IEEE TRANSACTIONS ON
EMERGING TOPICS
IN COMPUTING

Received 15 October 2019; revised 5 April 2020; accepted 30 May 2020.
Date of publication 5 June 2020; date of current version 16 September 2021.

Digital Object Identifier 10.1109/TETC.2020.3000352

Virtual Reality Simulation of a Quadrotor to Monitor Dependent People at Home

LIDIA M. BELMONTE, ARTURO S. GARCÍA, EVA SEGURA, PAULO NOVAIS, (Member, IEEE), RAFAEL MORALES, AND ANTONIO FERNÁNDEZ-CABALLERO

L.M. Belmonte, E. Segura, and R. Morales are with the Departamento de Ingeniería Eléctrica, Electrónica, Automática y Comunicaciones, Universidad de Castilla-La Mancha, 02071 Albacete, Spain, and also with the Instituto de Investigación en Informática de Albacete, 02071 Albacete, Spain
A.S. García is with the Departamento de Sistemas Informáticos, Universidad de Castilla-La Mancha, 02071, Albacete Spain, and also with the Instituto de Investigación en Informática de Albacete, 02071 Albacete, Spain
P. Novais is with the Departamento de Informática, Universidade do Minho, Campus de Gualtar4710-057 Braga, Portugal
A. Fernández-Caballero is with the Departamento de Sistemas Informáticos, Universidad de Castilla-La Mancha, 02071 Albacete, Spain, and the Instituto de Investigación en Informática de Albacete, 02071 Albacete, Spain, and CIBERSAM (Biomedical Research Networking Centre in Mental Health), 28029 Madrid, Spain
CORRESPONDING AUTHOR: ANTONIO FERNÁNDEZ-CABALLERO (antonio.fdez@uclm.es)

ABSTRACT Unmanned aerial vehicles (UAVs) represent an assistance solution for home care of dependent persons. These aircraft can cover the home, accompany the person, and position themselves to take photographs that can be analyzed to determine the person's mood and the assistance needed. In this context, this work principally aims to design a tool to aid in the development and validation of the navigation algorithms of an autonomous vision-based UAV for monitoring dependent people. For that, a distributed architecture has been proposed based on the real-time communication of two modules, one of them in charge of the dynamics of the UAV, the trajectory planning and the control algorithms, and the other devoted to visualizing the simulation in an immersive virtual environment. Thus, a system has been developed that allows the evaluation of the behavior of the assistant UAV from a technological point of view, as well as to carry out studies from the assisted person's viewpoint. An initial validation of a quadrotor model monitoring a virtual character demonstrates the advantages of the proposed system, which is an effective, safe and adaptable tool for the development of vision-based UAVs to help dependents at home.

INDEX TERMS K.4.2.b assistive technologies for persons with disabilities, L.2.0.k virtual reality, I.2.9.a autonomous vehicles

I. INTRODUCTION

Unmanned aerial vehicles (UAVs)are promising candidates for several vital applications. In this sense, public safety and emergency communications could be enhanced using UAVs as mobile stations that are rapidly deployed to improve the network coverage after a natural disaster or malevolent attacks in which the communication infrastructures are damaged and dysfunctional [1]. In such a situation where the Internet would not be available, UAVs could be used as a communication link to assist disaster management with possible victims by means of smartphone apps connected to the new networks to allow communications [2].

Similarly, cooperation between multiple UAVs exchanging data with each other in the air can be a key support for carrying out various types of missions, while providing the necessary assistance to ground networks [3]. Improving communications in vehicular ad hoc networks where critical road information could be shared [4] or monitoring roads to detect incidents and share information to alert and guide emergency vehicles using a global view of the road are some of the future applications of UAV networks [5].

The above are mainly aimed at reducing the response time in emergency situations, helping to limit material damage or even save human lives. However, UAVs can also be used to aid people to improve their quality of life on a regular basis. Among others, drones could enhance attendance in rural areas for people with chronic diseases by delivering and collecting medical supplies, while reducing travel and patient care costs [6]. Regarding home care, UAVs represent a future model of personal robots to assist dependent persons like elderly people living alone. Unlike solutions designed for nursing homes that require a specific network infrastructure, where patients use a physical device to monitor their health status, UAVs can contribute to monitor dependents and enable them to continue living in the comfort of their homes.

Indeed, a UAV equipped with an on-board camera allows, unlike static vision systems or other ground robots, overcoming possible barriers at home, such as stairs, as well as accessing to remote points, avoiding dead angles, positioning itself in front of the person, and accompanying the person in real time [7], [8]. In this manner, it would be possible to take snapshots that are processed with computer vision techniques to determine the person's state (both physical and emotional). Thus, by non-invasive methods such as automatic recognition of emotions [9], [10], [11], which studies speech and/or facial expressions, among others, it is plausible to analyze the person's mood as an initial step towards assisting him/her at each situation or condition.

Our current research is focused on the development of such UAVs based on computer vision for assisting care-dependent persons at home [12]. This is a fairly new research field. In this regard, we can mention a proposal of a customized quadcopter to supervise patients indoors using images, sonar and voice recognition strategies [13]. At a very early stage of development, this work compiles several existing algorithms for some of the technical problems that need to be addressed to achieve fully autonomous navigation, but it does not develop any solution beyond the description of the hardware selected for the air vehicle.

We believe that the development of an autonomous vision-based UAV to assist dependents entails multiple challenges, both of a more technical nature and others related to human factors [14]. In the first group, the need of a trajectory planner to guide the motion of the UAV during the monitoring process must be highlighted. Here, the ability of autonomous navigation is primordial. In the second group, the acceptance of the technology by the dependent person and his/her family is essential for these new assistant robots to be viable. At this point, a trust relationship between the assistant UAV and the assisted person should be encouraged. Similarly, it is necessary to respect the personal space of the person when planning routes during monitoring and to hinder as little as possible the usual routine of the person at home.

The technical and human factors, which have been briefly summarized, lead to focus on these two aspects during the development phases of the assistant UAVs. This way, it is necessary to test the different engineering solutions, but also people must be given the opportunity to assess this type of assistance system before bringing it to real homes. In order to achieve this objective, we rely on the use of virtual reality technology, as it allows testing in a controlled environment prior to building any physical device. Therefore, this paper presents a framework based on the integration of precise numerical simulations of the dynamics of the UAV, including the control algorithm and the path planner, complemented with a virtual reality simulation of the scenarios in which the target users will face the technology.

Virtual reality (VR), despite several adverse symptoms suffered by some people [15], has traditionally been used for simulating environments due to their capacity to reduce time and cost [16], eliminate risk for people and equipment [17], [18], training [19], etc. This is specially true for robotics [20], [21], since it allows rapid prototyping and algorithm development, and system verification, among other things. The rationale behind the use of VR in this work is twofold. On the one hand, we wanted to use the virtual environment as a test bed for our autonomous navigation algorithms, reducing the risk of crashing the UAV or hurting the participants. On the other hand, it is possible to carry out user-centered studies to assess the acceptance of the UAV. By using immersive VR technology, the assisted person experiences the feeling of having a UAV flying around while, for example, they are required to perform common activities of daily living in a 3D environment that recreates a house. This approach enables carrying out studies and analyses centered on both the assisted person and the assistant robot. The conclusions drawn from the tests will allow modifications and improvements, mainly related to the physical characteristics of the UAV and the design parameters of the flight planner.

The contributions of this work are (i) the implementation of a development and validation tool based on the real-time communication of a UAV simulator and a VR visualizer to allow evaluations both from the technical and human points of view. This will lay the foundation of our research on fully autonomous vision-based UAVs to assist dependent persons in real homes; (ii) a detailed description of the current implementation of the trajectory planning algorithm for the flight of a quadrotor equipped with a camera to monitor a person, which was briefly introduced in a previous work [22], and which has been extended to consider turns of the head to improve the capture of facial images; (iii) new simulation results using the VR tool developed to evaluate the performance of the new trajectory planner and the control algorithm for the monitoring process as well as to verify the proper communication of the two modules which compose the platform; (iv) a preliminary study on the acceptance of the assistant UAV by several participants using the developed tool and the immersive VR technology.

The remainder of the paper is as follows. Section II describes the architecture proposed for the VR validation of the assistant UAV, and introduces the two next modules. The UAV Simulator is presented in Section III, which contains the dynamic model of the UAV, the control algorithm and the trajectory planner. On the other hand, the VR Visualizer is depicted in Section IV. Next, the communication between these two modules is described in Section V. The simulation results of the UAV monitoring a virtual person by using the VR validation tool are presented in Section VI, while the acceptance evaluation of the assistant UAV is detailed in VII. Finally, Section VIII concerns the conclusions of the work and future research.

II. VIRTUAL REALITY VALIDATION OF THE UAV

In order to achieve our objective of providing a VR platform for the validation of assistant UAVs in a home environment,

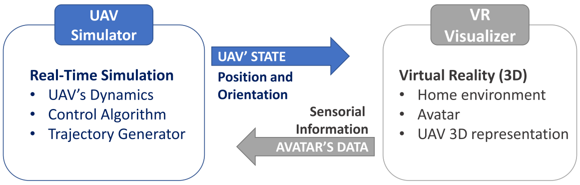

FIGURE 1. High-level view of the system proposed.

a system is proposed based on the conceptual distributed architecture depicted in Figure 1. The main aim is to decouple the UAV simulation from the visualization, since these two processes require significant computing resources [21]. Moreover, such an approach opens the door to future configurations where several UAVs and users may connect to the same simulation.

Figure 1 shows the two main components, an *UAV Simulator* and a *VR Visualizer* that exchange information between them. The UAV Simulator is in charge of controlling the behavior of the UAV, simulating its in-flight dynamics and creating trajectories for its movement inside the virtual house. On the other hand, the VR Visualizer renders the environment and the UAV for the target user to either test the behavior of the UAV or to assess their experience with it. Therefore, both modules produce information that is needed for each other. This exchange requires a fast and reliable transportation of the information between the components involved in the simulation.

