An exclusive human-robot interaction method on the TurtleBot platform

Chuantang Xiong and Xu Zhang

Abstract— **Recently, the exclusive human-robot interaction is increasingly required because robots are accepted by non-technical individuals as part of their lives. In this paper, we presented an exclusive human-robot interaction method on the TurtleBot platform. The 2D and 3D face recognition are integrated with skeleton information. The identity information and human-robot interaction message are always bound together, and the identity recognition has priority to human-robot control. Experiments show that the TurtleBot is able to robustly react on the skeleton signals from its human interaction partner while ignoring other signal sources.**

I. INTRODUCTION

The field of human-robot interaction is developing rapidly as robots are readily accepted by non-technical individuals as part of their lives. Robots become more capable of operating with people in natural environments because their sensing, cognitive and actuating capabilities are greatly enhanced. These natural and intuitive interactive ways greatly facilitate the interaction between robots and people, thus improve the effectiveness of communication, especially for non-expert client. On the other hand, the mature of human-robot interaction technology greatly broadens the range of robot application. For example, the service robot can act as a guard of a family, a partner of a lonely man, or even the pet of a kid, rather than as an autonomous cleaning 'machine' in the supermarket.

Recently, the exclusive human-robot interaction is required. The communication channel should be invalid without authorization or identification. This trend is motivated by actual demand. When a kid is taking his puppy robot for a walk, by no means does the kid want the puppy robot listen commands to everyone on the street, or if the guard robot can be easily controlled by the thief, this robot seems useless or even disastrous to any family. Those conditions demand the exclusive control as a key factor of the human-robot interaction. It means that robots can hardly be achieved with skeleton control or voice control from any wrong person.

In this paper, an exclusive human-robot interaction method is proposed, which can achieve the following functions: 1) When multiple persons are standing in front of the robot, it should be ensured to be controlled by the correct person. 2) The robot should be ensured to be controlled by the real person, not the accessory such as a picture of the person. In this method, identity recognition has priority to human-robot control, and the identity information and human-robot interaction message are always bound together. The proposed method is implemented on the TurtleBot platform. 2D and 3D face recognition is adopted as identification, and the body skeleton is the human-robot interaction channel. The binding condition is they all belong to the same person. Thus, the TurtleBot just responds to its own host command.

The main subtasks of this method can be divided into the following sections: 1) Integrating the 2D and 3D face recognition modules. 2) Applying face recognition modules on skeleton control module. 3) Calibrating the coordinate data of image with skeleton. Data are exchanged between nodes under ROS (Robot Operation System). They are organized as followed in this paper.

Section II presents the relative work of this paper. The detailed description of our method is presented in section III. Section IV introduces the experiments. Conclusion and future work are followed in section V.

Chuantang Xiong is with the University of Michigan - Shanghai Jiao Tong University Joint Institute, Shanghai Jiao Tong University, China. E-mail:davidxiong@sjtu.edu.cn

Xu Zhang is with faculty of mechanical engineering and automation, Shanghai University, China, and a researcher of State Key Lab. of Digital Manufacturing Equipment and Technology in Huazhong University of Science and Technology. Corresponding author, phone number: 086-21-56331365, E-mail: xuzhang@shu.edu.cn

II. RELATED WORK

A. Human-robot interaction

Human-robot interaction has attracted much attention. The traditional interface is the master-slave prototype in which a robot duplicates the same physical motion of its operator. R. M. Voyles [1] teaches a robot how to maneuver through complex configurations using guarded moves. J. Saunders [2] taught a robot to build blocks in a controlled environment where he shows the required behaviors of building blocks to the robot and the robot replicates the behaviors.

More recently, numerous research prototypes have been equipped with speech synthesizers and recognizers. Ogawa, H. [3] presented a simple way of interaction: the operator talks to the robot, while the robot reacts by nodding or shaking head. Bischoff [4] proposed a type of robot which can generate speech as well as understand spoken language. By using the adaptive noise reduction and voice activity detection methods, R Brueckmann [5] improved the verbal human-robot interaction.

