IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 19, 2019, accepted October 9, 2019, date of publication October 23, 2019, date of current version November 4, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2948880

A Novel Natural Mobile Human-Machine Interaction Method With Augmented Reality

GUANGLONG DU[1], BO ZHANG[1], CHUNQUAN LI[2], AND HUA YUAN[1]

[1] School of Computer Science and Engineering, South China University of Technology, Guangzhou 510641, China
[2] School of Information Engineering, Nanchang University, Nanchang 330029, China

Corresponding author: Chunquan Li (lichunquan@ncu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61973126, Grant 61863028, Grant 81660299, and Grant 61503177, in part by the Guangdong Natural Science Funds for Distinguished Young Scholar under Grant 2017A030306015, in part by the Pearl River S&T Nova Program of Guangzhou under Grant 201710010059, in part by the Guangdong Special Projects under Grant 2016TQ03X824, in part by the Fundamental Research Funds for the Central Universities, and in part by the Science and Technology Department of Jiangxi Province of China under Grant 20161ACB21007, Grant 20171BBE50071, and Grant 20171BAB202033.

ABSTRACT This paper proposes a novel teleoperation method that allows users to guide robots hand by hand along with speech. In this method, the virtual robot modeled according to the remote real robot is projected into the real local environment to form a 3D operation interface. In this case, users can directly interact with virtual objects by their hands. Furthermore, since the Leap Motion is attached to the augmented reality (AR) glasses, the operation space is greatly extended. Therefore, users can observe the virtual robot from an arbitrary angle without blind angle in such a mobile pattern, which enhances the users' interactive immersion and provides more natural human-machine interaction. To improve the accuracy of the measurement, an unscented Kalman filter (UKF) and an improved particle filter (IPF) are used to estimate the position and orientation of the hand, respectively. Furthermore, Term Frequency-Inverse Document Frequency (TF-IDF) and maximum entropy model are adopted to recognize the speech and gestures instructions of the user. The proposed method is compared with the three human-machine methods on various experiments. The results verified the effectiveness of the proposed method.

INDEX TERMS Teleoperation, augmented reality, hand by hand, unscented Kalman filter, term frequency - inverse document frequency.

I. INTRODUCTION

Nowadays, robots have played an increasingly pivotal role in the development of technology. They are employed not only in manufacture, but also in other various domains such as search and rescue, mine and bomb detection, scientific exploration, entertainment and hospital care. Particularly, many new application areas need to provide some interactions between human and machines. For example, human and robots share the workspace to improve the quality and efficiency of complex tasks.

Currently, various available methods have been proposed to implement human-machine interactions. Some methods are focusing on contact interaction between human and machine [1], [2]. Hyunki *et al.* [3] developed a soft wearable robot to replace a full-body rigid-frame exoskeleton device, improving the comfort of human-machine interaction. In [4], Zhang *et al.* proposed inertial/magnetic sensor module for improving the pedestrian tracking of the human-machine interaction. Gabriele *et al.* [5] attached the magnetic inertial unit to the waist for providing the effective sensor data fusion. Zhao *et al.* [6] used a brain-machine interface to capture the EMG signals, which were classified to generate robot commands. Xu *et al.* [7] used a haptic device named Phantom Device to control the remote manipulator, while force feedback was used to aid the operator. Rebelo *et al.* [8] developed a wearable arm exoskeleton master to achieve end-to-end control of the robot. Hou *et al.* [9] used a haptic joystick to control an aerial robot remotely. In these methods, the operator's movement is limited to a certain interactive space due to the contact device, and thus such interactions may not be sufficiently friendly and natural.

Non-contact human-machine methods [10], [11] are developed for handling interaction problems. In these methods,

The associate editor coordinating the review of this manuscript and approving it for publication was Sara Dadras.

the operators do not need to be in direct contact with the devices, but their human motion is measured in a non-contacting way such as the vision-based method. In [12], the physical markers were placed on the hand and two cameras were used to capture the hand-arm motion and control the remote robot. Liu *et al.* [13] employed a webcam to recognize 13 markers attached on human's body to control humanoid robot. Zinchenko *et al.* [14] developed a novel interface using speech to control surgical robots. Miura *et al.* [15] utilized genetic algorithms and Bayesian networks to segment the gestures collected by the Kinect body motion sensor, making the measurement accuracy. In the approach proposed by Browatzki *et al.* [16], the operator could use posture estimation methods to generate commands for interacting with the robot.

Note that the above non-contact methods adopted markers to detect human motion. Once the markers are occluded, those methods may fail. Therefore, the markerless way seems to be a better method for no-contact human-machine interaction, where hand information is generally obtained from unmarked devices. Kofman *et al.* [17] proposed a label-free vision-based tracking method for remote operation in 3D space. Such a method avoids hindrance of natural motion, but the background must be dark. In [18], the gestures of the human hand tracked by a Kinect can naturally control dual robot manipulators, however, the clear border between virtual software interface and reality world still affects user's experience. Furthermore, Du *et al.* [19] used a Leap Motion to capture gestures with Interval Kalman Filter (IKF) and Particle Filter. However, IKF diverges over time and does not perform well in a nonlinear system.

Although the contact methods have the advantage of high precision and the non-contact methods show the feasibility of natural interactions, both types of methods still need to be further improved. In the contact methods, the operator's movement is easily restricted by the contact device so that the human-machine interaction may be unnatural and even inefficient. On the other hand, in the non-contact methods, an operator can obtain information feedback via videos or 3D models. However, the limitation of the vision may cause incorrect operation. In addition, there are clear borders between virtual scenes and the real word, which affects the users' operating experience and degenerates the interactive immersion.

To address the above problems, this paper proposes a novel natural mobile human-machine interactive method. In this method, a mobile gesture sensor and an augmented reality (AR) wearable device are employed to establish an effective fusion between virtual robots and real hands so that an operator with the gestures and speech can naturally interact with the virtual robot. Because the movement of the virtual robots can be transmitted to the real robots in the remote environment via the internet, the operator can effectively teleoperate the real robots. More specifically, the AR wearable device can display the virtual robots that are regarded as a 3D operation interface and projected from the remote real robots; the mobile gesture sensor named Leap Motion is fixed on the AR wearable device not only to detect the hand movement but also to greatly expand the operating space; by combining the AR wearable device with Leap Motion, the operator can obtain a highly immersive virtual-real interaction scene where the operator can directly interact with the virtual robots by integrating his or her gestures and speech; the motion of the virtual robot can be reproduced by the real robot in the remote environment through the Internet. Therefore, the operator can naturally interact with the remote real robots by interacting with the virtual robots.

The main contributions of this paper are summarized as follows:

1) We propose a natural mobile human-machine interactive method, which adopts the augmented reality technology to avoid the clear borders between the virtual robots and the real hands of the operator. This can enhance the operator's interactive immersion and obtain more natural human-machines interaction between the hands and the remote real robots in a variable working environment.

2) By fixing the Leap Motion on the AR wearable device, the interactive space between the operator and the virtual robot has been greatly expanded. In particular, when the operator moves, he or she can naturally interact with the virtual robot, regardless of the matching relationship between their hands and the virtual robot or the occlusion factor. Thus, the operator can provide effective teleoperation between his or her hands and the real robot in such a mobile pattern.

3) To improve the accuracy of the measurement, an unscented Kalman filter (UKF) and an improved particle filter (IPF) are applied to estimate the positions and orientations of the hands and obtain the corresponding gestures, respectively. Furthermore, Term Frequency-Inverse Document Frequency (TF-IDF) and maximum entropy model are adopted to recognize the speech and gesture instructions of the operators for accurately controlling the virtual robot.

The rest of this paper is organized as follows: Section II is an overview of the whole mobile human-machine interactive method. Section III illustrates the way of the virtual-real fusion. Section IV demonstrates how to detect the position and posture of the hands. The fusion of speech and gesture is given in Section V. To validate the proposed method, experiments are designed in Section VI. Section VII and VIII provide the discussion and conclusion, respectively.

II. OVERVIEW

The intuitive interaction process of the proposed mobile human-machine interactive method is shown in Fig. 1, where two real robots (see Fig. 1(e)) in the remote environment has been projected into two virtual robots (see Fig. 1(a)). Such virtual robots can be observed by an operator with the AR wearable device (see Fig. 1(a)). As shown in Fig. 1(b) and 1(c), the operator can also employ the commands of the gestures and speech to guide the virtual robots in the virtual scene. Fig. 1(d) is a composite picture of Figs. 1(a), 1(b), and 1(c) for the purpose of visually

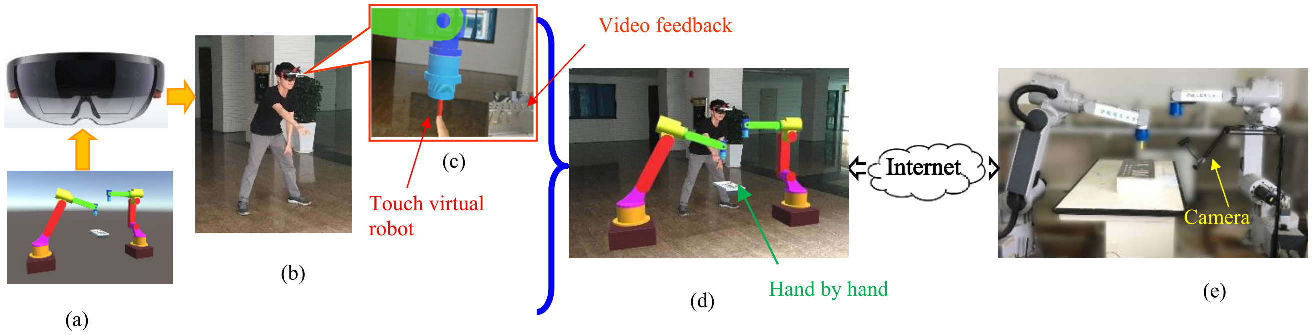


FIGURE 1. **The intuitive interaction process of the proposed mobile human-machine interactive method.**

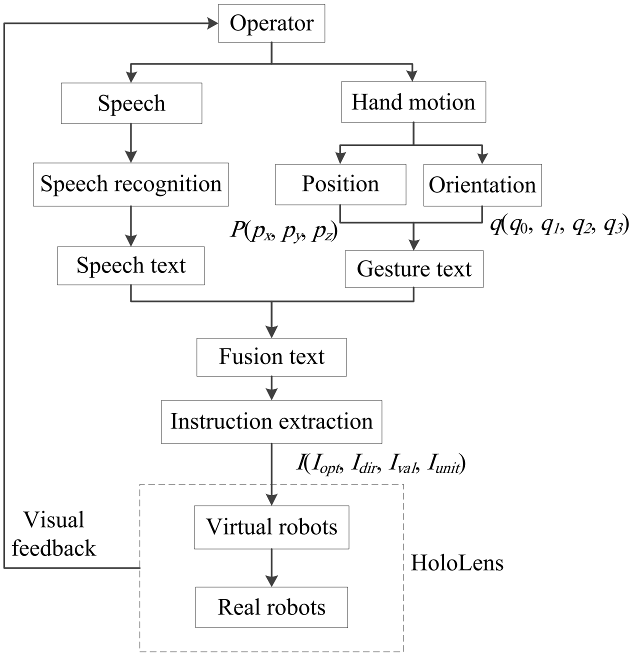


FIGURE 2. **The whole framework of the proposed mobile human-machine interactive method.**

showing how the operator interacts with the virtual robots. Both Figs. 1(d) and 1(e) display that the movement of the virtual robot can be passed to the real robots in the remote environment. Thus, the operator can provide teleoperation for the real robots by directly interacting with the virtual robots. In addition, the motion of the real robot can also be transmitted to the operator by the camera and the Internet.

