Robotica: (2019) volume 37, pp. 1663–1676. © Cambridge University Press 2019
doi:10.1017/S0263574719000183

Obstacle Avoidance through Gesture Recognition: Business Advancement Potential in Robot Navigation Socio-Technology

Xuan Liu†, Kashif Nazar Khan‡, Qamar Farooq†¶*, Yunhong Hao† and Muhammad Shoaib Arshad‡

† *Zhejiang Gongshang University, SBA, Hangzhou, Zhejiang, P. R. China*
‡ *COMSATS Institute of Information Technology, Islamabad, Pakistan*
¶ *Air University, Multan Campus, Multan, Pakistan*

(Accepted January 19, 2019. First published online: March 20, 2019)

SUMMARY

In the present modern age, a robot works like human and is controlled in such a manner that its movements should not create hindrance in human activities. This characteristic involves gesture feat and gesture recognition. This article is aimed to describe the developments in algorithms devised for obstacle avoidance in robot navigation which can open a new horizon for advancement in businesses. For this purpose, our study is focused on gesture recognition to mean socio-technological implication. Literature review on this issue reveals that movement of robots can be made efficient by introducing gesture-based collision avoidance techniques. Experimental results illustrated a high level of robustness and usability of the Gesture recognition (GR) system. The overall error rate is almost 10%. In our subjective judgment, we assume that GR system is very well-suited to instruct a mobile service robot to change its path on the instruction of human.

KEYWORDS: Control mechanism; Body tracking; Dynamic time wrapping; Human–computer interaction.

1. Introduction

Innovations and rapid developments in science of robotics are bringing swiftness in change management and business style. Robots are becoming so public in daily business that nowadays robots are performing the job of cook and waiters in modern restaurants. The increasing trend of robot deployment demands continuous innovation and improvement in the existing automation technology. Innovation performance is the channel to dampen the detrimental effects of certain variables on sustainability,[1] hence, it is important not only for routine trades but also for robot technology and business. For instance, in the common business use, entrepreneurs and managers are keen to use intelligent robots for the trade dealings. A robot is termed as Intelligent Robot if it is adept to avoid collisions by sensing and overcoming the obstacles in the path. This is the most important need of mobile robotics.

With central significance, scientists and system engineers focus on the obstacle avoidance of mobile robots. For this purpose, they design various algorithms. This has been done by them by fusing multi-sensor data to eliminate head gesture reference drift through complementary filter 2.[2] For effective performance of robots, advance gesture recognition is necessary. Optimal utilization of robotics is still looked-for routine business activities because the rapid pace of developments and

* Corresponding author. E-mail: f4farooq@outlook.com

design of algorithms for gesture recognition in robot navigation mandate incessant writings on the issue for updating the readership.

Our research assumes that the robot can move easily and find the path itself. Our focus is to enable a robot to avoid the obstacles that are dynamic. For instance, during human movement, people come across each other and cross each other without saying the other person to get aside. The context of this article is to improve the performance of robots like human decision as by observing the gestures of each other. Hence, to capture the gestures of a human being and to decide which way the robot should move in order to avoid a collision, a slight change of robot path is needed but the path does not need to be recalculated by the robot. In our research, we generate a signal for the robot to get a little aside to avoid the collision but continue its movement in the same direction just like a car driving and a car crossing, we get our car a bit aside just to avoid the collision.

2. Literature Review

Gesture recognition has been an active research area for a couple of decades which involves state-of-the-art machine learning. Gesture is a particular body movement used for reinforcement of verbal message or specific emotion or thought. There are several types of gestures that can be made with the movement of any body part like shoulders, legs, feet, head, arm, etc. Gesture is especially used by deaf and dump people, as they have to communicate and convey their messages. Hands assist a person while speaking and conveying the message more accurately and depict more precise meaning to an audience. The sign language developed by the ancient Indians of North America enabled people of different areas and regions speaking entirely different languages to communicate with each other. How did they do that without saying a word from mouth? Definitely they did it by recognizing gestures of each other.

Usage of Kinect Sensor, camera and image processing is also a useful processing technique for tracking the face by the robot. This not only detects the hand gestures but also avoid collisions by detecting obstacles.[3] In addition to aggregation of data on robot performance, remote server use for tracking and target recognition has also been considered. In order to follow the navigation and targets, fuzzy logic supplies the control mechanism.[4]

In robotics, turn decision is needed in the position of rest after movement and the decision is based on space, left half and right half available in front, known as "Openness" which is calculated through the sums of pixel depth from the left half and the right half space. Successful deployment of mobile robots at homes or workplace has the prerequisite of assurance of collision avoidance. Possible algorithms to avoid obstacles include the Base Algorithm, the algorithms to detect movement of obstacle that is, Asteroid Avoidance (AA) and Human Motion Model Avoidance (HMMA). Interactive collision prevention (ICP) is another algorithm pertaining to gesture for avoiding the obstacles.[5]

The idea of using gestures to communicate with a service robot is not new. Research on this topic was started almost two decades ago. In 1995, Huber and Kortenkamp[6] described a system that recognized the human gestures and humans by using optical flow and stereo vision. That system was capable enough to recognize six gestures and interpret these in significant signals. Gesture Recognition (GR) system of Kortenkamp et al.[7] runs in a physical robot and relies on static gestures that do not involve motion.

The system "Perseus" by Kahn et al.[8] described the features to recognize pointing gestures. It finds the objects pointed by humans. Different techniques have been used including feature maps, like intensity feature maps, motion feature maps, etc. to solve the problem of gestures. Perseus has also been applied to perform the following tasks like:

a. Object pickup task.
b. Recognizing an object.
c. Detection of person in the scene. Moving objects are humans.
d. Segmentation of humans from the scene or background.
e. Tracking of person by its clothes color.
f. Knowledge about the task and the environment.
g. Detection of body, hands, and head.
h. Tracking of body, hands and head.

i. Pointing gesture is recognized by the movement of hand.
j. Detection of area being pointed.

