

Optimal Robot Selection By Gaze Direction in Multi-Human Multi-Robot Interaction

Lingkang Zhang and Richard Vaughan
Autonomy Lab, Simon Fraser University
{lza90, vaughan}@sfu.ca

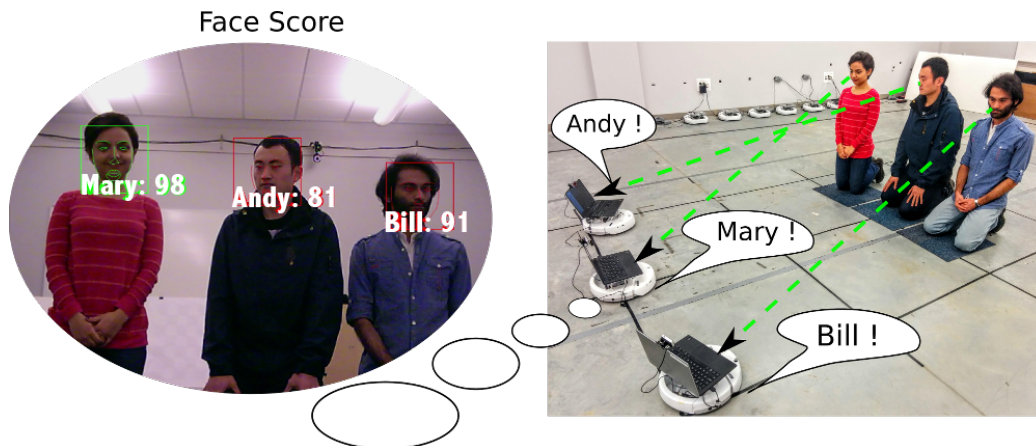


Fig. 1: Multi-Human Multi-Robot Interaction

Abstract—This paper presents a computer vision based system for interaction between multiple humans and multiple robots. Each human can “select” (obtain the undivided attention of) a robot by simply looking directly at it. This extends previous work whereby a single human can select one or more robots from a population. Each robot optimally assigns human identities to tracked faces in its camera view using a local Hungarian algorithm. The gaze-direction and location of the faces are estimated via vision, and a score for each robot-face pair is assigned. Then the system finds the global optimal allocation of robot-to-human selections using a centralized Hungarian algorithm. A useful feature of this method is that robots can be selected by people they cannot see. This is the first demonstration of optimal many-to-many robot-selection HRI.



Fig. 2: “Indirect” gaze based interaction

I. INTRODUCTION

We seek a system of multiple humans and robots where each human can select a single robot to work with by simply looking directly at it as in Figure 1. In humans and other social animals, gaze-direction serves an important role in regulating the communication between individuals [1]. In a multiple human interaction scenario as shown in Figure 2, a human can be told that someone else is seeking his attention even when he can not see that person himself. In our multi-robot system, information about detected user locations and gaze directions is shared among the robots via a centralized server. Thus a robot can be aware that a user is gazing at it even when that person is not visible from its own camera.

II. RELATED WORK

Previous HRI work on attracting the attention of robots has examined single-human multi-robot interaction [2], [3], [4], [5] and multi-human single-robot interaction [6].

The work in [5] and [6] uses sensors such as Kinect, laser scanners and microphones to enable the robot to detect the human’s attention. In [2], [3] and [4], a proxy for gaze direction is estimated from a template-matching face detector, and used to decide how likely the human is to be paying attention to a robot. Recently a face detection method based on face landmarks was introduced in [7], which provides a direct estimate of gaze direction. Both methods only require a monocular camera for a robot, which

is low-cost, simple to install and easy to use.

In a multiple robot system, information sharing among robots can help during the human robot interaction. A method of multiple robot collaboration to learn hand gesture effectively during human robot interaction is introduced by Nagi in [8], where the multiple viewpoints of the robots are used to disambiguate observed gestures. Gerkey [9] shows that the Multi-Robot Task Allocation problem (MRTA) can be optimally solved by the Hungarian algorithm.