The whole framework of the proposed mobile human-machine interactive method is shown in Fig. 2, where the hand motion or speech can be first converted into speech text or gesture text, separately. Subsequently, these two types of text can be merged into instruction extraction for guiding or interacting with the virtual robots. Further, the motion of the virtual robots can be sent to the real robots in the remote environment through the internet. Therefore, the real robots are able to accomplish similar tasks that the virtual robots are doing. In addition, the motion of the virtual and real robots can be visually fed back to the AR wear device. In short, the above framework can provide a mobile human-machine interactive method in a natural and friendly pattern.

Actually, the above mobile human-machine interaction consists of three main parts: the virtual-real fusion, the position and orientation estimation, and the multimodal instruction generation, elaborated in the following sections III, IV, and V, respectively.

III. VIRTUAL-REAL FUSION

The virtual-real fusion is intended to implement the natural interaction between the real hand and the virtual robots, avoiding the clear borders between those. The implementation of the virtual-real fusion involves two main parts. One is the coordinate registration; the other is the interactive detection.

A. COORDINATE REGISTRATION

When the hand interacting with the virtual robots, its positions and orientations can be recorded in different coordinate systems using the Leap Motion and AR wearable device, respectively. To achieve the accurate interaction between them, an effective coordinate registration between different coordinate systems are indispensable. The implementation of the coordinate registration is detailed as follows.

As shown in Fig. 3, $\{X_wY_wZ_w\}$, $\{X_lY_lZ_l\}$, and $\{X_hY_hZ_h\}$ represent the world coordinate system, the coordinate system of the Leap Motion, and the coordinate system of the hand, respectively. Note that $X_lY_lZ_l$ follows a right-hand Cartesian coordinate criteria and its origin is located in that of the Leap Motion controller. Furthermore, Fig. 3 shows that the Leap Motion can build a virtual human skeleton model, and the center of the palm is defined as the origin of $\{X_hY_hZ_h\}$. By the Leap Motion, the corresponding position and orientation of the hand in $\{X_hY_hZ_h\}$ can be transferred into those in $\{X_lY_lZ_l\}$ as follows:

$$[P_x^L, P_y^L, P_z^L]^T = T_{H2L}[P_x^H, P_y^H, P_z^H]^T \tag{1}$$

where $[P_x^H, P_y^H, P_z^H]^T$ represents the position of any point on the hand in the coordinate $\{X_hY_hZ_h\}$; T_{H2L}stands for the transformation matrix from $\{X_hY_hZ_h\}$ to $\{X_lY_lZ_l\}$, which can be obtained through the Leap Motion own program;

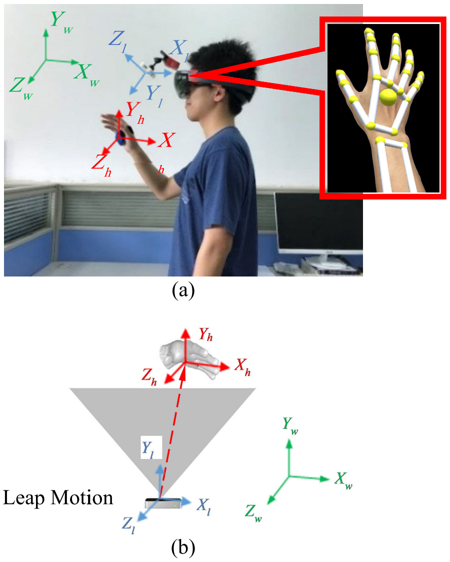


FIGURE 3. Construction of different coordinate systems.

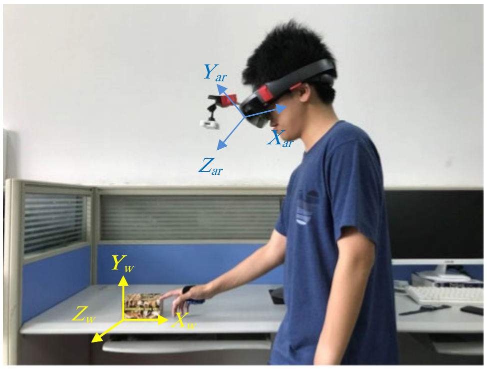


FIGURE 4. The coordinates between HoloLens and real world.

$[P_x^L, P_y^L, P_z^L]^T$ denotes the position of $[P_x^H, P_y^H, P_z^H]^T$ in the coordinate $\{X_l Y_l Z_l\}$.

Subsequently, $[P_x^L, P_y^L, P_z^L]^T$ in the$X_l Y_l Z_l$ can be transferred into the following position in the world coordinate $\{X_w Y_w Z_w\}$:

$$[P_x^W, P_y^W, P_z^W]^T = T_{L2W}[P_x^L, P_y^L, P_z^L]^T \tag{2}$$

where $[P_x^W, P_y^W, P_z^W]^T$ denotes the position of any point on the hand in the coordinate $\{X_w Y_w Z_w\}$; T_{L2W} is the transformation matrix from $\{X_l Y_l Z_l\}$ to $\{X_w Y_w Z_w\}$.

In fact, the certain corner point of a calibration box is employed as the origin of the world coordinate system $\{X_w Y_w Z_w\}$, shown in Fig. 4. In addition, Fig. 4 displays that $\{X_{ar} Y_{ar} Z_{ar}\}$ denotes the coordinate system of the AR wearable device, where the position and orientation of the virtual robots displayed in the AR wearable device are described. To implement the accurate interaction between the hand and the virtual robots, both their positions and orientations should be converted to the same coordinate system for the calculation. For simplicity, we also convert the positions and orientations of the virtual robots to the same coordinate $\{X_w Y_w Z_w\}$. Hence, the position of any point on the virtual robot can be rewritten as

$$[P_x^W, P_y^W, P_z^W]^T = T_{AR2W}[P_x^{AR}, P_y^{AR}, P_z^{AR}]^T \tag{3}$$

where $[\mathrm{P}_x^{AR}, P_y^{AR}, P_z^{AR}]^T$ denotes the position of any point in the coordinate $\{X_{ar} Y_{ar} Z_{ar}\}$; T_{AR2W} is the transformation matrix from $\{X_{ar} Y_{ar} Z_{ar}\}$ to $\{X_w Y_w Z_w\}$.

The operator utilizes the AR wearable device to observe the calibration box. The calibration program [20] from the AR wearable device can obtain the stereoscopic data of the calibrated box by capturing the images of the multiple surfaces of the calibrated box. Thus, we are able to establish the coordinate relation T_{AR2W} between $\{X_{ar} Y_{ar} Z_{ar}\}$ and $\{X_w Y_w Z_w\}$.

After observing the calibration box, the AR wearable device can capture and calculate the position of the four corner points (P_1, P_2, P_3, P_4) of the upper surface in the coordinate $\{X_{ar} Y_{ar} Z_{ar}\}$. With the fingertip of the user's index finger touching the four corner points, the position (P_1', P_2', P_3', P_4') in the coordinate $\{X_l Y_l Z_l\}$ can be calculated by Leap Motion. Therefore, the following coordinate relations are established:

$$\left[\, P_1^{AR}\ P_2^{AR}\ P_3^{AR}\ P_4^{AR} \,\right] = T_{L2AR} \cdot \left[\, P_1^{L}\ P_2^{L}\ P_3^{L}\ P_4^{L} \,\right] \tag{4}$$

with

$$P_n^{AR} = [P_{x,n}^{AR}, P_{y,n}^{AR}, P_{y,n}^{AR}]^T, P_n^L = [P_{x,n}^L, P_{y,n}^L, P_{z,n}^L]^T, \quad n = 1, 2, 3, 4 \tag{5}$$

where $[P_{x,n}^{AR}, P_{y,n}^{AR}, P_{y,n}^{AR}]^T$ is the coordinate of the point n in the coordinate $\{X_{ar} Y_{ar} Z_{ar}\}$; $[P_{x,n}^L, P_{y,n}^L, P_{z,n}^L]^T$ is the coordinate of the point n in the coordinate $\{X_l Y_l Z_l\}$; T_{L2AR} represents the transformation matrix from the Leap Motion coordinate system to the AR wearable device coordinate system. In order to acquire a more accurate result, we move the box and repeat the above process. The least square method [21] is used to calibrate T_{L2AR}. Therefore, T_{L2W} can be obtained as follows:

$$T_{L2W} = T_{L2AR} \cdot T_{AR2W} \tag{6}$$

B. INTERACTIVE DETECTION

To achieve the natural interactive effective, the bare hand is allowed to direct touch or interact with the virtual robots. Such an accurate interaction can be implemented by detection collision techniques. In [31], the continuous collision detection (CCD) algorithm with two moving ellipsoids is employed to execute the interactive detection. Reference [32] presented a fast CCD algorithm for general rigid and articulated models based on conservative advancement. However, this paper only focuses on the collision detection between the hand and the virtual robots. To achieve the real-time interactive performance, we develop a simple collision detection method as follows.

