que vão caindo através do movimento dos seus membros. O rastreamento do utilizador é feito através do modelo do esqueleto. A aplicação reconhece determinados comandos de voz que permitem ativar certas opções como alterar a o tamanho das figuras, alterar a velocidade, fazer pausa, entre outros. A aplicação fornece *feedback* por cada comando de voz reconhecido e sugere palavras como forma de auxiliar o utilizador. Verifica-se no entanto a formação de falsos positivos com alguma facilidade. Isto deve-se ao facto de a aplicação aceitar um grande numero de comandos de voz, o que leva à existência de erros de reconhecimento para palavras idênticas.

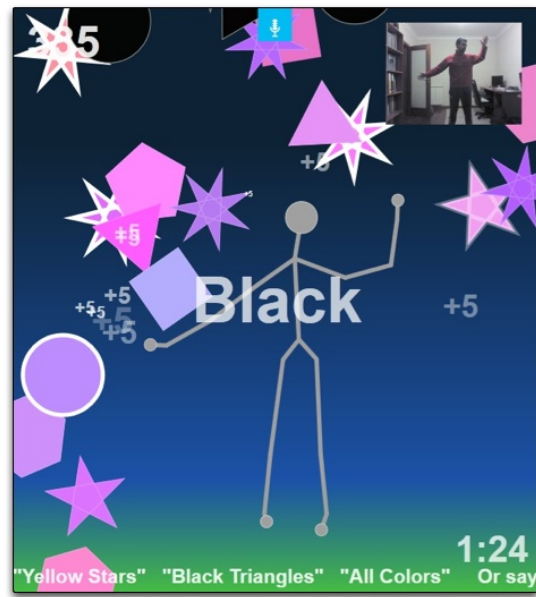


Figura 4.42 – Ambiente de interação da aplicação *Shape Game*.

CAPÍTULO 5

Interface *HosNui*

Ideias chave:

- O *HosNui* é um protótipo de UI multimodal que utiliza as potencialidades do *Kinect*.
- O protótipo pretende avaliar se a introdução de gestos como forma de interação com as interfaces são uma mais-valia para os utentes.

Na sequência do estudo dos novos paradigmas de UI para aplicações na área da saúde, foi desenvolvida uma UI multimodal denominada de *HosNui*, para implementação em ambientes hospitalares, mais propriamente no serviço de urgência. A interface foi desenvolvida tendo em conta o uso das potencialidades do dispositivo *Kinect*, de modo a garantir ao utente uma interação natural e intuitiva característica das NUI. A interface destina-se aos utentes e visa ser um ponto informativo onde podem ser consultados diferentes tipos de informações relativamente à instituição de saúde, problemas de saúde, entre outros. Pretende-se com isso verificar se a aplicação desenvolvida é uma mais-valia para os utentes.

O seu desenvolvimento visa observar qual o efeito da introdução deste tipo de interfaces nos utentes em ambientes hospitalares. Pretende-se verificar se a introdução de gestos como forma de interação com as interfaces são uma mais-valia para os utentes, auxiliando nas suas tarefas.

O desenvolvimento da UI teve como base a aplicação no serviço de um hospital fictício, desenvolvido para o caso em estudo. As informações abordadas nos diferentes temas presentes na interface referentes ao hospital são fictícias.

A interface foi desenvolvida na linguagem C#, recorrendo ao *Visual Studio 2010 Ultimate*. O seu desenvolvimento teve como base a aplicação *Basic Interactions* disponível no navegador do SDK, de modo a obter e simular os controladores de gestos e voz.

Previamente foi feita uma análise ao utilizador que a interface visava, assim como às características do tipo de ambiente que iria ser inserido. Escolheu-se o serviço de urgência de uma instituição de saúde. Este serviço recebe diariamente todo o tipo de utilizadores com diferentes faixas etárias, características físicas, níveis de conhecimento, entre outros. Assim, era importante que a interface fosse o mais simples e intuitiva possível, de modo a não colocar entraves a qualquer que fosse o utilizador. Da mesma forma, deveria conter toda a informação necessária para um utente que se encontre num serviço de urgência. Posteriormente, tendo em conta as possíveis soluções, realizou-se o planeamento da interface e desenvolveu-se um protótipo. Optou-se por dividir a informação por categorias onde são abordados diferentes temas. Os temas estão associados a subcategorias com intuito a facilitar o acesso à informação, de modo refinar a busca do utilizador. A organização foi realizada do seguinte modo:

- Hospital – Pretende dar a conhecer melhor a instituição através dos temas:
 - Apresentação – Breve apresentação da instituição;
 - Conheça melhor – Informações sobre a constituição das instalações;
 - Valores – Quais os valores defendidos pela instituição;
 - Voluntariado.
- Guia de Utente – Visa essencialmente o utente, fornecendo informações sobre:
 - Transportes – Quais as ligações existentes de linhas de transportes urbanos, assim como informações de táxis;
 - Taxas Moderadoras;
 - Parques de Estacionamento – discriminação dos locais e respetivas taxas;
 - Direitos e Deveres dos Utentes.
- Especialidades – Visa caracterizar os serviços que a instituição disponibiliza através dos temas:
 - Serviços Clínicos – Lista de todos os serviços clínicos presentes na instituição;
 - Serviço de urgência – Caracterização do seu funcionamento;
 - Triagem – Breve explicação do processo de triagem ao utente;
 - Visitas - Horário de visitas e suas condições.

- Serviços Complementares – Constituído por informações úteis sobre:
 - Assistência religiosa – Serviços religiosos oferecidos pela instituição e contatos;
 - Cantina, Cafeteria e Banco – Horários de funcionamento.
- Previna-se – Visa abordar vários problemas de saúde, aconselhando e informando os utentes através dos seguintes temas (informação retirada de [85]):
 - Alcoolismo – Como realizar o tratamento;
 - Deixe de Fumar – Sensibilização e conselhos para deixar de fumar;
 - Como se trata a obesidade – Como realizar o tratamento e quais os seus benefícios;
 - O que é o cancro? – Breve explicação sobre a doença e sintomas associados;
 - Diabetes – Breve explicação sobre a doença e sintomas associados;
 - Tuberculose – Breve explicação sobre a doença, sintomas e formas de tratamento;
 - Depressão - Breve explicação sobre o transtorno e formas de tratamento;
 - Síndrome da Imunodeficiência Adquirida – Breve explicação sobre a doença, formas de transmissão e teste diagnóstico.
- Cidade de Braga – Pretende dar a conhecer melhor a cidade aos utentes através do desenvolvimento dos temas (Informação retirada de [86]):
 - A Cidade – Breve apresentação da cidade de Braga;
 - Eventos tradicionais – Apresentação de alguns eventos que se realizam anualmente na cidade;
 - Gastronomia – dar a conhecer melhor a gastronomia típica da região;
 - Música – dar a conhecer a música típica da região.
- Comunicação – Categoria dedicada à publicação de notícias e eventos ligados à instituição;
- Os nossos parceiros – Seção de apresentação dos parceiros da instituição.

De seguida procedeu-se à programação da interface.

5.1 – Descrição das Funcionalidades

O *HosNui* é composto por mecanismos de reconhecimento de gestos e voz, tendo como objetivo avaliar qual o seu impacto no utente quando implementada no serviço de urgência. Inicialmente a interface encontra-se no modo inativo, enquanto não detetar o utilizador no campo de interação. Nesta fase a interface limita-se a exibir imagens abordadas pelo seu conteúdo (Figura 5.1). Para o utilizador iniciar a sua interação, basta colocar-se em frente ao dispositivo até este o reconhecer dentro do seu campo de interação.



Figura 5.1 – Janela exibida pelo *HosNui* quando o utilizador não se encontra a interagir com este.

A partir do momento em que o utilizador é identificado, a interface exibe a sua forma através dos seus contornos, lançando uma janela de boas vindas (Figura 5.2). Esta janela permite realizar o acesso à UI através de duas maneiras: do comando de voz “*Start*” ou então clicando no botão, recorrendo ao gesto.



Figura 5.2 – Janela de boas vindas do *HosNui*.

Para a interação através da voz, é utilizada o idioma Inglês como forma de reconhecimento. Não foi possível implementar o idioma Português devido ao dispositivo *Kinect* possuir as funcionalidades necessárias para o reconhecimento.

A partir do momento em que a interface é iniciada, os microfones do dispositivo estão sempre ligados e a receber dados, reconhecendo somente palavras-chave para desempenhar as respetivas funções.

O sistema possui um mecanismo para inativar a interface caso o utilizador deixe de ser detetado durante algum tempo. Quando tal acontece, a aplicação retoma à janela inicial, ilustrada na figura 5.1.

As interações são realizadas com base em gestos de apontar. Estes movimentos da mão pretendem substituir as funcionalidades do rato. Caso o utente pretenda selecionar uma categoria, tema ou outro objeto, basta apontar e deixar o cursor em cima durante uns segundos de modo a realizar o clique (Figura 5.3). Esta operação transmite *feedback* ao utilizador de forma a este ter uma melhor noção do desenrolar da interação, transmitindo uma sensação de controlo. Uma vez que o utilizador alvo é o utente, escolheu-se esta forma de clique por achar ser a mais simples e intuitiva, de modo a melhorar a sua experiência com a interface. No entanto, com o avançar da interação pelas sucessivas janelas, este mecanismo pode revelar-se ineficiente uma vez que pode criar conflito nas escolhas, confundindo o utente.

É exemplo disso o caso em que o utente pretende seleccionar uma categoria. Ao fazê-lo, será imediatamente transferido para uma janela com os temas. Caso o utente se encontre ainda com a mão levantada, o cursor poderá imediatamente iniciar o processo de seleção do tema que esteja em cima, o que poderá não ser o pretendido.

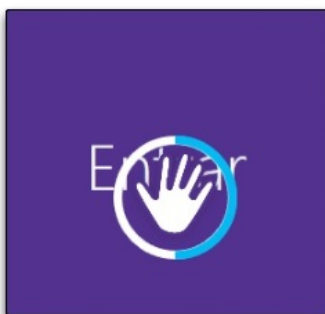


Figura 5.3 – Mecanismo de clique do *HosNui*.

Em todas as janelas, a interface transmite *feedback* do rastreamento do corpo. O modelo do utilizador que se encontra a interagir com a interface é exibido com um tom roxo escuro (Figura 5.4). Conforme existam mais pessoas dentro do campo de interação, estes são identificados com um tom de roxo mais leve, não influenciando a interação do utilizador principal. Assim, conforme a intensidade do tom, os utilizadores tem percepção de quem se encontra a interagir com a interface.



Figura 5.4 – *Feedback* transmitido sobre o estado do rastreamento do utilizador durante a interação o *HosNui*.