Work on gesture recognition was geared up by researchers and scientists in 1990s. Main research in the related area was done by Boehme et al.[9] during 1998. They developed a system to control their own robot "Milva" by using gesture recognition. Their approach recognizes static pose gestures only by using a neural network algorithm for gesture recognition.

Another approach for gesture recognition using the movement of hands was tried with service robots by Cui and Weng.[10] The same approach was used and enhanced by Triesch and Von Der Malsburg[11] in 1997. They worked to track the hand by taking the still images from a stereo camera. They used the features of color, motion, and depth to detect and localize the hand. Elastic graph matching algorithm was used to recognize the posture of hand. Since it was a complex analysis, it could not be implemented successfully in real time service robots. Moreover, it was also dependent on light and background conditions.

Wren et al.[12] made improvements in the existing project named Artificial Live (ALIVE) to produce a model of people for a variety of tasks. This project created a virtual world of objects in which a human interaction was made possible. This project allows a human to interact with objects by performing gestures like to wave hand to keep an object away from him. Human can look its image also along with those objects on a live TV screen. In 1997, Wren et al.[12] used their algorithm to detect humans by finding their head, leg, arm, and torso using Gaussian distribution associated with it. With the help of these distributions, a human is being tracked and its various body parts.

Representation of American Sign Language (ASL) project was contributed in the same era by Pentland et al.[13] They used the human hand to get the spatial statistics and associated them together with Hidden Markov Models, to interpret a 44 word subset of the ASL. They produced a real-time ASL interpreter with sign recognition accuracy. In 1999, Wilson and Bobick[14] presented a model where gestures are represented as motions of the human body. They used Hidden Markov Model to model gestures and introduced a system that uses the 3D position of the head and hands.

As the gestures are intended to communicate or transfer signals to objects, and it is a bit natural as in daily life humans perform gestures to communicate, authors of this article assume that humans will actively try to use this way with respect to the recipient. Han[15] presented his work in 2007 to explain robot navigation problems and suggested a free path planning algorithm. He, in his work, discussed the algorithms that can be used to solve the robot navigation problems and then chose two of them that generate better results.

In 2012, Ryan[16] presented a work to create and recognize the gestures by fingers of hands using Kinect Sensor. The work with the name "Finger and gesture recognition with Microsoft Kinect" is the extension of Kinect SDK provided by Microsoft in which fingers of hands cannot be tracked. In this research the author suggested a method to track the fingers of hand to perform gestures and to recognize them. The recognition of social wants of a community is important for businesses today[17] and this significance extends to robotics business as well. In 2011, Albrektsen[18] presented his work about the functionality of social robots by introducing Kinect sensor in it. The emphasis on social responsibility and usage of e-commerce in service delivery industry[19] also indicates the need and importance of socialism in robots.

In 2012, LauriBax[5] presented her work about Human robot ICP. She explained that one of the requirements of mobile service robots is their ability to navigate freely such that it is safe and collision free, thus comfortable for humans. Previous research also explores the method which can detect and create a gesture on the basis of movement of all five fingertips for each hand at any time. The resulting images are processed to build a recognizable gesture which is then compared to the existing gesture recognition classes to classify the performed gesture. Once gesture is recognized, relevant signal is sent to robot wirelessly to perform the action. Thus the robot can interact with humans in more efficient manner through the GR system.

Research is conducted in NCCA Bournemouth University about skeleton extraction of human body using Kinect Sensor. In this research a relatively novel approach is proposed, using image-processing techniques, mesh extraction, line skeleton, 2D Skeleton extraction, and depth detection methods, resulting into a 3D skeleton reconstruction. Recently in 2018, research on live skeleton data has strengthened these methods.[20]

3. Planned Work

3.1. Collision avoidance by gesture recognition

Inspiration comes by the way people avoid each other when they are on a collision track. By using subtle body language, people signal each other about their intentions of movement. When two people are on a collision course, they use this information to avoid in an early stage. Robots could use such information too, to better predict the path of people in their vicinity. They can then adjust their path such that a collision is avoided while the amount of distance traveled and time lost is minimal.

Unfortunately, detecting this kind of body language is very hard, especially for a moving robot. Therefore, the user's intentions have to be signaled in another, more explicit way that the robot can more easily detect. There are many ways to interact with the robot, but the use of gestures is the most logical and natural choice. Gestures are a natural form of body language, like the subtle cues. However, gestures are better detectable and easier to classify than the cues people give to each other. Furthermore, people use gestures in their everyday lives too, to communicate their intentions in and around their ways. This concept of interactively solving the collision situation with the other party is to be further explored.[5]

3.2. New approach

Our approach is to incorporate human motions into Human–Computer Interaction (HCI) applications by detecting and using body joints motions, i.e., gestures. A number of different techniques have been proposed for gesture recognition, ranging from the use of Rules based on systems to Dynamic Time Warping (DTW) and Hidden Markov Models. To recognize gestures, DTW algorithm has been studied and implemented. The DTW algorithm recognizes similarities between two time series which do not need to be synchronized in time. A user can perform a gesture slower or faster than the template (recorded gesture) and DTW can easily find the similarity between them.

