Research Article

Research on Intention Flexible Mapping Algorithm for Elderly Escort Robot

Xiaoyu Qiu ^{1,2}, Zhiquan Feng ^{1,2}, Tao Xu ^{1,2}, Xiaohui Yang ^{1,2}, Ya Hou ^{1,2}
and Xin Zhang ^{1,2}

¹College of Information Science and Engineering, University of Jinan, Jinan, Shandong 250022, China

²Shandong Provincial Key Laboratory of Network Environment Intelligent Computing Technology, University of Jinan, Jinan, Shandong 250022, China

Correspondence should be addressed to Zhiquan Feng; ise_fengzq@ujn.edu.cn

Received 25 January 2021; Revised 30 March 2021; Accepted 9 May 2021; Published 26 May 2021

Academic Editor: Pengwei Wang

Copyright © 2021 Xiaoyu Qiu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the development of science and technology and the intensification of the aging of the world's population, elderly care robots have begun to enter people's lives. However, the current elderly care system lacks intelligence and is just a simple patchwork of some traditional elderly products, which is not conducive to the needs of the elderly for easy understanding and easy operation. Therefore, this paper proposes a flexible mapping algorithm (FMFD), that is, a gesture can correspond to a flexible mapping of multiple semantics in the same interactive context. First, combine the input gesture with the surrounding environment to establish the current interactive context. Secondly, when the user uses the same gesture to express different semantics, the feature differences formed due to different cognitions are used as the basis to realize the mapping from one gesture to multiple semantics. Finally, four commonly used gestures are designed to demonstrate the results of flexible mapping. Experiments show that compared with the traditional gesture-based human-computer interaction, the proposed flexible mapping scheme greatly reduces the number of gestures that users need to remember, improves the fault tolerance of gestures in the human-computer interaction process, and meets the concept of elderly caregiver robots.

1. Introduction

With the continuous development and progress of science and technology and medical care, my country's aging population has also continued to grow simultaneously, and elderly care has become a social problem. At the same time, with the increasing aging and declining birthrate, more and more elderly people are living in "empty nests" or living alone. According to statistics, more than 70% of the elderly have a sense of loneliness, especially those who are frail, disabled, living alone, elderly, and widowed. However, the development of China's elderly care industry is still in its infancy, facing serious shortages in the service supply side, etc. In recent years, with the rapid development of artificial intelligence and robotics technology, intelligent elderly care with artificial intelligence nursing robots as the core has shown a blowout research and development trend around the world [1].

The intelligent escort robot is a multifunctional service robot, which mainly assists the life of the elderly, with functions such as service, safety monitoring, human-computer interaction, and entertainment. With the continuous development of artificial intelligence, escort robots have begun to integrate into people's work and life. At present, the prospect of escort robots is broad, but there are still many problems, which are manifested in the following aspects. First, most escort robots are equivalent to a mobile tablet computer. Except being able to move, their operation methods and functions are not innovative and do not meet the requirements of ease of use for the elderly [2]. Second, some escort robots lack intelligent interactive functions, and the anti-interference of voice recognition is not strong; especially, in a more chaotic environment, the recognition accuracy is very low [3]. Third, most escort robots lack the ability to perceive the environment and cannot perceive their own environment in a timely and effective manner,

which leads to the problem of unity and limitation in the scope of use [4]. At the same time, when elderly people interact with robots with gestures, we noticed a problem. Traditional human-computer interaction is basically where the user inputs an instruction, and then, the robot performs related actions [5]. This one-to-one mapping method requires a lot of control gestures to achieve the purpose of perfect function. However, for the elderly, too many gesture commands will cause a serious memory burden, which is not conducive to the interaction between the elderly and the escort robot, and does not meet the requirements for ease of operation.

In summary, this article proposes a flexible intention mapping algorithm to escort robots for these situations. The algorithm takes the feature difference formed by different cognitions when the user uses the same gesture to express different semantics as the basis of flexible mapping, perceives the interactive context through target object detection, and uses visual-based gesture recognition technology to perform human-machine interaction which realizes the mapping from one gesture to multiple semantics in the same interactive context. The main advantage of this method is that it gets rid of the traditional human-computer interaction mode in which one instruction corresponds to one operation. In this way, only a few gestures are required to complete multiple functions. At the same time, the cognitive burden of the elderly is reduced, making the use of the escort robot easier and more flexible.

2. Related Work

Gesture is a technology that can provide a more natural, creative, and intuitive communication and interaction with computers. Therefore, gesture recognition technology has gradually become a research hotspot in the field of human-computer interaction [6], with the development of touch screen and sensor technology, gesture interaction has been widely used in various fields, such as intelligent teaching systems, assisted driving systems, and smart TV interactive systems [7].

Gesture recognition technology is mainly divided into sensor-based gesture recognition technology and vision-based gesture recognition technology. Chen et al. [8] (2018) applied a wearable gesture recognition sensor to a smart home, using a wearable wristband camera sensor to recognize hand trajectories, and using a dynamic time warping algorithm for 15 experimenters in 3 different scenarios. The 10 gestures and 1,350 gesture samples were classified, which achieved good results and realized the natural interaction between man and machine. Zhang et al. [9] proposed an adaptive update strategy for pressure parameters and developed a prototype system with a wearable gesture sensing device containing four pressure sensors and a corresponding algorithm framework, which realized real-time interaction based on gestures. Yu et al. [10] applied the deep belief network (DBN) to the field of Chinese sign language recognition based on wearable sensors, studied three sensor fusion strategies, and applied deep learning methods to the field of CSL recognition based on wearable sensors.