A janela principal de interação é composta por várias categorias, nas quais o utilizador pode seleccionar os objetos através do gestos ou simplesmente invocando o nome do tema. Todas as janelas transmitem *feedback* sobre os comandos de voz disponíveis ao utilizador na sua

parte superior. Como é possível observar pela figura 5.5, a UI sugere ao utilizador que invoque o nome da categoria. Mais uma vez, visto que o dispositivo não permite reconhecer o idioma Português, este pode apresentar algumas dificuldades de interpretação dos nomes das categorias.

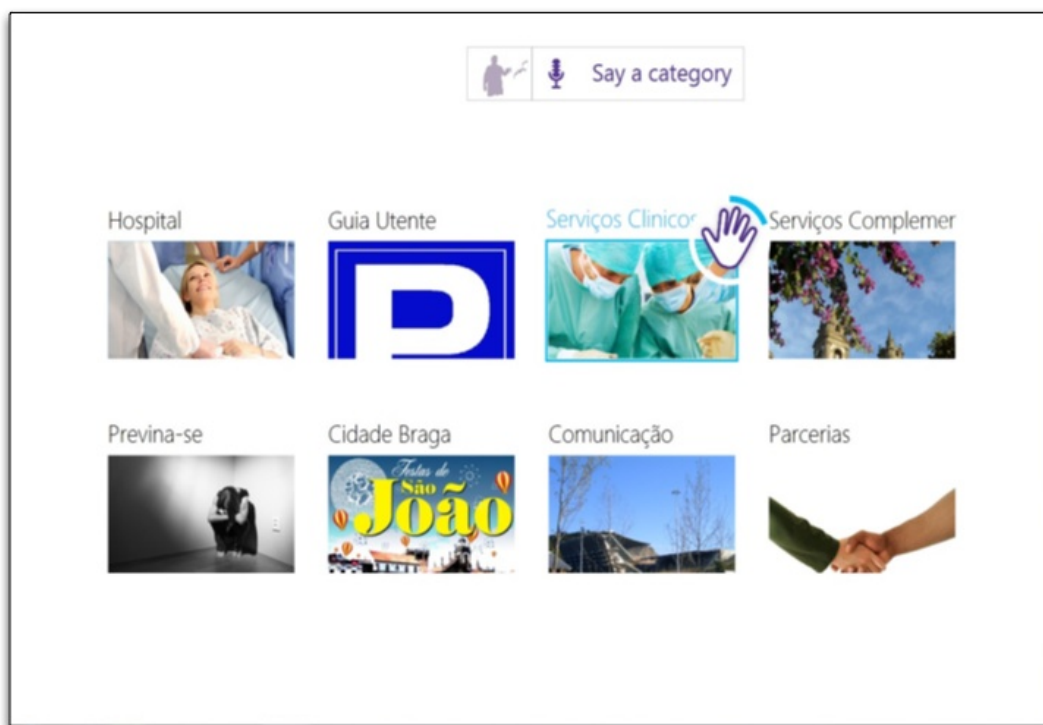


Figura 5.5 – Janela de escolha de categorias do *HosNui*.

Dependendo da categoria selecionada, a aplicação direciona o utilizador para uma segunda janela onde se encontram expostos os respetivos temas (Figura 5.6). O processo de seleção de cada tema através de gestos mantém-se, alterando apenas a forma como a interação com comandos de voz é realizada. Os temas surgem numerados, bastando apenas ao utilizador invocar o seu número em Inglês. Nesta fase, a janela fornece *feedback* sobre os comandos de voz disponíveis ao utilizador de modo a facilitar a sua interação. O retrocesso à janela anterior pode ser realizado ou através do comando de voz “*Home*” ou, então, através da seleção do objeto.

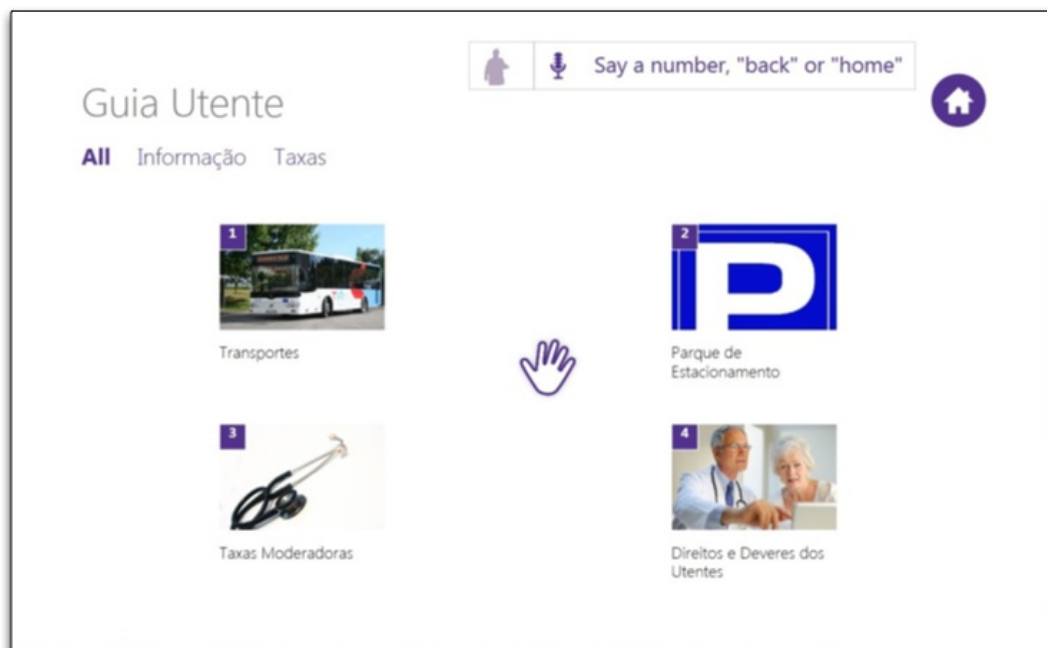


Figura 5.6 - Janela de escolha de temas dentro de uma determinada categoria do *HosNui*.

Dependendo do tema selecionado, a UI direciona o utilizador para uma janela onde se encontra exposta a informação pretendida. Esta janela disponibiliza mecanismos para mover verticalmente a janela, de modo a observar a totalidade da informação. Esta operação é realizada através de um botão presente na barra lateral (Figura 5.7). Em todos os temas existem mecanismos de avaliação da informação, expressando se a informação é útil ou não. A avaliação é feita através da seleção de dois objetos para tal. Posteriormente, se o utente pretender regressar à janela de seleção de temas, basta selecionar o objeto presente no canto superior esquerdo da janela ou então invocar o comando “back”. Do mesmo modo, se pretender regressar à janela de seleção de temas, basta selecionar o objeto presente no canto superior direito da janela ou então invocar o comando de voz “home”. Quando o utente pretender deixar de interagir com a UI, basta sair do campo de interação do dispositivo, levando a UI a desativar a interação e voltar a janela inicial.



Figura 5.7 - Exemplo de uma janela do *HosNui*, onde é abordado o tema dos direitos e deveres do utente.

5.2 – Estudo Estatístico

Posteriormente aplicação foi posta em operação de modo a tentar perceber qual o impacto junto dos utentes. Pretendeu-se analisar o nível de aceitação do *HosNui*, assim como o seu impacto quando disposta num meio simulado de serviço de urgência de um meio hospitalar. Para isso procedeu-se à recolha de dados estatísticos. Esta recolha de dados consistiu na realização de inquéritos por parte de potenciais utentes.

Apesar de os testes não terem sido realizados num ambiente de um serviço de urgência real de uma instituição de saúde, tentou-se recriar as mesmas condições durante a interação dos possíveis utentes. Para isso, introduziu-se um certo nível de ruído de fundo normalmente presente em abundância, de modo a tentar estabelecer da melhor forma possível o ambiente alvo. Com esta alteração pretendeu-se essencialmente verificar qual a sua influência no reconhecimento de voz da UI.

O estudo incidiu sobre uma amostra de trinta utilizadores de uma população. Esta amostra pretendia representar os utentes que podem frequentar o serviço de urgência. Não foi possível calcular o tamanho da amostra ideal, uma vez que não se conseguiu obter o número médio de utentes que frequentam diariamente o serviço de urgência. Assim, a escolha da

amostragem foi realizada de uma forma simples e aleatória, ou seja, a escolha de um sujeito não condiciona a seleção do sujeito seguinte.

Inicialmente os utilizadores foram incentivados a testarem a UI em estudo, sem qualquer tipo de ajuda. Seguidamente, a amostra de utilizadores procedeu à realização de um questionário (Anexo E) de modo a avaliar o desempenho da UI. Cada questão é possível de ser avaliada dependendo da opinião do utilizador. A avaliação é feita segundo um intervalo que vai desde um a cinco valores. O valor um corresponde a “discordo completamente” e o valor cinco corresponde a “concordo completamente”.

Análise dos Resultados

Da análise dos resultados obtidos através do questionário, foi possível verificar em que aspetos a UI mais se destaca e vice-versa, assim como o nível de satisfação dos utilizadores com a sua interação. O resultados obtidos encontram-se discriminados na Tabela 5.1.

Tabela 5.1 – Resultados obtidos através dos dados recolhidos no inquérito.

Percentagem(%)	1	2	3	4	5
1) A informação está bem organizada de uma forma clara	0%	0%	0%	87%	13%
2) Design é agradável	0%	0%	7%	50%	43%
3) A tarefa que está a ser executada é facilmente identificada	0%	0%	23%	57%	20%
4) A linguagem utilizada é simples e adequada	0%	0%	10%	77%	10%
5) A interface está estruturada de uma forma simples e clara	0%	0%	7%	67%	27%
6) A consulta de dados é rápida	0%	0%	20%	70%	10%
7) A interação é natural e intuitiva	0%	0%	3%	17%	80%
8) Os gestos são adequados	0%	0%	3%	43%	53%
9) Reconhecimento de gestos é preciso	0%	3%	13%	73%	10%
10) Comandos de voz são	0%	23%	57%	10%	10%

adequados					
11) Reconhecimento de voz é preciso	0%	37%	47%	7%	7%
12) Interação está bem repartida entre modo de reconhecimento de voz e gestual	0%	10%	60%	23%	7%
13) Interface transmite <i>feedback</i> necessário	0%	0%	13%	67%	20%
14) Interação é agradável	0%	0%	0%	30%	70%
15) A interface é útil	0%	0%	0%	77%	23%
16) No geral, está satisfeito com a interface	0%	0%	0%	80%	20%

Como é possível observar pelos dados presentes na tabela, de um modo geral, a UI obteve bons resultados junto dos possíveis utentes, exceto na interação através de comandos de voz. Maioritariamente, os utilizadores concordaram que a informação presente na UI estava bem organizada, de uma forma clara, assim como o seu design era agradável. As tarefas que foram executadas são facilmente identificadas, a linguagem utilizada é simples e adequada e a consulta de dados é rápida. De um modo geral, a UI está estruturada de uma forma simples e clara.