First, the virtual model of the human hand is built in the virtual scene by the AR wearable device to realize the interaction between the hand and virtual robots. The hand skeleton diagram can be obtained by the Leap Motion and

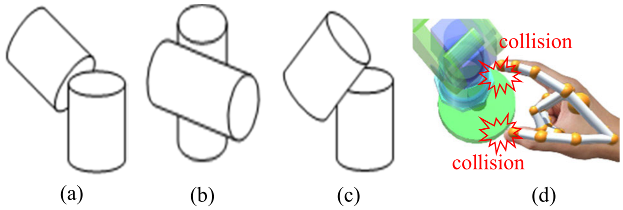


FIGURE 5. Collisions between the operator's hand and the virtual objects.

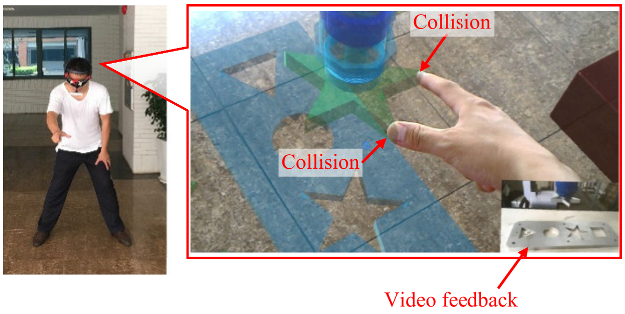


FIGURE 6. The application scenario of collision detection.

the hand's adjacent joint can be regarded as cylinders. Thus, the virtual hand consists of many cylinders shown in Fig. 5(d). Those cylinders are used as the bounding boxes of the interactive detection between the hand and the virtual robots for improving the detection efficiency. Furthermore, the virtual robots have been established beforehand and they can be seen as a combination of geometries.

Second, from the above analysis, we can effectively approximate the interactive collision between the virtual hand and the virtual robots as the collision between the cylinders and the geometries. Based on the relative positions between the hand and virtual robots, Figs. 5(a)-5(c) show three types of different interactive collisions between them including bottom-intersection, side-quadrature, and side-heterotrophic. The details involving the various collision detection of the cylinder can be referred to [33]. By detecting the interactive collision between the hand and robots, the operator can apply the bare hand to guide the virtual robot work.

As an example, Fig. 6 shows the application scenario of the operator guiding the virtual robot. Specifically, the operator can not only observe the virtual robots in the screen of the AR wearable device, but also see his or her real hand because of the transparent screen of the AR wearable device. When the Leap Motion detects the motion of the real hand and transfers its motion to the virtual hand of the AR wearable device, the virtual hand can guide or interact with the virtual robots by the interactive detection algorithm. Particularly, the virtual hand is not displayed in the screen of the AR wearable device so that the operator can only observe the real hand directly interacting with the virtual robots. In fact, the real hand cannot directly interact with the virtual hand in the actual interactive process.

In brief, such a virtual-real fusion method can provide an extremely realistic immersion for the interaction between the hand and the virtual robots.

IV. POSITION AND ORIENTATION ESTIMATION

In the natural interaction process, the operator generally often walks, shakes, and turns his/her head, which inevitably introduces noise into the Leap Motion and influences its measurement accuracy of the positions and orientation. To address this problem, the unscented Kalman filter (UKF) [22] and the improved particle filter (IPF) [23] are adopted in this paper.

A. POSITION ESTIMATION WITH UKF

In our previous work [19], the IKF has been used to estimate the position of the hand and eliminate noise effects. However, the IKF diverges in nonlinear systems over time. In general, the UKF is an effective method to estimate the positions in the nonlinear systems. Strictly speaking, the state-space model for estimating the motion of the hand is nonlinear. Therefore, we introduce the UKF to improve the accuracy of the positions of the hand.

Assume that the position of the hand measured by the Leap Motion is described as $[P_x^H, P_y^H, P_z^H]^T$ in the hand coordinate system. Based on the equations (1) and (2), the position of $[P_x^H, P_y^H, P_z^H]^T$ can be transformed into the world coordinate system as

$$\begin{aligned} P &= [P_x^W, P_y^W, P_z^W]^T = T_{L2W}\cdot T_{H2L}\cdot [P_x^H, P_y^H, P_z^H]^T \\ &= T_{H2W}.[P_x^H, P_y^H, P_z^H]^T \\ &= \begin{bmatrix} M_{H2W} & L_{H2W} \\ 0 & 1 \end{bmatrix} [P_x^H, P_y^H, P_z^H]^T \end{aligned} \tag{7}$$

where T_{H2W} means the transformation matrix from the hand coordinate system to the world coordinate system, and it consists of rotation matrix M_{H2W} and translation matrix L_{H2W}. At time t_k, M_{H2W} is described as

$$M_{H2W,k} = \begin{bmatrix} m_{X_x,k} & m_{Y_x,k} & m_{Z_x,k} \\ m_{X_y,k} & m_{Y_y,k} & m_{Z_y,k} \\ m_{X_z,k} & m_{Y_z,k} & m_{Z_z,k} \end{bmatrix} \tag{8}$$

where $m_{i_j,k} = \cos(\theta_{i_j})$; θ_{i_j} ($i, j \in (X, Y, Z)$) means the angle between the i-axis in the hand coordinate and the j-axis in the world coordinate.

In the world frame, P_{k+1} denotes the position of the hand at time t_{k+1} and it is calculated as:

$$p_{k+1} = p_k + v_k t + \frac{1}{2} a_k t^2 \tag{9}$$

where p_k, v_k and a_k are the position, velocity, and acceleration of the hand at time t_k.

The hand acceleration component on each axis of the world coordinate system can be calculated as follows:

$$\begin{aligned} a_k &= \begin{bmatrix} a_{k,x} \\ a_{k,y} \\ a_{k,z} \end{bmatrix} \\ &= \begin{bmatrix} m_{X_x,k}\cdot A_{x,k} + m_{Y_x,k}\cdot A_{y,k} + m_{Z_x,k}\cdot A_{z,k} \\ m_{X_y,k}\cdot A_{x,k} + m_{Y_y,k}\cdot A_{y,k} + m_{Z_y,k}\cdot A_{z,k} \\ m_{X_z,k}\cdot A_{x,k} + m_{Y_z,k}\cdot A_{y,k} + m_{Z_z,k}\cdot A_{z,k} - |g_l| \end{bmatrix} \end{aligned} \tag{10}$$

where $|g_l|$ stands for the magnitude of the local gravity vector and $(A_{x,k}, A_{y,k}, A_{z,k})$ is the acceleration measurement component on each axis of the hand coordinate system at time t_k.

Then, the velocity component $(v_{k,x}, v_{k,y}, v_{k,z})$ on each axis of the world coordinate system is described as:

$$v_k = \begin{bmatrix} v_{k,x} \\ v_{k,y} \\ v_{k,z} \end{bmatrix} = \begin{bmatrix} v_{k-1,x} + a_{k-1,x} \cdot t \\ v_{k-1,y} + a_{k-1,y} \cdot t \\ v_{k-1,z} + a_{k-1,z} \cdot t \end{bmatrix} \tag{11}$$

Based on (9), (10) and (11), we define the hand-position state x_k estimated by the UKF at timet_k as:

$$x_k{=}[p_{x,k}, v_{x,k}, A_{x,k}, p_{y,k}, v_{y,k}, A_{y,k}, p_{z,k}, v_{z,k}, A_{z,k}]^T \tag{12}$$

where $p\ i{,}k$, $v\ i{,}k$, and $A\ i{,}k$ represent the position, velocity, and acceleration estimation of the hand in i axis ($i = X$, Y, or, Z).

According to reference [22], the state-space model of the human hand can be built as follows:

$$\begin{cases} x_k = f(x_{k-1}, u_{k-1}) = \Phi_k \cdot x_{k-1} + \Gamma_k + u_{k-1} \\ y_k = h(x_k, w_k) = H_k(x_k) + w_k \end{cases} \tag{13}$$

where x_k and y_k are the state vector and the measurement vector at time t_k, respectively; $f(\cdot)$ and $h(\cdot)$ are the state transition function and the observation function, respectively and they are highly non-linear; u_{k-1} and w_k represent the process noise and the observation noise, respectively and they are the independent Gaussian white noise; Φ_k, H_k, and Γ_k are the state transition matrix, the observation matrix, and the system input matrix, respectively.

According to (9), (10) and (11), the state transition matrix Φ_k can be given as

$$\Phi_k = \begin{bmatrix} 1 & t & m_{X_x,k-1} \cdot t^2/2 & 0 & 0 & m_{Y_x,k-1} \cdot t^2/2 & 0 & 0 & m_{Z_x,k-1} \cdot t^2/2 \\ 0 & 1 & m_{X_x,k-1} \cdot t & 0 & 0 & m_{Y_x,k-1} \cdot t & 0 & 0 & m_{Z_x,k-1} \cdot t \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & m_{X_y,k-1} \cdot t^2/2 & 1 & t & m_{Y_y,k-1} \cdot t^2/2 & 0 & 0 & m_{Z_y} \cdot t^2/2 \\ 0 & 0 & m_{X_y,k-1} \cdot t & 0 & 1 & m_{Y_y,k-1} \cdot t & 0 & 0 & m_{Z_y,k-1} \cdot t \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & m_{X_z,k-1} \cdot t^2/2 & 0 & t & m_{Y_z,k-1} \cdot t^2/2 & 1 & t & m_{Z_z,k-1} \cdot t^2/2 \\ 0 & 0 & m_{X_z,k-1} \cdot t & 0 & 0 & m_{Y_z,k-1} \cdot t & 0 & 1 & m_{Z_z,k-1} \cdot t \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{14}$$

Because the gravity vector can be predetermined, the acceleration measurements are affected by the gravitational force. The Z axis of the world frame is parallel to the gravity vector so that the system input matrix is written as

$$\Gamma_k = [0, 0, 0, 0, 0, 0, -|g_l| \cdot t^2/2, -|g_l| \cdot t, 0]^T \tag{15}$$

The Leap Motion sensor is calibrated and initialized, so the observation function H_k is expressed as:

$$H_k(x_k) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot x_k \tag{16}$$

Therefore, the estimated hand-position state x_k at time t_k can be determined by the following steps:

1) Initialization

$$\hat{x}_0 = E\,[x_0] \tag{17}$$

$$P_0 = E\left[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T\right] \tag{18}$$

where $\hat{x}_0$ and P_0 are the mean and covariance of the initial state vector x_0, respectively.