Therefore, the main information needed for the exchange includes the position and orientation of the UAV inside the virtual environment at each particular time, which is provided by the UAV Simulator. At the same time, this module use the real-time information provided by the VR Visualizer regarding the position and orientation of the user inside the virtual environment in order to adjust the trajectories. Similarly, the VR Visualizer uses the information transmitted by the UAV Simulator to render the aircraft within the virtual environment at a location and with an orientation according to the physical simulation received.

III. UAV SIMULATOR

This section describes the actual implementation of the UAV Simulator module as described in Figure 1. The aim is to reproduce the flight of the UAV during the monitoring process in a precise way, considering its dynamics, the action of the controller and the trajectory planner. In the end, this determines the reference trajectories for the position and orientation variables of the UAV to achieve the desired movement of the aircraft for monitoring the person. For this purpose, the *MATLAB/Simulink* environment was selected for the following reasons: (i) it allows high performance real-time simulations of dynamic systems using the Simulink Desktop Real-Time tool [23]; (ii) it is possible to measure and graphically represent the system's states, control inputs, and references, allowing different algorithms to be evaluated and compared; (iii) it has been used to test robust control algorithms designed by our research group for different UAV models and platforms, such as the TRMS laboratory helicopter [24], [25]; and, (iv) it facilitates the integration with the VR visualization module by means of MQTT in MATLAB [26] (which implements the Message Queue Telemetry Protocol). However, and despite these advantages, at this point, we should mention that the use of other environments, such as Robot Operating System (ROS) and Gazebo which has also been used for simulating UAVs [27], [28], [29], is not ruled out for the final implementation of our system.

A complete diagram of the UAV Simulator module is represented in Figure 2. It is composed of the dynamic model of the rotatory-wing UAV selected (a quadrotor), the generalized proportional integral (GPI) controller [30], and the trajectory generator. As can be observed, the trajectory planner calculates the smooth reference trajectories for the position and yaw angle of the quadrotor. These references are used by the GPI algorithm to modify the control inputs to guide the flight of the air vehicle. During the flight, the main objective of the UAV is to monitor the dependent person, i.e., to position itself correctly to capture snapshots of the person with the camera on board. The images are sent to a base station to determine the assistance required in each moment after analyzing the person's mood.

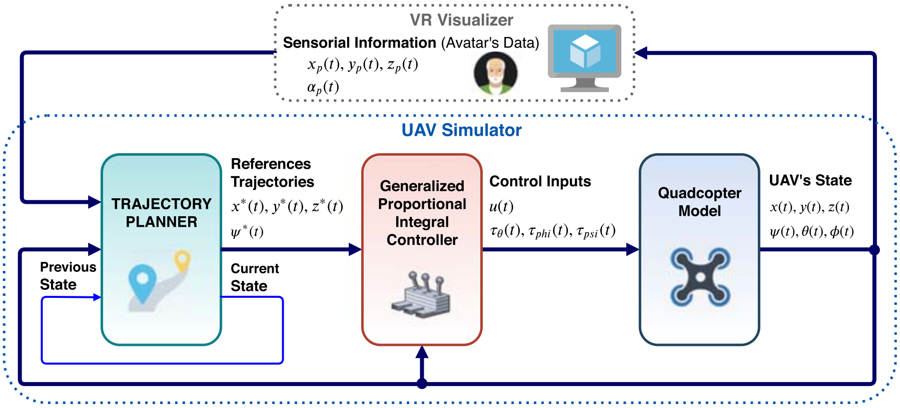

FIGURE 2. General control scheme.

Next, we describe the dynamics of the UAV, the control algorithm and the states that form the trajectory planning.

A. THE DYNAMICS OF THE QUADROTOR

A quadrotor is a rotatory-wing UAV formed by four rotors arranged in the shape of a cross and equidistant from the center of mass of the aircraft [30]. Such a vehicle allows vertical take-off and landing, and is characterized by a high maneuverability, agility, and versatility. In addition, it can move at low speed, reducing the risk of collision in flight, and improving the quality of the image recorded by a camera aboard. For all these reasons, it has been considered suitable for the proposed approach. The quadrotor's thrust is generated by the four fixed-angle propellers of the rotors. The lift forces are modified by changing the propellers rotation speed, thus achieving the three possible movements, namely, pitch, roll, and yaw. The system of equations that model its dynamic behavior is obtained through the Euler-Lagrange approach [31], resulting in:

$$\begin{aligned} m\ddot{x} &= -u\sin\theta & \ddot{\psi} &= \tau_{\psi} \\ m\ddot{y} &= u\cos\theta\sin\phi & \ddot{\theta} &= \tau_{\theta} \\ m\ddot{z} &= u\cos\theta\cos\phi - mg & \ddot{\phi} &= \tau_{\phi}, \end{aligned}$$

where m is the mass, g is the gravity acceleration, x and y are coordinates in the horizontal plane, z is the vertical position, the angles ϕ, θ and ψ express the independent orientation angles, u is defined as the total thrust and τ_{ψ}, τ_{θ} and τ_{ϕ} denote the angular moments (yawing moment, pitching moment and rolling moment, respectively). Moreover, the following assumption has been considered: orientation angles θ and ϕ are upper and lower bounded in intervals $-\frac{\pi}{2} < \phi < \frac{\pi}{2}$ and $-\frac{\pi}{2} < \theta < \frac{\pi}{2}$.

B. THE CONTROL

A control scheme is mandatory to regulate track the trajectory generated by the planner in order to perform a precise flight that monitors a dependent. For this, a GPI controller, based on the theory of differential flatness, has been selected, which has demonstrated good performance in the control of nonlinear systems. GPI control sidesteps the need for traditional asymptotic state observers and directly utilizes structural state estimates in place of state variables [32]. The impact of non-modeled dynamics is suitably compensated in the feedback control law by means of integrals of the output tracking errors.

The complete design of the GPI controller was designed in a previous work of our research group [30], and the results of that research demonstrated the effectiveness of the approach in comparison with the classical PID control in the following terms: (a) stabilization and trajectory tracking tasks; (b) performance when the measured signals are corrupted by noise; and (c) dynamic response when atmospheric disturbances trouble the quadrotor.

C. THE PLANNING OF THE TRAJECTORY

Trajectory planning problematic has attracted the interest of the research community in recent years [33], [34], [35], [36], [37] as it is one of the main technical aspects to sort out for autonomous navigation of mobile robots and the control of mechatronic systems. In this sense, this section explains the trajectory planner implemented for the quadrotor. The objective of trajectory generation is to position the UAV with an on-board camera in order to capture facial images of the care-dependent person. Thus, the planner calculates the flight path for the quadrotor so that it performs the monitoring process.

In this regard, an initial algorithm was recently introduced [22], which is now explained in detail. In addition, the new version of the trajectory generator improves the capture data state, so that when it detects that a person turns his/her head (thus remaining out of sight of the on-board camera), the UAV corrects its position (which means that it moves in the most convenient direction around the person to look for his/her face again). Also, as a novelty with respect to the previous work, it should be noted that now the trajectory planning algorithm is tested on the designed VR platform. The planner receives the data of the person in real-time, while the initial planner was only tested by means of numerical simulations in which the position and orientation of the person was fixed before running the simulation.

The following assumptions have been considered during the development of the trajectory planner: (i) the sensory information, in this case the VR Visualizer, provides the position of the person (defined by the coordinates of his/her head) and the face's direction (used to determine when the UAV is in front of the person's face and detect if he/she turns the head during the facial images capture); (ii) the dependent person does not walk during the monitoring flight; (iii) there are no obstacles in the UAV's working space; (iv) a safety radius is defined around the person to avoid unexpected collisions with the person. Energy limitations are not considered at this time since the UAV is located at its base until the monitoring process begins and returns when it ends. Therefore, it is assumed that the energy level is sufficient to complete the process in this short period of time.

As can be observed in the general control scheme of Figure 2, under the assumption that the person's position $(x_p(t), y_p(t), z_p(t))$ and direction (α_p) are known, the planner defines the reference trajectories for the position coordinates $(x^*(t), y^*(t), z^*(t))$ and yaw angle $(\psi^*(t))$ of the UAV. These references are used by the GPI control law to create the necessary control actions in order to both achieve the stabilization of the system and the tracking of the references and, consequently, to govern the motion of the UAV to monitor the care-dependent person.

The overall description of the monitoring process is the following. The UAV takes off up to the height defined by the person's face. Then, the UAV approaches the person until a safety position (defined by a safety radius R). Next, a circular movement is performed around the person in order to localize his/her face, stopping for a while to emulate the capture of facial images. In addition, and as a novelty in relation to the previous work [22], the UAV adapts its flight to correct its position when changes in the person's face direction are detected until it completes the facial images' capture. After that, the UAV completes the circular motion and returns to its base. Therefore, the monitoring flight is composed of a series of maneuvers. To generate the reference trajectories for each of them, the trajectory planner is implemented as a state machine.

The states that make up the planner are described next in detail. Considering the person's position and direction, the UAV's output variables and the previous state, the trajectory planner calculates the reference signals for the quadrotor's position (x, y, z) and yaw angle (ψ). In this case, a total of twelve states has been defined so that the UAV monitors the dependent person. An illustration of the states is represented in Figure 3. It should be noted that: (i) the planner defines the references so that the UAV camera's focus points towards the UAV's forward direction or the person; (ii) the UAV's initial state is defined by the base's position (x_b, y_b, z_b) and the initial yaw angle $(\psi(0))$; (iii) the configurable parameters of the trajectory planner are detailed in Table 1. Below, the calculation of the reference trajectories for each state is detailed, using the following notation:

- t_{in}: initial time of state n
- t_{if}: final time of state n
- $x_{in} = x(t_{in})$: initial value of x coordinate at the beginning of state n (at instant t_{in})
- $y_{in} = y(t_{in})$: initial value of y coordinate at the beginning of state y (at instant t_{in})
- $z_{in} = z(t_{in})$: initial value of z coordinate at the beginning of state n (at instant t_{in})
- $\psi_{in} = \psi(t_{in})$: initial value of ψ angle at the beginning of state n (at instant t_{in})

State 0: Home. This is the initial state of the trajectory planner. As can be observed in Figure 3(a), the UAV is located on its base $((x_{i0}, y_{i0}, z_{i0}) = (x_b, y_b, z_b))$, with an initial orientation $(\psi_{i0} = \psi(0))$, waiting for instructions. This way, when the UAV receives the instruction to start the monitoring process, the planner transits to state 1.