In this paper we extend Couture-Beil’s method for single-human multi-robot selection [2] to multiple humans by incorporating Emami’s face recognition to determine identity¹ and Kazemi’s face landmark detection [10] together with EPFL CHILI Lab’s attention tracker² to determine face location and gaze direction. In a first stage, on each robot individually, we find the optimal allocation of human identities to tracked faces, then in a second stage, we pass this information to a central server to globally optimally allocate each robot to the human who is looking at it most directly.

Although gazed-based attention-getting by explicit collaboration has been shown before, this is the first demonstration of gaze-based many-to-many robot-selection HRI using the information shared ‘in the cloud’. We also have the unique and appealing feature that a robot can be selected by a person it can not see. Indeed we show that a globally correct allocation can be achieved in practice even when no robot could see the person that selected it.

III. SYSTEM DESIGN

We describe a multi-human multi-robot allocation process based on face identity and landmark information (Figure 3).

We use three iRobot Create generic mobile robots equipped with low-cost laptops (Intel i5, 4GB RAM). One of the laptops is also used as the centralized server. Robots are globally localized using an external Vicon motion capture system, though any reasonably accurate localization system could be substituted (Figure 4).

The system workflow is:

- 1) A robot detects faces in the video stream from its camera, and the identity of each detected face is recognized using a pre-trained face identity classifier using the ROS face_recognition package¹. For each face, a confidence is assigned for each identity.
- 2) Each detected face is tracked over time. Each tracker is optimally assigned an identity using the Hungarian algorithm (see below). This idea is due to Felix [11] who applied the Hungarian algorithm to identify tracked vehicle types.
- 3) The 6-DOF pose of each tracked face is estimated from the current image.
- 4) Each robot learns its global pose via WiFi from the external motion capture system.

¹ROS face_recognition Package, Author: Pouyan Ziafati, Access on: http://wiki.ros.org/face_recognition

²Attention Tracker, CHILI Lab, Access on: <https://github.com/chili-epfl/attention-tracker>

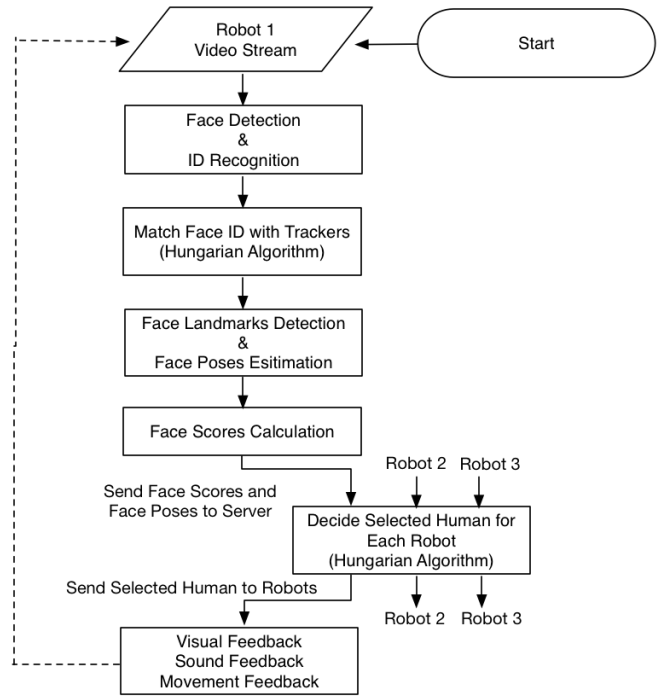


Fig. 3: Flowchart of the multi-human multi-robot allocation process

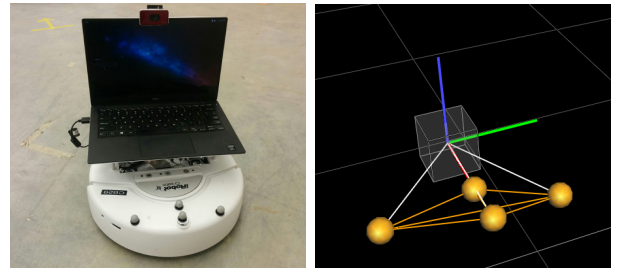


Fig. 4: Robot platform and Vicon motion capture system

- 5) Each robot calculates a “face score” for each tracked face for every robot. The face score is an estimate of how directly a face is looking at that robot (details Section III-B).
- 6) All robots send their face scores for each human identity to a central server, which uses the Hungarian algorithm to determine which robot is most likely being looked at by which human. Face scores below a threshold are ignored, so humans not looking at a robot are not assigned a robot.
- 7) Each selected robot gives audio, visual and movement feedback indicating which human has selected it (if any).