2) Building sigma points

A group of $2n{+}1$ sigma point can be generated by the following formulas (19):

$$\begin{cases} \chi_{k-1}^0 = \hat{x}_{k-1} \\ \chi_{k-1}^i = \hat{x}_{k-1} + \left(\sqrt{(n+\lambda)P_{k-1}}\right)_i, i = i, \ldots, n \\ \chi_{k-1}^i = \hat{x}_{k-1} - \left(\sqrt{(n+\lambda)P_{k-1}}\right)_i, i = n+1, \ldots, 2n \end{cases} \tag{19}$$

with

$$\lambda = \alpha^2(n+\kappa) - n \tag{20}$$

where χ_{k-1}^i represent the i^{th} sigma points, i =0,…..,2n; $\hat{x}_{k-1}$ and P_{k-1} are the estimated state vector and covariance of the process noise at time t_{k-1}; $\left(\sqrt{(n+\lambda)P_{k-1}}\right)_i$ represents the i^{th} column of the matrix square root of $\left(\sqrt{(n+\lambda)P_{k-1}}\right)_i$; n is the dimension of the estimated sate vector $\hat{x}_{k-1}$; constants α and κ are employed to manipulate the spread of the sigma points.

3) Time update

The sigma points can be propagated and projected via the transition function f and the observation function h respectively as follows:

$$\chi_{k|k-1}^i = f(\chi_{k-1}^i), i = 0, \ldots.., 2n \tag{21}$$

$$y_{k|k-1}^i = h(\chi_{k|k-1}^i), i = 0, \ldots.., 2n \tag{22}$$

The weighted sigma points can be recombined to generate the following estimated state vector $\hat{x}_{k|k-1}$ and measurement vector $\hat{y}_{k|k-1}$, respectively:

$$\hat{x}_{k|k-1} = \sum_{i=0}^{2n} w_i^{(m)} \cdot \chi_{k|k-1}^i \tag{23}$$

$$\hat{y}_{k|k-1} = \sum_{i=0}^{2n} w_i^{(m)} \cdot y_{k|k-1}^i \tag{24}$$

Based on equations (21)-(24), the predicted state covariance P_k^{xx} and measurement covarianceP_k^{yy} can be obtained, separately as follows:

$$P_k^{xx} = \sum_{i=0}^{2n} w_i^{(c)} (\chi_{k|k-1}^i - \hat{x}_{k|k-1})(\chi_{k|k-1}^i - \hat{x}_{k|k-1})^T + Q_k \tag{25}$$

$$P_k^{yy} = \sum_{i=0}^{2n} w_i^{(c)} (y_{k|k-1}^i - \hat{y}_{k|k-1})(y_{k|k-1}^i - \hat{y}_{k|k-1})^T + R_k \tag{26}$$

where Q_k and R_k denote the covariance matrix of the process and observation noise, respectively; the weights of the state and covariance are computed by:

$$\begin{cases} w_0^{(m)} = \lambda/(n+\lambda) \\ w_0^{(c)} = \lambda/(n+\lambda) + (1-\alpha^2+\beta) \\ w_i^{(m)} = w_i^{(c)} = 1/[2(n+\lambda)],\, i = 1, \ldots, 2n \end{cases} \tag{27}$$

where β is linked with the distribution of the state vector.

4) Measurement update

According to equations (21)-(24) and (27), we can compute the state-measurement cross-covariance matrix P_k^{xy} as follows:

$$P_k^{xy} = \sum_{i=0}^{2n} w_i^{(c)} (\chi_{k|k-1}^i - \hat{x}_{k|k-1})(y_{k|k-1}^i - \hat{y}_{k|k-1})^T \tag{28}$$

Furthermore, according to equations (26) and (28), the optimal Kalman gain K_k can be computed as follows:

$$K_k = P_k^{xy} \cdot (P_k^{yy})^{-1} \tag{29}$$

Eventually, we can obtain the following estimated system state and covariance of the position of the hand, respectively:

$$\begin{cases} \hat{x}_k = \hat{x}_{k|k-1} + K_k(y_k - \hat{y}_{k|k-1}) \\ P_k = P_k^{xx} - K_k P_k^{yy} K_k^T \end{cases} \tag{30}$$

B. ORIENTATION ESTIMATION USING IPF

In general, the orientations of the human hand can be described using the Euler angles (roll, pitch, and yaw) measured via the Leap Motion in its own coordinate system $\{X_l Y_l Z_l\}$. The roll, pitch, and yaw of the human hand are defined as the angle of the rotation around X_l, Y_l, and Z_l, respectively. Particularly, the factored quaternion algorithm (FQA) [25] is employed to assist in estimating the orientation of the hand. Therefore, the Euler angles of the human hand can be converted into a unit quaternion vector in FQA as follows:

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} \cos(\frac{\phi}{2})\cos(\frac{\theta}{2})\cos(\frac{\psi}{2}) + \sin(\frac{\phi}{2})\sin(\frac{\theta}{2})\sin(\frac{\psi}{2}) \\ \sin(\frac{\phi}{2})\cos(\frac{\theta}{2})\cos(\frac{\psi}{2}) - s\cos(\frac{\phi}{2})\sin(\frac{\theta}{2})\sin(\frac{\psi}{2}) \\ \cos(\frac{\phi}{2})\sin(\frac{\theta}{2})\cos(\frac{\psi}{2}) + \sin(\frac{\phi}{2})\cos(\frac{\theta}{2})\sin(\frac{\psi}{2}) \\ \cos(\frac{\phi}{2})\cos(\frac{\theta}{2})\sin(\frac{\psi}{2}) - \sin(\frac{\phi}{2})\sin(\frac{\theta}{2})\cos(\frac{\psi}{2}) \end{bmatrix} \tag{31}$$

where ϕ, θ, and Ψ stand for the roll, pitch and yaw of the Euler angles, respectively; q_0, q_1, q_2, and q_3 are the quaternion components that follows the equation below:

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \tag{32}$$

However, FQA is only suitable for situations where the hand remains stationary or moves slowly. In a dynamical case, FQA does not work well. Therefore, the particle filter (IPF) [19] algorithm with angular information is used to improve the FQA method. As a state estimator, the IPF can estimate the true posterior using a finite number of random state samples among their corresponding normalized weights [24]. At time t_k, the posterior density can be approximate as

$$p\,(x_k|\, z_{1:k}, u_{0:k-1}) \approx \sum\nolimits_{i=1}^{N} w_k^i \delta(x_k - x_k^i) \tag{33}$$

Here, $\delta(*)$, x_k^i, w_k^i, and N stand for the Dirac delta function, thei^{th} state particle at time t_k, the normalized weight of the i^{th} particle, and the number of samples, respectively.

The orientation of the human hand can be represented by using N particles; each particle is written as $x_{PF,k}^i = \left[\, q0_k^i \; q1_k^i \; q2_k^i \; q3_k^i \,\right]^T, i = 1,2\ldots., N$; $\left[\, q0_k^i \; q0_k^i \; q2_k^i \; q3_k^i \,\right]^T$ is a unit quaternion including four components and they follow equations (31) and (32).

The Leap Motion can provide the measurements for the angular. At timet_{k+1}, the quaternion components of each particle can be obtained by the following:

$$\begin{bmatrix} q0_{k+1}^i \\ q1_{k+1}^i \\ q2_{k+1}^i \\ q3_{k+1}^i \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 2 & -\omega_{x,k}\cdot t & -\omega_{y,k}\cdot t & -\omega_{z,k}\cdot t \\ \omega_{x,k}\cdot t & 2 & \omega_{z,k}\cdot t & -\omega_{y,k}\cdot t \\ \omega_{y,k}\cdot t & -\omega_{z,k}\cdot t & 2 & \omega_{x,k}\cdot t \\ \omega_{z,k}\cdot t & \omega_{y,k}\cdot t & -\omega_{x,k}\cdot t & 2 \end{bmatrix} \cdot \begin{bmatrix} q0_k^i \\ q1_k^i \\ q2_k^i \\ q3_k^i \end{bmatrix} \tag{34}$$

where t and $\omega_{axis,k}$ means the sampling time and the angular velocity component measured by the Leap Motion in the corresponding coordinate axis, respectively.

The rotation matrix ${}_b^f C_k^i$ of the i^{th} particle from the hand coordinate to the world coordinate at time t_k is demonstrated in [26]. Therefore, the acceleration${}^f\dot{V}$ of each particle can be written as:

$${}^f\dot{V} = {}_b^f C \cdot {}^b A + {}^f g \tag{35}$$

where ${}^f V$ represents the velocity in the world frame, ${}_b^f C$ is given in [26], ${}^b A$ characterizes the acceleration measurement in the hand coordinate, and${}^f g$ denotes the local gravity vector. The acceleration ${}^f\dot{V}$ in the world coordinate is calculated with a large error if there is a large error in the rotation matrix ${}_b^f C_k^i$. Therefore, the weight of each particle can be assigned by the accumulated differences between the computed position of each particle and the estimated position via Kalman Filter (KF). The position differences are illustrated as follows:

$$D_s^i = \sum_{j=(s-1)\cdot M_s+1}^{M_s\cdot s} \Big\{ (x_{p,j}^i - x_{k,j}^i)^2 + (y_{p,j}^i - y_{k,j}^i)^2 + (z_{p,j}^i - z_{k,j}^i)^2 \Big\} \tag{36}$$

Here, $Di\ s$ denotes the accumulated position difference of the i^{th} particle at the s^{th} orientation iteration and $M_s = \Delta T_s/t$; $x^i_{p,j}$, $y^i_{p,j}$, and $z^i_{p,j}$ are the positions of the i^{th} orientation particle in axes X, Y, and Z at time t_j, respectively and they can be computed with the acceleration calculation from (35) without utilizing position measurement information; $x^i_{k,j}$, $y^i_{k,j}$, and $z^i_{k,j}$ are the positions of the i^{th} particle in axes X, Y, and Z, respectively at time t_j and they are estimated by KF. For a given particle, the smaller the $\mathrm{PE}^{\mathrm{i}}_{\mathrm{s}}D^i_s$ is, the higher probability that it represents the correct orientation. From the $\mathrm{PE}^{\mathrm{i}}_{\mathrm{s}}$ value of D^i_s, the weight of the i^{th} particle is resampled during each period of ΔT_s as follows:

$$w^i_k \propto \exp\left(\frac{-\left(D^i_s - \arg\min(D^i_s)\right)^2}{2 \times \left(\sigma\left(D^i_s\right)\right)^2}\right) \tag{37}$$

where w^i_k is the normalized weight; argmin(D^i_s) is defined as the most probable value.