DTW is a template matching algorithm to find the best match for a pattern out of the referenced patterns, where the patterns are represented as a time sequence of measurements. Any data that can be converted into a (linear) order can be analyzed with DTW, which includes data types such as text, video, audio, or general time series. DTW is one of the techniques used in gesture recognition.[21] To recognize a gesture, DTW warps a time sequence of joint positions (one gesture) to referenced time sequences of joint positions (group of gestures) and produces a similarity value. However, all body joints are not equally important in computing the similarity of two sequences.[22]

3.3. Applications of DTW and related work

DTW is being used in different applications – for example, DTW is used in speech recognition to warp speech in time and to be able to cope with different speaking speeds. In addition to speech recognition, DTW has been found in numerous applications in a wide range of fields including walking pattern recognition, data mining, speaker recognition, information retrieval, signature verification, bioinformatics, chemical engineering, signal processing, gesture recognition, and computer graphics. DTW has also been used to analyze and align motion data in the field of computer animation.

At Microsoft Research, a group of researchers used DTW to recognize dance gestures using the Kinect. Their gesture classifier has an average accuracy of 96.9% with which it can be said that DTW can be used to achieve high accuracy (Fig. 1).[23]

3.4. DTW working

The DTW algorithm is a dynamic programming algorithm, which uses a recursive update of DTW cost by adding the distance between mapped elements of the two sequences at each recursion step. The distance between two elements is oftentimes the Euclidean distance, which gives equal weights to all dimensions of a sequence sample (Figs. 2 and 3).

Above equations are to calculate Euclidean distance between two sequences. First equation is to calculate the Euclidean distance between the two points of a sequence and second equation is the accumulated distance of all the points. If the two sequences are identical then the accumulated distance (total cost) D must be 0 as shown in Fig. 4.

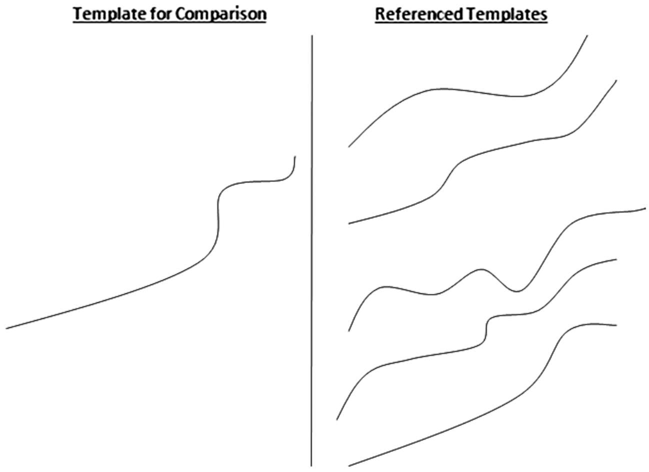


Fig. 1. To find the best match for the pattern at left side out of referenced patterns at right side.

$$d(i, j) = \| r(i) - t(j) \|$$

Fig. 2. Equation to calculate distance between two points using Euclidean distance.

$$D = \sum_{k} d(ik, jk)$$

Fig. 3. Accumulated distance using Euclidean distance.

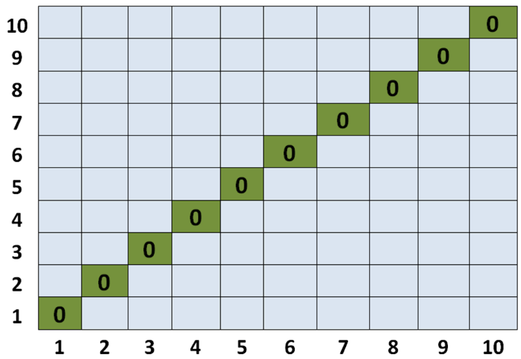


Fig. 4. Simple Euclidean distance.

$$P(i, j) = D(i, j) + MIN[D(i-1, j), D(i, j-1), D(i-1, j-1)]$$

Fig. 5. Equation to calculate each node cost.

However, DTW calculates the cost with a little variation to Euclidean distance. DTW calculates the distance between the two points of two different time series using Euclidean distance and addition of lowest cost of its three neighbor cells from which this cell can be reached. A cell can be reached from three different ways; from left, bottom or the diagonal down cell. The total cost can be calculated by accumulating the smallest cost cells from last cell to first cell, that is, (0, 0). In other words, the path from last cell to first cell with smallest cost (distance) is the accumulated cost and the desired path (Figs. 5 and 6).

https://doi.org/10.1017/S0263574719000183 Published online by Cambridge University Press

$$D = \sum_k p(ik, jk)$$

Fig. 6. Accumulated cost using DTW.

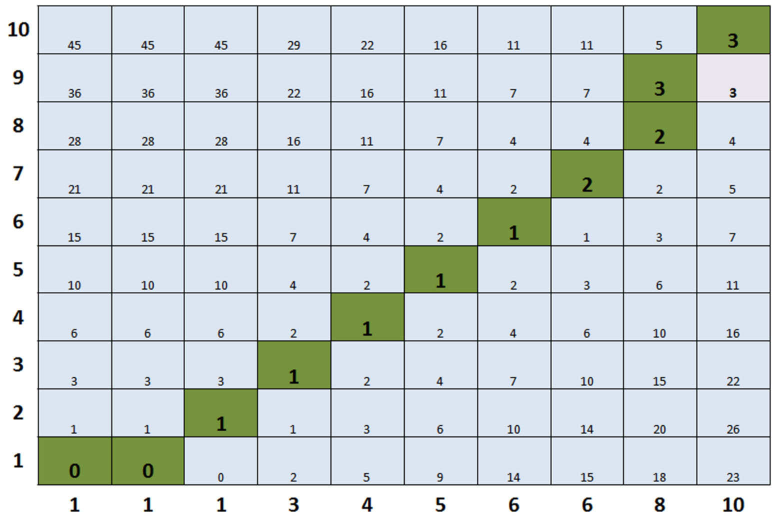


Fig. 7. Calculated cost using DTW.