However, wearable sensors have the problem of inconvenience to wear, and the range of motion is limited by wireless signals or data lines and cannot be widely used [13, 14].

With the development of optical sensing and depth cameras, gesture recognition technology based on machine vision has received more attention. Wu et al. [13] proposed a multimodal gesture recognition method based on Hidden Markov Model (HMM). This method uses bone joint information, depth, and RGB images as multimodal input observation values, which are used to simultaneously segment and recognize gestures, which greatly improves the performance of gesture recognition. Dawar and Kehtarnavaz [14] applied gesture interaction to smart TV systems, and its research focuses on the user's preference for the gesture type and style of each operation command. Through experiments, the target user operation commands required by the smart TV are extracted, and the corresponding 9 gesture commands are selected to realize the gesture control on the smart TV. Zhang and Zhang [15] proposed a new human-machine 3DTV interaction system based on a set of simple freehand gestures and direct touch interaction. The combination of the system and virtual interface makes the user experience more comfortable. Jiang et al. [16] proposed a full-featured gesture control system (Givs) in response to the inconvenience and insecurity of touchpads in the driving environment. It uses the latest motion sensing technology, overcomes the technical limitations of motion sensors, realizes human-machine interaction in the driving environment, and improves driving convenience and safety. In summary, gesture recognition technology basically realizes the human-computer interaction in the current environment, which greatly facilitates people's lives. However, all the above methods are based on the mapping of one instruction for one semantic, and the use of the system requires a large number of control commands, and there is a heavy burden of memory.

So, in terms of the mapping between gestures and commands, the previous method can only map gestures to fixed commands. If the number of gestures is large, it may cause the gestures to conflict with each other in the feature space [17]. At present, most of the mapping methods are based on the frequency ratio to define the mapping between gestures and commands, that is, which kind of semantics is more probabilistic for a certain gesture. Therefore, the high-frequency gestures of each command are selected and estimated based on subjective measurement, which may lead to ignorance of meaningful gestures due to different cognitions [18]. This highlights the common problems of lack of intelligence and inflexible mapping in current gesture recognition technology, so that a gesture can only correspond to one semantic, which greatly increases the gesture vocabulary and brings heavy cognitive and operational burdens for human-computer interaction [19]. In order to achieve flexible mapping between gesture and semantics, Feng et al. [20] proposed flexible mapping using finger folding speed, gesture's global motion speed, trajectory diameter, movement time, and movement depth as semantic-oriented differential features, which achieved certain results and significantly reduced operators' cognition and operational

load. Feng et al. [21] targeted at two basic problems in intelligent interactive interface, namely, error in interface change caused by gesture recognition error and gesture recognition failure. An intelligent teaching interface based on gesture interaction is designed and implemented, and a flexible mapping interaction algorithm based on multiple gestures corresponding to the same semantics is proposed. This algorithm can effectively reduce the user load. And, it has been used in an intelligent teaching system interface based on gesture interaction. Therefore, it is feasible to realize a gesture expressing multiple semantics in the same interactive context through different characteristics of gestures. However, in the application, it is found that the mapping process of the above method is more complex, and there are problems of mapping delay and less mapping semantics.

In response to the above problems, this article aims to propose a flexible intention mapping algorithm for escort robots. The algorithm is based on the difference in characteristics due to different cognitions when the same gesture expresses different semantics in the same interactive situation and realizes the flexible mapping from one gesture to multiple semantics. At the same time, key and representative features are selected for flexible mapping to reduce the number of required features, achieve the purpose of real-time mapping and more mapping semantics, and get rid of the traditional human-computer interaction mode of one instruction corresponding to one operation, making the use of the elderly care robot more convenient and simple.

3. The Foundation and Overall Framework of Flexible Mapping

3.1. Cognitive Foundation. In order to find the feature differences of the same gesture when expressing different semantics, this paper conducts statistical analysis of data in the form of cognitive experiments. First of all, 10 elderly people were invited and the interaction scene was set as home care life. Second, ask them to wear data gloves and tracking wristbands and use the same gesture to express different semantics. Observation found that, for some different semantics, they can be expressed by the same gesture, for example, the same gesture can express hunger and stomachache; a gesture can express semantics such as drinking water, taking medicine, and pouring water. By observing these gestures, it is found that when the elderly use the same gestures to express different semantics, most of the differences are reflected in their cognition or behavior habits. For example, the gesture of “holding” can express the semantics of drinking water, taking medicine, pouring water, and grabbing. When expressing drinking water, the gesture is usually a natural squeeze but not clench; for taking medicine, most elderly people will express it by making a large fist to leave a gap the size of a pill; for pouring water, the angle of the fist is the same as the angle while drinking water, but people can distinguish by gesture direction: inward means drinking water and outward means pouring water. In summary, combined with the analysis and comparison of hand data, this paper proposes to use the curvature of the

fingers and the gesture direction as new gesture features to conduct flexible mapping research.

3.2. Flexible Mapping Overall Framework. Flexible mapping is the process of extracting features from visual information and transforming it into intent. This paper uses YOLOV3 to detect objects in the surrounding environment. Different scenarios are determined based on the set of detected objects. In order to solve the problem of low recognition rate under complex backgrounds and strong illumination changes, this study uses a combination of Gaussian skin color model and interframe difference method for gesture segmentation. This method can roughly find the position of the hand through motion detection and then perform fine segmentation, reducing the influence of the background on the gesture segmentation. In order to reduce the influence of noise on the image, this paper uses bilateral filtering to filter the binary image. Figure 1 shows the overall frame structure of flexible mapping.