Meanwhile, many visual human-robot interaction methods were also proposed. S Waldherr [6] presented a gesture based interface for human-robot interaction, then M Hasanuzzaman [7] worked on this field, and proposed a more robust method, which used multi-cluster approach, and combined computer vision and knowledge-based approaches in order to adapt to new users, gestures and robot behaviors.

B. Human recognition

Human recognition is another independent area in computer vision. During decades, many recognition methods including fingerprint recognition [8], vein recognition [9], iris recognition [10], and face recognition [11], [12], [13] have been presented. However, in terms of the fingerprint, vein, or iris recognition, the controller has to maintain a very close distance with the robot. Due to this reason, the recognition process will be implemented only once at the beginning of the control process which makes the whole process inconvenient or unsafe. Given the features of different human recognition methods, we adopted the face recognition to achieve the exclusive control.

III. EXCLUSIVE HUMAN-ROBOT INTERACTION

In this paper, an exclusive human-robot interaction is proposed and it includes four parts, such as, 2D face recognition, 3D face recognition, skeleton detection, and data calibration and skeleton control. These modules cooperate together to control a robot just from the right person. The whole method is described in Fig. 1.

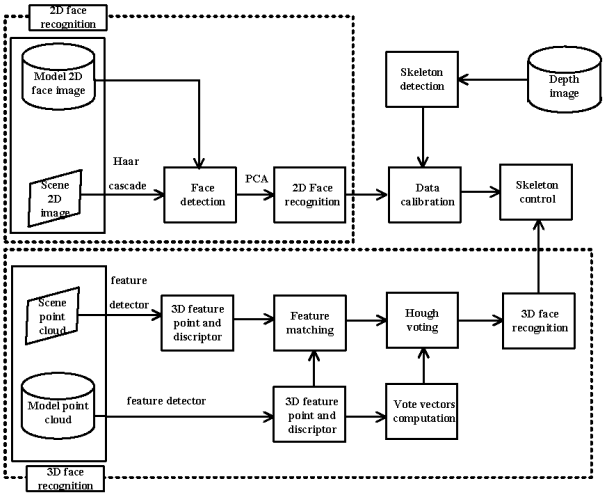


Fig. 1: The framework of the proposed exclusive human-robot interaction

First, three types of image data such as 2D image, point cloud image and depth image are input independently to the 2D face recognition, 3D face recognition and skeleton detection modules. After the process of those data, messages, containing information such as whether the correct person has been recognized or the coordinate positions of skeleton data and face data, immediately is sent to the skeleton control module. Then, the head frame of the skeleton is calibrated with the position of face. This bonding connection ensures the control message of the skeleton and the face identification from the same person. After the correct person is found, the control is valid and the robot can react to the order.

A. 2D face recognition

In 2D face recognition, PCA (Principal component analysis) is firstly adopted to reduce to the dimension. Then, the face is recognized with Haar cascade [14]. 250 images for 10 persons were pre-stored. Two submodules work corporately on this module. One is responsible for reading image data and commands, and recognizing the correct person, the other is provided for demonstration and test. In order to reduce the effect of random factors, only if more than 15 pictures in every 20 pictures were recognized, this recognition can be defined as effective.

B. 3D face recognition

Hough voting algorithm [15] is adopted to recognize the 3D face from the point cloud. Interest points are extracted both from scene and model point cloud, for example, the blue circles in Fig. 2. A 3D descriptor containing local neighborhood information is defined for each interest point. Typically, the detection and extraction of model point cloud will not be processed every time, they can just be performed once for all the off-line. A group of correspondences (green arrow in Fig. 2) can be determined by defining the threshold like the distance between their descriptors. Due to the affection of noise, cluttered background and partial occlusions or spikes of point cloud, the matching process will also include wrong correspondences such as red arrow in Fig. 2.