State 1: Takeoff. Before searching the person, the UAV must take off until reaching the altitude level defined by the person's height (coordinate z_p). For this takeoff maneuver, the reference path of the z coordinate is calculated as a ramp function whose slope is the configurable speed parameter v_z, and the initial and final positions are the altitude of the base (z_b) and person (z_p), respectively. For its part, the UAV's position at horizontal plane (x, y) and its yaw angle (ψ) are maintained during the takeoff (see Figure 3(a)). This way, the reference trajectories for state 1 are defined as follows:

$$\begin{aligned} &x_1^*(t) = x_{i1}; \qquad y_1^*(t) = y_{i1} \\ &z_1^*(t) = z_b + \left(\tfrac{t-t_{i1}}{t_{f1}-t_{i1}}\right) \cdot (z_p - z_b) \qquad \psi_1^*(t) \end{aligned} \tag{1}$$

where $t_{f1} = t_{i1} + \left|\frac{z_p - z_b}{v_z}\right|$ is the time in which state 1 ends. At that moment, when the UAV reaches the position (x_b, y_b, z_p), the planner transits to state 2.

State 2: Person Search. In this maneuver, the UAV is requested to modify its yaw angle maintaining the 3D position in the space. The objective is that the on-board camera's focus points towards the person (see Figure 3(a)). Since the position of the person is known, it is possible to determine the angle formed by the imaginary line (in the horizontal plane) joining the UAV's center with the person face's center, $\alpha = arctan(\frac{y_p - y_{i2}}{x_p - x_{i2}})$. Thus, the UAV's final yaw angle (ψ_{f2}) should be the difference between α and the camera's angle (α_{camera}) so that the on-board camera focuses on the person. For such adjustment in the yaw angle, a ramp function is defined whose final time is determined by the state's start time, the initial and final angular positions, and the ω_ψ velocity parameter, $t_{f2} = t_{i2} + \left|\frac{(\alpha - \alpha_{camera}) - \psi_{i2}}{\omega_\psi}\right|$.

This way, the reference trajectories for state 2 results as follows:

$$\begin{aligned} &x_2^*(t) = x_{i2}; \qquad y_2^*(t) = y_{i2}; \qquad z_2^*(t) = z_{i2} \\ &\psi_2^*(t) = \psi_{i2} + \left(\tfrac{t-t_{i2}}{t_{f2}-t_{i2}}\right) \cdot (\psi_{f2} - \psi_{i2}). \end{aligned} \tag{2}$$

When the camera's center is aligned with the person, that is, when the UAV's yaw angle reaches an angular position equal to $\alpha - \alpha_{camera}$, the trajectory planner transits to state 3.

State 3: Approximation. Once the camera's focus points towards the person, the next step is the approach maneuver. Maintaining the flight height, the UAV must reach the *safety position* in the circle of radius R around the person. The safety position's coordinates in the horizontal plane $z = z_p$ are determined from the initial distance between the UAV's position and the person's position, $d = \sqrt{(x_p - x_{i3})^2 + (y_p - y_{i3})^2}$, as follows:

$$\begin{aligned} x_{sp} &= x_{i3} + (x_p - x_{i3}) \cdot \tfrac{d-R}{d} \\ y_{sp} &= y_{i3} + (y_p - y_{i3}) \cdot \tfrac{d-R}{d}. \end{aligned}$$

The UAV must, therefore, make a movement in a straight line towards the person to travel the diagonal $h = d - R$ (see Figure 3(b)). For this purpose, the reference paths for the horizontal position of the UAV $(x^*(t), y^*(t))$ are defined as ramp functions whose slopes are given by the x and y components of the velocity v_d (configurable parameter), while the altitude coordinate and yaw angle are constants in state 3:

$$\begin{aligned} &x_3^*(t) = x_{i3} + v_d \cdot \tfrac{(x_p - x_{i3})}{d} \cdot (t - t_{i3}) \\ &y_3^*(t) = y_{i3} + v_d \cdot \tfrac{(y_p - y_{i3})}{d} \cdot (t - t_{i3}) \\ &z_3^*(t) = z_{i3}; \qquad \psi_3^*(t) = \psi_{i3}. \end{aligned} \tag{3}$$

When the UAV reaches safety position (x_{sp}, y_{sp}, z_p), the planner transits to state 4.

State 4: Waiting in Safety Position. Before starting the circular motion around the person, the UAV remains static during a brief period of time in state 4 (see Figure 3(c)). This way, the reference trajectories for the position and yaw angle

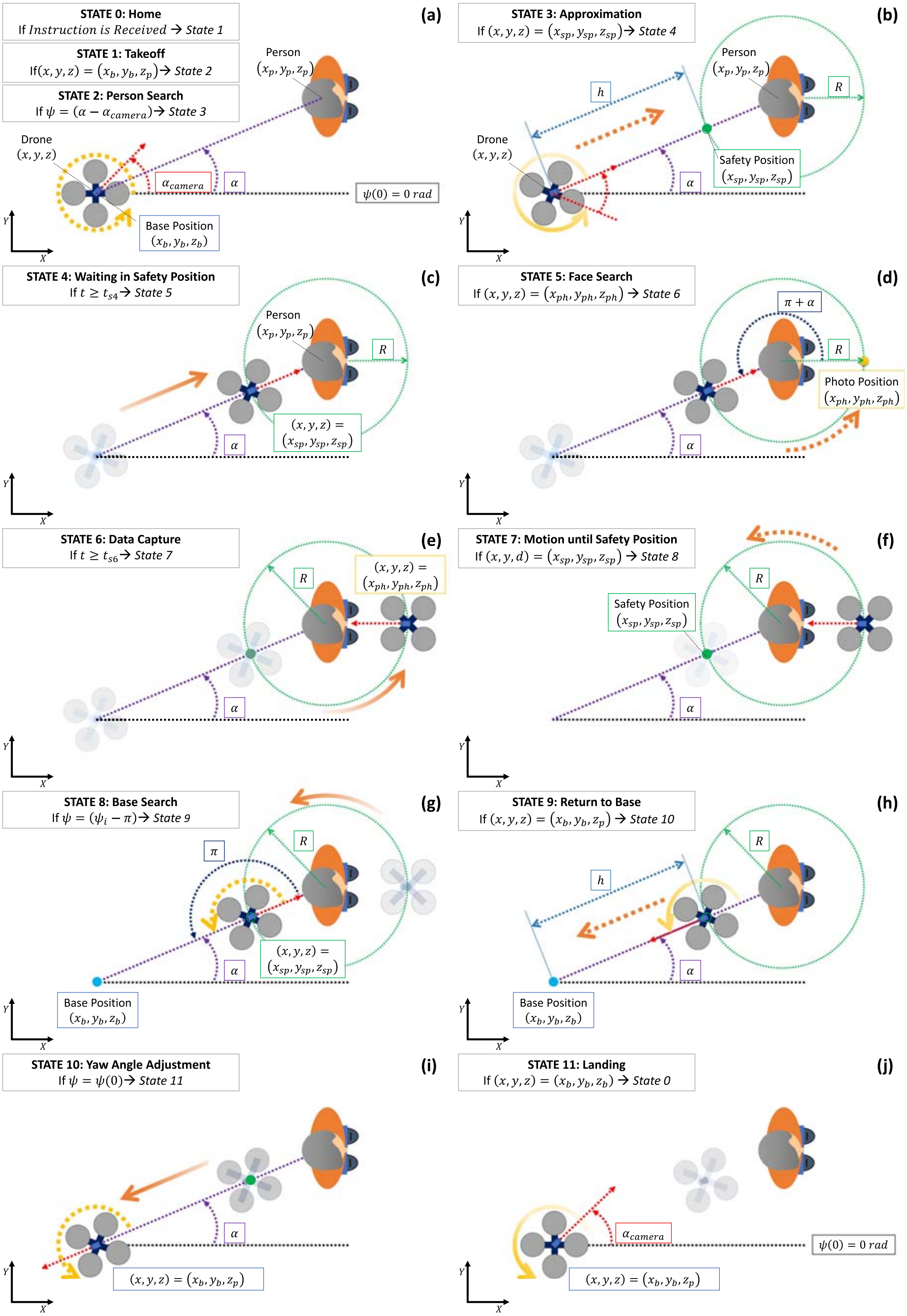

FIGURE 3. Graphical representation of the trajectory planner's states.

TABLE 1. **Parameters defined in the UAV simulator (MATLAB/Simulink environment).**

DESCRIPTION	VALUE
UAV's Parameters	
Initial Position (Base Position)	$(x_b, y_b, z_b) = (0,0,0)$ [m]
Initial Yaw Angle	$\psi(0) = 0$ [rad]
Camera's Angle	$\alpha_{camera} = \pi/4$ [rad]
Mass	$m = 1$ [Kg]
Planner's Parameters	
Safety Radius	$R = 2$ [m]
Velocity in Z axis	$v_z = 8.16 \cdot 10^{-2}$ [m/s] (takeoff); $v_z = 4.08 \cdot 10^{-2}$ [m/s] (landing)
Velocity in Diagonal Motion at Horizontal Plane (x, y)	$v_d = 0.1$ [m/s] (state 3) ; $v_d = 0.2$ [m/s] (state 9)
Angular Velocity for Yaw Adjustment	$\omega_\psi = 3 \cdot \pi/50$ [rad/s]
Angular Velocity for Circular Motion	$\omega_{circle} = 9 \cdot \pi/250$ [rad/s]
Period of time for State 4 (Waiting in Safety Position)	$t_{s4} = 5$ [s]
Period of time for State 6 (Data Capture)	$t_{s6} = 15$ [s]
Simulation Parameters	
Sample Time	$Ts = 0.01$ [s]
Simulation Time	$t = 250$ [s]

are constants equal to the initial value of these parameters at the beginning of the state.

$$x_4^*(t) = x_{i4}; \quad y_4^*(t) = y_{i4}; \quad z_4^*(t) = z_{i4}; \quad psi_4^*(t) = \psi_{i4}. \tag{4}$$

When the programmed timeout (t_{s4}) elapses, the trajectory planner transits to state 5.