C. ELIMINATE THE EFFECT OF SHAKING HEAD

To obtain the natural interactive pattern, the operators should be allowed to shake their heads, nod their heads, or face another orientation. However, such behaviors may result in some inaccurate or even false detections of the Leap Motion. This can be attributed to such a fact that the Leap Motion fixed on the head only detects the movement of the human hand relative to itself so that the movement of the head can introduce the relative deviation into its measurement.

Here, we employ an Inertial Measurement Unit (IMU) to distinguish the motion of the head. Once the measured motion data between IMU and Leap Motion is significantly different, we think the head is moving. Then the motion information of the human hand measured by the Leap Motion will be discarded. In this way, we can eliminate the impact of human head motion on the measurement accuracy of the Leap Motion.

Fig. 3(a) shows that this IMU worn on the center of the palm can detect the speed of the hand in the hand coordinate system $\{X_h Y_h Z_h\}$, which is then transformed into the Leap Motion coordinate system $\{X_l Y_l Z_l\}$ and expressed as $v_h = \left[\, v_{hx}\ v_{hy}\ v_{hz} \,\right]^T$. At the same time, the Leap Motion can also do that in $\{X_l Y_l Z_l\}$ as $v_l = \left[\, v_{lx}\ v_{ly}\ v_{lz} \,\right]^T$. Because the Leap Motion has a higher sampling rate of the hand-motion than IMU, it is reasonable to view v_l as a fitting curve. Next step is just to make a regression analysis in vector space between v_l and v_h as time grows. The curve keeps updating once new data arrives from IMU. New frame is always added to analysis once a system time is set.

Assuming that the IMU provides n measured values $\{v_{h1}, v_{h2} \ldots, v_{hn}\}$ and their corresponding prospect values are defined as $\{v_{l1}, v_{l2} \ldots, v_{ln}\}$. According to the regression analysis in statistics [27], the residual between the two sets is defined as$e_i = v_{li} - v_{hi}$, i =1, 2,..., n. The average measurement of the Leap Motion is written as

$$\overline{v_l} = \frac{1}{n}\sum\nolimits_{i=1}^{n} v_{li} \tag{38}$$

The sum of squares of the Leap Motion values is given as

$$SS_{tot} = \sum_{i=1}^{n} (v_{li} - \overline{v_l})^2 \tag{39}$$

The regression sum of squares of the IMU values is defined as

$$SS_{reg} = \sum_{i=1}^{n} (v_{hi} - \overline{v_l})^2 \tag{40}$$

The residual sum of squares of the two sets of data is

$$SS_{res} = \sum_{i=1}^{n} (v_{li} - v_{hi})^2 = \sum_{i=1}^{n} e_i^2 \tag{41}$$

Thus, the decision coefficient R^2 is employed to determine whether the fitting result is effective as follows:

$$R^2 = \frac{SS_{reg}}{SS_{tot}} = 1 - \frac{SS_{res}}{SS_{tot}} \tag{42}$$

Note that the closer the value of R^2 is to 1, the better the fitting result. In other words, when the value of R^2 is close to 1, we deem that there is no movement of the head. On the other hand, when the value of R^2 is much less than 1, the data involving the speed of the hands detected by the Leap Motion should be discarded, and the velocity data measured by IMU should be adopted.

V. MULTIMODAL INSTRUCTION GENERATION

Either using the gesture or the speech cannot efficiently convey an interactive instruction to the robot. For example, considering one instruction that makes the robot move in the left, the speech is not as effective as the gesture. Conversely, taking into account the other that enables the robot to move 10 cm in the left, the speech is more accurate than the gesture. In fact, the combination of the speech and gesture has advantages over one of them. Therefore, the combination is converted into the multimodal instruction text to guide the virtual robots. Meanwhile, the motion of the virtual robots is also passed to the real robots via the internet for tele-operating the real robots in the remote environment.

When the input is the gesture, the Leap Motion can first recognize it by estimating the position and orientation of the hand. This gesture can be further translated into the gesture text. On the other hand, while the input is the speech, a built-in microphone array in the AR wearable device is employed to receive the speech and a Microsoft Speech SDK translates the speech into the speech text. Eventually, the gesture text and speech text is fused and converted into a vector. Its features are extracted for classification by Term Frequency-Inverse Document Frequency (TF-IDF), detailed in [28]. The maximum entropy model is adopted as the classifier to classify the gestures and speech into the corresponding instruction, illustrated in [29].

Table 1 shows some examples of mapping natural interaction instructions into robot commands. We employ four

TABLE 1. Example of mapping of natural interaction instructions and robot movement commands.

Type	Speech	Gesture	Instructions			
			I_{op}	I_{dir}	I_{val}	I_{unit}
Simple speech command	"Pause"	--	--	--	0	--
Complex speech command	"Move 10 cm in the left (L)"	--	Move	L	10	cm
Speech and Gesture (static)	"Move 5 cm in this direction"	☞ Direction: $\vec{D}$	Move	$\vec{D}$	5	cm
Speech and Gesture (dynamic)	"Follow my hand"	∿ $D_0, D_1, \ldots, D_n$	Move	$\overrightarrow{D_0 - D_{current}}, \overrightarrow{D_1 - D_0}, \ldots \overrightarrow{D_n - D_{n-1}}$	$\|D_0\text{-}D_{current}\|, \|D_1\text{-}D_0\|, \ldots \|D_n\text{-}D_{n-1}\|$	cm

attributes [I_{op},I_{dir}, I_{val}, I_{unit}] to characterize an interactive instruction. I_{op} represents the motion of the robot such as "Move"; I_{dir} indicates the direction of the gesture; I_{val} and I_{unit} represent the distance the robot moves and its corresponding unit, respectively. For a simple speech instruction like "PAUSE", we only need to set the value of I_{val}. As for a complex speech instruction, for example, when the operator says "Move 10 cm in the left", the instruction is set as [I_{op} = MOVE, I_{dir} = L,I_{val} = 10, I_{unit} = cm]. L represents the left direction in the text.

Furthermore, we also consider an instruction involving both the speech and the gesture. As an example, the operator says "Move 5 cm in this direction", and he/she also points at one direction with his/her gesture, simultaneously. In this case, the Leap Motion will first capture the gesture and convert it into the text "Direction $\vec{D}$"; subsequently, the speech and the gesture are fused into "Move 5 cm in this direction: $\vec{D}$"; finally, the instruction is set as [I_{op} = MOVE, I_{dir} = $\vec{D}$, I_{val} = 5, I_{unit} = cm]. Note that if the speech does not involve the specific distance information but "Follow my hand" with a set of continuous gestures, the instruction consists of some vectors and its four attributes listed in the fourth row of Table 1.

VI. EXPERIMENT

A. EXPERIMENT SETUP

During experiments, the operator with the AR wearable device (Holoens) stood outside the laboratory and employed the speech and gesture to implement the remote operation for two robots (GOOGOL GRB3016) in the laboratory. The Leap Motion (version I) and IMU (LPMS-B2, Bluetooth transmission) are employed to measure the hand's position and orientation, respectively. The communication bandwidth between the Hololens and remote robots is 45Kb/s, which leads to the delay of 1.5s-3s.

In experiments, we also tested our interface when there were noise and packet loss on the network. Some methods were used to deal with them.

1) NETWORK NOISE

For each packet, we set a check digit to check whether the data is contaminated by noise. If the check result of the receiver is incorrect, the packet is discarded and the sender is requested to resend the data.

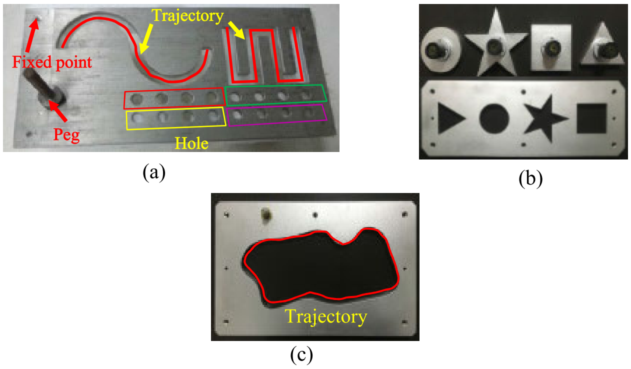


FIGURE 7. (a): Steel plate with trajectories, holes, and a peg. (b): The wire-cut shaped objects with their corresponding holes. (c): Irregular trajectory.

2) PACKET LOSS

When data loss occurs, the system will judge how much data is lost. If the number of consecutive drops is small, the lost data will be fitted according to the data received before and after. When more packets are lost, the sender will be asked for retransmission, and subsequent received data will be cached until the retransmission of the lost data is completed.

The above can be concluded as the following rule:

Rule 1:

If N≤N_0, fit the lost trajectory based on context. (N: the number of consecutive drops; N_0: the threshold representing the maximum number of consecutive packet drops that can be tolerated)

Otherwise:

Wait for retransmission.

B. EXPERIMENTAL TASK

First, the proposed method was compared with Method [18], Method [19], and Method [30] to validate its effectiveness on three experiments. The three experiments were conducted without noise and packet drop on the network.

In the first experiment, the operator remotely guided the robots with the fingertip to track the semicircular and the square trajectory on a steel plate shown in Fig. 7(a), where the width of the two trajectories is 20 mm; the radius of the semicircular trajectory is 100 mm; the size of the square trajectory is 110 mm × 140 mm [see Fig. 7(a)]. The operation error and operation time are used to evaluate the accuracy and efficiency of these compared methods. Note that the shafts of the robots are vertical by default. Furthermore, Fig. 7(c) shows that a steel plate involving an irregular trajectory was also used in this experiment to further test the stability of the above compared methods. Note that the external rectangle of the steel plate is 450 mm × 210 mm. Additionally, the operator regulated the end-effectors of the robots based on the speech and gesture while notably deviating from the reference trajectory.