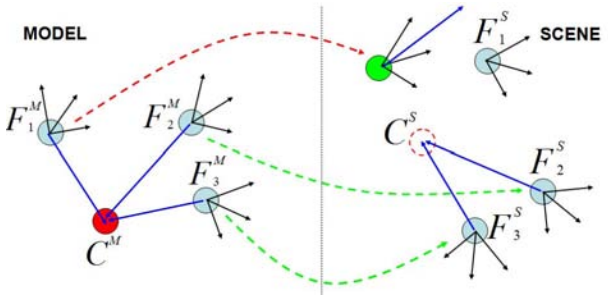


Fig. 2: Example of 3D Hough Voting based on local reference [15]

A sample of 3D Hough voting algorithm can be described in Fig. 3. Green lines stand for correct correspondences, while red lines stand for wrong correspondences. A unique reference point is calculated at off-line mode (red circle in Fig. 2), and the center of the model point cloud is chose as the reference point. Supposing all the point coordinates of the model are given in the same global reference, the vector between F_i^M and C^M can be obtained from each feature point F_i^M:

$$V_{i,G}^M = C^M - F_i^M \tag{1}$$

Due to the vector between a feature point and reference point should be calculated under local reference, the vector $V_{i,G}^M$ is changed to local reference (see Fig. 4):

$$V_{i,L}^M = R_{GL}^M \cdot V_{i,G}^M \tag{2}$$

Where $R_G L^M$ represents the rotation matrix from the global reference to local reference:

$$R_{GL}^M = [L_{i,x}^M \; L_{i,y}^M \; L_{i,z}^M]^T \tag{3}$$

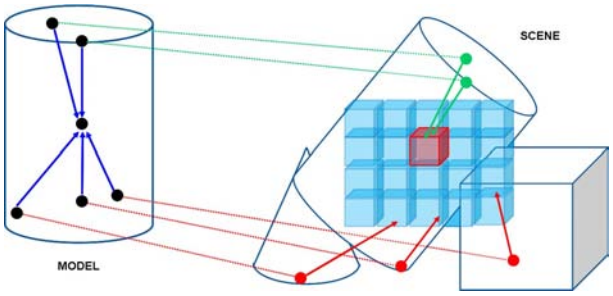


Fig. 3: Example showing the 3D Hough Voting algorithm [15]

The on-line mode is responsible for the correspondence of feature points and estimation of pose between model point cloud and scene point cloud. Each correspondence in model point cloud and scene point cloud ($F_i^M \leftrightarrow F_j^S$) casts a vote for the position of the reference point in the scene. Because the calculation of local reference is invariant, this allows the transformation shown in Fig. 4 as R_L^{MS}, making $V_{i,L}^S = V_{i,L}^M$. Then we can transform $V_{i,L}^S$ into global reference by the relationship:

$$V_{i,G}^S = R_{LG}^S \cdot V_{i,L}^S + F_j^S \tag{4}$$

Where R_{LG}^S represents the rotation matrix from the local reference to the global reference:

$$R_L G^S = [L_{j,x}^S \; L_{j,y}^S \; L_{j,z}^S]$$

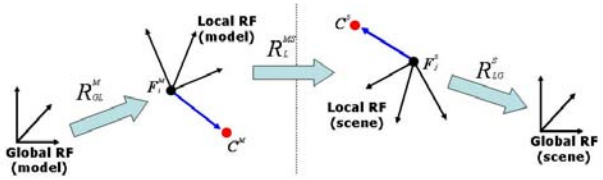


Fig. 4: Transformations by the use of local references [15]

After a set of correspondence has been selected in model point cloud and scene point cloud, the correct 3D face can be recognized through an appropriate threshold value.

C. skeleton detection

The skeleton control program can detect as many as 16 skeletons, and show them with different colors and numbers on the screen. Each skeleton contains 15 fixed frames like: head, neck, torso, shoulder, elbow, hand, etc. The first skeleton being recognized owns the power to control the robot when multiple persons standing in front of the TurtleBot. In this paper, we draw a rectangle on the head frame of the skeleton of the controller (see Fig. 5).

D. data calibration

Both 2D and 3D face recognition communicate with skeleton control program under ROS. This unique mechanism encourages the use of Node and Topic. A node is an executable that uses ROS to communicate with other nodes. In this paper, both the 2D face recognition and 3D face recognition programs are nodes. However, nodes do not communicate with each other directly, they publish messages on a topic and subscribe to it to receive messages (see Fig. 6).

After the 2D face recognition, it would publish a message containing the name of the person on a specific topic. The skeleton control node subscribes to the same topic all the time, receiving the name information of the person, and compare it with the controller's name. Only after the 2D identification will the skeleton control node show a message like "2D recognition successfully!" on the screen. The 3D face recognition node performs in the similar way, after the recognition, this node will publish a message like "3D recognition successfully!" to a specific topic. Once the skeleton control node received the 2D and 3D recognition messages, it switches control mode from "off" to "on", and allows the skeleton control.