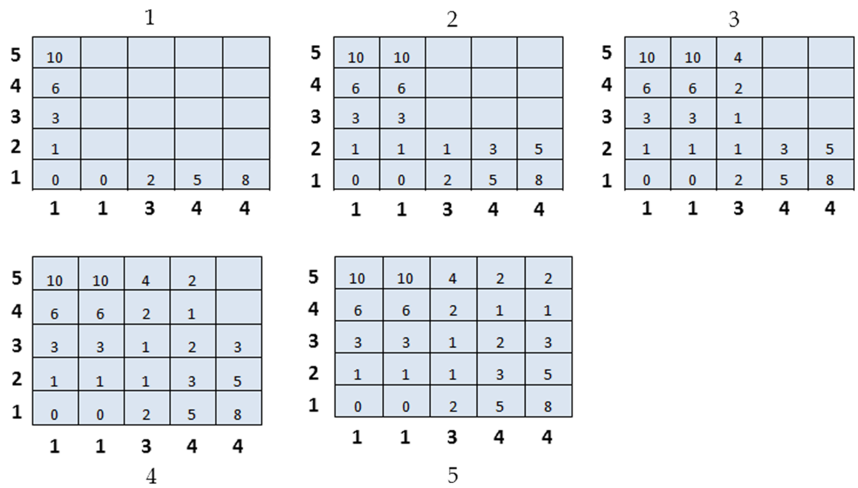


Fig. 8. Calculating each node cost using DTW.

This whole process is represented in Fig. 7.

Step by step representation of DTW is shown Fig. 8

In first step, cost of each node is calculated, as shown in above figures, started from the first cell to the last cell, cost of each cell is calculated. The cost of a cell is the cost of cell itself plus the cheapest cost of one of its three neighboring cells that can be reached from this cell. In the second step, shortest path with lowest accumulated cost starting from the last cell ([5, 4]) to the first cell ([0, 0]) is calculated. Beginning at the last cell, the cheapest cell of the three can be chosen from (left, down and the diagonal down cell) as the next cell. The accumulated cost for each cell in the path is added together to be the total path cost (Fig. 9).

https://doi.org/10.1017/S0263574719000183 Published online by Cambridge University Press

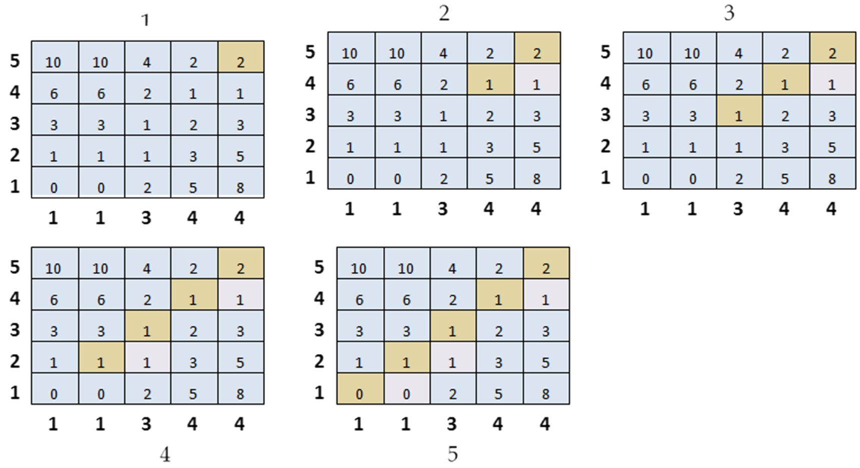


Fig. 9. Calculating accumulated cost and smallest cost path.

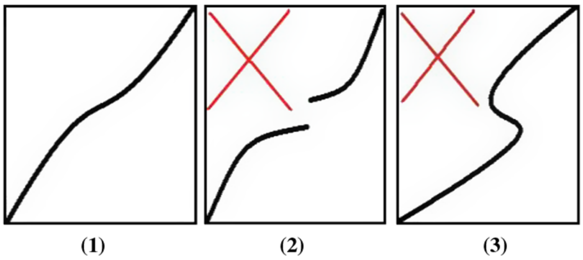


Fig. 10. Constraints of DTW.

4. DTW Constraints

Three basis constraints of DTW (Fig. 10) are:

1. Path should include beginning and ending.
2. Path should not have any jumps.
3. Path cannot go back in time.

5. Experiments and Results

Result of this project is to generate the signals on the basis of hand gestures for the robot. Initially three signals have been generated using right and left hand's movement. The algorithm is tested by applying the signals on a MS power point presentation. Signal generated by using right hand moves the presentation one down, whereas signal generated using left hand moves the presentation one up, and to stop the presentation cross of both hands is used. Other signals can also be generated by following the same technique.

5.1. Gesture recognition experiments

Using the acquired data and defined rules, following gesture recognition experiments are performed.

1. Individual recognition

All three gesture classes are trained and tested by each individual itself. A distinction is made between the three gesture classes and nine commands.

2. Collective recognition

Using same setup, the trained data (templates) is used by all the individuals.
The results of each gesture recognition class are described below:

https://doi.org/10.1017/S0263574719000183 Published online by Cambridge University Press

Table I. Performance of each user on all three GR classes with normal speed.

Users	Move Left	Missed	Move Right	Missed	Stop	Missed
U 1	9	1	8	2	7	3
U 2	9	1	9	1	9	1
U 3	8	2	7	3	8	2
U 4	9	1	8	2	8	2
U 5	7	3	9	1	7	3

Table II. Percentage of user's performance with normal speed.