The second experiment was a peg-into-hole task. Fig. 7(a) shows 16 round holes on the steel plate and their diameters are 20mm, 19mm, 18mm, and 17mm, corresponding to the red frame, yellow frame, green frame, and purple frame, respectively. The peg was fixed on the end effector (EE) of the robot and its diameter is 15mm. Similar to the first experiment, the operator with his/her finger guided the robot to insert the peg into the holes.

The third experiment was a more difficult and complex task for the robots. As shown in Fig. 7(b), the four different metal objects including the triangle, circle, star, and, square were cut from a steel plate, respectively. The circumradius of the triangle and star object are 50 mm and 70 mm, respectively; both the radius of the circle object and the side length of the square object are 40 mm. The operators guided the robots to manipulate the four metal objects into the holes of their corresponding shape. In this case, the operator employed his/her hands to guide the robots as shown in Fig. 6. In the third experiment, the rotation angles of the metal objects directly determine whether they can be successfully inserted in the corresponding holes so that the operations with more dimensions need to be introduced to improve the experimental accuracy.

In addition, in order to test the robustness of our method, it is employed to perform experiments 1 and 3 in the cases of network with noise and packet drop. For experiment 2 or 3, since the trajectory is mainly used to guide the robot end-effector to the steel plate, the number of lost packets has less effect on the insertion result, as shown in Figs. 8(a) and 8(c). Thus, N_0 can be set to a large value, such as 6. For experiment 1 (trajectory tracking), when the number of the consecutive drops is small, the data fit is still applicable (see Fig. 8(b)). However, when the number is large, the fitted trajectory will seriously deviate from the reference trajectory, as shown in Fig. 8(d). So N_0 should be set to a small value, such as 3. We simulated the network conditions with different noise and packet loss through network tool, and the operation time of task is recorded for analysis.

In all experiments, the operator controlled the virtual robot in real time and observed the execution of the virtual robot. Moreover, the motion of the real robot shot by the remote camera was also transmitted to the operator, which was displayed in the lower right corner of the glasses.

C. EXPERIMENT RESULTS

Fig. 9 shows intuitive comparisons between the proposed method, Method [18], Method [19], and Method [30] on the first experiment. Figs. 9(a) and 9(b) show the results of the robots tracking the semicircular and square trajectory with the four methods, respectively. Fig. 9(c) presents the tracking results in the irregular trajectory using the four methods. Particularly, Figs. 9(d) and 9(e) are the resolved trajectories of Fig. 9(c) in axes X and Y, respectively. From the results, an interesting observation is that our method constantly acquires the best tracking performance relative to the reference trajectory among all the compared methods.

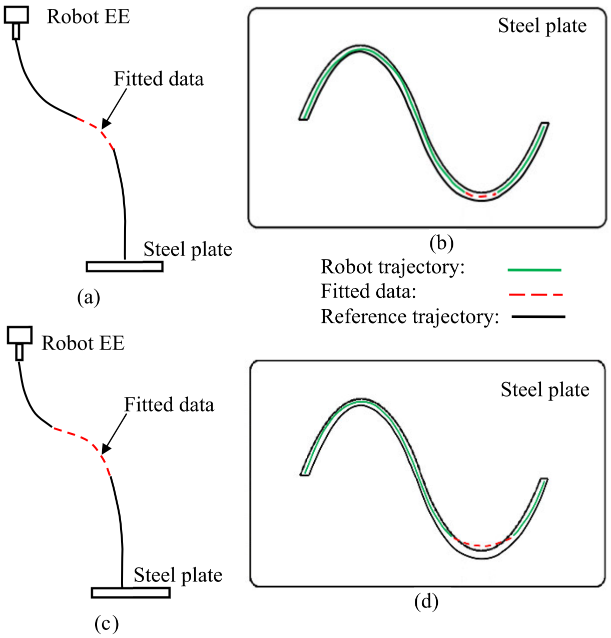


FIGURE 8. The fitting of lost data (a) Peg-into-hole experiment with few packets lost (b) Semicircular trajectory tracking with few packets lost (c) Peg-into-hole experiment with many packets lost (d) Semicircular trajectory tracking with many packets lost.

Fig. 10 shows the mean results between the proposed method, Method [18], Method [19], and Method [30] on the first experiment with the different number of trials. Specifically, Fig. 10(a) displays the tracking errors of the semicircular and square trajectory from the end effectors of the compared four methods in different number of trials; Fig. 10(c) does that of the irregular trajectory. Furthermore, Fig. 10(b) presents the average operating time of the compared four methods in the semicircular and square trajectory with different number of trials; Fig. 10(d) does that in the irregular trajectory. Interestingly, the proposed method consistently outperforms the other three methods on the above results.

In experiment 2, the operators use the proposed method, Method [18], Method [19], and Method [30] to control the robots' EE inserting the peg to 16 holes. In experiment 3, we adopt the above methods to carry out the tests of placing the different shaped objects into the corresponding targets. Figs. 11(a) and 11(b) show the average operation time for different methods on experiments 2 and 3 in the different number of trials, respectively. The polyline at the bottom represents our method, which has the shortest time to complete

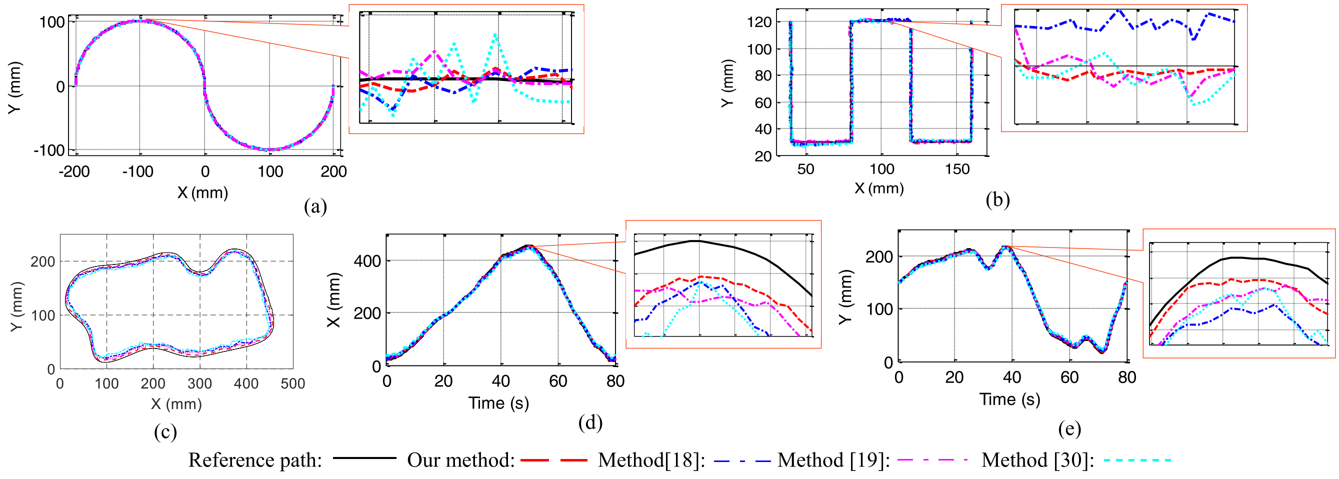


FIGURE 9. (a) Tracking results of semicircular trajectory. (b) Tracking results of square trajectory. (c) The trajectory of irregular trajectory. (d) The irregular trajectory in the x direction. (e) The irregular trajectory in the y direction.

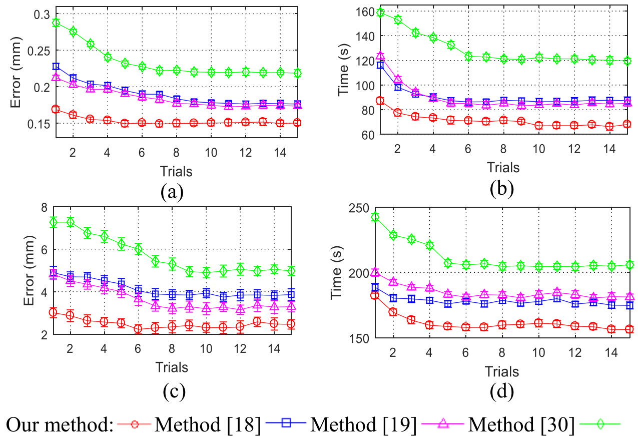


FIGURE 10. (a): Tracking error of the semicircular trajectory and the square trajectory, during 15 trials (b): Average operating time of the semicircular trajectory and the square trajectory, during 15 trials. (c): Tracking error of the irregular trajectory, during 15 trials. (d): Average operating time of the irregular trajectory, during 15 trials. The error bars show standard deviations.

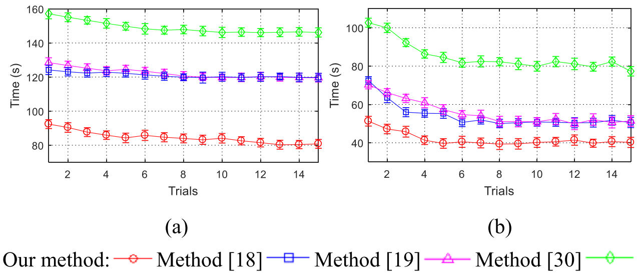


FIGURE 11. (a) Peg-into-hole time of experiment 2, (b) Placing workpieces time of experiment 3, during 15 trials. The error bars show standard deviations.

the corresponding tasks on experiments 2 and 3 among all compared methods. This indicates that our method is more efficient than Method [18], Method [19], and Method [30].

In addition, we executed a pairwise t-test [34] with a significance level of 0.05 for the average operating time and error between the proposed interface and the other three methods. We set the null hypothesis as "there is no significant difference between the proposed interface and methods [18], [19], [30] on experiments". According to the pairwise t-test theory, if $p \geq 0.05$, the null hypothesis holds; otherwise $p<0.05$, the null hypothesis does not hold. The proposed interface was paired with the other interfaces [18], [19], [30], and p was calculated based on the difference. A software called "Statistical Product and Service Solutions" [35] was used to achieve statistical results.

The statistical significance tests of experiments 1-3 are shown in Figure 12, where the bars represent the difference between our methods and the other three methods. The corresponding p value is marked above the bar. Note that $p<0.05$ denotes that there is a significant difference in the same experiment between our method and the other three methods. As shown in Fig. 12, all p values are less than 0.05, indicating that our method is superior to the other three methods in experiments 1-3.