In 2D face recognition, a rectangle on the face of the correct person is drawn on the image and the central point of the rectangle is picked up as a reference point. Under the ROS mechanism, the coordinate data of the reference point is published to another specific topic, and is also received by the skeleton control. This coordinate data is compared with the position of head frame of the skeleton. These two position data are calibrated with a suitable threshold. Only if the correct person is recognized as controller, the message "You are the right person" would be shown on screen. Meanwhile, the person's control is valid with skeleton.

E. Skeleton control

Several poses are defined as commands to control the different action of the TurtleBot (see Fig. 7): 1) Raising left head means stop. 2) Raising right hand means begin. 3) Stretching out right hand from forward to $45°$ right deflection means going forward (see Fig. 8). 4) Stretching out right hand from backward to $45°$ left deflection means going backward (see Fig. 9). 5) Stretching out right hand from side direction to $45°$ left deflection means rotating counter-clockwise (see Fig. 10). 6) Stretching out right hand from side direction to $45°$ right deflection means rotating

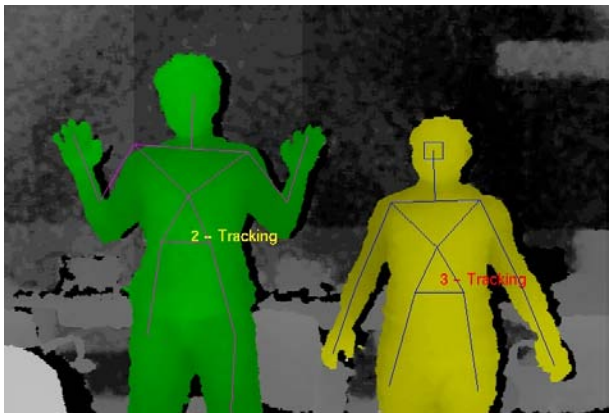


Fig. 5: Skeleton detection

Fig. 6: ROS communication mechanism

clockwise (see Fig. 11). The further the controller stretches his hand, the faster the TurtleBot moves.