No que diz respeito à natureza da interação da UI, 80% dos utilizadores concorda plenamente que é natural e intuitiva e 17% concorda, sendo que apenas 3% é indiferente (Figura 5.8). Estes dados permitem concluir que a forma de interação apresentada pela UI em estudo não representa um obstáculo para os utilizadores.

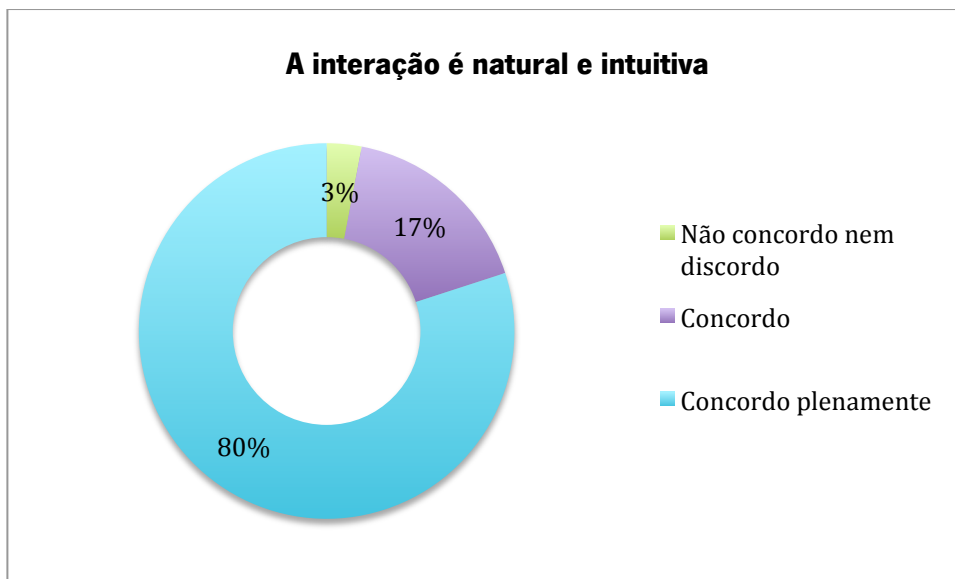


Figura 5.8 – Natureza da interação (questão nº7.)

No que diz respeito aos gestos utilizados na interação, estes revelaram ser adequados. Como é possível verificar pela figura 5.9, 53% dos utilizadores concordam plenamente que os gestos são adequados, 43% concordam e 3% são indiferentes. Este caso não reuniu a plenitude de concordância dos utilizadores devido à forma como o clique é realizado. Para uma grande parte dos utilizadores, o clique deveria ter uma mecanismo diferente, uma vez que com a forma colocada induz em erro. Uma das sugestões seria implementar o gesto de “empurrar”.

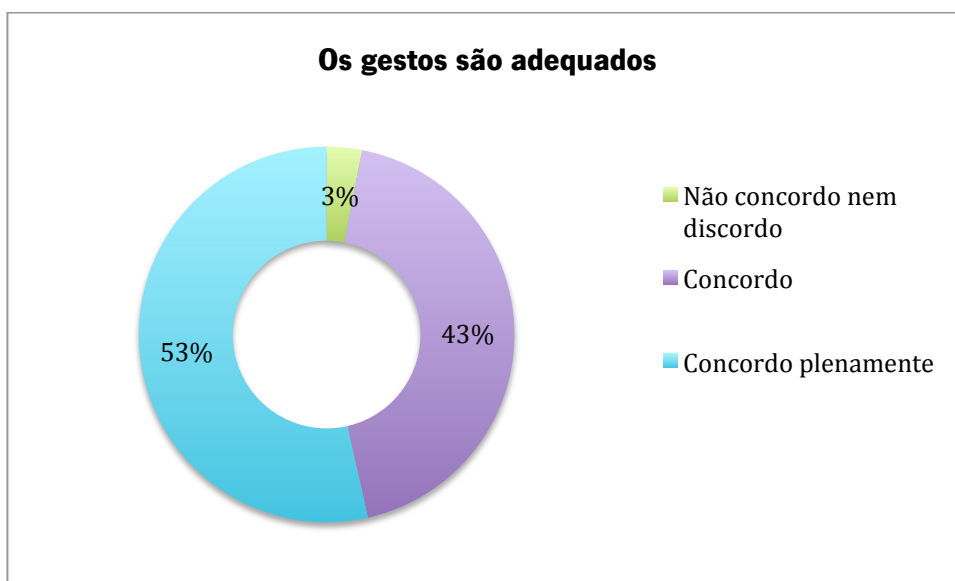


Figura 5.9 - Adequação dos gestos (questão nº8).

Quanto ao reconhecimento de gestos, de um modo geral, os utilizadores concordam que este é preciso (Figura 5.10). Da amostra de utilizadores, 73% concorda que é preciso, 13% é indiferente, 10% concorda plenamente e 3% discorda. De facto, verifica-se que, em certos momentos da interação, o reconhecimento apresenta algumas falhas. Em alguns casos, verifica-se quando se realiza um movimento mais rápido. Este fator pode dever-se ao *software* utilizado, otimização da UI e limitações do dispositivo. Mesmo assim, a maioria concorda que é fiável para a tarefa em questão, não pondo em causa a sua interação.

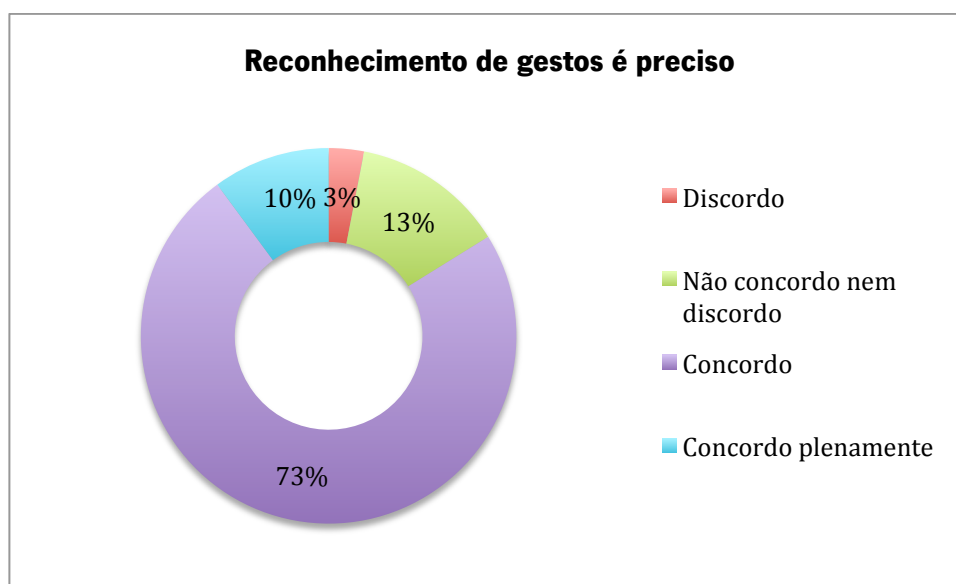


Figura 5.10 – Precisão do reconhecimentos de gestos (questão nº9).

Sobre os comandos de voz, foi possível verificar que 57% dos utilizadores é indiferente à adequação dos comandos de voz, sendo que 23% não concorda (Figura 5.11). Para a obtenção destes valores, teve influência o facto do idioma usado pela UI ser o Inglês, uma vez que o dispositivo não suporta o Português. Este fator pode condicionar a utilização de comandos de voz como forma de interação com a UI.

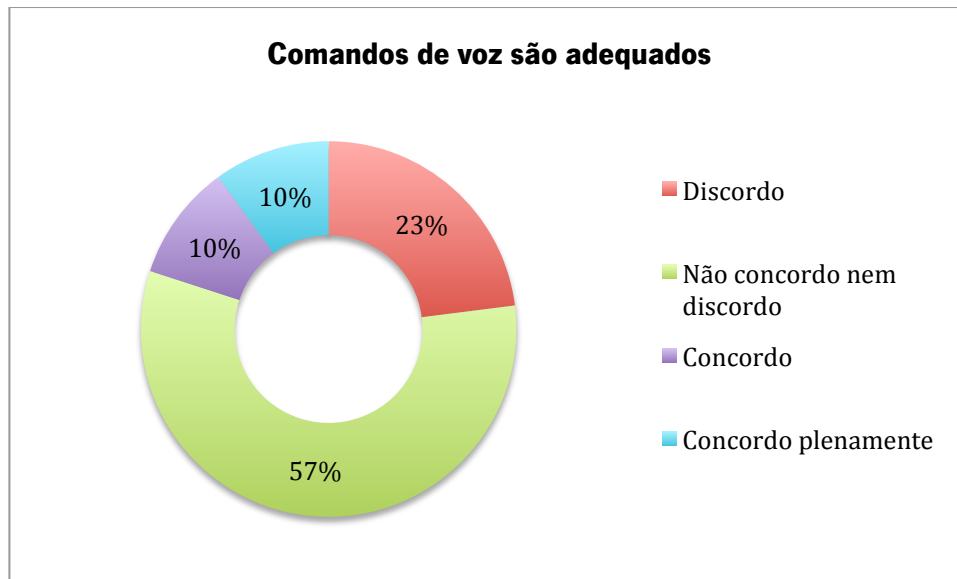


Figura 5.11 – Adequação para comandos de voz (questão nº10).

No que diz respeito à precisão do reconhecimento de voz, 47% dos utilizadores é indiferente e 37% discorda (Figura 5.12). Foi possível verificar a existência de uma certa dificuldade em validar os comandos de voz, sendo necessário repetir mais do que uma vez na maioria dos casos. A existência de ruído de fundo condicionada o reconhecimento de voz. Este fator revelou ser um inconveniente para os utilizadores, influenciando na quebra de utilização de comandos de voz.