A C++ implementation of Hungarian algorithm² is used to solve the allocation problem in both step 2) and step 6). The algorithm guarantees a globally optimal allocation of n tasks to n bidders. While the time complexity of the Hungarian algorithm is $O(n^3)$, the running time is acceptable for the

small values of n considered here.

A. Face Identity Recognition and Tracking

Each robot is equipped with a camera in this system. The ROS face_recognition package¹ is used for multiple face detection and face identity recognition. The package provides human face identity classifier training. We implemented a simple procedure for the robot to automatically collect the human face data and train a classifier for face identity recognition. The user enters her name before training then shows her face to the robot. It takes less than two minutes to train a classifier for new user. The classifier obtained by one robot is shared via wireless network with other robots.

The ROS face_recognition package¹ takes more than 300 msec to process a single frame when we want to detect multiple faces in an image. To improve the performance for real-time application, we replaced the multiple object detector which comes with this package with the multiple object detector in Dlib³ which reduces the processing time for a single image from 300 msec to 30 msec. Though the working distance of the multiple object detector in Dlib is around 2 meters comparing to the 4 meters of the ROS package, it meets our requirements for a short range indoor HRI.

The ROS face_recognition package¹ detects faces and recognizes the face identity in the video stream by checking the confidence of each detected face on different identities, and a face identity will be assigned to a face if it has the highest confidence on this face, with confidence above a threshold. This method is simple but leads to problems when dealing with multiple-face recognition in the real-time video stream. First, it is possible that multiple faces in the image be assigned with the same identity. Second, the recognition result can be unstable since the face identity assignment is done on a per-frame basis.

To address these problems, we implemented a face tracking method which optimally matches the face identity with the faces detected. When a new face is detected in the image, we initiate a new face tracker. The face tracker is a rectangular bounding box which searches the area of twice the face size in the previous frame. Instead of using the result from the ROS face_recognition package to assign the face identity to a face tracker directly, we maintain a window in each face tracker to buffer the identity recognition in five continuous frames as shown in Figure 5. Thus we can obtain a face confidence matrix as Table I to indicate how many times the classifier thinks the face in a certain face tracker matches a face in five continuous frames. Then we use the Hungarian algorithm to optimally match each face in the face trackers and the identities. The result is shown in Table II. The face tracker is counted as effective when it appears in at least 5 continuous frames, and a face tracker will be removed if it cannot be detected in three continuous frames.

²Kuhn-Munkres (Hungarian) Algorithm in C++, Author: John Weaver, Access on: <https://github.com/saebyn/munkres-cpp>

³Dlib, Author: King, DE, Access on: <http://dlib.net>

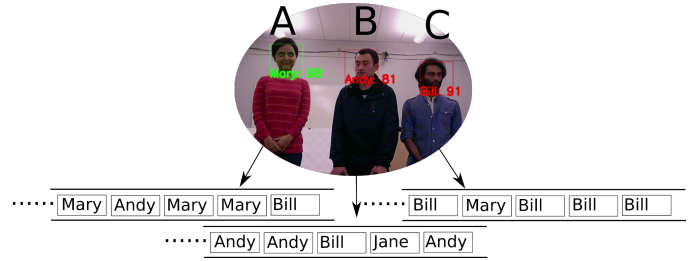


Fig. 5: Face trackers and identities matching

The advantage of using this method is that it greatly reduces the frequency of false positives caused by transient detections of faces on background objects, and it maintains a stable continuous detection of multiple faces and face identity recognition.