Users	Successful performed gestures	Successful (%)	Missed	Missed (%)
U 1	24	80.00	6	20.00
U 2	27	90	3	10.00
U 3	23	76.67	7	23.33
U 4	25	83.33	5	16.67
U 5	23	76.67	7	23.33

5.1.1. Gesture recognition performance. A dataset was collected for testing the gesture recognition (GR) software. For six participants, data were acquired. These data were later used to test (offline) the DTW-based GR systems. Below, the recognition results for the DTW-based GR systems have been described.

5.1.2. Performance of DTW-based system. Five users were chosen for this activity to test GR system. A test environment was created and signal generated by the system is shown by the help of MS Power Point Slides. We established that on a successful Move Right Gesture Power Point Slide will go next one slide, whereas on successful Move Right Gesture, Power Point slide will go back one slide and it will go home on successful Stop Gesture. Each user was asked to choose which gesture style is comfortable for him/her, keeping in mind the basic limitations of each gesture class. After understanding and recording, user was asked to perform the gesture to check the performance of system. All three classes of gestures were loaded in the system and the user was asked to perform 10 gestures for each class. Data are captured by individual performance of users. We used Microsoft Kinect and Microsoft SDK to capture the movement of a human body and retrieve three gestures regarding arm and hand movement.

The performance of GR system is tested using DTW while keeping all the settings same as described above. The System is tested on same pattern once with Individual Recognition and again with Collective Recognition. Same users were asked to perform the experiment with this system. The overall results and performance of GR system with DTW were satisfactory with normal speed but a decline in performance was noted when gestures are performed in slow and fast motion. We again compared the results performed in normal speed and fast speed. Results show that DTW can be used as a basic gesture recognition method.

5.1.3. Individual recognition. The results of experiments are presented in Tables I and II and Figure 11 which were generated through the data obtained from the users by selecting the templates at their own. They were asked to generate the template after a couple of exercises. Once they were comfortable with their movements to produce the gestures they produced a template to be used in experiments.

Results show that 81% is the success rate, which is quite good performance of DTW. By analyzing, it is found that since there is no starting point defined to record the gesture in the system, hence once the user gets itself in front of the camera, it starts capturing and generates frame for DTW. The efficiency can be improved to more than 90% by introducing a controlled mechanism by defining a starting and ending point of gestures. Improvements can also be made to ignore irrelevant gestures in our system to avoid the missing gestures. Another important factor that hinders in achieving the

Table III. Performance of each user on all three GR classes with speed.

Users	Move Left	Missed	Move Right	Missed	Stop	Missed
U 1	8	2	7	3	5	5
U 2	7	3	6	4	7	3
U 3	6	4	7	3	6	4
U 4	8	2	5	5	8	2
U 5	5	5	6	4	7	3

Table IV. Percentage of user's performance with speed.

Users	Successful performed gestures	Successful (%)	Missed	Missed (%)
U 1	20	66.67	10	33.33
U 2	20	66.67	10	33.33
U 3	19	63.33	11	36.67
U 4	21	70.00	9	30.00
U 5	18	60.00	12	40.00

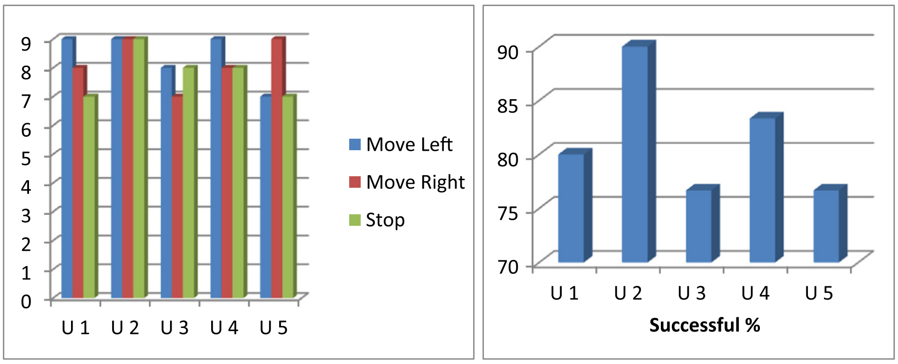


Fig. 11. Graph (left): Performance of each user on all three GR classes with normal speed. (right): Percentage of user's performance with normal speed.

desired result can be the user-defined template. Although users were asked to train themselves and be familiar with the working of system prior to generate the final template, but still, when we analyzed the quality of gestures it was not found up to the mark. This flaw was covered when collective recognition was experimented where the templates were made by the joint efforts of some users and finally altered by the experts.

Another important factor which was identified during investigation is that the template comprises of around 30–34 frames. When user performs gesture in test environment to test the working of GR system, gesture that comprises of at least seven frames is provided to the GR system for recognition. Tables III and IV and Figure 12 show that unnaturally even before the complete performance of the gesture by the users, GR system recognized the seven frames, which caused the system to miss a gesture.

Results of GR system when gestures are performed in fast speed are decreased in performance than normal speed. The overall performance drops down to 65% success rate. The possible reason and factors of this decline are similar as described above. The success rate can be increased if the minimum threshold of similarity increased a little bit more.

5.2. *Collective recognition*

Results shown below are the experiment that is conducted in second phase of DTW-based GR system. All the settings of the environment were kept same and same users were asked to perform this second

https://doi.org/10.1017/S0263574719000183 Published online by Cambridge University Press

Table V. Performance of each user on all three GR classes with speed.