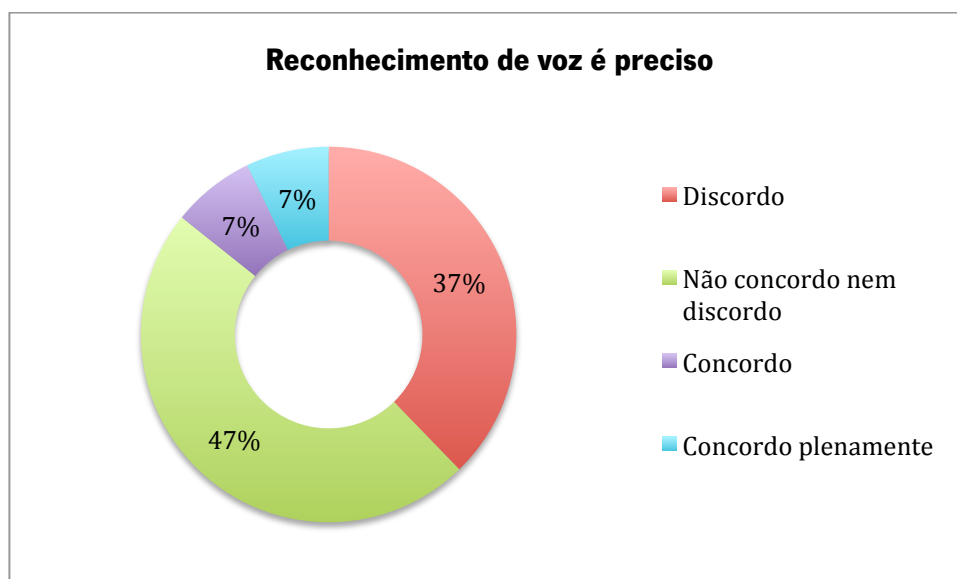


Figura 5.12 - Precisão do reconhecimentos de voz (questão nº11).

Relativamente ao modo como a UI reparte a interação entre os modos de reconhecimento, 23% concorda que está bem feita e 60% é indiferente (Figura 5.13). De salientar que 10% dos utilizadores não concorda. Foi possível verificar que os utilizadores sentiam maior conforto e curiosidade em utilizar os gestos que voz, deixando um bocado de lado os comandos de voz. Além disso verificou-se que o utilizador sentia-se muito mais à vontade em manusear a UI através de gestos do que com comandos de voz. A utilização de comandos de voz leva os utilizadores a sentirem-se constrangidos na presença de público. Este facto leva a crer que a sua aplicabilidade, no cenário de serviços de urgência, iria cair em desuso.

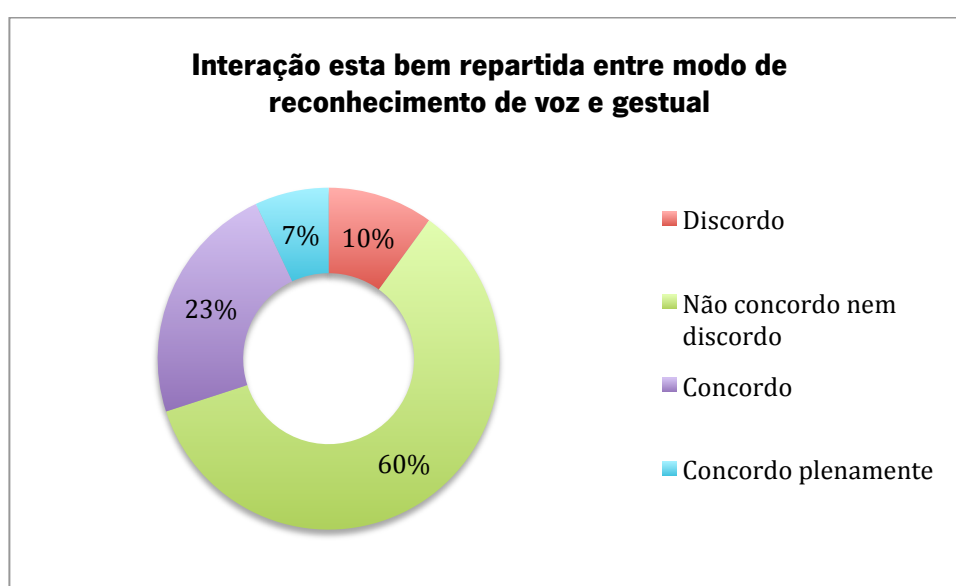


Figura 5.13 – Repartição do reconhecimento de voz e gestual (questão nº12).

Sobre o facto de a interação ser agradável, a maioria dos utilizadores está de acordo, sendo que 30% concorda e 70% concorda completamente (Figura 5.14). Da interação de todos os utilizadores, é possível notar um certo à vontade e conforto em manusear a UI, sendo um dos pontos onde a maioria das opiniões coincidiram.

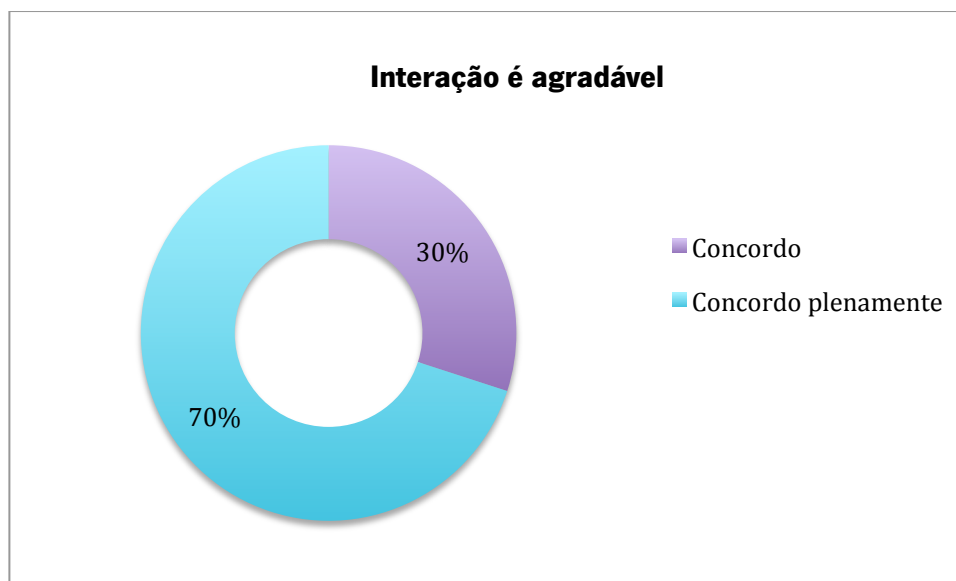


Figura 5.14 – Nível de satisfação com a interação (questão nº14).

No que diz respeito à utilidade da UI, 77% dos utilizadores concorda que é útil, sendo que 23% concorda plenamente (Figura 5.15). Por último, 80% dos utilizadores ficou satisfeito com a UI e 20% muito satisfeito (Figura 5.16). Apesar destes dados revelarem que os utilizadores obtiveram uma experiência muito positiva, aquando da utilização da UI, seria muito difícil prever se iria obter a aceitação dos utentes, quando disposta num meio indicado. Poderia, por exemplo, verificar-se, numa fase inicial, uma grande afluência dos utentes à UI por mera curiosidade, mas, com o passar do tempo, o seu uso caísse. Deste modo seria necessário realizar uma implementação experimental na instituição de saúde para tentar perceber se este iria beneficiar da implementação da UI.

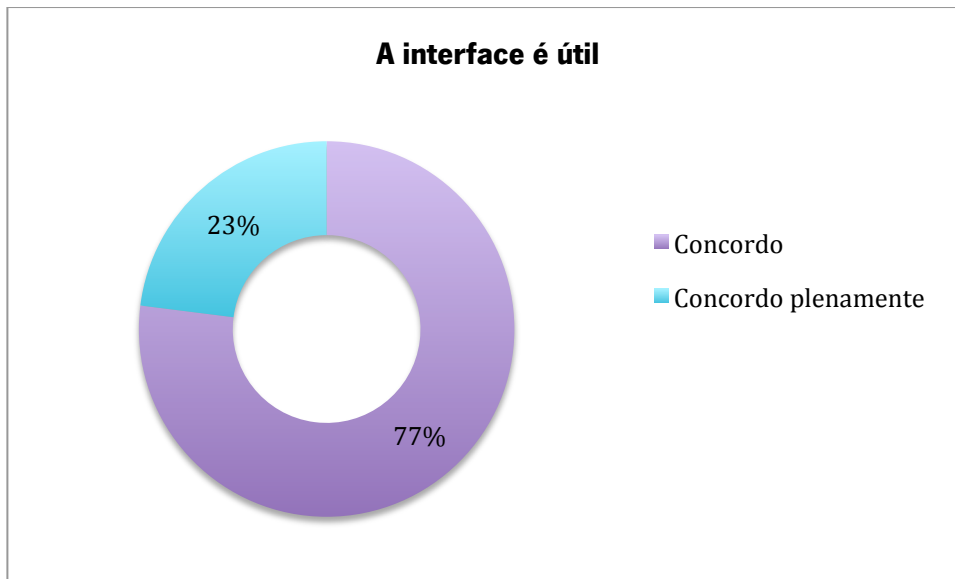


Figura 5.15 – Utilidade da interface (questão nº15).

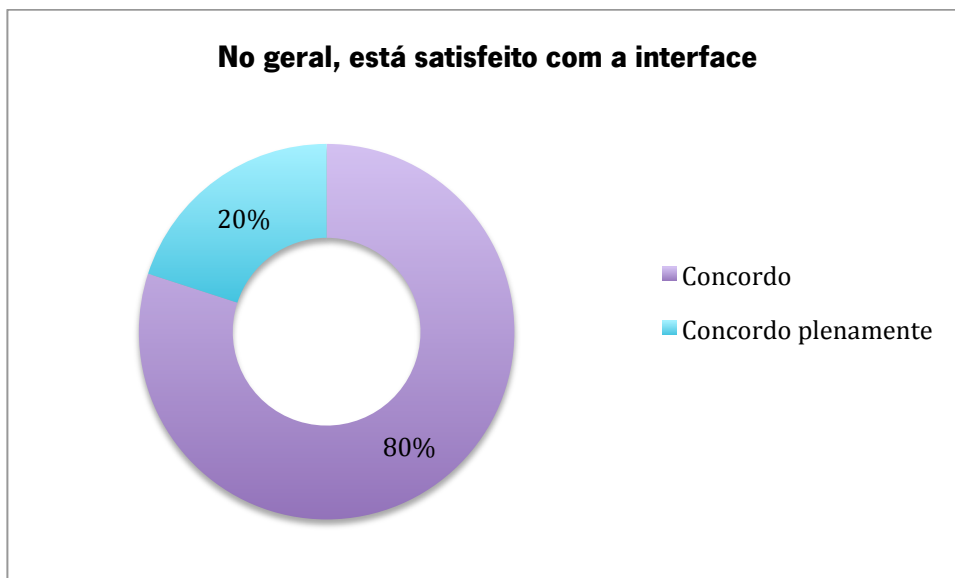


Figura 5.16 – Satisfação com interface (questão nº16).