State 5: Face Search. In the fifth state (see Figure 3(d)), the UAV is requested to make a circular motion around the person while varying the yaw angle, so that the on-board camera continues to point to the person. For its part, the flight altitude is maintained constant. This way, the reference trajectory for the UAV's horizontal position (coordinates x and y) are calculated as a uniform circular motion of radius R from the initial angular position defined by $\beta_5 = \psi_{i5} + \pi + \alpha_{camera} = \pi + \alpha$, where α is the angle defined above in state 2. At the same time, the yaw angle is gradually modified according to the increment in the angular position as a result of the circular motion.

Therefore, the definition of the reference trajectories for state 5 is as follows:

$$\begin{aligned} x_5^*(t) &= x_p + R \cdot cos(\omega_c(t - t_{i5}) + \beta_5) \\ y_5^*(t) &= y_p + R \cdot sin(\omega_c(t - t_{i5}) + \beta_5) \\ z_5^*(t) &= z_{i5}; \qquad \psi_5^*(t) = \psi_{i5} + \omega_c(t - t_{i5}). \end{aligned} \tag{5}$$

When the UAV is in front of the person's face, that is, when the UAV reaches the photo position, the trajectory planner transits to state 6.

State 6: Data Capture. Once the UAV is positioned in front of the person's face, in position (x_{ph}, y_{ph}, z_{ph}), it is necessary to maintain it to capture facial images (see Figure 3 (e)). The images are sent to a base station responsible for analyzing facial expressions in order to determine the person's mood. Thus, and in a similar way to state 4, the reference trajectories for the UAV's position and yaw angle are constants equal to the initial value of these parameters at the beginning of the state.

$$x_6^*(t) = x_{i6}; \quad y_6^*(t) = y_{i6}; \quad z_6^*(t) = z_{i6}; \quad psi_6^*(t) = \psi_{i6}. \tag{6}$$

When data capture time t_{s6} elapses without detecting significant changes in the person's head direction (i.e., the person's face remains in the field of view of the UAV's camera), the data capture process is considered completed and the trajectory planner transits to state 7. Conversely, when the person turns his/her head before the capture is completed, the UAV must again turn the person over to look for his/her face. To do this, the trajectory planner returns to state 5 (and adapts the sign of the angular velocity so that the UAV navigates around the person in the most convenient direction according to the head's spin).

State 7: Motion to Safety Position. After data capture, the UAV must continue the circular motion around the person and reach again the safety position (x_{sp}, y_{sp}, z_{sp}) (see Figure 3 (f)). Thereby, and in a similar way to state 5, the reference trajectories for the UAV's horizontal position (coordinates x and y) are calculated as a uniform circular motion of radius R. Nevertheless, in this case, the initial angular position is defined by $\beta_7 = \psi_{i7} + \pi + \alpha_{camera}$. For its part, the altitude is maintained constant while the yaw angle is incremented again, so that the UAV's camera still focuses on the person. Hence, the reference paths in state 7 for the UAV's position and yaw movement are the following:

$$\begin{aligned} x_7^*(t) &= x_p + R \cdot cos(\omega(t - t_{i7}) + \beta_7) \\ y_7^*(t) &= y_p + R \cdot sin(\omega(t - t_{i7}) + \beta_7) \\ z_7^*(t) &= z_{i7}; \qquad \psi_7^*(t) = \psi_{i7} + \omega(t - t_{i7}). \end{aligned} \tag{7}$$

When the UAV reaches safety position (x_{sp}, y_{sp}, z_{sp}), the planner transits to state 8.

State 8: Base Search. This state aims to rotate the UAV so that its camera's focus points towards the base (see Figure 3 (g)). Since the UAV is located at the safety position, the turn ratio to adjust the yaw angle is π[rad]. The 3D position is maintained during the maneuver. This way, the references trajectories for state 8 are defined as follows:

$$\begin{aligned} x_8^*(t) = x_{i8}; \qquad y_8^*(t) = y_{i8}; \qquad z_8^*(t) = z_{i8} \\ \psi_8^*(t) = \psi_{i8} + \left(\tfrac{t-t_{i8}}{t_{f8}-t_{i8}}\right) \cdot ((\psi_{i8} - \pi) - \psi_{i8}), \end{aligned} \tag{8}$$

where $t_{f8} = t_{i8} + \left|\frac{(\psi_{i8}-\pi)-\psi_{i8}}{\omega_\psi}\right|$ is the time in which state 8 ends. At that moment, when the camera focuses towards the base (the UAV's yaw angle has already turned -π [rad]), the planner transits to state 9.

State 9: Return to Base. Similarly to state 3, but in opposite direction, the UAV travels the distance $h = \sqrt{(x_b - x_{i9})^2 + (y_b - y_{i9})^2}$, maintaining the flight altitude, to position itself over the base (see Figure 3(h)). Hence, the reference paths are defined in a similar way. $x^*(t)$, $y^*(t)$ are ramp functions whose slopes are given by the x and y components of velocity v_d, while $z^*(t)$ and $\psi^*(t)$ are constants:

$$\begin{aligned} x_9^*(t) &= x_{i9} + v_d \cdot \tfrac{(x_b - x_{i9})}{h} \cdot (t - t_{i9}) \\ y_9^*(t) &= y_{i9} + v_d \cdot \tfrac{(y_b - y_{i9})}{h} \cdot (t - t_{i9}) \\ z_9^*(t) &= z_{i9} \qquad \psi_9^*(t) = \psi_{i9}. \end{aligned} \tag{9}$$

When the UAV reaches the position over the base (x_b, y_b, z_p), the planner transits to state 10.

State 10: Yaw Angle Adjustment. Before landing, the UAV is requested to modify its yaw angle so that it can position on the base correctly and be ready for the next monitoring process (see Figure 3(i)). Thereby, maintaining the position, the UAV has to modify the yaw angle to its initial value, $\psi(0)$:

$$\begin{aligned} x_{10}^*(t) = x_{i10}; \qquad y_{10}^*(t) = y_{i10}; \qquad z_{10}^*(t) = z_{i10} \\ \psi_{10}^*(t) = \psi_{i10} + \left(\tfrac{t-t_{i10}}{t_{f10}-t_{i10}}\right) \cdot (\psi(0) - \psi_{i10}), \end{aligned} \tag{10}$$

where $t_{f10} = t_{i10} + \left|\frac{\psi(0)-\psi_{i10}}{\omega_\psi}\right|$ is the time in which the state ends and the planner transits to state 11.

State 11: Landing. The last maneuver is to land the UAV (see Figure 3(j)). For this purpose, a reference path for the altitude coordinate is necessary, while the horizontal position and yaw angle are constants. Thereby, the reference trajectories are similar to state 1, but with the difference of the initial and final positions of the UAV. In this case, the altitude is modified from the person face's height (z_p) to the level of the base (z_b).

$$\begin{aligned} & x_{11}^*(t) = x_{i11}; \qquad y_{11}^*(t) = y_{i11}; \\ & z_{11}^*(t) = z_p + \left(\tfrac{t-t_{i11}}{t_{f11}-t_{i11}}\right) \cdot (z_b - z_p); \qquad \psi_{11}^*(t) = \psi_{i11}. \end{aligned} \tag{11}$$

where $t_{f11} = t_{i11} + \left|\frac{z_b - z_p}{v_z}\right|$ is the time in which state 11 ends. At that moment, when the UAV lands on the base (x_b, y_b, z_b), the planner returns to state 0, waiting for the next monitoring process.

IV. VR VISUALIZER

The VR Visualizer was implemented using Unity 3D (version 2018.3.14f1) for rendering the virtual house environment, the UAV and the representation of the user in the virtual environment (avatar). The movement of the (virtual) UAV is achieved by periodically updating its position and orientation according to the values received from the UAV Simulator module. Moreover, the up-to-date position of the user is sent back to this module, which then calculates the new position of the UAV in order to follow the person at a certain distance.

User interfaces were designed for this module using VR devices, a traditional screen, a keyboard and a mouse (see a video under Supplementary Material, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TETC.2020.3000352.). Unity 3D allows the creation of both types of interfaces. The former is devoted to the first person experience of having a UAV flying around, while the latter is used for testing the UAV flight inside the environment, mainly because it is easier to visualize the trajectory using a third person perspective and allowing free camera movement. When using the VR interface, the user is only able to walk and look around, and switch on/off the lights, using the HTC Vive head-mounted display (HMD) and its controllers to navigate. Spatial sound is used to simulate the buzzing sounds of the UAV. The traditional interface is divided into three frames that can interchangeably display different camera views, namely the first person view of the avatar, the UAV camera, and a selection of camera positions placed around the room, including a free viewpoint camera. It is also possible to show the trajectory followed by the UAV visible inside the virtual environment to assess the correct functioning of the system, as this trajectory can be compared to the one generated by the UAV Simulator.

V. COMMUNICATION BETWEEN THE MODULES

Figure 4 shows the implementation of the high-level architecture described in Section II and depicted in Figure 1. It includes the tools used to implement each component, which were described in depth in Sections III and IV, and introduces the component that grants the communication between them. Regarding the UAV Simulator, the figure summarizes the three main tasks for which this component is responsible, namely the UAV's dynamics, the control algorithm and the trajectory generator. For the VR Visualizer, it is in charge of rendering the virtual environment (home), the avatar and the UAV.