The human-computer interaction process based on flexible mapping can be divided into three parts: information input, flexible mapping, and intention acquisition and task assignment. In the process of information input, use Kinect to obtain various data in the visual interactive scene. In the process of flexible mapping, first, perform gesture recognition on the depth image of the hand to obtain the gesture type. At the same time, the feature value of the hand data is extracted and the feature threshold value is compared to obtain the gesture feature value. Second, obtain various objects in the current scene and obtain the current interactive scene through target detection. Finally, the gesture feature value, gesture type, and interaction scene are combined, and the intention is understood by matching with the behavior-intention database. In the process of intention acquisition and task allocation, the feasibility analysis of the obtained user intentions will be carried out, and if feasible, the robot will be assigned to start the task to implement the elder care system.

4. Flexible Mapping Algorithm Based on Feature Difference (FMFD)

The flexible mapping algorithm based on feature differences (hereinafter referred to as FMFD) requires flexible one-to-many mapping between gestures and semantics. Therefore, it is first necessary to establish a gesture data set and a behavior-intent database. The gesture data set is composed of gesture serial number, gesture feature, and interaction context. The behavior-intention database is composed of conditions such as gesture serial number, semantics, finger curvature, and gesture direction. It describes the mapping relationship between gestures and semantics under certain conditions. The key of the FMFD algorithm is to compare the calculated feature value with the threshold value obtained by the cognitive experiment, so as to realize the gesture classification and complete the mapping. Therefore, this section first introduces the detection and extraction of gesture features and then uses the extracted gesture features

FIGURE 1: Flexible mapping overall framework.

to calculate finger curvature and gesture direction. Finally, the obtained feature numbers are combined into an intention determination set, and mapping is done through matching behavior-intent database.

4.1. Algorithm Framework. The flexible mapping process is shown in Figure 2. First, perform gesture recognition on the acquired gesture image to obtain the gesture type r . Secondly, perform fingertip detection and centroid detection on gesture images and then use the acquired center of mass and the position information of each fingertip to calculate the curvature of the finger and the displacement of the gesture. At the same time, according to the comparison of the threshold values of equations (1) and (2), the bending degree set of each finger $(h_1, h_2, h_3, h_4, h_5)$ and the gesture direction serial number d are obtained. Finally, combining these feature sequences with the gesture type r and the interaction scene s , we can get the intention determination set $I[r, (h_1, h_2, h_3, h_4, h_5), d, s]$ and combine it with the behavior-intention. The database is matched to obtain the user's intention and realize flexible mapping.

4.2. Gesture Recognition. In order to be able to apply the complex background and dark conditions, to meet the user's human-computer interaction needs. In this paper, the dynamic interactive gesture recognition algorithm based on Kinect was adopted to recognize gestures [22]. The method uses Kinect camera to obtain gesture images, and after gesture segmentation, the value of tangent angle of centroid

motion trajectory is used for uniform quantization encoding. By setting probability threshold model and encoding type, the undefined gesture is excluded, and the hidden Markov model is established to recognize dynamic gesture. The operation is simple. Finally, the gesture type r is obtained according to the gesture features.

4.3. Eigenvalue Calculation. The calculation of feature values is mainly divided into gesture curvature calculation and gesture direction calculation. Among them, the calculation of gesture curvature is particularly important. It is mainly divided into three parts: fingertip detection, centroid detection, and finger bending degree detection. The calculation of the gesture direction is mainly based on the movement distance and direction of the gesture contour centroid.

4.4. Fingertip Detection. Aiming at the current problems of fingertip detection [23, 24], this paper uses a combination of YCbCr color space [25] and the background difference method for gesture segmentation. This method can first find the position of the hand region from the image and then perform gesture segmentation and refinement in this area, thereby reducing the influence of complex background on gesture detection and recognition. The fingertip is detected by first searching for fingertip candidate points by curvature and then using the center of gravity distance method for fingertip candidate points, which can reduce the time and space complexity and the false recognition rate of fingertip detection.

Through the extraction of the gesture contour, it can be found that there are obvious curvature features not only at the fingertips but also at the joints, finger indentations, and wrist joints, as shown in Figure 3. Therefore, the points processed by curvature are called fingertip candidate points.

After finding the fingertip candidate points, the next step is to find the real fingertip point from these candidate points through the center of gravity distance method. For gestures with open fingers, the center of gravity distance method can find the fingertips by looking for the farthest point from the center of mass, but for those gestures with clenched fist, the distance from the hollow of the finger to the center of mass is greater than the distance from the fingertips to the center of mass. Therefore, this paper proposes a method to find fingertips through dynamic gestures. At the beginning of the gesture, the candidate fingertip points were selected once, and at the end of the gesture, the fingertip points were detected again, and the distance from the center of mass was calculated. If the distance from the center of mass remained unchanged, the point was the depression point or the node, which was deleted. Finally, the remaining points are the actual fingertip points (x_i, y_i) (Algorithm 1).