CAPÍTULO 6

Conclusões

Ideias chave:

- Não se pretende obter uma solução definitiva para a problemática exposta mas contribuir com soluções pontuais.
- A introdução de reconhecimento de voz pode não ser o mais apropriado em determinados casos.
- Os desenvolvimentos apresentam características que permitem auxiliar na prevenção e controlo de infeções hospitalares, através da redução da necessidade de contato direto com a interface de utilizador.
- Possibilidade futura de criação de diferentes protótipos para auxílio das tarefas dos profissionais de saúde, como médicos e enfermeiros.

O presente capítulo enquadra e caracteriza de forma sinóptica o trabalho de pesquisa efetuado e apresenta os resultados e conclusões fruto deste trabalho. Procurou-se contribuir para o aumento do conhecimento na área da Saúde e da Informática Médica, mais concretamente nos novos paradigmas de UI para aplicações em ambientes hospitalares.

Não se pretendeu obter uma solução definitiva para a problemática exposta, mas sim contribuir com algumas soluções pontuais, sendo, neste caso, o estudo realizado sobre as potencialidades do dispositivo *Kinect* e o desenvolvimento do *HosNui*, a face mais visível das contribuições aqui apresentadas.

Pretende-se, assim, que as soluções apresentadas possam contribuir para um despertar do desenvolvimento de novas formas de interação de UI aplicadas à área da saúde, em futuros estudos.

Findo este trajeto, é de mencionar a grande satisfação pessoal que o contributo deste trabalho possa ter para o auxílio da melhoria contínua dos serviços de saúde prestados.

6.1 – Síntese

Com o surgimento de novas tecnologias de informação e comunicação, têm vindo a ser desenvolvidos novos tipos de UI. É neste sentido que surge a necessidade de direcionar estes novos avanços na área da saúde, de modo a obter um serviço mais eficiente e com maior qualidade, sendo nestes pontos que incide a presente dissertação. Pode dizer-se, ainda, que estes desenvolvimentos apresentam características que permitem auxiliar na prevenção e controlo de infeções hospitalares, reduzindo a necessidade de contato direto com os objetos e consequentemente a sua propagação.

Com a efetivação deste trabalho, ambiciona-se realizar o estudo das potencialidades do reconhecimento gestual aplicadas à UI para implementação na área da saúde. Em adição, será desenvolvido um protótipo de modo a validar as UI propostas, tendo em conta a utilização das tecnologias recentemente introduzidas no mercado. Para tal foi adquirido um modelo do dispositivo *Kinect* com vista a possibilitar o seu desenvolvimento.

No primeiro capítulo foram apresentadas notas introdutórias relativas ao surgimento de novas tecnologias e consequente aposta das instituições de saúde em novos dispositivos e UI, como meio de colmatar o aumento significativo de dados de informação. Também foi abordada a problemática da prevenção e controlo de infeções hospitalares e a necessidade crescente de otimização dos processos de acesso e utilização da informação. Expuseram-se os objetivos do trabalho e a organização da dissertação.

No segundo capítulo abordou-se a temática específica das UI. Facultou-se uma breve introdução, assim como se abordou o seu processo de desenvolvimento, focando pontos como: a análise de domínio, usabilidade, modelo mental, design das janelas virtuais, desenvolvimento do protótipo, programação e testes. Foram referidos os principais tipos de UI existentes. Por último, abordou-se o processo de implementação das UI em meios hospitalares.

No terceiro capítulo, abordou-se a temática do reconhecimento gestual. Foi apresentado o estado da arte das UI de reconhecimento gestual, referindo alguns recursos a tecnológicas existentes. Foram expostos alguns conceitos de linguagem gestual para a correta

implementação nas interfaces, assim como foram expostas as diferentes formas de sistemas de reconhecimento gestual, dividindo-as nas duas principais categorias: superfícies táteis e *Free-Forms*. Foram, também, apresentadas as características das UI de reconhecimento gestual. Abordou-se, ainda, a temática das UI multimodais, facultando uma breve explicação do processo de reconhecimento de voz e quais as vantagens na sua implementação em conjunto com o reconhecimento de gestos. Por fim, destacaram-se os diferentes tipos de dispositivos de reconhecimento gestual existentes, enunciando as suas características essenciais.

No quarto capítulo foi elaborada uma introdução técnica ao dispositivo *Kinect*, utilizado no âmbito do tema da dissertação. Enunciou-se as características técnicas do sensor, essenciais para realizar o reconhecimento gestual e voz, assim como seu funcionamento. Foi abordada a forma como o dispositivo procede à deteção e rastreamento do utilizador e consequente reconhecimento gestual, de voz e facial. Foram, ainda, descritos os testes técnicos realizados às funcionalidades do dispositivo como sensor RGB, sensor de profundidade e microfones. Mostrou-se aplicabilidade prática da capacidade de reconhecimento gestual e de voz conferida pelo dispositivo, sugerindo possíveis aplicações e quais as melhores características a explorar na UI, de modo a serem implementadas em ambientes hospitalares.

Por último apresentaram-se as etapas de desenvolvimento do *HosNui*, tendo em conta as potencialidades do dispositivo *Kinect* para a implementação num possível ambiente de um serviço de urgência. Foram especificadas quais as ferramentas utilizadas para o desenvolvimento e a forma como foi organizada a informação presente na interface.

Elaborou-se um levantamento das principais funcionalidades, apresentando posteriormente o estudo estatístico realizado de modo a validar a interface. Por último, são apresentados os resultados e respetivas interpretações.

6.2 – Conclusões

Uma das principais conclusões a que se chegou foi a de que as UI de reconhecimento gestual apresentam enormes potencialidades, tendo sido possível verificar este facto numa primeira fase dos testes do dispositivo *Kinect*.

Numa fase posterior, em que o *HosNui* já se encontrava desenvolvida, procedeu-se à avaliação do seu desempenho. Para isto, recorreu-se à análise estatística do conteúdo de uma série de

questionários aplicados a uma amostra de possíveis utilizadores. Da informação recolhida nestes questionários, foi possível perceber que, de um modo geral, os utilizadores revelaram um elevado grau de satisfação na interação com a UI, destacando-se a facilidade e a simplicidade com que é realizada esta interação. No entanto, este processo não é perfeito, pelo que, em certos momentos é possível diagnosticar algumas imperfeições no reconhecimento estabelecido pela UI.

Foi possível notar que um maior controlo dos fatores de interação por parte do utilizador, permitem-lhe obter uma melhor experiência com a interface, reduzindo assim a probabilidade de interações interrompidas. Ainda assim, o estudo revelou que a aplicação do reconhecimento de voz na interface não era uma mais-valia, pelo contrário. Segundo o que se conseguiu apurar, o reconhecimento de voz não é muito eficiente, relevando-se, por vezes, ineficaz, nomeadamente quando o dispositivo se encontra inserido num meio com algum ruído de fundo. Considerando que o ambiente de um serviço de urgência é caracterizado por algum ruído, esta imperfeição do dispositivo inviabiliza a sua utilização num ambiente deste género. Por outro lado, foi também possível verificar a existência de um certo constrangimento entre os utilizadores, no que diz respeito à utilização de comandos de voz como forma de interação. É importante ter em consideração que este fator pode influenciar redução da aceitação da interface ou até mesmo inviabilizar a sua utilização.

Ainda neste rol de desvantagens, pode-se referir o facto de o dispositivo *Kinect* não oferecer suporte para o idioma Português. No caso do *HosNui* este aspeto revelou ser uma grande desvantagem, já que nem todos os utilizadores se encontravam familiarizados com idioma Inglês.

Em suma, embora esteja comprovado que o reconhecimento de voz auxilia o utilizador na interação com a UI, neste caso prático, o reconhecimento de voz pode tornar-se um inconveniente para o sistema.

É também importante salientar que, de um modo geral, os utilizadores revelaram uma receção positiva a esta interação através de gestos livres, não apresentando qualquer tipo de dificuldade na sua utilização. Assim sendo, e como se constatou pelo estudo, a interface obteve muito boa aceitação por parte dos utilizadores, o que é um bom indicador para a sua implementação em futuras aplicações.

Contudo, apesar do *feedback* positivo recebido por parte dos utilizadores em relação ao *HosNui*, é imprudente afirmar que esta iria vingar quando disposta num serviço de urgência,

tendo como base apenas estes dados. Assim, o mais prudente seria realizar uma implementação experimental na instituição de saúde para tentar perceber que impacto a UI teria junto dos utentes e consequente nível de aceitação.

6.3 – Novos Desafios

Como em qualquer investigação, o embrenhar no tema fez com que despoletassem inúmeras ideias e sugestões para implementação futura. Neste sentido, seguem-se alguns apontamentos para projetos de aplicação direta no sistema desenvolvido.

Em primeiro lugar seria interessante testar a aplicação da UI num ambiente hospitalar real. O objetivo seria, pois, verificar se os resultados correspondiam ou não aos apresentados neste trabalho. Além disso, esta experiência permitiria observar a reação do utilizador relativamente à interface.

Além desta aplicação da interface no ambiente real seria também estimulante, para trabalho futuro, o desenvolvimento de aplicações não só para pacientes, mas também para os profissionais de saúde, nomeadamente médicos e enfermeiros. No fundo, o que se pretendia com isto seria a aplicação da tecnologia em outras e novas formas de interfaces, de modo a auxiliar os profissionais de saúde na execução das suas tarefas.

Referências

- [1] P. Tang and V. Patel, "Major issues in user interface design for health professional workstations: summary and recommendations," *International journal of bio-medical computing*, 1994.
- [2] R. W. Picard, A. Wexelblat, and C. I. N. I. Clifford I. Nass, "Future interfaces," in *CHI '02 extended abstracts on Human factors in computing systems - CHI '02*, 2002, p. 698.
- [3] C. M. Johnson, T. R. Johnson, and J. Zhang, "A user-centered framework for redesigning health care interfaces.," *Journal of biomedical informatics*, vol. 38, no. 1, pp. 75–87, Feb. 2005.
- [4] Organização Mundial da Saúde, *Prevenção de infecções Adquiridas no hospital*, 2ª Edição. World Health Organization, 2002.
- [5] G. Waloszek, "Interaction Design Guide for Touchscreen Applications," *Engineering Applications of Artificial Intelligence*, 2000.
- [6] F.-G. Wu, H. Lin, and M. You, "Direct-touch vs. mouse input for navigation modes of the web map," *Displays*, vol. 32, no. 5, pp. 261–267, Dec. 2011.
- [7] Y.-C. Lu, Y. Xiao, A. Sears, and J. A. Jacko, "A review and a framework of handheld computer adoption in healthcare," *International Journal of Medical Informatics*, vol. 74, no. 5, pp. 409–422, 2005.
- [8] J. Rhoads and E. Drazen, "Touchscreen Check-In : Kiosks Speed Hospital Registration Touchscreen," *California Healthcare Foundation*, no. March, 2009.
- [9] M. Rehm, N. Bee, and E. André, "Wave like an Egyptian: accelerometer based gesture recognition for culture specific interactions," in *Proceedings of the 22nd British HCI Group Annual Conference on People and Computers: Culture, Creativity, Interaction - Volume 1*, 2008, pp. 13–22.
- [10] T. Leyvand, C. Meekhof, Y.-C. Wei, J. Sun, and B. Guo, "Kinect Identity: Technology and Experience," *Computer*, vol. 44, no. 4. pp. 94–96, 2011.