The communication between the two modules is implemented using the Message Queue Telemetry Transport (MQTT) protocol, a light-weight messaging protocol based on a subscription model and topic publication. The modules implement an MQTT client that connects to the *message*

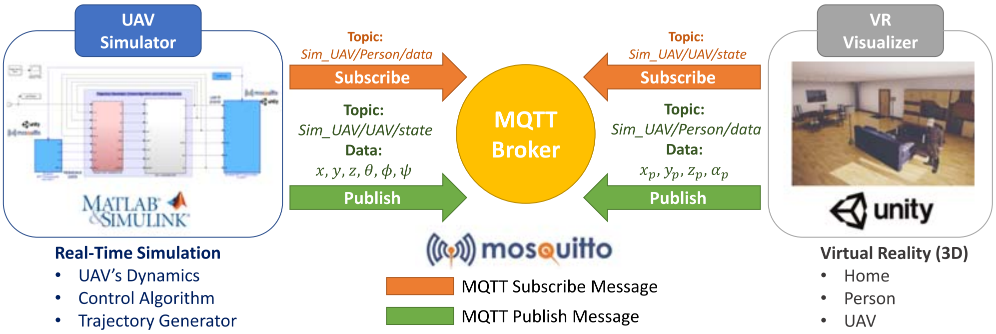

FIGURE 4. **Implementation of the architecture.**

broker, which is in charge of managing the connection and transmitting the messages. Mosquitto is the open source broker selected for this implementation. The UAV Simulator publishes the data regarding the UAV in the *Sim_UAV/UAV/state* topic, while it is subscribed to *Sim_UAV/Person/data*. Conversely, the VR Visualizer is subscribed to *Sim_UAV/UAV/state* and publishes the data about the person location inside the virtual environment in *Sim_UAV/Person/data*. The data published in the topic *Sim_UAV/UAV/state* regarding the UAV are the position (x, y, z) and the orientation angles (θ, ϕ, ψ), while the data published in *Sim_UAV/Person/data* regarding to the avatar are the position (x_p, y_p, z_p) and the head orientation angle (α_p).

To initiate a connection, the UAV Simulator subscribes to the topic *SIM_UAV/Session* and waits for VR Visualizer to publish the settings for the current simulation (UAV selected, virtual environment in which the simulation will take place, user to be followed by the UAV, etc.) and creates an instance of the simulation. During the session, the UAV simulation generates a trajectory based on the control algorithms, the UAV's dynamics and the location of the user, and publishes the position and orientation of the UAV in the virtual environment. At the same time, the VR Visualizer updates the position and orientation of the virtual representation of the UAV and publishes the location and the head orientation of the user inside the virtual environment.

VI. SIMULATION RESULTS

For the initial validation of the system proposed, we designed a living room in which a character (depicted by either an aging man or woman) was able to move around. The base of the UAV, as the starting point for the aircraft, was located in the south-west corner of the room. Furniture was added to make the living room look more realistic. As it has already been mentioned, the movement of the character was controlled by either a person using VR devices or an operator using a keyboard and a mouse.

In this scenery, the mission of the virtual UAV was to effectively perform the avatar monitoring process, i.e., to perform a supervision flight to approach the character and position correctly in order to capture of facial images from its simulated camera. At this point, some advantages provided by the designed tool must be highlighted. First, it is possible to verify the operation of the UAV from different points of view, including that of the camera itself aboard the UAV. This allows us, therefore, to observe the images that could be taken by the UAV camera, as well as to perceive how the assisted person would see the assistant UAV during monitoring.

Another advantage of the proposed system is that we continue to have the potential of MATLAB/Simulink to perform precise real-time simulations in order to measure the behavior of the trajectory planner and the GPI controller. By means of graphs, it is possible to represent (along the time) the UAV's state, the control input, and the reference trajectories, as well as to make comparisons and measure the different parameters. Therefore, not only can we observe in the virtual world the work of the assistant UAV, but we are able to precisely measure and evaluate the algorithms designed for the navigation of the UAV during the monitoring process.

All this is due to the fact that the tool designed is based on the integration in real time of the two modules that have been previously commented: the VR Visualizer module that builds the virtual 3D environment of the dependent person's home, and the UAV Simulator module, which is implemented in MATLAB/Simulink and is responsible of the simulation of the dynamic model of the quadrotor, the designed GPI control algorithm and the trajectory planner. In the correct communication of these two modules lies the key to the proper functioning of the proposed system. It is essential that data regarding the status of the UAV (position and orientation) be transmitted periodically to update the position and orientation of the virtual UAV (so as to get a realistic flight). On the other hand, the information concerning the position and direction of the person is used by the trajectory generator to calculate the precise references for the flight of the UAV. These references will be the inputs of the GPI algorithm, which determines the control actions required for stabilization and tracking of the reference trajectories, thus achieving the monitoring flight by the part of the UAV.

The parameters defined for the simulation are summarized in Table 1 for a test performed to evaluate the proposed system and to verify the performance of the trajectory planning algorithm and the GPI control for the assistant UAV. The test is as explained next.

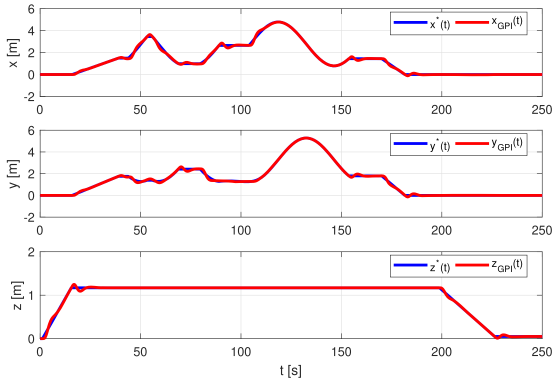

FIGURE 5. **Position and reference variables of the center of mass of the quadrotor.**

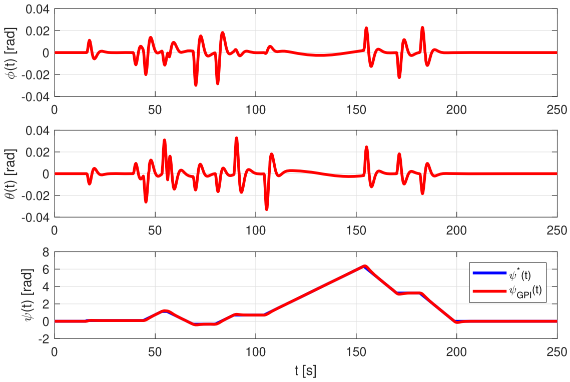

FIGURE 6. **Attitude variables of the quadrotor.**

A virtual character represented by an elderly man is in the living room (virtual) and when he sits on his sofa, the UAV starts the monitoring process from its initial position in the base. The UAV must, therefore, take off to the height defined by the *z*-coordinate of the avatar's position (representing the height of the center of the face). After this, the UAV will modify its orientation so that it can later approach the person with the focus of the camera already on him. Once the safety position has been reached, within the radius defined by parameter *R*, the UAV will stop briefly to precisely start the circular movement around the person until it is positioned in front of his face. At that time, it will stop for a while to emulate image capture. In case the person turns his/her head before the programmed data capture time elapses, the UAV turns again around the person in the appropriate direction to position itself in front of his/her face. Once the UAV reaches the correct position and the programmed time remains without detecting significant changes, the face's snapshot is considered to be taken. Finally, the UAV will complete the movement around the person to return to the safe position and from there go back to its base in the corner of the room. The video under Supplementary Material, available online, illustrates this process.

Let us explain the graphs obtained from the MATLAB/ Simulink environment in detail. First, Figure 5 represents the temporal evolution of the UAV's position coordinates. Here, it is possible to observe the trajectory defined by the planner (in color blue) and the real trajectory performed by the UAV (in red). These references are constructed on the basis of the information relative to the person. In this case, it is the avatar's position and direction, the UAV's outputs variables, and the different states which that make up the planner. We observe how the first and last maneuvers are the takeoff and landing of the UAV (*z*-coordinate), while the intermediate maneuvers are those performed at the horizontal plane (coordinates *x* and y). This figure also allows us to verify the action of the GPI controller. We can check that effectively the UAV is capable of following the position references by the action of the control algorithm.

On the other hand, it is necessary to control the UAV's yaw angle to correctly perform the monitoring process, as illustrated in Figure 6. Again, we observe how the controller is able to follow the required angle at each moment, while the independent orientation angles are bounded to a small range (lower than ±0.05 rad). This meets the assumption considered in the quadrotor's dynamic model. Thereby, the GPI algorithm is responsible for calculating the necessary control actions to govern the quadrotor's flight. The time evolution of these control inputs is represented in Figure 7.