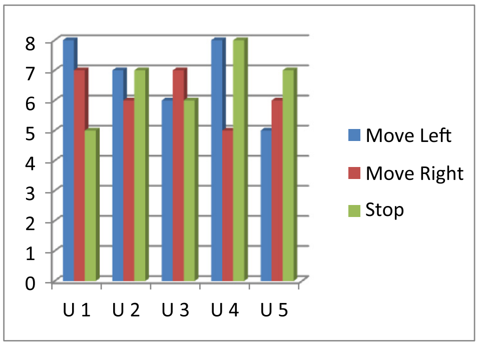
Users	Move Left	Missed	Move Right	Missed	Stop	Missed
U 1	9	1	9	1	8	2
U 2	10	0	10	0	9	1
U 3	8	2	8	2	9	1
U 4	10	0	9	1	10	0
U 5	9	1	10	0	9	1

Table VI. Percentage of user's performance with normal speed.

Users	Successful performed gestures	Successful (%)	Missed	Missed (%)
U 1	26	86.67	4	13.33
U 2	29	96.67	1	3.33
U 3	25	83.33	5	16.67
U 4	29	96.67	1	3.33
U 5	28	93.33	2	6.67

Table VII. Performance of each user on all three GR classes with speed.

Users	Move Left	Missed	Move Right	Missed	Stop	Missed
U 1	9	1	8	2	9	1
U 2	8	2	9	1	7	3
U 3	9	1	7	3	8	2
U 4	9	1	9	1	9	1
U 5	8	2	10	0	10	0



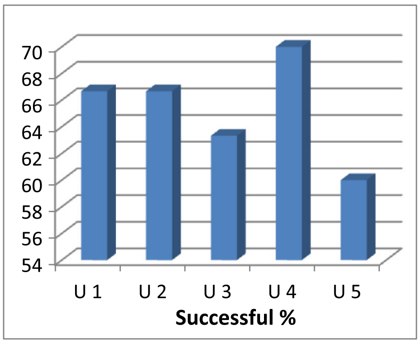


Fig. 12. Graph (left): Performance of each user on all three GR classes with fast speed. (right): Percentage of user's performance with fast speed.

round of tests. The difference between the Individual Recognition and Collective Recognition is that here users are given a pre-defined template of each gesture class. The template was created by the gesture inputs of users and refined by experts.

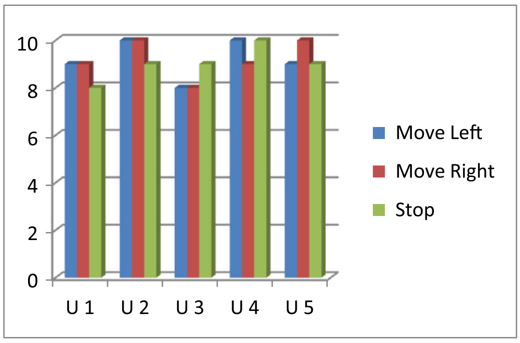
Experiments are conducted in two scenarios, normal speed and fast speed. Results are more favorable and show that DTW can be a better technique to be used in gesture recognition. When experiments performed in normal speed the overall performance calculated is above 90% success and in fast speed it is around 85%, which is quite satisfactory (Tables V, VI, VII and VIII; Figs. 13 and 14).

Table VIII. Percentage of user's performance with speed.

Users	Successful performed gestures	Successful (%)	Missed	Missed (%)
U 1	26	86.67	4	13.33
U 2	24	80.00	6	20.00
U 3	24	80.00	6	20.00
U 4	27	90.00	3	10.00
U 5	28	93.33	2	6.67

Table IX. Confusion matrix for normal speed and fast speed for all three GR classes.

	Normal speed			
	ML	**MR**	**ST**	**Missed**
ML	88			12
MR		87		13
ST			84	16
Missed	12	13	16	
	Fast speed			
	ML	**MR**	**ST**	**Missed**
ML	77			23
MR		74		26
ST			76	24
Missed	23	26	24	



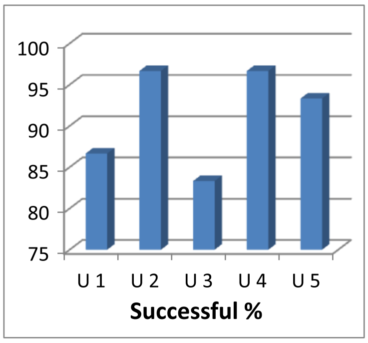


Fig. 13. Graph (left): Performance of each user on all three GR classes with normal speed. (right): Percentage of user's performance with normal speed.

5.3. *Confusion matrix*

A confusion matrix is developed to judge the DTW-based GR system's performance for all three GR classes: Move Left (ML), Move Right (MR), and Stop (ST). Two matrices are made separately, one for normal speed and other for fast speed (Table IX).

Above confusion matrix shows that the performance of DTW-based GR system is quite efficient. Efficiency is better in performance in normal speed than performance in fast speed. The reason of better performance in normal speed can be that the templates were created mostly on the basis of normal speed. But the difference is not significant between the two speeds and the overall performance of GR system remained very good. The good thing of the system is that it did not falsely recognize any Gesture Class. When considering Move Left class, 12 attempts are missed in normal speed and 23 in fast mode. Only the gestures which could not be performed according to the instructions got missed or those which could not be started and completed properly by the user. Around 15% of the

https://doi.org/10.1017/S0263574719000183 Published online by Cambridge University Press

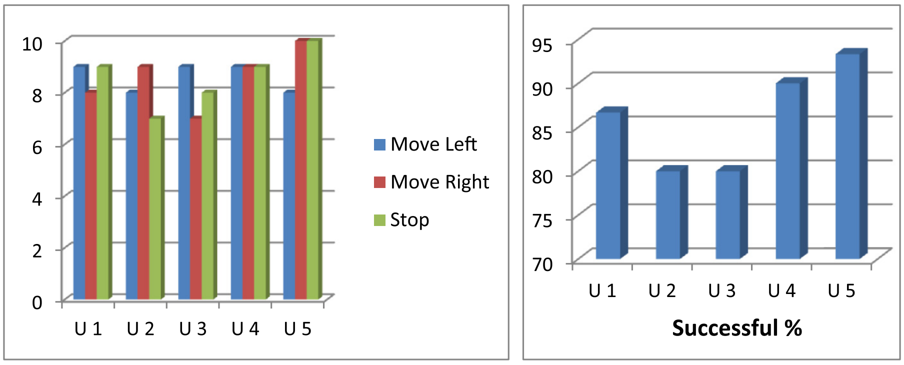


Fig. 14. Graph (left): Performance of each user on all three GR classes with fast speed. (right): Percentage of user's performance with fast speed.