- [11] S. Engelhart, E. Fischnaller, A. Simon, and J. Gebel, "Microbial contamination of computer user interfaces (keyboard, mouse) in a tertiary care centre under conditions of practice," *Hyg Med*, pp. 504–507, 2008.
- [12] D. Pittet, B. Allegranzi, H. Sax, S. Dharan, C. L. Pessoa-Silva, L. Donaldson, and J. M. Boyce, "Evidence-based model for hand transmission during patient care and the role of improved practices.," *The Lancet infectious diseases*, vol. 6, no. 10, pp. 641–52, Oct. 2006.
- [13] B. Hartmann, M. Benson, A. Junger, L. Quinzio, R. Röhrig, B. Fengler, U. W. Färber, B. Wille, and G. Hempelmann, "Computer keyboard and mouse as a reservoir of pathogens in an intensive care unit.," *Journal of clinical monitoring and computing*, vol. 18, no. 1, pp. 7–12, Feb. 2004.
- [14] W. J. Spicer, *Clinical microbiology and infectious diseases: an illustrated colour text*, 2nd Editio. Elsevier, 2008, p. 251.
- [15] S. Lauesen, *User Interface Design: A Software Engineering Perspective*. Addison Wesley, 2005, p. 624.
- [16] B. Shneiderman, *DESIGNING THE USER INTERFACE*, 4th Editio. Pearson Addison Wesley, 2004.
- [17] S. Lauesen and M. B. Harning, "Virtual windows: linking user tasks, data models, and interface design," *IEEE Software*, vol. 18, no. 4, pp. 67–75, Jul. 2001.
- [18] J. Tidwell, *Designing Interfaces*, 2nd Editio. O'Reilly Media, 2011.
- [19] L. Mathis, *Designed for Use: Create Usable Interfaces for Applications and the Web*, 1st ed. Pragmatic Bookshelf, 2011.
- [20] M. J. Davidson, L. Dove, and J. Wertz, "Mental Models." [Online]. Available: http://www.lauradove.info/reports/mental_models.htm. [Accessed: 08-May-2012].
- [21] R. M. Young, "Surrogates and mappings: two kinds of conceptual models for interactive devices," in *Mental Models*, D. Gentner and A. L. Stevens, Eds. Lawrence Erlbaum Associates, 1983.
- [22] B. Shneiderman and C. Plaisant, *Designing the User Interface: Strategies for Effective Human-Computer Interaction (4th Edition)*. Pearson Addison Wesley, 2004.
- [23] D. Wigdor and D. Wixon, *Brave NUI World: Designing Natural User Interfaces for Touch and Gesture*, 1st ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011.
- [24] Y. Wu and T. S. Huang, "Vision-Based Gesture Recognition: A Review," in *Proceedings of the International Gesture Workshop on Gesture-Based Communication in Human-Computer Interaction*, 1999, pp. 103–115.

- [25] P. K. Bhardwaj, D. Kleiman, B. Barber, and K. Andreou, *How to Cheat at Windows System Administration Using Command Line Scripts*. Syngress Publishing, 2006.
- [26] J. Blake, *Natural User Interface in .Net*. Manning Publications, 2012, p. 375.
- [27] K.-W. Lee and Y.-C. Lee, "Design and validation of an improved graphical user interface with the 'Tool ball'.," *Applied ergonomics*, vol. 43, no. 1, pp. 57–68, Jan. 2012.
- [28] P. Atkinson, "The Best Laid Plans of Mice and Men: The Computer Mouse in the History of Computing," *Design Issues*, vol. 23, no. 3, pp. 46–61, Jul. 2007.
- [29] M. PORTA, "Vision-based user interfaces: methods and applications," *International Journal of Human-Computer Studies*, vol. 57, no. 1, pp. 27–73, Jul. 2002.
- [30] W. O. Galitz, *The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques*. Wiley Publishing Inc., 2007.
- [31] D. Saffer, *Designing Gestural Interfaces: Touchscreens and Interactive Devices*. O'Reilly Media, Inc., 2008.
- [32] C. M. Johnson and J. P. Turley, "The significance of cognitive modeling in building healthcare interfaces.," *International journal of medical informatics*, vol. 75, no. 2, pp. 163–72, Mar. 2006.
- [33] V. L. Patel and A. W. Kushniruk, "Interface design for health care environments: the role of cognitive science," in *Proc AMIA Symposium*, 1998, pp. 29–37.
- [34] J. Nielsen, *Usability engineering*. Academic Press, 1993.
- [35] Corepoint Health, "What is Your Healthcare Interface Method?," 2009 [Online]. Available: <http://www.corepointhealth.com/sites/default/files/whitepapers/healthcare-clinical-interface-solution.pdf>. [Accessed: 3-April-2012].
- [36] Informatica, "Integration Engine – Solutions for Healthcare Providers." [Online]. Available: <http://www.informatica.com/us/solutions/industry-solutions/healthcare/healthcare-solutions-for-providers/integration-engine/>. [Accessed: 18-Jul-2012].
- [37] Corepoint Health, "MEDITECH, MUSE and the Unique Challenges of Point-to-Point Interfaces." [Online]. Available: <http://www.corepointhealth.com/geni/meditech-muse-and-unique-challenges-point-point-interfaces>. [Accessed: 19-Jul-2012].
- [38] HL7 International, "Health Level Seven International - Homepage." [Online]. Available: <http://www.hl7.org/index.cfm>. [Accessed: 18-Jul-2012].

- [39] OrionHealth, “HL7 Interface Engine.” [Online]. Available: http://www.hl7.com/interface_engine/. [Accessed: 18-Jul-2012].
- [40] Bill Buxton and M. Billinghurst, “Gesture Based Interaction,” 2011. [Online]. Available: <http://www.billbuxton.com/input14.Gesture.pdf>. [Accessed: 18-Jul-2012].
- [41] E. De Vries, “Touchscreen interaction for healthcare applications,” University of Twente, 2009.
- [42] D. Brown, C. Steinbacher, T. Turpin, R. Butler, and C. Bales, “Company history from Elographics to Elo TouchSystems.” [Online]. Available: <http://www.elotouch.com/AboutElo/History/>. [Accessed: 17-May-2012].
- [43] M. Naumis, “Touch Screen Interfaces and their Advantages in Personal Computing.” [Online]. Available: http://www.mikenaumis.com/files/CCT400_Essay.pdf. [Accessed: 18-May-2012].
- [44] V. I. Pavlovic, R. Sharma, and T. S. Huang, “Visual interpretation of hand gestures for human-computer interaction: a review,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 677–695, Jul. 1997.
- [45] G. Demming, “Sony EyeToy™: Developing Mental Models for 3-D Interaction in a 2-D Gaming Environment,” *Citeseer*, pp. 575–582.
- [46] Microsoft, “Kinect - Xbox.com.” [Online]. Available: <http://www.xbox.com/en-US/KINECT>. [Accessed: 11-May-2012].
- [47] Microsoft, “Develop for Kinect | Microsoft Kinect for Windows.” [Online]. Available: <http://www.microsoft.com/en-us/kinectforwindows/>. [Accessed: 02-May-2012].
- [48] J. P. Wachs, H. I. Stern, Y. Edan, M. Gillam, J. Handler, C. Feied, and M. Smith, “A gesture-based tool for sterile browsing of radiology images,” *Journal of the American Medical Informatics Association : JAMIA*, vol. 15, no. 3, pp. 321–3.
- [49] G. C. S. Ruppert, L. O. Reis, P. H. J. Amorim, T. F. de Moraes, and J. V. L. da Silva, “Touchless gesture user interface for interactive image visualization in urological surgery,” *World journal of urology*, May 2012.
- [50] K. K. Matsumura and R. Sonnino, “FUSION4D - INTERFACE NATURAL E IMERSIVA PARA MANIPULAÇÃO DE OBJETOS 3D FUSION4D - INTERFACE NATURAL E IMERSIVA PARA MANIPULAÇÃO DE OBJETOS 3D,” Escola Politécnica da Universidade de São Paulo, 2011.
- [51] A. Mulder, “Hand gestures for HCI.” [Online]. Available: <http://xpsasm.com/x/sfu/vmi/HCI-gestures.htm#ergotic>. [Accessed: 26-Jul-2012].
- [52] C. Cadoz, *Les réalités virtuelles*. Dominos, 1994, p. 125.

- [53] F. K. H. Quek, "Toward a vision-based hand gesture interface," in *Proceedings of the conference on Virtual reality software and technology*, 1994, pp. 17–31.
- [54] A. Holzinger, "Finger instead of mouse: touch screens as a means of enhancing universal access," in *Proceedings of the User interfaces for all 7th international conference on Universal access: theoretical perspectives, practice, and experience*, 2003, pp. 387–397.
- [55] C. Moussette and D. Fallman, "Designing for touch: Creating and building meaningful haptic interfaces," *Proceedings of IASDR*, 2009.
- [56] Planar, "Touchscreen technology for Embedded Displays." [Online]. Available: <http://www.planarembded.com/technology/touch/>. [Accessed: 25-May-2012].
- [57] C. Hopping, "Infographic: Capacitive vs resistive vs infrared touchscreens." [Online]. Available: http://www.knowyourmobile.com/blog/988344/infographic_capacitive_vs_resistive_vs_infrared_touchscreens.html. [Accessed: 30-Jul-2012].
- [58] J. Y. Han, "Low-cost multi-touch sensing through frustrated total internal reflection," in *Proceedings of the 18th annual ACM symposium on User interface software and technology*, 2005, pp. 115–118.
- [59] P. Garg, N. Aggarwal, and S. Sofat, "Vision based hand gesture recognition," *World Academy of Science, ...*, pp. 972–977, 2009.
- [60] C. Harshith, K. R. Shastry, M. Ravindran, M. V. V. N. . Srikanth, and N. Lakshmikanth, "Survey on Various Gesture Recognition Techniques for Interfacing Machines Based on Ambient Intelligence," *International Journal of Computer Science & Engineering Survey*, vol. 1, no. 2, pp. 31–42, Nov. 2010.
- [61] A. Stojanov, "Gesture-based Human-computer Interaction," Jacobs University Bremen, 2009.
- [62] C.-Y. Kao and C.-S. Fahn, "A Human-Machine Interaction Technique: Hand Gesture Recognition Based on Hidden Markov Models with Trajectory of Hand Motion," *Procedia Engineering*, vol. 15, pp. 3739–3743, Jan. 2011.
- [63] F.-S. Chen, C.-M. Fu, and C.-L. Huang, "Hand gesture recognition using a real-time tracking method and hidden Markov models," *Image and Vision Computing*, vol. 21, no. 8, pp. 745–758, Aug. 2003.
- [64] "Designing Intuitive Gesture-Based Human Interface Systems." [Online]. Available: http://www.silabs.com/Support_Documents/TechnicalDocs/Designing-Gesture-Based-Human-Interface-Systems.pdf. [Accessed: 20-Jul-2012].
- [65] M. A. Anusuya and S. K. Katti, "Speech Recognition by Machine, A Review," *CoRR*, vol. abs/1001.2, 2010.