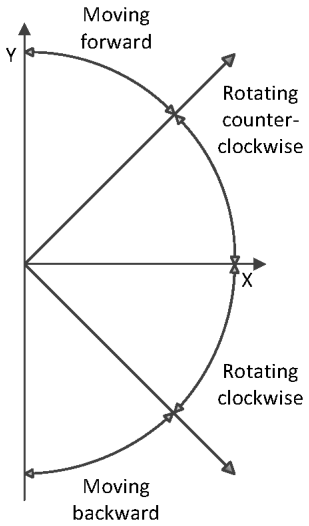


Fig. 7: Top view of commands by hand stretching

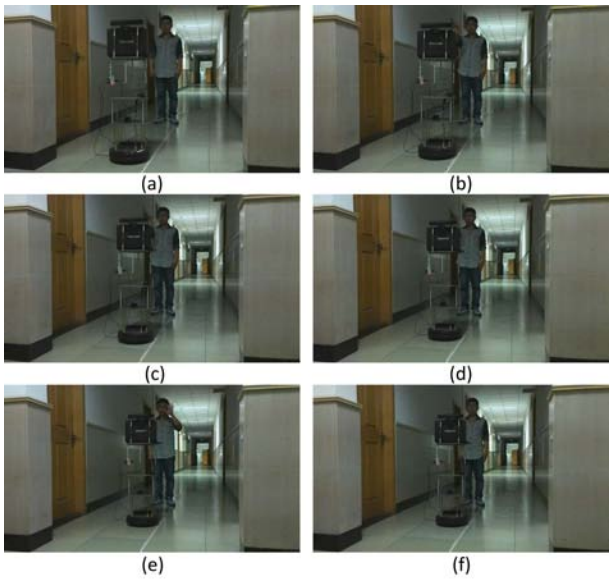


Fig. 8: Moving forward

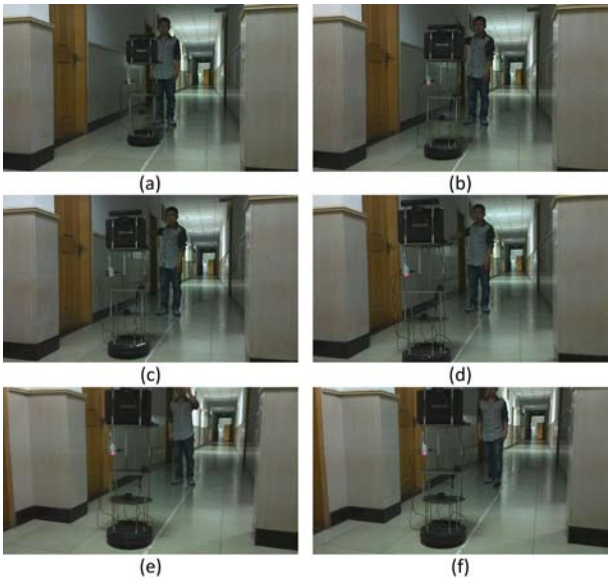


Fig. 9: Moving backward

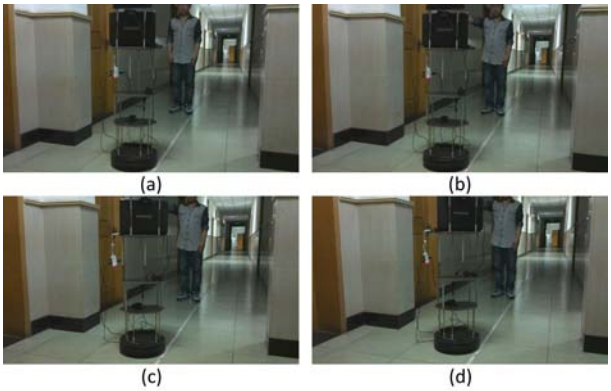


Fig. 10: Rotating counter-clockwise

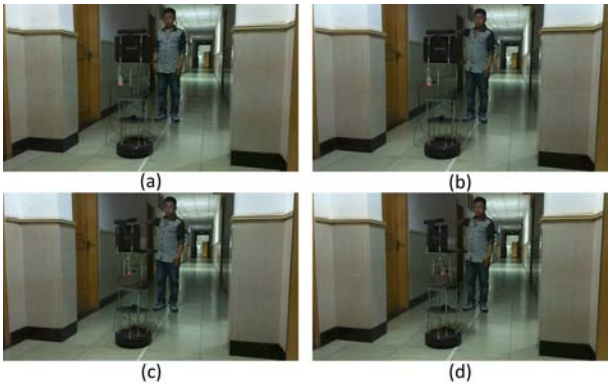


Fig. 11: Rotating clockwise

IV. EXPERIMENTS AND RESULTS

Our proposed method is implemented on the TurtleBot which is operated under ROS. First, the face recognition is testified in different conditions: 1) the target person changes his pose and position, 2) multiple persons stand in front of the camera. Second, the exclusive human-robot interaction is verified in real situation.