Lastly, Figure 8 shows the trajectory of the UAV both in the UAV Simulator and in the VR Visualizer. The left side of Figure 8 illustrates the reference trajectory generated by the planner (in blue) and the trajectory performed by the UAV (in red), both in a 3D representation. In this picture, the planner's key points are highlighted; (i) the base position where the quadrotor is waiting until the monitoring begins (and where it returns at its conclusion), (ii) the safety position to approach the person (and where it returns after completing the circular motion), (iii) the photo position in which the UAV stops to capture the data (on the first and second attempt, the data capture process was not completed because the person turned his/her head), and (iv) the avatar's head position and direction, i.e., where he/she is and where he/she is looking at during the monitoring (the direction at the time of photo capture is represented in green, while the two failed photo attempts are represented in gray). Finally, the right side of Figure 8 shows in pink the trajectory followed inside

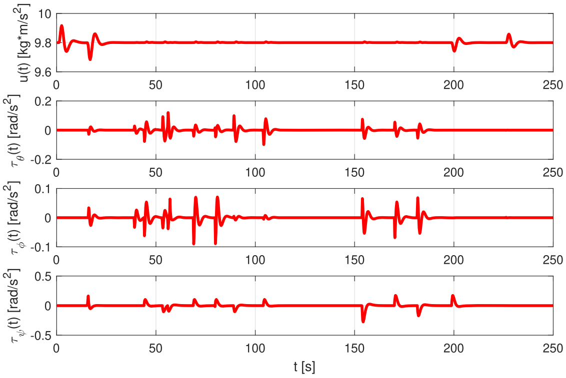

FIGURE 7. **Applied control inputs.**

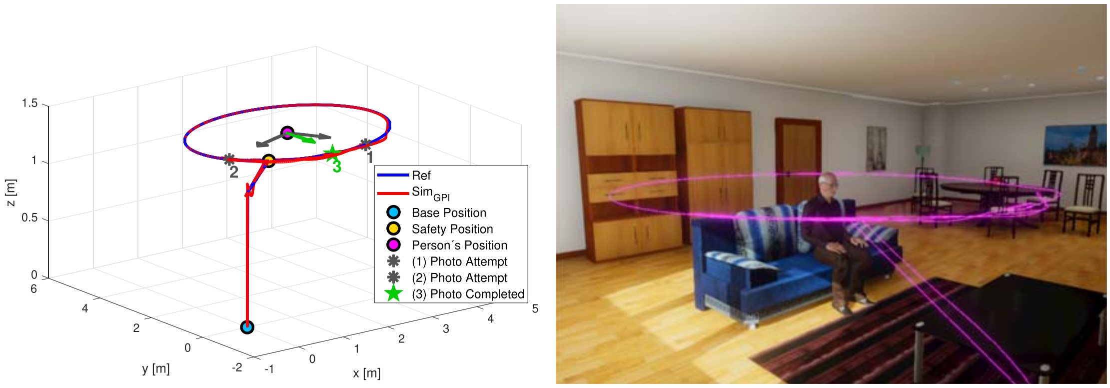

FIGURE 8. **Trajectory simulated in the UAV Simulator (left) side by side with the trajectory displayed in the VR Visualizer (right).**

the virtual living room environment. Even though the perspective is not the same in both pictures, it is easy to see that the UAV behaves similarly in both the mathematically simulated environment and the virtual reality environment.

Thereby, it can be concluded that the initial tests show the effectiveness of the VR system for the simulation of the quadrotor in monitoring a (virtual) person. According to the data received from the VR Visualizer, the state-machine-based trajectory planner is capable of constructing reference paths for the coordinates of the position and yaw angle so that the GPI algorithm determines the control actions for both stabilization and trajectory tracking by the UAV. On the contrary, the position and orientation of the quadrotor model are sent from the UAV Simulator to the VR Visualizer to recreate the flight in a realistic manner. Therefore, the communication between the modules works correctly.

VII. PRELIMINARY ACCEPTANCE EVALUATION

Since the use of assistant UAVs in a home environment is a new application of the drone technology, we wanted to assess its acceptance by the end users. However, at this early stage, we decided not to focus directly on the dependent people but to approach first people who have (or have had) experience with dependent people. The reason for this choice is twofold. First, we did not want to disturb dependent people with their participation in an evaluation at this early prototype stage. Second, we considered that people that have first-hand experience with a dependent person (either a member of the family at home or a friend) are aware of the special needs and the difficulties they face in their daily lives. Of course, this evaluation does not replace a future evaluation with dependent people, as their opinion is vital for the adoption of assistant UAVs.

A. EVALUATION METHOD AND PARTICIPANTS

The task that the users had to perform was easy, as they just needed to sit and wear the HMD for about 3 minutes. In the virtual environment, they remained seated in a sofa in front of a TV, pretending they were watching it while the UAV was performing its normal operation, as described in Section VI. Even though the participants could move around, we preferred them to remain seated for the sake of simplicity. Once the UAV had completed the monitoring task, the users took off the HMD and filled in a short questionnaire.

A standard questionnaire for the acceptance of a UAV could not be found in the literature. Although there are questionnaires for the acceptance of robots, most of them focus on their appearance or on their social interaction skills [38], [39]. For that reason, we decided to design our own questionnaire, inspired in other approaches [40], [41], [42], to better represent the context of the assistant UAV related to this study. First, the participants had to fill a demographic questionnaire and, after that, they had to reply to the 14 questions that composed the acceptance questionnaire (see Table 2). The responses were measured using a 5-point Likert scale ranging from strongly disagree (1) to strongly agree (5).

Twenty-two participants (50 percent females, 50 percent males) took part in the evaluation. They all had hands-on experience in the care of dependent people, although only one was a professional caregiver. The average age was 39.59 years ($SD = 11.18$).

B. RESULTS AND DISCUSSION

Figure 9 and Table 2 show the results of the questionnaire. The table provides median values along with the mean values for each question. Moreover, Figure 9 presents the histograms depicting the frequency of each response.

In general terms, the assistant UAV proposal was found interesting (Prop1 $Mdn = 5$, being 5 -strongly agree- the 77 percent of the responses). 91 percent of the respondents agreed or strongly agreed with Prop2 (values 4 and 5), which was related to the belief that an assistant UAV would improve the quality of life of the dependent people and their family. Moreover, the same percentage of the respondents would also recommend the use of this technology to people in a dependent position (Prop3). Similarly, 91 percent of the respondents agreed or strongly agreed with Prop4, which stated that they would feel good at home having an assistant UAV if they needed one.

Regarding the questions belonging to the safety block, the results were similar. All the responses for both Saf1 and Saf3 questions lie within the rage 4 to 5. This indicates that the respondents felt safe during the monitoring process (Saf1) and

TABLE 2. Questionnaire used for the assessment of the acceptation of assistant drones at home. Descriptive statistics are added for each question and block of questions.

Question	Mean	SD	Median	IQR
Proposal	4.70	0.59	5	0
Prop1: I find interesting the use of a assistant UAVs for the monitoring of dependent people	4.77	0.43	5	0
Prop2: An assistant UAV would help improve the quality of life of dependents (and their families)	4.68	0.65	5	0
Prop3: I would recommend this technology to family and/or friends in a dependency situation	4.73	0.63	5	0
Prop4: I would feel good at home if I was a dependent and had a care UAV	4.64	0.66	5	0.75
Safety	4.38	0.95	5	1
Saf1: I felt safe during the monitoring process by the UAV	4.73	0.46	5	0.75
Saf2: I found the UAV's monitoring distance to be correct	4.55	0.67	5	1
Saf3: I found the speed of the UAV adequate	4.77	0.43	5	0
Saf4: I think having a UAV flying in a house is not a risk for the property	3.45	1.30	4	1.75
Privacy	3.69	0.93	4	1
Priv1: I find it appropriate to have a UAV monitoring my activity at home	3.77	0.61	4	1
Priv2: I do not feel that there is too much supervision	3.55	0.96	3.5	1
Priv3: I would not mind if a computer processed the images captured by the UAV	3.91	0.87	4	1.75
Priv4: I do not think the UAV would distract me from my daily routine	3.55	1.18	4	2.5
Self-Concept	3.98	1.10	4	2
Self1: Using an assistance robot would not make me feel less self-sufficient	4.14	0.94	4	1
Self2: I would not be embarrassed if other people knew I was using a care UAV	3.98	1.02	4	1.25

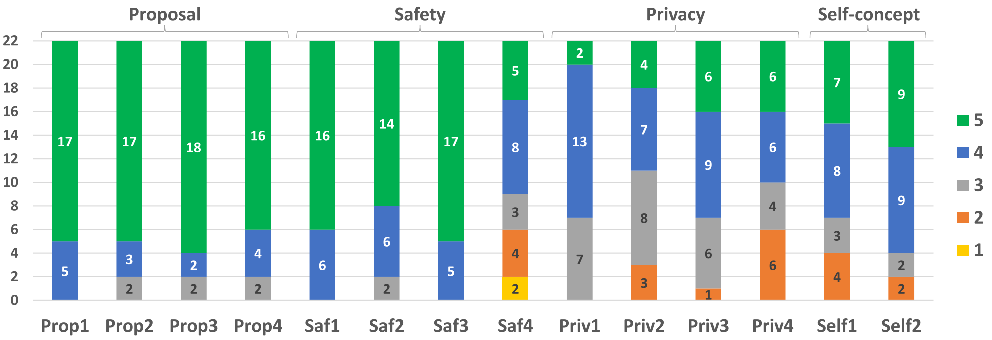

FIGURE 9. Frequency of responses to each item in the questionnaire used to assess the acceptance of UAVs. Responses ranged from *strongly disagree* (1) to *strongly agree* (5).

found the speed of the UAV adequate (Saf2). The percentage of answers within the 4–5 range for Saf2 was lower, but still high enough (91 percent) to claim that the participants found the monitoring distance to be correct. The case of Saf4 is different. 58 percent of the respondents did not consider the fact of having a flying robot at home a risk for the property (answers from 4 to 5), but, in this case, 27 percent of the responses lie within the range 1 to 2 (strongly disagree and disagree). The median value for this response is 4, but with a high dispersion ($IQR = 1.75$), showing a variety of answers.

The questions about privacy obtained lower ratings in general, with values centered around 4 (third row in Figure 9). This can be interpreted as follows. Although the participants did not fully agree with the idea of being monitored by a robot, they still would accept it (88 percent of the responses lie within the positive range from 3 to 5, while only 12 percent of them are negative). The block named as self-concept is related to how the participants would see themselves if they had to use this assistant technology. Question Self1 is related to not feeling less self-sufficient ($Mdn = 4$) and Self2 is about not feeling embarrassed about people knowing that they use an assistant UAV ($Mdn = 3.98$). Having a look at the results, it appears that the participants would not worsen their self-image if they had to use this technology.

As expected, there was no significant difference in the responses between genre and age groups (divided by 40.5 as the median of the age of the participants). This was tested using the Mann-Whitney U Test with 5 percent as the level of significance, as the data did not follow a normal distribution.