- [66] I. Poupyrev, S. Maruyama, and J. Rekimoto, "Ambient touch: designing tactile interfaces for handheld devices," in *Proceedings of the 15th annual ACM symposium on User interface software and technology*, 2002, pp. 51–60.
- [67] E. Ammenwerth, A. Buchauer, B. Bludau, and R. Haux, "Mobile information and communication tools in the hospital," *International Journal of Medical Informatics*, vol. 57, no. 1, pp. 21–40, Jan. 2000.
- [68] S. D. Burdette, T. E. Herchline, and R. Oehler, "Practicing medicine in a technological age: using smartphones in clinical practice.," *Clinical infectious diseases : an official publication of the Infectious Diseases Society of America*, vol. 47, no. 1, pp. 117–22, Jul. 2008.
- [69] Toshiba America Information Systems, "Tablet PC: Making an Impact in Hospitals and Doctors' Offices." [Online]. Available: http://www.toshibadirect.com/content/pc/b2c/downloads/MS_HealthI_WP_FINAL.pdf. [Accessed: 15-May-2012].
- [70] D. Nicholas, P. Huntington, and P. Williams, "Three years of digital consumer health information: a longitudinal study of the touch screen health kiosk," *Information Processing & Management*, vol. 39, no. 3, pp. 479–502, May 2003.
- [71] Sony, "PlayStation®Eye." [Online]. Available: <http://pt.playstation.com/ps3/peripherals/detail/item78905/PlayStation®Eye/>. [Accessed: 02-Aug-2012].
- [72] T. Schlömer, B. Poppinga, N. Henze, and S. Boll, "Gesture recognition with a Wii controller," *Proceedings of the 2nd ...*, pp. 1–4, 2008.
- [73] J. C. Lee, "Hacking the Nintendo Wii Remote," *IEEE Pervasive Computing*, vol. 7, no. 3, pp. 39–45, Jul. 2008.
- [74] Nintendo, "Wii Official Site at Nintendo." [Online]. Available: <http://www.nintendo.com/wii>. [Accessed: 23-May-2012].
- [75] Leap Motion, "Leap." [Online]. Available: <https://leapmotion.com/>. [Accessed: 25-May-2012].
- [76] R. Baldwin, "Why the Leap Is the Best Gesture-Control System We've Ever Tested." [Online]. Available: <http://www.wired.com/gadgetlab/2012/05/why-the-leap-is-the-best-gesture-control-system-weve-ever-tested/>. [Accessed: 25-May-2012].
- [77] C. Martins, "Como funciona o Leap Motion?" [Online]. Available: <http://abertoatedemadrugada.com/2012/05/como-funciona-o-leap-motion.html>. [Accessed: 03-Oct-2012].

- [78] T. Ogasawara, "Kinect For Windows: Beyond Games And Gorilla Arms." [Online]. Available: <http://www.informationweek.com/byte/personal-tech/peripherals/kinect-for-windows-beyond-games-and-gori/232600318>. [Accessed: 17-May-2012].
- [79] R. Miles, *Start Here! Learn the Kinect API*. Microsoft Press, 2012, p. 272.
- [80] Microsoft, "Kinect for Windows SDK." [Online]. Available: <http://msdn.microsoft.com/en-us/library/hh855347.aspx>. [Accessed: 09-May-2012].
- [81] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," *Cvpr 2011*, pp. 1297–1304, Jun. 2011.
- [82] Microsoft, "Human Interface Guidelines: Kinect for Windows v1.5.0." Microsoft Corporation, 2012.
- [83] D. Catuhe, *Programming with the Kinect for Windows Software Development Kit*. Microsoft Press, 2012, p. 207.
- [84] F. T. Cerezo, "3D Hand and Finger Recognition using Kinect," Universidad de Granada.
- [85] Ministerio da Saúde, "Portal da Saúde." [Online]. Available: <http://www.portaldasaude.pt/portal>. [Accessed: 04-Sep-2012].
- [86] Municipio de Braga, "Seção Turismo." [Online]. Available: <http://www.cm-braga.pt/>. [Accessed: 07-Nov-2012].
- [87] D. Fernandez, "Camera Fundamentals," Channel9. [Online]. Available: <http://channel9.msdn.com/Series/KinectQuickstart/Camera-Fundamentals>. [Accessed: 19-Apr-2012].

Anexo A

Desenvolvimento de um projeto em C#

Para iniciar um novo projeto com as características do dispositivo *Kinect* e bibliotecas do SDK em C# é necessário numa primeira fase escolher entre as *Windows Forms* e WPF presentes no IDE *Visual Studio* (Figura A.1).

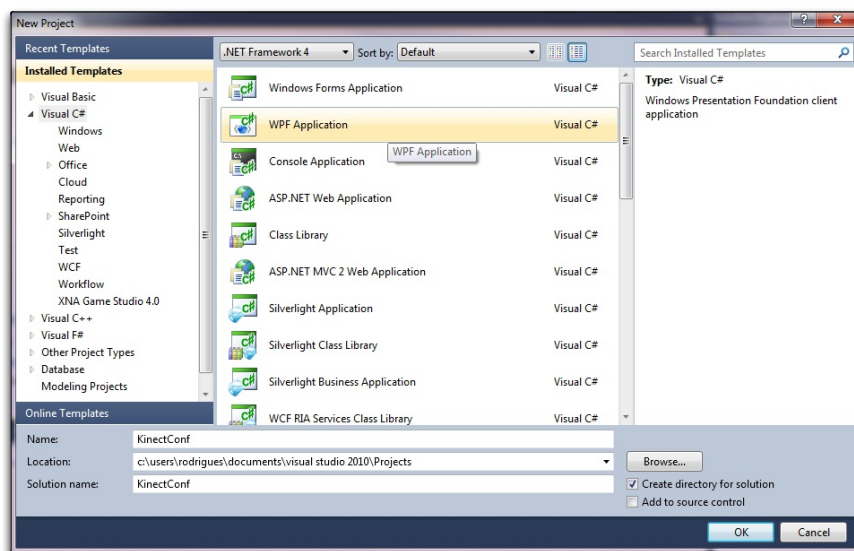


Figura A.1 – Janela de seleção dos *templates* disponíveis no *Visual Studio*.

Posteriormente é necessário adicionar as referencias das bibliotecas específicas do dispositivo *Kinect* presentes no SDK (Figura A.2)

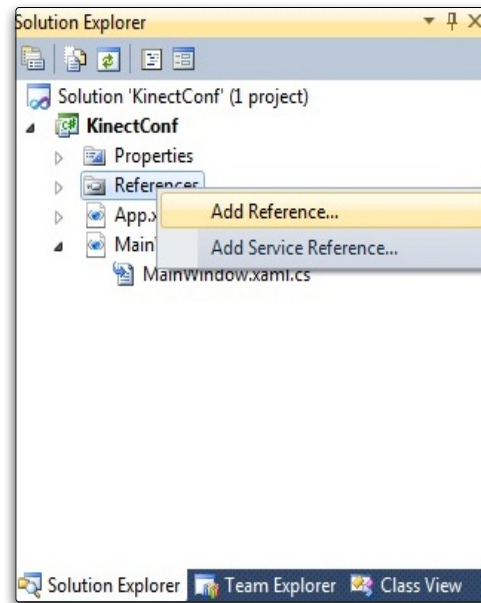
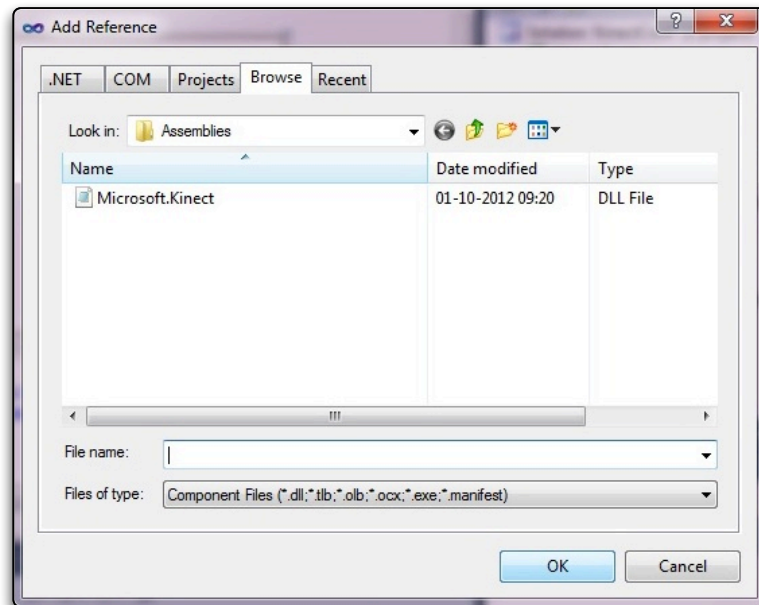
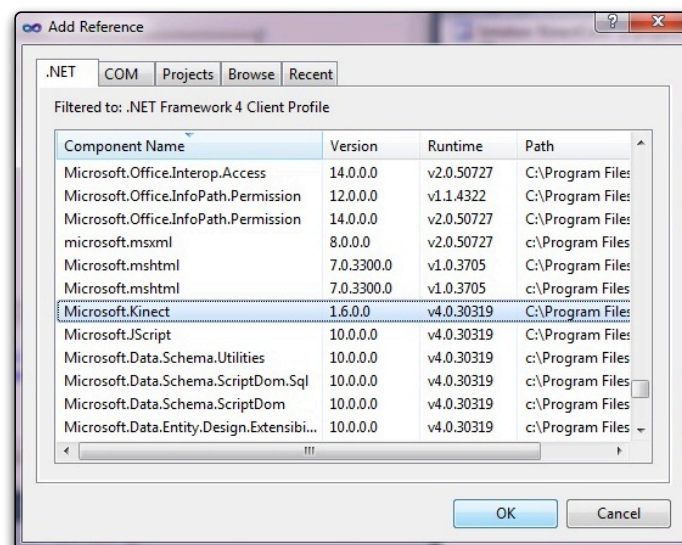


Figura A.2 – Processo de adição de referência no *Visual Studio*.