4.5. Centroid Detection. After obtaining the binary image of the gesture contour, in order to obtain the coordinates of the centroid, the contour of the gesture needs to be obtained first, that is, edge detection is performed. This paper uses the Canny algorithm [26] to detect the edge of the gesture binary

FIGURE 2: Flexible mapping process.

timage. The test results are shown in Figure 4. After the edge detection of gesture, the contour of gesture can be extracted and the contour moment and contour centroid can be calculated. In this paper, `findContoursz()` function in `opencv` is used to find the gesture contour. This method is stored in the form of point vector, that is, the point type vector is used to express the contour moment and contour centroid, which are convenient for the calculation of the following contour moment and contour centroid. The results of centroid detection are shown in Figure 5 and are recorded as $(\bar{x}, \bar{y})$.

4.6. Judgment of Finger Bending. According to the above process, the fingertip point (x_i, y_i) and the centroid point $(\bar{x}, \bar{y})$ can be obtained. In this paper, the Euclidean distance is used to express the distance between the fingertip point and the centroid point, so as to determine the curvature of

the finger. Calculate the distance L_i from each fingertip point to the center of mass point, as shown in Figure 6. The calculation formula is as follows:

$$L_i = \sqrt{(x_i - \bar{x})^2 + (y_i - \bar{y})^2}. \quad (1)$$

Suppose the finger curvature threshold is $T(t_1, t_2, t_3)$. Among them, t_1 is the minimum threshold and much smaller than t_2 , and t_2 and t_3 are intermediate thresholds. By comparing c with the finger bending threshold, the bending degree of each finger can be obtained. As shown in formula (2). When $L_i > t_3$, it means that the degree of finger bending is low, that is, slightly bent. When the value of L_i is between t_2 and t_3 , it means that the finger is normally bent. When $t_1 < L_i < t_2$, it means that the degree of finger bending is high. When $L_i < t_1$, it means that the fingers are completely bent, that is, a fist gesture. The threshold comparison process is as follows:

FIGURE 3: Fingertip candidate points.

- (1) Get a frame of image at the beginning of the gesture
- (2) While (point P_i is not empty)
 - (1) According to the step length k , points P_{i+k} and P_{i-k} are selected to form the vector P_1 and the vector P_2 .
 - (2) Find the curvature at point P_i according to the obtained vector and save it.
 - (3) Point P_i subscript moves down $i+1$
- (3) The curvature of each point is compared with the threshold T , the points with $K \geq T$ are taken out as fingertip candidate points, and the distance to the centroid point is calculated.
- (4) Set the delay time TD. If the gesture has no action within TD time, it is judged as the end of the gesture, and a frame of the image at the end of the gesture is obtained.
- (5) Fingertip detection and center of gravity distance calculation were carried out again on the acquired images, and the points with constant distance were eliminated.
- (6) Get real fingertips.

ALGORITHM 1: Fingertip selection.

$$\begin{cases} L > t_3, \\ L \in T(t_2, t_3), \\ t_1 < L_i < t_2, \\ L_i < t_1. \end{cases} \quad (2)$$

4.7. Gesture Direction Judgment. In the process of fingertip detection, the coordinates of each fingertip point and the center of mass at the beginning and end of the gesture are recorded, respectively. Therefore, this paper adopts the method of coordinate subtraction to determine whether the gesture is moving and the direction of the gesture. Set the movement determination threshold to ν . When the movement amount is less than ν , the gesture is determined to have no displacement, and when the movement amount is greater than ν , the gesture is determined to have displacement. When it is determined that the movement occurs, according to the coordinate subtraction and formula (3), the movement amount Δx and Δy of the centroid point in the x and y directions are obtained, respectively. Finally, the direction of

FIGURE 4: Gesture edge detection map.

gesture movement is determined according to the positive and negative of Δx and Δy . $\Delta x < 0$ indicates that the gesture moves to the side of the body, $\Delta x > 0$ indicates that the

FIGURE 5: Centroid detection result.

FIGURE 6: Distance between fingertip and centroid.

gesture moves away from the body, $\Delta y < 0$ indicates that the gesture moves upward, and $\Delta y > 0$ indicates that the gesture moves downward. The formula is defined as follows:

$$\begin{cases} |\text{final}.x - \text{start}.x| > \nu, \\ |\text{final}.y - \text{start}.y| > \nu, \\ \Delta x = \text{final}.x - \text{start}.x, \\ \Delta y = \text{final}.y - \text{start}.y, \end{cases} \quad (3)$$

among them, final represents the coordinates at the end of the gesture and start represents the coordinates at the beginning of the gesture. When the displacement of the centroid point on the x -axis or the y -axis is greater than ν , the calculation of Δx and Δy is performed, and the direction of the gesture movement is determined according to their positive or negative, and finally, the gesture movement direction d is obtained.

4.8. Flexible Mapping Algorithm. The design of flexible mapping algorithm is a process of one-to-many mapping between gestures and semantics. In this paper, the finger curvature of the gesture and the direction of gesture movement are used as the difference features at the same time. The intention determination set is obtained by combining the gesture recognition result and the feature difference and finally matches the user's real intention with the behavior-intention database. The algorithm is described as follows (Algorithm 2):

4.9. Analysis of Algorithms. The flexible mapping algorithm (FMFD) proposed in this paper realizes the flexible mapping from one gesture to multiple semantics in the same interactive context. On the one hand, compared with the traditional human-computer interaction model, such as the gesture smart TV system, the algorithm proposed in this