VIII. CONCLUSION

In our ongoing research project devoted to the development of autonomous UAVs for assistance to dependent-care people, mainly elderly people, we have identified the need to address not only technical aspects but also the human factor. The first one concern the engineering solutions for autonomous navigation of the UAV. In this sense, this paper has described in detail the planning algorithm designed for the flight of a quadrotor in the monitoring process. This consists of an autonomous flight so that the UAV, from time to time, positions correctly to enable the on-board camera to take facial images of the person. These images will be processed through computer vision to determine the person's mood and the assistance action required.

With regards to the human factor aspects, principally it is necessary to respect the personal space during the monitoring process and to work towards the acceptance of the assistant UAV. These factors lead to evaluating the performance of the UAV also from the viewpoint of the assisted person. In this sense, this paper has presented a novel architecture for validation of the UAV assistants using virtual reality. This allowed us to conduct a first study on the perception of such technology by people who have a first-hand experience with dependent persons.

The VR platform consists of a real-time communication, using the MQTT protocol, between two modules: the UAV Simulator, implemented in MATLAB/Simulink, and the VR Visualizer, designed with Unity 3D. The first module is in charge of the real-time simulation of the UAV's dynamics, the GPI control algorithm and the trajectory planner. On the other hand, Unity 3D recreates the virtual home environment that is shared by an avatar, who can move freely emulating the real behavior of a person at home, and the 3D model of the quadrotor UAV. The motion of the virtual UAV is achieved by periodically updating the position and orientation according to the values received from the UAV Simulator. At the same time, the information relative to the person, in this case, the avatar, is sent from the VR Visualizer to the UAV Simulator to correctly calculate the flight path for the monitoring process.

Regarding the human factors, the results of a first study have been presented from the participation of people who have or have had experience with people in a dependency situation. In this evaluation, the users tested the platform using immersive VR technology so that they experienced the monitoring work of the assistant UAV in first person. According to the results, the acceptance of the assistant UAV proposal was considered positive, as well as the benefits that such a system would have for dependent people. Moreover, the aspects relative to safety and privacy during the monitoring process were also positively evaluated, although to a lesser extent for the latter.

It is worth noting that this type of study allows drawing conclusions in order to improve the design of the UAV in different aspects because, among others, the participants were questioned about some configurable parameters (such as the UAV's speed or monitoring distance). In addition, we received direct feedback from participants, which is very valuable to address future developments. For instance, in future studies, we would like to assess the convenience of raising the UAV's flight height so that it is not in the person's direct line of sight, as some participants complained about the UAV obstructing the TV.

At this point, it is important to emphasize that the developed architecture enables us to conduct both technical tests and user evaluations in a controlled and safe virtual environment, thus saving costs and enabling us to advance with the next steps of this research project that is still in its initial stages. In this way, this work lays the foundations to face future developments and overcome current limitations, which are mainly related to the assumptions considered in the trajectory planner. It is necessary to increase the level of detail of the planner to implement the transitions between states in order to adapt the UAV's flight to the possible movement of the person during the monitoring and to integrate a solution to detect and avoid obstacles. In this sense, at the height at which the UAV works, there are usually far fewer obstacles than at ground level, but it is necessary to consider also the appearance of unexpected obstacles in the flight path that may not be represented on a home's map, such as lamps, furniture or even another person accompanying the dependent (that should be treated as dynamic obstacles from the navigation point of view). All this entails the need for developing a reactive navigation solution to dynamically adapt the flight path defined by the planner to avoid obstacles than could be detected by the on-board camera.

Another limitation of this work is the evaluation performed. Even though it was useful to obtain valuable information regarding the experience of having a UAV monitoring a person at home and that the participants were important actors in the care of dependent people, the system was not actually tested by the end users. The main reason for this was the difficulty to have access to them (in many cases, it may be also difficult for them to answer a questionnaire) and the initial state of development of the system, as it is worth involving them once more functionality will be developed.

Therefore, in view of the future objectives, the VR validation tool presented here will be paramount. It will provide an adaptable, safe and save-cost environment to test the performance of the UAV in different scenarios and conditions. Moreover, the monitoring process will also be evaluated when the dependent person is walking and/or makes natural and habitual movements in the virtual environment. In this case, for the simulation it will be necessary to study wireless alternatives to the currently used HTC Vive HMD like the Oculus Quest VR headset. In addition, as progress will be introduced at technical level, it will be necessary to continue evaluations with dependents and/or family members in order to obtain feedback from the participants to improve the appearance and/or behavior of the UAV. Therefore, when more situations than only sitting will be considered, we will be able to verify if the results on user acceptance are modified or remain with respect to those obtained in this first study. This will allow drawing conclusions about other improvements in the design of the autonomous assistant UAV before approaching the experimental phase tests as the last stage in its development.

ACKNOWLEDGMENTS

This work was partially supported by Ministerio de Ciencia, Innovación y Universidades, Agencia Estatal de Investigación / European Regional Development Fund under PID2019-106084RB-I00 and DPI2016-80894-R grants, and by CIBERSAM of the Instituto de Salud Carlos III.

REFERENCES

[1] A. Merwaday, A. Tuncer, A. Kumbhar, and I. Guvenc, “Improved throughput coverage in natural disasters: Unmanned aerial base stations for public-safety communications,” *IEEE Veh. Technol. Magazine*, vol. 11, no. 4, pp. 53–60, Dec. 2016.

[2] P. Li, T. Miyazaki, K. Wang, S. Guo, and W. Zhuang, “Vehicle-assist resilient information and network system for disaster management,” *IEEE Trans. Emerg. Topics Comput.*, vol. 5, no. 3, pp. 438–448, Jul. 2017.

[3] O. S. Oubbati, M. Mozaffari, N. Chaib, P. Lorenz, M. Atiquzzaman, and A. Jamalipour, “ECad: Energy-efficient routing in flying ad hoc networks,” *Int. J. Commun. Syst.*, vol. 32, no. 18, 2019, Art. no. e4156. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/dac.4156

[4] O. S. Oubbati, N. Chaib, A. Lakas, P. Lorenz, and A. Rachedi, “Uav-assisted supporting services connectivity in urban vanets,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3944–3951, Apr. 2019.

[5] O. S. Oubbati, A. Lakas, P. Lorenz, M. Atiquzzaman, and A. Jamalipour, “Leveraging communicating UAVs for emergency vehicle guidance in urban areas,” *IEEE Trans. Emerg. Topics Comput.*, to be published, doi: 10.1109/TETC.2019.2930124.

[6] S. J. Kim, G. J. Lim, J. Cho, and M. J. Côté, “Drone-aided healthcare services for patients with chronic diseases in rural areas,” *J. Intell. Robot. Syst.*, vol. 88, no. 1, pp. 163–180, 2017.

[7] J. M. Gascueña and A. Fernández-Caballero, “Agent-oriented modeling and development of a person-following mobile robot,” *Expert Syst. Appl.*, vol. 38, no. 4, pp. 4280–4290, 2011.

[8] J. wan, M. Li, M. O’Grady, X. Gu, M. A. A. H Alawlaqi, and G. O’Hare, “Time-bounded activity recognition for ambient assisted living,” *IEEE Trans. Emerg. Topics Comput.*, to be published, doi: 10.1109/TETC.2018.2870047.

[9] J. Martinez-Gomez, A. Fernández-Caballero, I. Garcia-Varea, L. Rodriguez, and C. Romero-Gonzalez, “A taxonomy of vision systems for ground mobile robots,” *Int. J. Advanced Robot. Syst.*, vol. 11, no. 7, 2014, Art. no. 111.

[10] E. Lozano-Monasor, M. T. López, F. Vigo-Bustos, and A. Fernández-Caballero, “Facial expression recognition in ageing adults: from lab to ambient assisted living,” *J. Ambient Intell. Humanized Comput.*, vol. 8, no. 4, pp. 567–578, 2017.

[11] I. Bisio, A. Delfino, F. Lavagetto, M. Marchese, and A. Sciarrone, “Gender-driven emotion recognition through speech signals for ambient intelligence applications,” *IEEE Trans. Emerg. Topics Comput.*, vol. 1, no. 2, pp. 244–257, Dec. 2013.

[12] L. M. Belmonte, R. Morales, and A. Fernández-Caballero, “Computer vision in autonomous unmanned aerial vehicles—a systematic mapping study,” *Appl. Sci.*, vol. 9, no. 15, 2019. [Online]. Available: https://www.mdpi.com/2076-3417/9/15/3196

[13] C. Todd *et al.*, “A proposed UAV for indoor patient care,” *Technol. Health Care*, vol. 2015, pp. 1–8, 2015.

[14] L. M. Belmonte, R. Morales, A. S. Garcőa, E. Segura, P. Novais, and A. Fernández-Caballero, “Assisting dependent people at home through autonomous unmanned aerial vehicles,” in *Proc. Advances Intell. Syst. Comput.*, 2019, pp. 216–223.

[15] J. Mayor, L. Raya, and A. Sanchez, “A comparative study of virtual reality methods of interaction and locomotion based on presence, cybersickness and usability,” *IEEE Trans. Emerg. Topics Comput.*, to be published, doi: 0.1109/TETC.2019.2915287.

[16] A. Seth, J. M. Vance, and J. H. Oliver, “Virtual reality for assembly methods prototyping: A review,” *Virt. Real.*, vol. 15, no. 1, pp. 5–20, 2011.

[17] D. J. Roberts, A. S. Garcia, J. Dodiya, R. Wolff, A. J. Fairchild, and T. Fernando, “Collaborative telepresence workspaces for space operation and science,” in *Proc. IEEE Virt. Reality*, 2015, pp. 275–276.

[18] X. Li, W. Yi, H.-L. Chi, X. Wang, and A. P. Chan, “A critical review of virtual and augmented reality (VR/AR) applications in construction safety,” *Autom. Construction*, vol. 86, pp. 150–162, 2018.