Na aba *Browse* é possível selecionar a referência através da navegação pela pasta específica da instalação SDK (Figura A.3). Esta deve-se encontrar na localização similar a:

C:\Program Files\Microsoft SDKs\Kinect\v1.6\Assemblies \Microsoft.Kinect.dll

Na aba *.Net* é possível selecionar a biblioteca relativa à *Kinect* para proceder ao desenvolvimento de aplicações (Figura A.4).

Figura A.3 - Processo da seleção da referência da *Kinect* no *Visual Studio*Figura A.4 - Processo de referência da *Kinect* no *Visual Studio* através da aba *.Net*.

Depois de ter adicionado as referências *Kinect* para o projeto, estas irão aparecer discriminadas no *Solution Explorer* do projeto (Figura A.5).

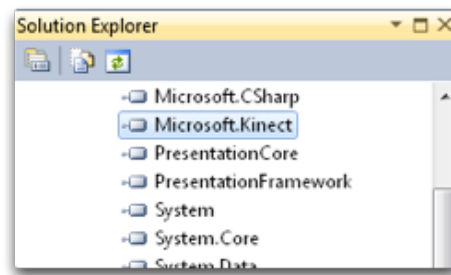


Figura A.5 – Referência citada no *Solution Explorer*.

Adicionadas as referências das bibliotecas ao projeto, é possível adicionar a instrução da *Kinect*. Assim é possível usar as classes *Kinect* no projeto. Este passo evita ao utilizador ter que se referir às classes *Kinect* pelo seu nome totalmente qualificado de cada vez que for necessário. Visto que a diretiva “*using*” foi adicionada, o compilador C# irá procurar automaticamente o *namespace* para classe.

```
using Microsoft.Kinect;
```

Adição deste elemento projeto permitem que o código possa fazer uso das classes *Kinect*. Uma vez que o projeto contém referências para as bibliotecas *Kinect* presentes apenas na máquina onde são carregadas, estas não funcionarão num sistema que não contenha o SDK instalado. Este processo deve ser seguido para criar projetos que utilizem as potencialidades do dispositivo *Kinect* [79].

Anexo B

Iniciar o Funcionamento do Dispositivo Kinect

Posteriormente à devida referenciação das bibliotecas essenciais, é necessário obter controle do dispositivo de modo a dar início ao seu funcionamento. Para interagir com o dispositivo é necessário implementar a instância da classe *KinectSensor*, responsável por controlar as conexões com a *Kinect*. O *KinectSensor* é declarado na classe *MainWindow* da aplicação. Deste modo, de cada vez que o dispositivo precisar de executar uma tarefa, utiliza esta variável.

Para iniciar as funções é necessário definir a variável *myKinect* de modo a referir a instância da classe *KinectSensor* que será usada no controle do dispositivo. Estes passos são definidos no evento *Window_Loaded*. Este evento é executado quando aplicação é iniciada, tendo associadas as funções para inicializar o dispositivo assim como quais os tipos de dados a processar através do método “*Enable*”.

Também é necessário definir qual a utilização dos dados obtidos através do dispositivo. Para isso, a aplicação utiliza o evento “*EventHandler*”, responsável por informar a variável *myKinect* sobre a utilidade dos dados obtidos.

Por último adiciona-se o método *myKinect.Start*, responsável por acionar o sensor.

```
public partial class MainWindow : Window
{
    KinectSensor kin;
    public MainWindow()
    {
        InitializeComponent();
    }
    private void Window_Loaded(object sender, RoutedEventArgs e)
    {
        // Atribui a variavel ao primeiro dispositivo Kinect
        kin = KinectSensor.KinectSensors[0];
        // Ativa o color stream para video
        kin.ColorStream.Enable(ColorImageFormat.RgbResolution640x480Fps30);
        //Ativa o sensor de profundidade
        kin.DepthStream.Enable(DepthImageFormat.Resolution640x480Fps30);
        // Ativa o modelo de esqueleto
        kin.SkeletonStream.Enable()
        // Ativa controlador dos eventos vídeo responsável por obter dados
    }
}
```

```
kin.ColorFrameReady += new EventHandler<ColorImageFrameReadyEventArgs>
    (kin_ColorFrameReady);
// Inicia o sensor
kin.Start();
}
```

O método *StopKinect*, permite por termo à utilização do sensor.

```
private void StopKinect(KinectSensor sensor)
{
    if (sensor != null)
    {
        if (sensor.IsRunning)
        {
            //Desliga o sensor
            sensor.Stop();
            if (sensor.AudioSource != null)
            {
                sensor.AudioSource.Stop();
            }
        }
    }
}
```

Necessitando apenas de invocar o método e fornecer a variável *sensor_kin* referente ao sensor [79].

```
private void Window_Closing(object sender,
System.ComponentModel.CancelEventArgs e)
{
    StopKinect(sensor_kin);
}
```

Anexo C

Kinect – Câmera C#

A primeira fase para criar aplicação com funcionalidades de uma câmera RGB passa por criar um elemento de exibição de imagem na janela WPF. Posteriormente cria-se o código que vai levar os dados de vídeo a partir do fluxo de dados *Kinect*. O método *myKinect_VideoFrameReady* atribui à aplicação os *frames* de vídeo produzidos pela câmera.

A declaração *using* identifica o bloco que irá fazer uso de uma determinada variável. Esta variável denominada de *colorFrame* é removida quando deixar de ser usada. Este valor deve ser verificado se é válido. Para isso, aplicação testa se o *frame* com que esta a atuar é genuíno. Caso o teste seja válido, deve criar um *buffer* para armazenar os dados de vídeo recebidos. Estes são fornecidos sob a forma de um grande número de valores em *bytes* num array. A classe *ColorImageFrame* fornece uma propriedade denominada de *PixelFormatLength* que contém o número de *bytes* do vídeo por cada *frame*.

De seguida cria-se um *array* de *bytes* que suporte os dados da imagem. O método *CopyPixelFormatTo* ordena à instância *ColorImageFrame* para copiar os dados dos *frames* para o *array* do *buffer* fornecido.

Posteriormente cria-se um *bitmap WPF* para estes dados. Para isso utiliza-se o método *Create* no qual atribui-se um conjunto de informação sobre o tipo de *bitmap* a ser criado, juntamente com o bloco de dados que fornece as informações sobre a imagem. É construído o *BitmapSource* que é definido como a fonte da imagem vídeo na janela WPF [79].

```
void myKinect_ColorFrameReady(object sender, ColorImageFrameReadyEventArgs e)
{
    using (ColorImageFrame colorFrame = e.OpenColorImageFrame())
    {
        if (colorFrame == null) return;
        byte[] colorData = new byte[colorFrame.PixelDataLength];
        colorFrame.CopyPixelFormatTo(colorData);
        kinectVideo.Source = BitmapSource.Create(
            colorFrame.Width, colorFrame.Height, // dimensões
            96, 96, // resolução - 96 dpi para frames de vídeo
            PixelFormats.Bgr32, // formato de vídeo
            da imagem
```

```
        null,           // palette
        colorData,     // tipos dados de video
        colorFrame.Width * colorFrame.BytesPerPixel
    );
}
}
```

Anexo D

Controlador inclinação do dispositivo C#

O controlador define o valor de inclinação do dispositivo com base no valor definido pelo utilizador. Para isso é necessário atribuir um valor para o *ElevationAngle* de acordo com o pretendido. O valores de inclinação podem variar entre $\pm 27^\circ$ (Figura D.1) [87].

```
private void button1_Click(object sender, RoutedEventArgs e)
{
    button1.IsEnabled = false;

    // Define o ângulo para o valor de Slider1
    if (kinectSensorChooser1.Kinect != null &&
    kinectSensorChooser1.Kinect.IsRunning)
    {
        kinectSensorChooser1.Kinect.ElevationAngle = (int)slider1.Value;
        lblCurrentAngle.Content = kinectSensorChooser1.Kinect.ElevationAngle;
    }

    System.Threading.Thread.Sleep(new TimeSpan(hours: 0, minutes: 0, seconds:
    1));
    button1.IsEnabled = true;
}
```

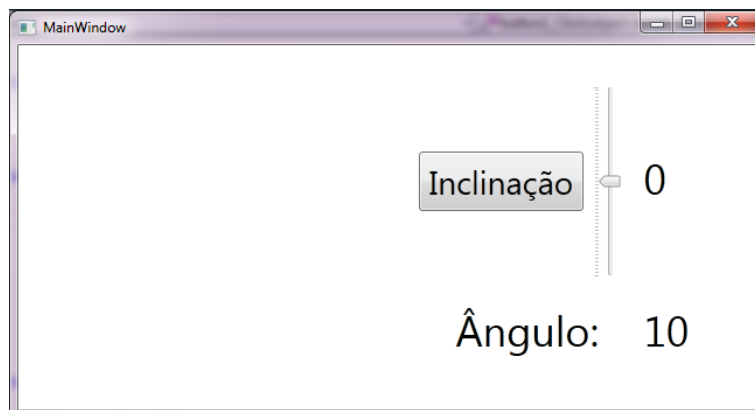


Figura D.1 – Painel de seleção do ângulo de inclinação.

Anexo E

Questionário avaliação desempenho do protótipo de interface HosNui

Questionário utilizado no âmbito do estudo do nível de aceitação do *HosNui* junto de possíveis utentes do serviço de urgência de uma instituição de saúde.

UI Gestual para Implementação em Ambientes Hospitalares

O presente questionário serve para avaliar a presente interface de utilizador(UI) de reconhecimento gestual em ambientes hospitalares. Agradecemos desde já a sua disponibilidade para participar no presente inquérito.

Classifique as seguintes afirmações sobre a UI segundo a sua opinião:

Utilize valores de 1 a 5, sendo que 1 representa 'Discordo plenamente' e 5 representa 'Concordo completamente'.

	1	2	3	4	5
1 - A informação está bem organizada de uma forma clara	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2 - Design é agradável	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3 - A tarefa que está a ser executada é facilmente identificada	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4 - A linguagem utilizada é simples e adequada	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5 - A interface está estruturado de uma forma simples e clara	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
6 - A consulta de dados é rápida	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7 - A interação é natural e intuitiva	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
8 - Os gestos são adequados	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
9 - Reconhecimento de gestos é preciso	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
10 - Comandos de voz são adequados	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

11 - Reconhecimento de voz é preciso	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
12 - Interação esta bem repartida entre modo de reconhecimento de voz e gestual	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
13 - Interface transmite feedback necessário	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
14 - Interação é agradavel	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
15 - A interface é util	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
16 - No geral, está satisfeito com a interface	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Powered by [Google Docs](#)[Report Abuse](#) - [Terms of Service](#) - [Additional Terms](#)