Tracker / Identity	Mary	Andy	Bill
A	3	1	1
B	1	3	1
C	1	0	4

TABLE I: Face identities confidence matrix

Identity	Mary	Andy	Bill
Tracker	A	B	C

TABLE II: Face identities and trackers matching result using Hungarian algorithm

B. Face Pose Estimation

The face landmark (Figure 8) is obtained using Dlib³ based on the method described in [10]. The face pose is estimated using the method described in [7]. Eight interesting feature points (right eye, left eye, right ear, left ear, sellion, menton, nose and stomion) are selected from the 68 face landmarks obtained from Dlib. Those 2D feature points are matched with 3D feature points of a virtual human head centered at the camera center as described in [7] using the *solvePNP* function implemented in OpenCV [12] to obtain a 6-DOF pose ($x, y, z, roll, pitch, yaw$) of the detected face relative to the 3D virtual human head.

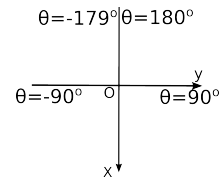


Fig. 6: Coordinate system

In the following we make the simplifying assumption that the humans and robots are aligned on a common vertical axis and thus we project cameras and face poses onto a 2D plane. Thus we are concerned with 3-DOF poses of x, y translation and yaw angle. The coordinate system we use is indicated in Figure 6, where x axis points forward direction and y axis points left direction, and yaw is ranging from

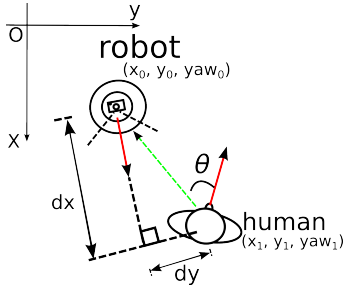


Fig. 7: Face pose estimation

-179° to 180° . All angle calculations are normalized to this range. Figure 7 shows the process of face pose estimation. The global pose of the robot is $P_0(x_0, y_0, yaw_0)$, and the global pose of the face is $P_1(x_1, y_1, yaw_1)$. The red vector indicates the facing direction of the robot and the face. The green vector, l , indicates the direction from the face to the robot. The angle between the gazing direction of the face and l is θ (θ is a negative value in the case shown in Figure 7). The translation of the face relative to the robot frame is dx and dy . The dx translation and dy translation are obtained using the method in [7] as described above. In order to get a better θ and yaw_1 angle here, instead of estimating the face's rotation relative to the 3D virtual human head, we moved the face to the center of the image and use the *solvePNP* again and get the Rotation Matrix R using the *Rodrigues* OpenCV function, then we use the method in [13] to calculate angle θ . The global pose of the robot P_0 is obtained from the Vicon motion capture system. Thus we can calculate out the global pose of the face P_1 by using Equation 1, 2 and 3. P_1 and θ are also part of the face tracker mentioned in Section III-A.

$$x_1 = x_0 + dx \times \cos yaw_0 - dy \times \sin yaw_0 \quad (1)$$

$$y_1 = y_0 + dx \times \sin yaw_0 + dy \times \cos yaw_0 \quad (2)$$

$$yaw_1 = yaw_0 + \theta + \text{atan2}(dy, dx) + 180^\circ \quad (3)$$

C. Face Score and Matching

The proposed face score is a more principled alternative to the original template-match count used in [2], and is used to indicate the directness of a face's gaze to a certain robot. Each robot calculates a "face score" for all faces it is tracking, based on the angle of the face as shown in Figure 8. The face score is calculated using Equation 4.

$$\text{score} = \max(100 - |\theta|, 0) \quad (4)$$

where θ is the angle in degree between vector l and the gaze direction of the human as mentioned in Section III-B, and the score will be assigned to zero if it is less than zero.