total gestures missed in ML class. Similarly, by looking at Move Right class again around 17.5% are missing which is quite an efficient performance. While analyzing Stop class it is found that around 20% of gestures gone missed. It is not a very bad result, but cannot be considered as a very significant as compared to ML and MR results. The reason of this decline may be due to the difficult way to perform ST gesture. There may be the possibility that users could not perform in an efficient manner and also users perform the stop gesture in the end and may be their behavior was a bit casual. Since the performance time is very short so if the users have to perform twice to instruct a robot, there is always plenty of time for a robot to recognize the instructions of user.

6. Discussion and Conclusion

An Obstacle Avoiding Robot detects the obstacles using Ultrasonic sensors and Microcontroller processing of data. In absence of this feature, movement of a robot becomes brittle or restricted. Smooth navigation, avoiding collisions, and acquainting in unknown environment turn into possibility through dynamic steering algorithm, enabling a robot to ensure continuity of movement in presence of obstacles by eluding collisions. The hulking one-armed Goliaths which dominated industrial assembly lines is the kind of robot which endured for the past half century. In current era, gesture has attained the importance in robotics. For NAO humanoid, Wang et al.[2] developed the strategy via head gestures on the basis of local and global real-time videos available on Google Glass by devising a module for connection establishment through Google Glass to subsequent detection of head gestures.

Robots are controlled using a special interface, such as an operative panel or a computer. To change a movement, robot's objective consists of stopping the robot using an on-board interface or a computer. However, one might imagine a scenario where a robot interacts with humans operating in a more fluent way. What if someone could tell the robot to perform a task, only using its own body? This research brings that scenario one step closer. Although humans may communicate using speech alone, gestures such as pointing or signaling actions are frequently used in daily life, especially when explaining actions. Hence, gesture recognition is an important task for robots and humans to coexist in the same environment.[18]

This research has described an interactive vision-based gesture interface for human–robot interaction. The approach presented consists of joints detection for recognizing hand gestures using DTW-based similarity finding method. DTW is a robust technique to find the similarity between two time series – it is employed to find out the similarity between a predefined hand movement template and a performed movement. If the similarity lies in between the defined threshold it is said to be a recognized gesture. The GR system has been tested with the help of Microsoft Power point slide (MPPT) software. The different gestures "Move Left", "Move Right", and "Stop" are mapped with three signals "Back", "Next", and "Home". On a successful recognized gesture one of the events on MPPT executed. For example, if Move Left gesture is recognized by the system it generates the signal of "Back" to MPPT that moves the slide one previous. Similarly it moves the MPPT one next on a successful "Move Right" gesture, and move MPPT to first slide or home on performing a successful "Stop" gesture.

Our interface goes beyond previous work in that it is capable of recognizing not only static pose gestures, but also dynamic motion gestures, which are commonly used for communication among people. Motion gestures are defined through specific temporal patterns of arm movements, which improve the classification accuracy and reduce the chances of accidentally classifying arm poses as gestures that were not intended as such.

Our study is an advanced research on robot navigation in which we assume that a robot is already there and working fine. Means, it can find its path in static environments and can determine a path to move freely. What we do is to generate a signal for the robot to move accordingly and it will do in the same way like we move the steering of a car to get away from the main road to avoid the collisions, but it remains on its path and continue its journey. Same functionality of movement has already been implemented in a robot. We want to control its movement as per the specific need. In our case for collision avoidance from humans coming straight in the path of a robot, the algorithm has been tested and it generates signal properly. We checked it with the help of PowerPoint and also in robot simulator "Gazebo."

Experimental results illustrate a high level of robustness and usability of the GR system. The overall error rate was almost 10%. In our subjective judgment, we assume that GR system is very well-suited to instruct a mobile service robot to change its path on the instruction of human. While this is only an example application designed to test the utility of gestures in human–robot interaction, we conjecture that the proposed interface transcend to a much broader range of upcoming service robots.

There are few natural ways to moderate the relationship of a media and communication[24] and real-time path planning has already been used as the method of obstacle avoidance in multiple mobile robots.[25,26] The relationships are explained through corporate social responsibility with respect to variation cultures,[27] as well as through scientific developments including robot navigational studies. This article has aided entrepreneurs in introducing a convenient alternate to obstacle avoidance in robot technology for commercialization in particular business activities. We believe that finding "natural" ways of communication between humans and robots is of utmost importance for the field of mobile service robotics. While this research exclusively presents the gestures as input, we believe it is worthwhile to augment the GR system by integrating a speech-based interface, so that both gestures and speech can be combined when instructing a mobile service robot. Gestures can help to clarify spoken commands and future research in the area of utilizing robots in customer-oriented businesses will mitigate the collision threats for competitive technological advantage.

Funding

This research was funded by Natural Science Foundation of Zhejiang Province (LQ19G020004) and Key research center of humanities and social sciences in universities of Zhejiang Province (JYTgs20151104).