Input: Gesture instructions and objects in the surrounding environment

Output: User intent

- (1) Perform gesture segmentation and binarization of the input gesture instructions to obtain the outline of the gesture to prepare for fingertip detection and centroid detection.
- (2) Perform gesture recognition and target detection with visual information to obtain the user's current gesture type r and interaction scene s .
- (3) hile (gesture recognition starts, get the outline of the gesture):
 - {
 - (1) According to Algorithm 1, the fingertip coordinates (x_i, y_i) of the gesture are detected.
 - (2) The gesture contour moment was calculated according to the obtained gesture contour, and the gesture centroid $(\bar{x}, \bar{y})$ was obtained by centroid detection.
 - (3) According to the obtained coordinates of the fingertip point and the gesture centroid point, the distance L between each fingertip and the centroid is calculated by formula (1).
 - }
- (4) The obtained distance L between each fingertip point and the center of mass is compared with the threshold value through formula (2), so as to obtain the corresponding feature intention serial number $(h_1, h_2, h_3, h_4, h_5)$.
- (5) By tracking the centroid of the gesture, use formula (3) to subtract coordinates to obtain displacements $|\Delta x|$ and $|\Delta y|$.
- (6) While $(|\Delta x| > \text{Displacement judgment threshold } v \text{ or } |\Delta y| > \text{Displacement judgment threshold } v)$
 - {
 - (1) Determine the sign of Δx and Δy .
 - (2) Combine the determination results of Δx and Δy to obtain the gesture movement direction d
 - }
- (7) Combine the obtained finger curvature serial number, gesture direction serial number, gesture type serial number, and interaction scene serial number to form an intention determination set $I[r, (h_1, h_2, h_3, h_4, h_5), d, s]$. Finally, compare the intention determination set I with the behavior-intention database; if there is a match, the mapping is successful, and the user intention R is obtained. Otherwise, the mapping does not exist.

ALGORITHM 2: Flexible mapping algorithm.

paper not only ensures the accuracy of gesture recognition but also realizes the mapping of one instruction to multiple semantics, which greatly reduces the number of gestures. On the other hand, compared with other flexible mapping algorithms, the algorithm proposed in this paper reduces the number of required gesture features, realizes a gesture expressing multiple semantics only through the two features of finger curvature and gesture movement direction, improves the real-time performance of human-computer interaction, and reduces the time complexity. At the same time, although the number of gesture features used is reduced, the mapping flexibility of this algorithm has been improved, that is, using fewer gestures can express more semantics. In summary, the FMFD algorithm in this paper conforms to the design concept of flexible mapping and convenient use and solves the problems of lack of intelligence and inflexible mapping in human-computer interaction.

5. Experimental Results and Analysis

5.1. Experimental Setup. In the experimental operation, a Kinect camera was used, which can obtain depth images and color images of gestures. The computer used is configured with Intel(R) Core(TM) i7-4712MQ quad-core CPU, 2.30 GHz processor. In order to make the experiment process similar to the daily life of the elderly, this paper builds a simulated home environment, which is required to fit the size of the living room. There is a table in the environment, and there are experimental props such as water

cups, medicine bottles, and apples on the table, as shown in Figure 7. In the experiment, the robot used the humanoid intelligent robot Pepper developed by SoftBank. The transmission of computer instructions to the Pepper robot is realized through socket communication between Python3 and Python2, and various functions are completed by the robot.

In order to verify that the same gesture can express different semantics and whether the proposed algorithm can successfully perform flexible mapping, we went to the elderly activity center for in-depth research. After talking and observing with the elderly, there are gestures that express multiple semantics. According to the services and common gestures that the elderly need in their lives, this paper defines the four common gestures shown in Table 1 for experimental verification of flexible mapping.

5.2. Data Collection and Experimental Process

5.2.1. Data Collection. We invited 10 elderly people in the 60–70 age group to participate in the experiment. First, the subjects were asked to wear data gloves and hand-tracking wristbands, and then, each semantic gesture was repeated 10 times. The computer used the fingertips of the obtained gestures. The position data and motion data calculate the average sample of finger curvature and gesture direction threshold to generate a behavior-intention database. The data collection process is shown in Figure 8. Through the collection of experimental data, this paper obtains the

FIGURE 7: Experimental scene graph.

TABLE 1: Four commonly used gestures.

Gesture name	Description
G1	Make a fist of the five open fingers
G2	Grab with the thumb, index finger, and middle finger
G3	Fold the fingers of the open four fingers except the thumb to the center of mass
G4	Five fingers slightly bent posture

FIGURE 8: Data collection.

average data sample of the curvature of each finger and the direction of the gesture when a gesture, as shown in Table 2, expresses different semantics.

After the data collection was completed, the subjects were asked to use Kinect as a gesture input device to verify flexible mapping in a simulated home environment. This article sets up four life situations, requiring the experimenter to complete different semantic instructions through a gesture in each interactive situation. When the robot recognizes the intention, it will perform the corresponding action, remembering the robot action and the user's real intention. If they are the same, the mapping is successful, otherwise it is failed.

5.2.2. Experimental Process. In order to verify the algorithm, this article has conducted a lot of experiments. Due to the limited space of this article, 4 typical experiments are listed randomly. Using Kinect as the input device, each group of experiments was completed by 10 elderly people. Taking into account that a large number of experiments may cause tiredness of the elderly and affect the results of the experiment, we use ten gesture interactions as a group and complete an experiment in three times, so as to achieve each

TABLE 2: Gesture feature data table.