[19] J. D. Bric, D. C. Lumbard, M. J. Frelich, and J. C. Gould, “Current state of virtual reality simulation in robotic surgery training: A review,” *Surgical Endoscopy*, vol. 30, no. 6, pp. 2169–2178, 2016.

[20] X. Zhang, G. Foderaro, C. Henriquez, and S. Ferrari, “A scalable weight-free learning algorithm for regulatory control of cell activity in spiking neuronal networks,” *Int. J. Neural Syst.*, vol. 28, no. 2, 2018, Art. no. 1750015.

[21] E. Rohmer, S. P. Singh, and M. Freese, “V-REP: A versatile and scalable robot simulation framework,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 1321–1326.

[22] L. M. Belmonte, R. Morales, A. S. Garcőa, E. Segura, P. Novais, and A. Fernández-Caballero, “Trajectory planning of a quadrotor to monitor dependent people,” in *Understanding the Brain Function and Emotions*. Berlin, Germany: Springer, 2019, pp. 212–221.

[23] MathWorks, “Simulink desktop real-time,” 2020. [Online]. Available: https://www.mathworks.com/products/simulink-desktop-real-time.html

[24] L. M. Belmonte, R. Morales, A. Fernández-Caballero, and J. A. Somolinos, “A tandem active disturbance rejection control for a laboratory helicopter with variable-speed rotors,” *IEEE Trans. Ind. Electron.*, vol. 63, no. 10, pp. 6395–6406, Oct. 2016.

[25] L. M. Belmonte, R. Morales, A. Fernández-Caballero, and J. A. Somolinos, “Robust decentralized nonlinear control for a twin rotor mimo system,” *Sensors*, vol. 16, no. 8, 2016. [Online]. Available: https://www.mdpi.com/1424-8220/16/8/1160

[26] MathWorks Internet of Things Team, “MQTT in MATLAB - file exchange - MATLAB Central,” 2020. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/64303

[27] J. Meyer, A. Sendobry, S. Kohlbrecher, U. Klingauf, and O. von Stryk, “Comprehensive simulation of quadrotor uavs using ros and gazebo,” in *Simulation, Modeling, and Programming for Autonomous Robots*, I. Noda, N. Ando, D. Brugali, and J. J. Kuffner, Eds. Berlin, Germany: Springer, 2012, pp. 400–411.

[28] M. Zhang *et al.*, “A high fidelity simulator for a quadrotor UAV using ROS and gazebo,” in *Proc. 41st Annu. Conf. IEEE Ind. Electron. Soc.*, 2015, pp. 002 846–002 851.

[29] M. Schmittle *et al.*, “OpenUAV: A UAV testbed for the CPS and robotics community,” in *Proc. ACM/IEEE 9th Int. Conf. Cyber-Physical Syst.*, 2018, pp. 130–139.

[30] A. Fernández-Caballero, L. M. Belmonte, R. Morales, and J. A. Somolinos, “Generalized proportional integral control for an unmanned quadrotor system,” *Int. J. Advanced Robot. Syst.*, vol. 12, no. 7, 2015, Art. no. 85.

[31] R. Lozano, *Unmanned Aerial Vehicles: Embedded Control*. Hoboken, NJ, USA: Wiley, 2010.

[32] R. Morales, H. Sira-Ramőrez, and V. Feliu, “Adaptive control based on fast online algebraic identification and GPI control for magnetic levitation systems with time-varying input gain,” *Int. J. Control*, vol. 87, no. 8, pp. 1604–1621, 2014.

[33] L. Campos-Macőas, D. Gómez-Gutiérrez, R. Aldana-López, R. de la Guardia, and J. I. Parra-Vilchis, “A hybrid method for online trajectory planning of mobile robots in cluttered environments,” *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 935–942, Apr. 2017.

[34] J. A. Chocoteco, R. Morales, and V. Feliu, “Enhancing the trajectory generation of a stair-climbing mobility system,” *Sensors*, vol. 17, no. 11, 2017, Art. no. 2608.

[35] W. Lim, S. Lee, M. Sunwoo, and K. Jo, “Hierarchical trajectory planning of an autonomous car based on the integration of a sampling and an optimization method,” *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 2, pp. 613–626, Feb. 2018.

[36] S. Liu *et al.*, “Planning dynamically feasible trajectories for quadrotors using safe flight corridors in 3-D complex environments,” *IEEE Robot. Autom. Lett.*, vol. 2, no. 3, pp. 1688–1695, Jul. 2017.

[37] X. Lv, Z. Yu, M. Liu, G. Zhang, and L. Zhang, “Direct trajectory planning method based on IEPSO and fuzzy rewards and punishment theory for multi-degree-of freedom manipulators,” *IEEE Access*, vol. 7, pp. 20 452–20 461, 2019.

[38] T. Nomura, T. Kanda, and T. Suzuki, “Experimental investigation into influence of negative attitudes toward robots on human–robot interaction,” *AI Soc.*, vol. 20, no. 2, pp. 138–150, 2006.

[39] C. Bartneck, D. Kulić, E. Croft, and S. Zoghbi, “Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots,” *Int. J. Soc. Robot.*, vol. 1, no. 1, pp. 71–81, 2009.

[40] K. Dockx *et al.*, “Fall-prone older people’s attitudes towards the use of virtual reality technology for fall prevention,” *Gerontology*, vol. 63, no. 6, pp. 590–598, 2017.

[41] T. Ninomiya, A. Fujita, D. Suzuki, and H. Umemuro, “Development of the multi-dimensional robot attitude scale: Constructs of people’s attitudes towards domestic robots,” in *Proc. Int. Conf. Soc. Robot.*, 2015, pp. 482–491.

[42] L. Chiari, R. van Lummel, K. Pfeiffer, U. Lindemann, and W. Zijlstra, “Deliverable 2.2: Classification of the user’s needs, characteristics and scenarios-update,” *Unpublished report from the EU Project (6th Framework Program, IST Contract no. 045622) Sensing and Action to support mobility in Ambient Assisted Living]. Department of Health*, 2009.

LIDIA M. BELMONTE received the BSc degree in industrial engineering (Honors) from Universidad de Castilla-La Mancha, Spain, in 2008, the MSc degree in industrial engineering from Universidad Nacional de Educación a Distancia (UNED), Spain, in 2013, and the PhD degree from Universidad de Castilla-La Mancha in 2019. She is an associate lecturer at Escuela Técnica Superior de Ingenieros Industriales, Universidad de Castilla-La Mancha, Albacete, Spain. Her research interests include the modeling and control of unmanned aerial vehicles (UAVs), assistance robotics and virtual reality applications.

ARTURO S. GARCÍA received the European PhD degree in computer science from the Universidad de Castilla-La Mancha, in 2010. He is a research fellow at the Universidad de Castilla-La Mancha (UCLM) in Spain. He was an R&D Manager at SymbiaIT, a Spin-Off of the UCLM (2009–2013), and then he was a research fellow with the University of Salford (2014–2017). In 2017, he joined the UCLM as a research fellow and assistant professor. His research interests include the development of collaborative virtual environments (CVEs), interaction in VEs and CVEs and the creation of affective virtual humans.

EVA SEGURA received the MSc degree in business from the Universidad de Castilla-La Mancha, Spain, in 2010, the MSc degree in economics from the Universidad Nacional de Educación a Distancia, Spain, in 2015, and the PhD degree from Universidad de Castilla-La Mancha in 2018. She is a researcher at Escuela Técnica Superior de Ingenieros Industriales, Universidad de Castilla-La Mancha, Albacete, Spain. Her research interests include the technical and economic study of robotics and renewable energy systems. She is the author of more than 30 papers published in refereed professional journals and international conference proceedings.

PAULO NOVAIS (Senior Member, IEEE) is full professor of computer science with the Department of Informatics, School of Engineering of the Universidade do Minho, Portugal, and a researcher with the ALGORITMI Centre, where he is the leader of the research group ISlab - Synthetic Intelligence lab -, and the coordinator of the research line in Computer Science and Technology (CST). His main research aim is to make systems a little more smart, intelligent and also reliable. He is the co-author of more than 350 book chapters, journal papers, conference and workshop papers, and books. He is the president of APPIA (the Portuguese Association for Artificial Intelligence) since 2016; (Institute of Electrical and Electronics Engineers); Member of the IFIP (International Federation for Information Processing) - TC 12 Artificial Intelligence.

RAFAEL MORALES received the MSc degree in industrial engineering and the PhD degree from Universidad de Castilla-La Mancha, Spain, in 2002 and 2006, respectively. He is full-time lecturer at Escuela Técnica Superior de Ingenieros Industriales, Universidad de Castilla-La Mancha, Albacete, Spain. From 2007 to 2008, he was a visiting researcher with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA. His research interests include the design, development and control of industrial robots, assistance robots and unmanned aerial vehicles (UAVs). He is the author of more than 80 papers published in refereed professional journals and international conference proceedings. He was awarded the Industrial Robot Highly Commended Award in 2007 and the Highly Commended Paper at the Literati Network awards for Excellence 2008 by the Industrial Robot Journal. He is a member of the Editorial Board of the International Journal of Advanced Robotics Systems and Mathematical Problems in Engineering.

ANTONIO FERNÁNDEZ-CABALLERO received the MSc degree in computer science from Universidad Politécnica de Madrid, Spain, in 1993, and the PhD degree from the Department of Artificial Intelligence, Universidad Nacional de Educación a Distancia, Spain, in 2001. He is full professor with the Department of Computer Science, Universidad de Castilla-La Mancha. He is head of the n&aIS (natural and artificial Interaction Systems) research group belonging to the Albacete Research Institute of Informatics. His research interests are in image processing, cognitive vision, mobile robotics, affective computing, and intelligent agents. Antonio Fernández-Caballero is associate editor of Pattern Recognition Letters, Topic editor-in-chief for Vision Systems of International Journal of Advanced Robotic Systems, and Specialty chief editor for Robot and Machine Vision of Frontiers in Robotics and AI, among other editorship tasks. He has authored more than 370 peer-reviewed papers.