Each robot also calculates face scores for each face it tracks with respect to all other robots based on global poses. The process of the calculation of these face scores is basically an inverse process of the face global pose calculation as mentioned in Figure 7. Given a robot global pose $P_0(x_0, y_0, yaw_0)$ and a face global pose $P_1(x_1, y_1, yaw_1)$,



Fig. 8: Face landmark and face score based on gaze-direction

we use Equation 5 and 6 to calculate θ which can be used to calculate the face score using Equation 4.

$$\theta = yaw_1 - \text{atan2}(y_0 - y_1, x_0 - x_1) \quad (5)$$

$$\theta = \begin{cases} \theta - 360^\circ, & \text{if } \theta > 180^\circ \\ \theta + 360^\circ, & \text{if } \theta < -180^\circ \\ \theta, & \text{otherwise} \end{cases} \quad (6)$$

Now that a robot can calculate the face score of a face to both itself and other robots, we can consider a multi-human multi-robot interaction scenario in Figure 9, where robot_0 can see Mary and Andy, robot_1 can see all the three humans, and robot_2 can only see Bill.

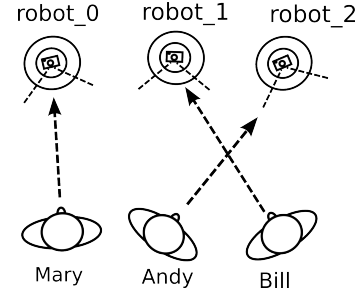


Fig. 9: Calculate face score in multi-human multi-robot scenario

	Mary	Andy	Bill		Mary	Andy	Bill
robot_0	95	20	-1	robot_0	93	24	70
robot_1	72	48	-1	robot_1	70	50	90
robot_2	15	90	-1	robot_2	17	94	40
	Mary	Andy	Bill				
robot_0	-1	-1	72				
robot_1	-1	-1	88				
robot_2	-1	-1	44				

TABLE III: Face score matrix given by Robot_0, Robot_1 and Robot_2

	Mary	Andy	Bill
robot_0	94	22	71
robot_1	71	49	89
robot_2	16	92	42

TABLE IV: Face score matrix in the centralized server

A robot gives a face score of -1 to all identities that is not currently tracking. The central server discards all negative scores, and averages all positive scores received for each

Face	Mary	Andy	Bill
Robot	robot_0	robot_2	robot_1

TABLE V: Robots and humans matching result using Hungarian algorithm

face. The Hungarian algorithm is then used to optimally match the robot to the identity of the person looking most directly at it. If the matching score is less than a threshold, the server will reject the match and not inform the robot of it, so that a face not gazing at a robot will not be matched with a robot. The face scores for a robot-face pair given by different robots are slightly different from each other as be discussed in Section IV-C. From Table III we can calculate the score matrix in the server which is shown in Table IV, and the matching results obtained by the Hungarian algorithm are shown in Table V.

D. Robots' Feedback

Once a robot is notified that a human is gazing it, the robot performs sound, visual and movement feedback.

The robot will speak the name of the matched human. The robot will also display his/her face in a green bounding box if the face is in its view as shown in Figure 1. And if the matched face is in the view of the robot, the robot will turn left or right to maintain the face is in the center of the view with a deviation of 10° , and drive toward the face until their distance is around 35 cm . Otherwise, if the robot cannot see the matched face as shown in Figure 10, the robot needs to calculate the angle α between the vector l from the robot to the face and the facing direction of the robot. The facing direction yaw_0 of the robot is obtained from Vicon motion capture system. The angle of l can be calculate using equation 7.

$$\alpha = \text{atan2}(y_1 - y_0, x_1 - x_0) - yaw_0 \quad (7)$$

Since we are using the coordinate system described in III-B, the robot should turn left if α is positive or right if α is negative. There exists a situation when the face is in the view of the robot while it is too far for the face to be detected and recognized. Thus we make the robot drive forward if $|\alpha|$ is less than 10 degrees. Once the robot can see the matched face, it will use the movement strategy mentioned in the paragraph above.

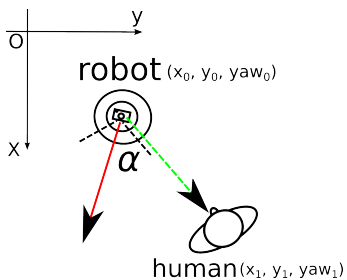


Fig. 10: Robot's movement feedback when it cannot see the matched face

IV. EVALUATION

A. Processing Time

Table VI illustrates the processing time of the whole system. The face recognition section is the most costly component. The average time to process a single video frame is 151 ms, so we have an update frequency better than 6 Hz.