Gesture name	Gesture semantics	Finger curvature ($h_1, h_2, h_3, h_4, h_5,$)	Gesture direction
G1	Drink water (s1)	(67.4559, 43.5514, 32.1578, 44.4142, 44.0455)	$\Delta x > 0,$ $\Delta y > 0$
	Take medicine (s2)	(42.6891, 92.1126, 58.3479, 71.6812, 68.5976)	$\Delta x > 0,$ $\Delta y > 0$
	Pour water (s3)	(67.4559, 43.5514, 32.1578, 44.4142, 44.0455)	$\Delta x < 0,$ $\Delta y < 0$
G2	Shut down (s1)	(68.7510, 42.5803, 30.4489, 65.6564, 53.1332)	$\Delta x < 0,$ $\Delta y < 0$
	Reduce (s2)	(66.9641, 17.5849, 14.7684, 64.8448, 61.9186)	$\Delta x < 0,$ $\Delta y < 0$
G3	Go fast (s1)	(46.6453, 66.5543, 43.8976, 47.8731, 44.8385)	$\Delta x > 0,$ $\Delta y > 0$
	Walk slowly (s2)	(51.3915, 41.5673, 25.6352, 34.6403, 34.2732)	$\Delta x > 0,$ $\Delta y > 0$
G4	Hungry (s1)	(58.6360, 15.5277, 4.3419, 15.5310, 25.0435)	$\Delta x = 0,$ $\Delta y = 0$
	stomachache (s2)	(63.6511, 19.8596, 14.2232, 26.2701, 32.7044)	$\Delta x = 0,$ $\Delta y = 0$

semantic gesture repeating 30 times. For each scene, the elderly are required to use a designated gesture to express different semantics. The experimental results are as follows.

- (1) Experiment 1: use G1 to express drinking water, taking medicine or pouring water.

The interactive situation is set for the elderly who want to take medicine and need a robot to bring water and deliver medicine for them. The actions of the robot are programmed by Choregraphe software [27]. There is a water cup and a medicine box in this scene. First, the subject uses gesture G1 to express drinking water. After recognition, the robot will pick up the water cup and deliver it to the elderly. Then, use the gesture G1 to express the medicine again. After recognition, the robot will grab the medicine bottle and deliver it to the elderly. Finally, after taking the medicine, use gesture G1 to express the meaning of pouring water. The robot takes the cup from the old man and goes to the designated place to complete the pouring. When the recognition result of the gesture is that the user expresses the real intention, it is recorded as the mapping success. The mapping success rate is shown in Figure 9(a), and the experimental effect is shown in Figures 10(a) and 10(b). Experiments show that the mapping success rate of the FMFD algorithm has reached more than 94%. In this example, the average sample of the

FIGURE 9: (a) The mapping success rate when using G1 gestures to express the semantics of s1, s2, and s3. (b) The mapping success rate when using G2 gestures to express the semantics of s1 and s2. (c) The mapping success rate when using G3 gestures to express the semantics of s1 and s2. (d) The mapping success rate when using G4 gestures to express the semantics of s1 and s2.

FIGURE 10: (a) The rendering of the robot sending water cup. (b) The rendering of the robot sending medicine box. (c) The rendering of the robot's rapid advance. (d) The robot judges the experimenter's stomach pain and automatically takes the medicine.

bending degree of each finger when each experimenter expresses a certain semantics is obtained, as shown in Figure 11. Therefore, we design the flexible mapping as h_1 [28, 40, 45], h_3 [34, 56, 70], h_4 [46, 70, 73], and h_5 [46, 67, 70], for semantics s1 and s2, design $\Delta x > 0$ and $\Delta y > 0$; for Semantic s3, design $\Delta x < 0$ and $\Delta y < 0$.

- (2) Experiment 2: use G2 to express close or decrease
- Set the interactive situation as the old man is using the Pepper robot's tablet to watch a program. He can control the robot to turn off the tablet or reduce the volume through gesture G2. First, the experimenter reduces the volume through gesture G2. After recognition, the robot will reduce the volume by five units and give voice prompts. Then, the experimenter uses the G2 gesture to express the closing instruction. After the robot recognizes it, it will remind the user through voice and turn off the tablet after 5 seconds. During the experiment, each gesture is repeated 30 times. If the command executed by the

robot meets the operator's intention, it is recorded as a successful operation. The mapping success rate is shown in Figure 9(b). According to Table 2, we design the flexible mapping as h_1 [48, 67, 70], h_2 [19, 41, 44], h_3 [16, 29, 32], h_4 [36, 64, 67], h_5 [33, 52, 55], $\Delta x < 0$, and $\Delta y < 0$.

- (3) Experiment 3: use G3 to express fast or slow walking
- Set the interactive situation that the old man needs to call the robot to his side. This scene is a relatively empty space, as shown in the figure, to simulate the emergency call of the robot when the old man falls. First, the experimenter uses the gesture G3 to express the call command in the normal state, and the robot will move to the elderly at a normal speed after recognition. Second, the experimenter is required to express an emergency call for the robot to come to the side by hand gesture G3 in the simulated fall. The robot will move to the side of the elderly at the fastest speed after recognition. The mapping success rate is shown in Figure 9(c), and the experimental effect is

FIGURE 11: The curvature of each semantic finger of G1 gesture.

shown in Figure 10(c). According to Table 2, we design the flexible mapping as h_1 [23, 45, 48], h_2 [43, 65, 68], h_3 [27, 42, 45], h_4 [36, 46, 49], h_5 [36, 43, 46], $\Delta x > 0$, and $\Delta y > 0$.