Actually, in teleoperation, the remote may only provide a single perspective for the operator, resulting in a visual illusion. For instance, the subfigure on the bottom right corner of Fig. 13(a) shows a real image shot by the remote camera in the remote robot's task execution. We can observe from this subpicture that the shaft on the robot's end-effector seems to be aligned with the hole on the steel plate. However, there is still a real deviation between the shaft and the hole. Interestingly, Fig. 13(a) shows that the proposed interface combining augmented reality can observe the virtual robot's operation from different perspectives so that the operator can timely and accurately determine whether the shaft is aligned with the holes. Therefore, our method can effectively improve the operational precision in human-machine interaction.

Furthermore, we employed the hand motion trajectory distance on the three experiments to evaluate the performance of the proposed interface and the other three other methods. Specifically, the operator employed the different methods to

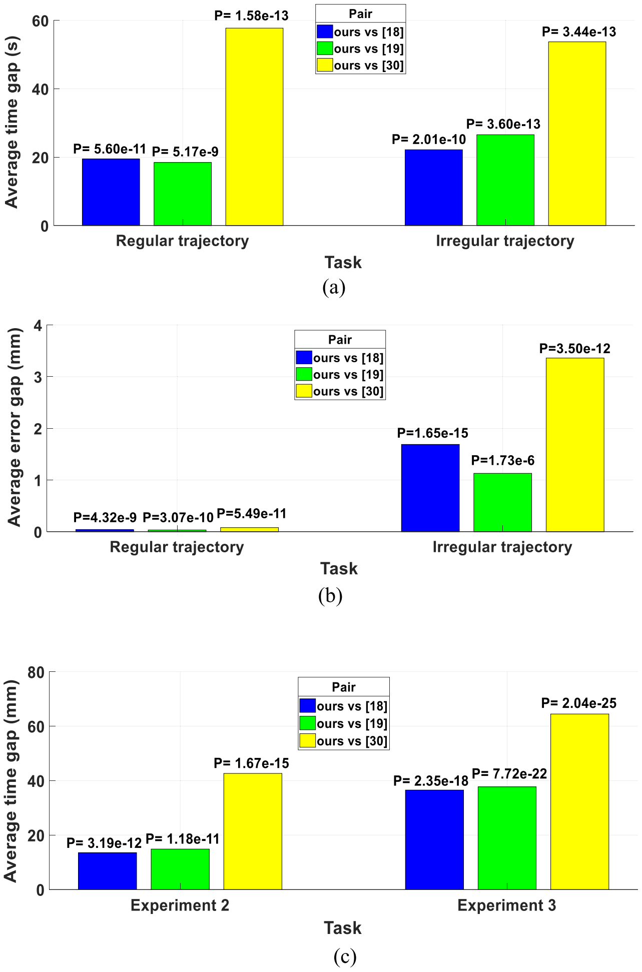


FIGURE 12. **Statistical significance tests between our method and three methods in three experiments. (a) Difference of average time in experiment 1. (b) Difference of average error in experiment 1. (c) Difference of average time in experiment 2 and 3.**

execute the three experiments, and the hand motion trajectory distance on each experiment could be recorded. The greater the hand motion trajectory distance, the lower the operational efficiency of this compared method. In the bar chart, each colored bar represents the average in 15 trials in one experiment. Each experiment was conducted in 15 trials. In each experiment, we provide the same task for different methods. We can clearly observe from Fig. 13(b) that the proposed interface can provide a shorter hand motion trajectory distance. This indicates that our method is more efficient than the other three methods.

In addition, we recorded the average completion time of placing workpieces, regular trajectory tracking, and irregular trajectory tracking to evaluate the performance of the proposed interface under different network conditions. Figure 14 shows the detailed results for the proposed interface. From Fig. 14(a), we can observe that with the increase of error rate, the average completion time on three tasks becomes longer. This is because more and more erroneous data needs to be retransmitted. From Fig. 14(b), we can see that in the experiment of placing workpieces, the packet loss rate has almost no effect on the average completion time of the task, and the increase in the packet loss rate

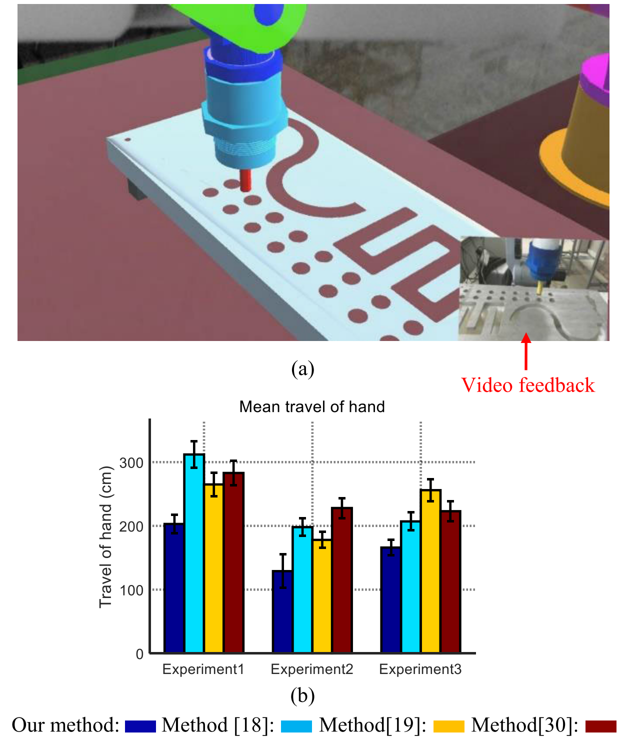


FIGURE 13. **(a) Image from the right rear direction. (b) Hand motion trajectory distance.**

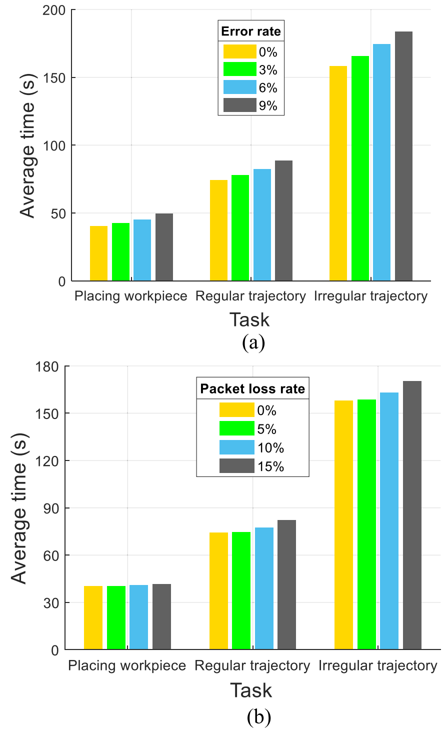


FIGURE 14. **(a) The results under different noises (Error rate represents the probability that the packet will be damaged by noise during transmission). (b) The results under different packet loss rates.**

does not cause a significant change in the average completion time. This is because the experiment of placing workpieces requires low similarity between the fitted trajectory and the original trajectory, and can tolerate more packet loss.

However, in the trajectory tracking tasks, excessive continuous packet loss leads to the intersection between the fitted trajectory and the trajectory on the steel plate, resulting in collision between the robotic end-effector and the steel plate. In Figure 14(b), as the packet loss rate increases, the average completion time of the two trajectory tracking experiments also increases. The reason is that the higher the packet loss rate, the more times the fitting fails, and the more data that needs to be retransmitted.

VII. DISCUSSION

This paper presented a natural mobile human-machine interactive method by using the augmented reality technology to avoid the significant borders between the virtual robots and the real hands of the operator. The Leap Motion can capture the motion of the hand and map it to the virtual world to guide the virtual robot. At the same time, the motion of the virtual robots is transmitted to the real robots in the remote environment through the Internet so that the operator can remotely guide the real robots. The experimental results confirm that the proposed approach can guide the robot to accomplish some complex tasks and it is more efficient and less time cost compared with the other methods [18], [19], [30].

The significant advantages of our approach are illustrated as follows: 1) The augmented reality technology can improve the operator's interactive immersion and obtain more natural human-machine interaction; 2) The Leap Motion is fastened on the AR wearable device so that the interactive space between the operators and the virtual robot has been greatly expanded. In this case, when moving, the operator interacts naturally with the virtual robot neither considering the matching relationship between the hands and the virtual robot nor doing the occlusion factor. 3) The UKF and IPF are introduced to improve the measurement accuracy of the proposed approach; TF-IDF and the maximum entropy model are employed to distinguish the speech and gesture instructions of the operators for accurately controlling the virtual robot.

However, the disadvantage of our approach is that the AR glasses are not very convenient to wear. Moreover, because interacting with the guided robot is virtual, the operator cannot achieve a real feedback force during the interactive process. In the future, we will introduce force feedback to improve the interactive perception of the operator in the virtual world. In addition, it is promising to extend the voice in the proposed interface to an intelligent voice-based synthetic assistant [36]. Such an assistant can be used to supervise the operations and provide assistance to voice users, thereby reducing the operational errors and improving task performance.

VIII. CONCLUSION

We present a natural human-machine interactive method using the AR wearable device and the Leap Motion, which covers the virtual-real fusion, the position and orientation estimation, and the multimodal instruction generation. The virtual-real fusion aims to implement the natural interaction between the real hand and the virtual robots and avoid the clear borders between them. Owing to the utilization of the UKF and IKF for the Leap Motion, the proposed method achieves the promising measure accuracy. By combining the gesture and speech, the multimodal instructions are established to further improve the interactive between human-machine. From the above reasons, the proposed method can provide effective natural human-machine interaction between the real and virtual world.

REFERENCES

[1] U. Keller, H. J. van Hedel, V. Klamroth-Marganska, and R. Riener, "ChARMin: The first actuated exoskeleton robot for pediatric arm rehabilitation," *IEEE/ASME Trans. Mechatronics*, vol. 21, no. 5, pp. 2201–2213, Oct. 2016.

[2] S. Oh, H. Woo, and K. Kong, "Frequency-shaped impedance control for safe human–robot interaction in reference tracking application," *IEEE/ASME Trans. Mechatronics*, vol. 19, no. 6, pp. 1907–1916, Dec. 2014.