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. Y. Hao, Y. Zhang and Q. Farooq, "The contribution of leading firms in environmental sustainability: Dampening the detrimental effect of political capital ties," *Int. J. Environ. Sci. Technol.* **15**(12), 2581–2594 (2018).
2. Z. Wang, X. Wen, Y. Song, X. Mao, W. Li and G. Chen, "Navigation of a humanoid robot via head gestures based on global and local live videos on google glass," **In**: *IEEE International Instrumentation and Measurement Technology Conference*, Turin, Italy (2017) pp. 1–6.
3. R. Berri, D. Wolf and F. Osório, "Telepresence robot with image-based face tracking and 3d perception with human gesture interface using kinect sensor," **In**: *2014 Joint Conference on Robotics: SBR-LARS Robotics Symposium and Robocontrol (SBR LARS Robocontrol),* IEEE, Sao Carlos, Brazil (2014) pp. 205–210.
4. P. Benavidez and J. Mo, "Mobile robot navigation and target tracking system," **In**: *International Conference on System of Systems Engineering*, Albuquerque, NM, USA (2011) pp. 299–304.
5. L. Bax, "Human-robot interactive collision prevention: Improved navigation in home environments," (2012). Master Thesis. Retrieved from https://theses.ubn.ru.nl/handle/123456789/194
6. E. Huber and D. Kortenkamp, "Using stereo vision to pursue moving agents with a mobile robot," **In**: *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, Nagoya, Japan (1995) p. V13.

7. D. Kortenkamp, E. Huber and R. P. Bonasso, “Recognizing and interpreting gestures on a mobile robot,” *AAAI/IAAI* **2** (1996) pp. 915–921. Retrieved from http://www.aaai.org/Library/AAAI/1996/aaai96-136.php
8. R. E. Kahn, M. J. Swain, P. N. Prokopowicz and R. J. Firby, “Gesture recognition using the perseus architecture,” **In**: *Proceedings CVPR’96, 1996 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE (1996) pp. 734–741.
9. H.-J. Boehme, U.-D. Braumann, A. Brakensiek, A. Corradini, M. Krabbes and H.-M. Gross, “User localisation for visually-based human-machine-interaction,” *Proceedings of the Third IEEE International Conference on Automatic Face and Gesture Recognition,* Nara, Japan (1998) p. 486.
10. Y. Cui and J. J. Weng, “Hand segmentation using learning-based prediction and verification for hand sign recognition,” **In**: *Proceedings CVPR’96, 1996 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE (1996) pp. 88–93.
11. J. Triesch and C. Von Der Malsburg, “Robotic gesture recognition,” **In**: *International Gesture Workshop*, Springer (1997) pp. 233–244.
12. C. R. Wren, A. Azarbayejani, T. Darrell and A. P. Pentland, “Pfinder: Real-time tracking of the human body,” *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(7), 780–785 (1997).
13. A. Pentland, B. Moghaddam and T. Starner, “View-based and modular eigenspaces for face recognition,” (1994). doi:10.1.1.47.3791
14. A. D. Wilson and A. F. Bobick, “Parametric hidden Markov models for gesture recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.* **21**(9), 884–900 (1999).
15. K. M. Han, *Collision Free Path Planning Algorithms for Robot Navigation Problem* (University of Missouri, Columbia, 2007). doi:10.32469/10355/5021
16. D. J. Ryan, *Finger and Gesture Recognition with Microsoft Kinect* (University of Stavanger, Norway, 2012).
17. Y. Hao, Q. Farooq and Y. Zhang, “Unattended social wants and corporate social responsibility of leading firms: Relationship of intrinsic motivation of volunteering in proposed welfare programs and employee attributes,” *Corporate Social Responsibility Environ. Manage.* **25**(6), 1029–1038 (2018).
18. S. M. Albrektsen, *Using the Kinect Sensor for Social Robotics* (Institutt for teknisk kybernetikk, Trondheim, Norway, 2011).
19. Q. Farooq, P. Fu, Y. Hao, T. Jonathan and Y. Zhang, “A review of management and importance of E-Commerce implementation in service delivery of private express enterprises of China,” *SAGE Open* **9**(1), (2019). doi:10.1177/2158244018824194
20. T. Hoxey and I. Stephenson, “Integrating live skeleton data into a VR environment,” (2018). Retrieved from http://eprints.bournemouth.ac.uk/30826/
21. M. Müller, “Dynamic time warping,” **In**: *Information Retrieval for Music and Motion* (Springer, Berlin, Heidelberg, 2007) pp. 69–84.
22. S. Celebi, A. S. Aydin, T. T. Temiz and T. Arici, “Gesture recognition using skeleton data with weighted dynamic time warping,” *VISAPP* (1), 620–625 (2013).
23. M. Raptis, D. Kirovski and H. Hoppe, “Real-time classification of dance gestures from skeleton animation,” **In**: *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation,* ACM (2011) pp. 147–156.
24. Y. Hao, Q. Farooq and Y. Sun, “Development of theoretical framework and measures for the role of social media in realizing corporate social responsibility through native and non-native communication modes: moderating effects of cross-cultural management”. *Corporate Soc. Respons. Environ. Manage.* **25**(4), 704–711 (2018).
25. F. Fahimi, C. Nataraj and H. Ashrafiuon, “Real-time obstacle avoidance for multiple mobile robots,” *Robotica* **27**(2), 189–198 (2009).
26. M. H. Tanveer, C. T. Recchiuto and A. Sgorbissa, “Analysis of path following and obstacle avoidance for multiple wheeled robots in a shared workspace,” *Robotica* **37**(1), 80–108 (2018).
27. Q. Farooq, Y. Hao and X. Liu “Understanding Corporate Social Responsibility with Cross-Cultural Difference: A Deeper Look at Religiosity,” *Corporate Social Responsibility and Environmental Management* (2019). doi:10.1002/csr.1736