- (4) Experiment 4: use G4 to express hunger or stomach pain

Set the interactive situation for the elderly to be hungry or have a stomachache at home, and the robot needs to send food or medicine boxes. In this scene, there is a table with food and medicine boxes on the table, and the old man performs gesture operations on the seat next to the table. First, the experimenter placed the gesture G4 next to the stomach to express the semantics of being hungry. After the robot recognized it, the food was picked up and delivered to the hands of the elderly. Then, the gesture G4 is used to express the semantics of stomach pain, and the robot will send a pill box for the elderly after recognition. The mapping success rate is shown in Figure 9(d), and the experimental effect is shown in Figure 10(d). According to Table 2, we design the flexible mapping as h_1 [50, 62, 65], h_2 [7, 18, 21], h_3 [6, 13, 16], h_4 [17, 25, 28], h_5 [17, 31, 34], $\Delta x = 0$, and $\Delta y = 0$.

5.3. Comparative Experiment. In order to further test whether the proposed FMFD algorithm achieves its design goals, this paper compares it with smart TV systems, FM

algorithms [20], and methods based on guided gesture interaction [25]. In order to make the comparison result more convincing, the scenarios selected in this paper are all interactive scenarios proposed in the smart TV system and other flexible mapping algorithms.

Experiment 1: in order to verify the advantages of FMFD algorithm in mapping flexibility and interaction compared with traditional human-computer interaction methods, this paper compares the FMFD algorithm with the gesture smart TV system proposed by Wu et al. [28]. The main idea of the gesture smart TV system is to define a set of gestures and build a unified framework based on this to realize the interaction between users and TV applications. This article selects 8 semantics from its system-defined functions, which are next channel, previous channel, volume up, volume down, turn on the TV, turn off the TV, confirm, and mute. These semantics correspond to a total of 8 gestures. In contrast, FMFD only uses 4 gestures to complete the above semantics, which greatly reduces the cognitive burden. At the same time, the average accuracy of smart TV systems is 92.33%, and the average accuracy of FMFD is 94.73%. In contrast, the accuracy of FMFD is increased by 2.4%.

Experiment 2: in order to verify the performance improvement of the proposed algorithm compared with other flexible mapping algorithms, this paper compares the FMFD algorithm with the FM algorithm proposed by Feng et al. [20]. The main idea of FM is to map the same gesture to several different semantics in the same context through SDFBM feature recognition. It studies five characteristics

of finger folding speed, global motion speed of gestures, trajectory diameter, movement time, and movement depth and has been applied in intelligent teaching interface and vehicle-mounted system. This article selects 4 gestures in its intelligent teaching interface for flexible mapping comparison. The four gestures are g1 (hand gesture for making a fist), g2 (stretching the fist posture of the thumb, index finger, and middle finger), g3 (grasping with the thumb, index finger, and middle finger), and g4 (opening the fist). Among them, the time cost of FMFD is between 1.2 s and 2.10 s, and the time cost of FM is between 3.0 s and 5.25 s. Compared with FM, the time cost of FMFD is reduced by 2.8 s–3.15 s. At the same time, both algorithms use four identical gestures. FM expresses a total of 8 semantics, while FMFD expresses a total of 11 semantics. In contrast, the mapping of FMFD is more flexible. Finally, when the two algorithms are performing flexible mapping experiments, the mapping success rates of different operators are compared, as shown in Figure 12. In contrast, the overall mapping success rate of FMFD is relatively high.

Experiment 3: in order to verify the superiority of the FMFD algorithm compared with other interaction methods, this paper compares the FMFD algorithm with the guided gesture interaction (hereinafter referred to as SG) proposed by Muthugala et al. [29] based on current environment settings and voice commands. The SG system mainly evaluates the spatial parameters and influential concepts of the current environment, then performs a fuzzy inference quantification of the uncertain spatial descriptor, and finally communicates it through a combination of gestures and speech. The interactive scene selected in this article is to control the robot to walk a certain distance forward through instructions. For the SG system, it needs to use a finger to point to a certain point on the ground while inputting voice to achieve the purpose of controlling the robot to walk a certain distance. For the FMFD algorithm, it only needs to use the gesture G3 to express it. According to the different finger curvature and direction, the robot can be controlled to move in three ways: short distance, medium distance, and long distance. In contrast, the control method of FMFD is simpler and the mapping is more flexible. At the same time, the method of combining voice and gesture is easily affected by noisy environments, causing errors in intention understanding, while the FMFD method is less affected by the environment. Figure 13 shows the comparison of the mapping success rate of the two methods in a normal environment and a noisy environment. It can be seen that the FMFD algorithm has a performance preference in terms of accuracy and anti-interference.

5.3.1. Analysis of Results. Experiments show that compared with the traditional human-computer interaction method, the FMFD algorithm proposed in this paper not only guarantees the accuracy of gesture recognition but also greatly improves the flexibility of gesture mapping and the anti-interference ability of the system. Compared with other

FIGURE 12: Mapping success rate of different operators.

FIGURE 13: Performance comparison between FMFD and SG.

flexible mapping methods, this algorithm has obvious advantages in time complexity, while the flexibility of mapping and the success rate of mapping have also been improved. It realizes the flexible mapping of one gesture to multiple semantics in the same interactive situation, reduces the user's memory burden, and is easy to realize intelligent interactive operations.