[3] H. In, B. B. Kang, M. Sin, and K. J. Cho, "Exo-glove: A wearable robot for the hand with a soft tendon routing system," *IEEE Robot. Autom. Mag.*, vol. 22, no. 1, pp. 97–105, Mar. 2015.

[4] Z.-Q. Zhang and X. Meng, "Use of an inertial/magnetic sensor module for pedestrian tracking during normal walking," *IEEE Trans. Instrum. Meas.*, vol. 64, no. 3, pp. 776–783, Mar. 2015.

[5] G. Ligorio, E. Bergamini, I. Pasciuto, G. Vannozzi, A. Cappozzo, and A. M. Sabatini, "Assessing the performance of sensor fusion methods: Application to magnetic-inertial-based human body tracking," *Sensors*, vol. 16, no. 2, p. 153, Jan. 2016.

[6] S. Zhao, Z. Li, R. Cui, Y. Kang, F. Sun, and R. Song, "Brain-machine interfacing-based teleoperation of multiple coordinated mobile robots," *IEEE Trans. Ind. Electron.*, vol. 64, no. 6, pp. 5161–5170, Jun. 2017.

[7] X. Xu, A. Song, D. Ni, H. Li, P. Xiong, and C. Zhu, "Visual-haptic aid teleoperation based on 3-D environment modeling and updating," *IEEE Trans. Ind. Electron.*, vol. 63, no. 10, pp. 6419–6428, Oct. 2016.

[8] J. Rebelo, T. Sednaoui, E. B. den Exter, T. Krueger, and A. Schiele, "Bilateral robot teleoperation: A wearable arm exoskeleton featuring an intuitive user interface," *IEEE Robot. Autom. Mag.*, vol. 21, no. 4, pp. 62–69, Dec. 2014.

[9] X. Hou and R. Mahony, "Dynamic kinesthetic boundary for haptic teleoperation of VTOL aerial robots in complex environments," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 5, pp. 694–705, May 2016.

[10] A. Poncela and L. Gallardo-Estrella, "Command-based voice teleoperation of a mobile robot via a human-robot interface," *Robotica*, vol. 33, no. 1, pp. 1–18, Jan. 2015.

[11] Y. Liu and Y. Zhang, "Toward welding robot with human knowledge: A remotely-controlled approach," *IEEE Trans. Autom. Sci. Eng.*, vol. 12, no. 2, pp. 769–774, Apr. 2015.

[12] J. Kofman, X. Wu, T. J. Luu, and S. Verma, "Teleoperation of a robot manipulator using a vision-based human-robot interface," *IEEE Trans. Ind. Electron.*, vol. 52, no. 5, pp. 1206–1219, Oct. 2005.

[13] H.-Y. Liu, W. Wang, R. Wang, C. Tung, P. Wang, and I. Chang, "Image recognition and force measurement application in the humanoid robot imitation," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 1, pp. 149–161, Jan. 2012.

[14] K. Zinchenko, C.-Y. Wu, and K.-T. Song, "A study on speech recognition control for a surgical robot," *IEEE Trans. Ind. Informat.*, vol. 13, no. 2, pp. 607–615, Apr. 2017.

[15] K. Miura, A. Matsui, and S. Katsura, "Synthesis of motion-reproduction systems based on motion-copying system considering control stiffness," *IEEE/ASME Trans. Mechatronics*, vol. 21, no. 2, pp. 1015–1023, Apr. 2016.

[16] B. Browatzki, V. Tikhanoff, G. Metta, H. H. Bülthoff, and C. Wallraven, "Active in-hand object recognition on a humanoid robot," *IEEE Trans. Robot.*, vol. 30, no. 5, pp. 1260–1269, Oct. 2014.

[17] J. Kofman, S. Verma, and X. Wu, "Robot-manipulator teleoperation by markerless vision-based hand-arm tracking," *Int. J. Optomechatron.*, vol. 1, no. 3, pp. 331–357, Sep. 2007.

[18] G. Du and P. Zhang, "A markerless human–robot interface using particle filter and Kalman filter for dual robots," *IEEE Trans. Ind. Electron.*, vol. 62, no. 4, pp. 2257–2264, Apr. 2015.

[19] G. Du, P. Zhang, and X. Liu, "Markerless human-manipulator interface using leap motion with interval Kalman filter and improved particle filter," *IEEE Trans. Ind. Informat.*, vol. 12, no. 2, pp. 694–704, Apr. 2016.

[20] Vuforia. *Multi-Target-Guide, [EB/OL]*. Accessed: Oct. 20, 2018. [Online]. Available: https://library.vuforia.com/content/vuforia-library/en/articles/Training/Multi-Target-Guide.html

[21] Y. Chen and Y. Z. Shen, "A simplified model of three dimensional datum transformation adapted to big rotation angle," *J. Geomatics Inf. Sci. Wuhan Univ.*, vol. 29, no. 12, pp. 1101–1105, Dec. 2004.

[22] H. M. T. Menegaz, J. Y. Ishihara, G. A. Borges, and A. N. Vargas, "A systematization of the unscented Kalman filter theory," *IEEE Trans. Autom. Control*, vol. 60, no. 10, pp. 2538–2598, Oct. 2015.

[23] S.-H. P. Won, W. W. Melek, and F. Golnaraghi, "A Kalman/particle filter-based position and orientation estimation method using a position sensor/inertial measurement unit hybrid system," *IEEE Trans. Ind. Electron.*, vol. 57, no. 5, pp. 1787–1798, May 2010.

[24] K. Y. Chan, C. K. F. Yiu, T. S. Dillon, S. Nordholm, and S. H. Ling, "Enhancement of speech recognitions for control automation using an intelligent particle swarm optimization," *IEEE Trans. Ind. Informat.*, vol. 8, no. 4, pp. 869–879, Nov. 2012.

[25] X. Yun, E. R. Bachmann, and R. B. Mcghee, "A simplified quaternion-based algorithm for orientation estimation from earth gravity and magnetic field measurements," *IEEE Trans. Instrum. Meas.*, vol. 57, no. 3, pp. 638–650, Mar. 2008.

[26] S. H. P. Won, W. Melek, and F. Golnaraghi, "Fastening tool tracking system using a Kalman filter and particle filter combination," *Meas. Sci. Technol.*, vol. 22, no. 12, Nov. 2011, Art. no. 125108.

[27] J. L. Devore, *Probability and Statistics for Engineering and the Sciences*, 8th ed. Boston, MA, USA: Cengage Learning, 2011.

[28] W. Wang, Q. Zhao, and T. Zhu, "Research of natural language understanding in human-service robot interaction," *Microcomput. Appl.*, vol. 31, no. 3, pp. 45–49, Mar. 2015.

[29] G. Du, M. Chen, C. Liu, B. Zhang, and P. Zhang, "Online robot teaching with natural human–robot interaction," *IEEE Trans. Ind. Electron.*, vol. 65, no. 12, pp. 9571–9581, Dec. 2018.

[30] D. Kruse, J. T. Wen, and R. J. Radke, "A sensor-based dual-arm tele-robotic system," *IEEE Trans. Autom. Sci. Eng.*, vol. 12, no. 1, pp. 4–18, Jan. 2015.

[31] Y.-K. Choi, J.-W. Chang, W. Wang, M.-S. Kim, and G. Elber, "Continuous collision detection for ellipsoids," *IEEE Trans. Vis. Comput. Graphics*, vol. 15, no. 2, pp. 311–325, Mar./Apr. 2009.

[32] M. Tang, D. Manocha, and Y. J. Kim, "Hierarchical and controlled advancement for continuous collision detectionof rigid and articulated models," *IEEE Trans. Vis. Comput. Graphics*, vol. 20, no. 5, pp. 755–766, May 2014.

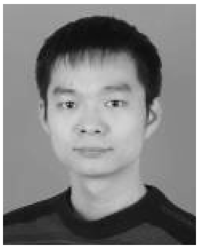
[33] S. Cheng and Y. Feng, "Fast collision detection algorithm of cylinders based on generatrices," *J. Jilin Univ. (Sci. Ed.)*, vol. 53, no. 2, pp. 291–296, Feb. 2015.

[34] H. A. David and J. L. Gunnink, "The paired t test under artificial pairing," *Amer. Statistician*, vol. 51, no. 1, pp. 9–12, Feb. 1997.

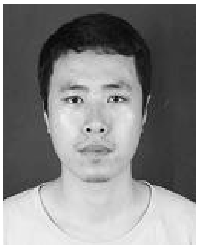
[35] IBM. (Feb. 22, 2019). *IBM SPSS Software*. [Online]. Available: https://www.ibm.com/analytics/spss-statistics-software

[36] P. Damacharla, P. Dhakal, S. Stumbo, A. Y. Javaid, S. Ganapathy, D. A. Malek, D. C. Hodge, and V. Devabhaktuni, "Effects of voice-based synthetic assistant on performance of emergency care provider in training," *Int. J. Artif. Intell. Educ.*, vol. 29, no. 1, pp. 122–143, Mar. 2019.

GUANGLONG DU received the Ph.D. degree in computer application technology from the South China University of Technology, Guangzhou, China, in 2013. He is currently an Associate Professor with the Computer Science and Engineering School, South China University of Technology. His research interests include intelligent robotics, human–computer interaction, artificial intelligence, and machine vision.



BO ZHANG received the B.S. degree from the School of Computer Science and Technology, Huazhong University of Science and Technology. He is currently pursuing the master's degree with the School of Computer Science and Engineering, South China University of Technology. His research interests include robot teaching, machine vision, and human–computer interaction.



CHUNQUAN LI received the B.Sc., M.Sc., and Ph.D. degrees from Nanchang University, Nanchang, China, in 2002, 2007, and 2015, respectively.

He has been with the School of Information Engineering, Nanchang University, since 2002, where he is currently an Associate Professor and a Young Scholar of Ganjiang River. He has published over 30 research articles. He is also a Visiting Professor with the Department of Systems and Computer Engineering, Carleton University, Ottawa, ON, Canada. His current research interests include computing intelligence, haptics, virtual surgery simulation, robotics, and their applications to biomedical engineering.

HUA YUAN received the B.S. degree from Harbin Engineering University, Harbin, China, and the M.S. and Ph.D. degrees from Sichuan University, Chengdu, China. She is currently an Associate Professor with the Computer Science and Engineering School, South China University of Technology, Guangzhou, China. Her research interests include image processing, video communication, big data processing, and the next generation network architecture.