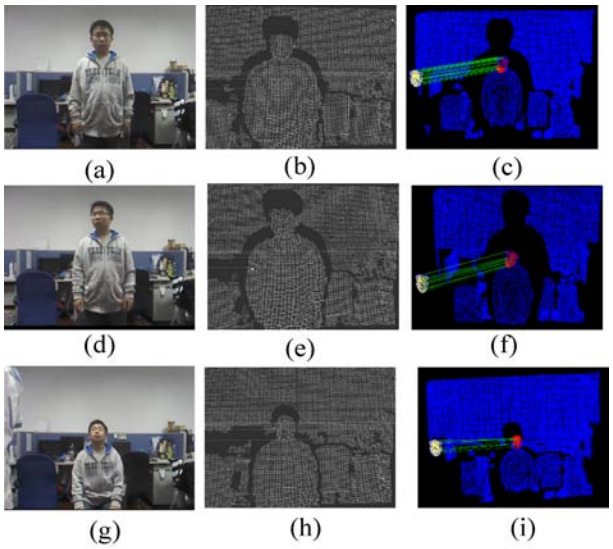


Fig. 12: Three circumstances of 3D face recognition. White face in (c) is the model point cloud shown in Fig. 10. Red face is the correspondence in scene point cloud. (a) A point cloud image of controller standing with head facing forward, (b) A point cloud image of controller standing with head facing forward, (c) 3D face recognition when controller stands with head facing forward, (d) A 2D image of controller standing with head rotating horizontally about 60°, (e) A point cloud image of controller standing with head rotating horizontally about 60°, (f) 3D face recognition when controller stands with head rotating horizontally about 60°, (g) A 2D image of controller sitting down with head rotating vertically about 45°, (h) A point cloud image of controller sitting down with head rotating vertically about 45°, (i) 3D face recognition when controller sits down with head rotating vertically about 45°.

A. Experiment of 3D face recognition

1) Experimental setup: In order to achieve the 3D face recognition, we captured a 3D image of the controller and stored it as PCD (Point Cloud Data) file format [16]. PCD file format has many advantages on flexibility and speed: 1) Higher ability to store and process organized data sets, which is of extreme importance for real time application. 2) Binary mmap/munmap data type to achieve the highest speed of saving and loading data to disk. 3) Many different data types including char, integer, float and double to store, which ensures the point cloud data to be flexible and efficient with respect to storage and processing. 4) n-D histograms for feature descriptor-very important for 3D perception and computer vision application. The camera was about 1 m in front of controller, and we used point cloud segmentation to pick out the face as model point cloud (see Fig. 13).

2) 3D face recognition on different poses and positions: In the first experiment, the controller performed three different poses and positions i.e. 1) Standing in front of the camera with head facing forward. 2) Standing in front of the camera with head rotating an angle about 60° horizontally. 3) Sitting in front of the camera with head rotating an angle about 45° vertically. The result shows a good recognition effect under the three circumstances (see Fig. 12).

Fig. 13: Model point cloud (face of controller)

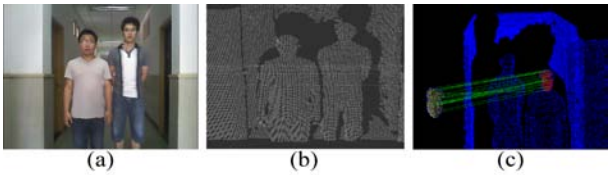


Fig. 14: 3D face recognition when multiple persons standing in front of the camera

3) 3D face recognition on multiple persons: In the second experiment, two persons stood in front of the camera with their head facing forward, our method was also testified to have a robust recognition effect (see Fig. 14).

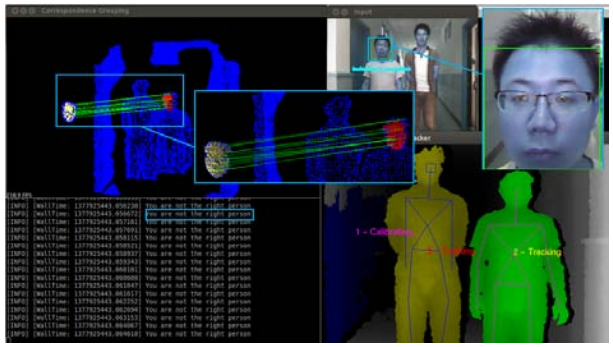
B. Experiment of skeleton control

This experiment demonstrated a clear image of the whole process of our method. We asked two persons to stand in front of the TurtleBot, and the TurtleBot drew a rectangle at the head frame of the first skeleton being detected. In the first case, the skeleton of the other person was detected at first. Because the face position and the head frame position does not match with each other, the massage "You are not the right person!" was printed on the screen, and TurtleBot was not controlled by the skeleton (see Fig. 15). In the other case, the skeleton of the correct person was detected at first. After being recognized successfully, the three messages: "2D face recognized successfully!", "3D face recognition successfully!" and "You are the right person" were printed on the screen, and the TurtleBot was easily controlled by the skeleton of the person (see Fig. 16).