5.3.2. User Evaluation. In order to further test whether the FMFD algorithm meets the purpose of the design, this paper uses the NASA cognitive load measurement method to evaluate the cognitive load of users in this experiment. In Figure 14, the user's evaluation of the smart TV system, FM algorithm, and FMFD algorithm for the mental requirements, physical requirements, and degree of frustration during the experiment is given. Among them, the mental requirements reflect the user's memory burden, the physical requirements describe the user's degree of fatigue during operation, and the degree of frustration reflects the user's degree of negativity due to the failure of flexible mapping in the operation. The NASA evaluation index

FIGURE 14: NASA-TLX scores of three mapping algorithms.

adopts a 5-point scale, and each index is divided into five grades. 0-1 indicates small cognitive burden, 1-2 indicates relatively small cognitive burden, 2-3 indicates moderate cognitive burden, 3-4 indicates relatively large cognitive burden, and 4-5 indicates large cognitive burden. As can be seen from the figure, compared with other algorithms, the FMFD algorithm proposed in this paper has a lower user cognitive burden, which meets the needs of easy-to-understand, easy-to-operate, and flexible mapping for the elderly escort robot.

6. Conclusions

Aiming at the cumbersome operation and lack of intelligence, which are common in the current escort robots, this paper designs a flexible intention mapping algorithm for elderly escort robots. The intention understanding algorithm based on flexible mapping proposed in this paper uses finger curvature and gesture direction as feature differences and realizes a gesture to express different semantics in the same interactive context. Experiments show that compared with the traditional human-computer interaction, the algorithm reduces the user's memory and operating burden so that the elderly can also use it easily, which meets the needs of easy-to-understand and easy-to-operate elderly caregivers.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This paper was supported by the National Key R&D Program of China (no. 2018YFB1004901) and the Independent In-novation Team Project of Jinan City (no. 2019GXRC013). This work was supported in part by the Shandong Provincial Natural Science Foundation (no. ZR2020LZH004).

References

- [1] T. M. Wang, Y. Tao, and H. Liu, "Current researches and future development trend of intelligent robot: a review," *International Journal of Automation & Computing*, vol. 15, no. 5, pp. 525–546, 2018.
- [2] B. Tony, K. James, R. Aditi, B. Scassellati, and F. Tanaka, "Social robots for education: a review," *Science Robotics*, vol. 3, no. 21, Article ID eaat5954, 2018.
- [3] Q. Lv, H. Lin, G. Wang, H. Wei, and Y. Wang, "ORB-SLAM-based tracing and 3D reconstruction for robot using Kinect 2.0," in *Proceedings of the 2017 29th Chinese Control and Decision Conference (CCDC)*, pp. 3319–3324, Chongqing, China, 2017.
- [4] M. Jiao, H. Wei, B. Zhang, J. Jin, Z. Jia, and J. Yan, "Path Planning of Escort Robot Based on Improved Quantum Particle Swarm Optimization," in *Proceedings of the Chinese Control and Decision Conference (CCDC)*, pp. 3730–3735, Nanchang, China, 2019.
- [5] J. Yu and C. W. Chen, "From talking head to singing head: a significant enhancement for more natural human computer interaction," in *Proceeding of the IEEE International Conference on Multimedia and Expo (ICME)*, pp. 511–516, Hong Kong, China, 2017.
- [6] K. Hu, L. Yin, and T. Wang, "Temporal interframe pattern analysis for static and dynamic hand gesture recognition," in *Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP)*, pp. 3422–3426, Taipei, Taiwan, 2019.
- [7] G. Yang, H. Lv, F. Chen et al., "A novel gesture recognition system for intelligent interaction with a nursing-care assistant robot," *Applied Sciences*, vol. 8, no. 12, 2018.
- [8] F. Chen, H. Lv, Z. Pang et al., "WristCam: a wearable sensor for hand trajectory gesture recognition and intelligent human-robot interaction," *IEEE Sensors Journal*, vol. 19, no. 19, pp. 8441–8451, 2019.
- [9] Y. Zhang, B. Liu, and Z. Liu, "Recognizing hand gestures with pressure-sensor-based motion sensing," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 13, no. 6, pp. 1425–1436, 2019.
- [10] Y. Yu, X. Chen, S. Cao, X. Zhang, and X. Chen, "Exploration of Chinese sign language recognition using wearable sensors based on deep belief net," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 5, pp. 1310–1320, 2020.
- [11] P. Jung, G. Lim, S. Kim, and K. Kong, "A wearable gesture recognition device for detecting muscular activities based on air-pressure sensors," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 2, pp. 485–494, 2015.
- [12] M. Wang, Z. Yan, T. Wang et al., "Gesture recognition using a bioinspired learning architecture that integrates visual data with somatosensory data from stretchable sensors," *Nature Electronics*, vol. 3, no. 9, pp. 563–570, 2020.
- [13] D. Wu, L. Pigou, and L. Shao, "Deep dynamic neural networks for multimodal gesture segmentation and recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 8, pp. 1583–1597, 2016.
- [14] N. Dawar and N. Kehtarnavaz, "Real-time continuous detection and recognition of subject-specific smart TV gestures via fusion of depth and inertial sensing," *IEEE Access*, vol. 6, pp. 7019–7028, 2018.
- [15] S. Zhang and S. Zhang, "A novel human-3DTV interaction system based on free hand gestures and a touch-based virtual interface," *IEEE Access*, vol. 7, pp. 165961–165973, 2019.