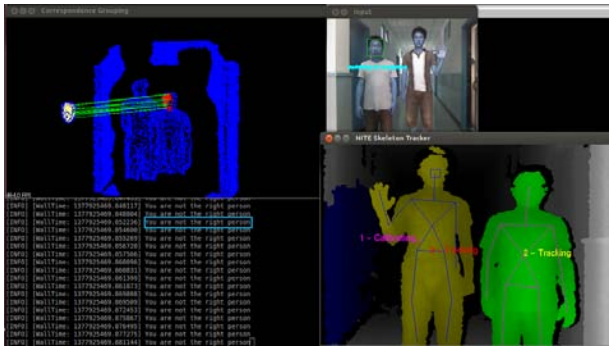
V. CONCLUSIONS AND FUTURE WORKS

In this paper, we proposed an exclusive human-robot interaction method on the TurtleBot platform. This method introduced several novelties including 1) By showing different poses of skeleton like raising hand or stretching hand, the TurtleBot platform receives different commands, thus reacts with the appropriate behaviors as the controller wishes. 2) With the cooperation of 2D and 3D face recognition, the TurtleBot is able to recognize the correct person, which makes it exclusive to control.

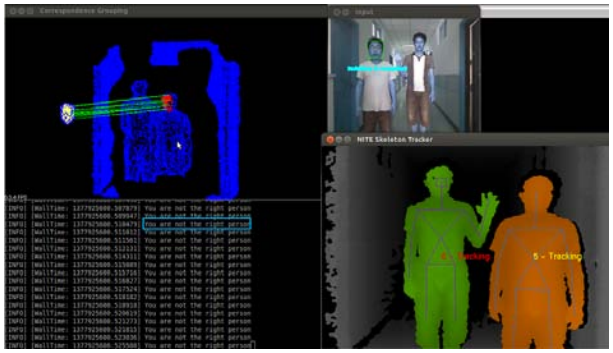
Although the 3D face recognition module of our method works well with common faces, it can fail when too many data points are lost. Due to the large volume of the data



(a)

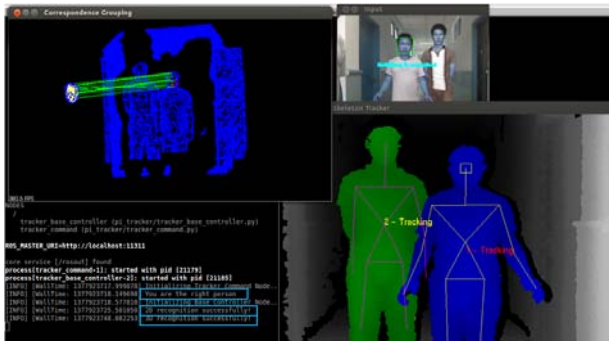


(b)

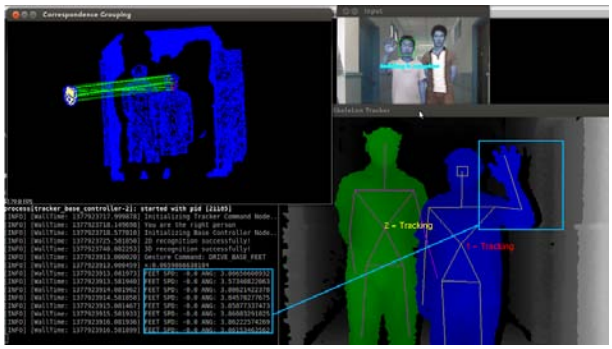


(c)

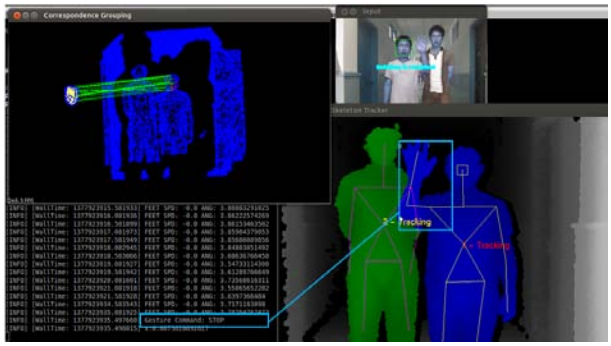
Fig. 15: Skeleton control failed. (a) Head frame position did not match with face position. (b) No reaction on TurtleBot when the person raises right hand. (c) No reaction on TurtleBot when the person raises left hand.



(a)



(b)



(c)

Fig. 16: Skeleton control by the correct person. (a) Successful recognition (b) TurtleBot moved forward when the person stretched out right hand. c) TurtleBot stopped when the person raised left hand.

calculation, it also spends a long time. Thus, in our method, the 3D face recognition is performed only once at the beginning of the control. In addition, the skeleton detection module is still not very robust when the TurtleBot is rocked or shook, which often happens in real application. Future work regarding the design includes: 1) speeding up the 3D face recognition, 2) enhancing the recognition accuracy, and 3) strengthening the robustness of skeleton detection so that the exclusive human robot interaction method can be achieved in a more complex and realistic experimental setting.

VI. ACKNOWLEDGMENTS

This work was partially supported by the National Natural Science Foundation of China under grants No.51205244, and Open Research Foundation of State Key Lab. of Digital Manufacturing Equipment and Technology in Huazhong University of Science and Technology.

REFERENCES

[1] Voyles, Richard M., J. Dan Morrow, and Pradeep K. Khosla. "Towards gesture-based programming: Shape from motion primordial learning of sensorimotor primitives." Robotics and Autonomous Systems 22.3 (1997): 361-375.

[2] Saunders, Joe, Chrystopher L. Nehaniv, and Kerstin Dautenhahn. "Teaching robots by moulding behavior and scaffolding the environment." Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction. ACM, 2006.

[3] Ogawa, Hiroki, and Tomio Watanabe. "InterRobot: a speech driven embodied interaction robot." Robot and Human Interactive Communication, 2000. RO-MAN 2000. Proceedings. 9th IEEE International Workshop on. IEEE, 2000.
[4] Bischoff, Rainer, and Volker Graefe. "Integrating vision, touch and natural language in the control of a situation-oriented behavior-based humanoid robot." Systems, Man, and Cybernetics, 1999. IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on. Vol. 2. IEEE, 1999.
[5] Brueckmann, Robert, Andrea Scheidig, and H-M. Gross. "Adaptive noise reduction and voice activity detection for improved verbal human-robot interaction using binaural data." Robotics and Automation, 2007 IEEE International Conference on. IEEE, 2007. 1858-1870.
[6] Waldherr, Stefan, Roseli Romero, and Sebastian Thrun. "A gesture based interface for human-robot interaction." Autonomous Robots 9.2 (2000): 151-173.
[7] Hasanuzzaman, Md, et al. "Adaptive visual gesture recognition for humanCrobot interaction using a knowledge-based software platform." Robotics and Autonomous Systems 55.8 (2007): 643-657.
[8] Coetzee, Louis, and Elizabeth C. Botha. "Fingerprint recognition in low quality images." Pattern Recognition 26.10 (1993): 1441-1460.
[9] Yang, Jinfeng, and Xu Li. "Efficient finger vein localization and recognition." Pattern Recognition (ICPR), 2010 20th International Conference on. IEEE, 2010.
[10] Daugman, John. "How iris recognition works." Circuits and Systems for Video Technology, IEEE Transactions on 14.1 (2004): 21-30.
[11] Lee, Yeung-hak, and Jae-chang Shim. "Curvature based human face recognition using depth weighted hausdorff distance." Image Processing, 2004. ICIP'04. 2004 International Conference on. Vol. 3. IEEE, 2004.
[12] nan, Tolga, and Ugur Halici. "3-d face recognition with local shape descriptors." Information Forensics and Security, IEEE Transactions on 7.2 (2012): 577-587.
[13] Spreeuwers, Luuk. "Fast and accurate 3d face recognition." International journal of computer vision 93.3 (2011): 389-414.
[14] Lienhart, Rainer, and Jochen Maydt. "An extended set of haar-like features for rapid object detection." Image Processing. 2002. Proceedings. 2002 International Conference on. Vol. 1. IEEE, 2002.
[15] Tombari, Federico, and Luigi Di Stefano. "Object recognition in 3D scenes with occlusions and clutter by Hough voting." Image and Video Technology (PSIVT), 2010 Fourth Pacific-Rim Symposium on. IEEE, 2010.
[16] $http://pointclouds.org/documentation/tutorials/pcd_file_format.php$