Section	Face recognition	Pose	Score	Matching	Total
Time(msec)	75	34	23	19	151

TABLE VI: Processing time

B. Face Pose Estimation Evaluation

The face pose estimation is evaluated with the method shown in Figure 11. A human is moving in the green trajectory in a $2m \times 2m$ area with rotational movement for 2 minutes. The human wears a helmet with markers on it to enable the Vicon motion capture system to obtain the ground truth pose of the face. Figure 12 shows the distance between the real location and estimated location of face over time with an average value of 17.68 cm. Figure 13 shows the real face angle comparing with the estimated face angle overtime with an average difference value of 12 degrees.

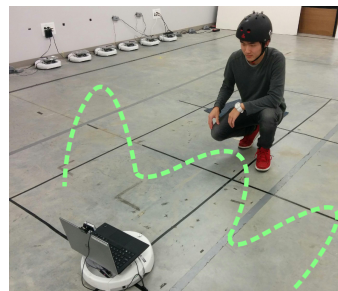


Fig. 11: Face pose estimation evaluation

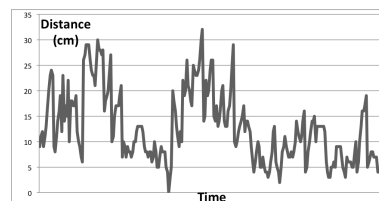


Fig. 12: Face localization error: distance between estimated face location and ground truth

C. Face Score Estimation Evaluation

The face score is evaluated as shown in Figure 14. robot_0 estimates the face score of the human face for robot_1. This experiment consists of four rounds in which robot_1 will be placed at four different locations respectively: A, B, C and D. The estimated face score for robot_1 from robot_0 and the score given by robot_1 itself are compared. For each round, the human will keep changing the face direction for 2 minutes, and the score error for each video frame is recorded. The average face score errors for each round are in Table VII.

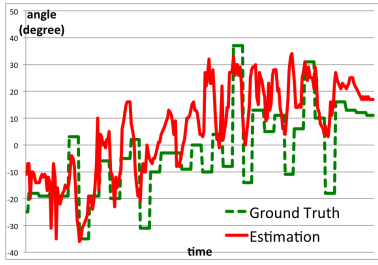


Fig. 13: Estimated face angle and ground truth

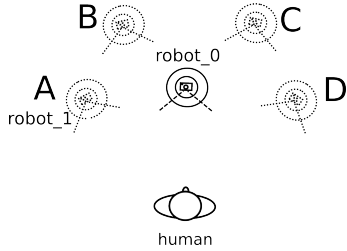


Fig. 14: Face score estimation evaluation

Location	A	B	C	D	average
Error	11.46	9.42	9.64	16.28	11.7

TABLE VII: Face score estimation error

D. Face Identities Matching Evaluation

The face identity tracker is evaluated as shown in Figure 15. All three humans are in the camera view of the robot. All the humans move their gaze direction during the experiment, and the robot logs the position data of each recognized face identity for every video frame. Three rounds of tests are performed with humans differently positioned in each round, and every round takes two minutes. The logged position data are compared with the labelled ground truth, and a video frame is counted as success when all of the three face identities are correctly matched with the faces. The success rate for each round is in Table VIII.

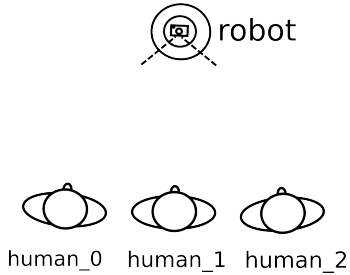


Fig. 15: Face tracker evaluation

Round No.	1	2	3	average
Success rate	84%	80%	79%	83%

TABLE VIII: Face identities matching success rate

E. Multi-human Multi-robot Experiments

Our test scenarios are shown in Figure 16, with three humans and three robots. We consider six scenarios where in a, b and c, all the robots can see all the people, in d, e and f, at least one robot can **not** see the person that selects it.

- a. Humans gaze at the robots directly in front of them.
- b. Two humans gaze at robots not in front of them, and one human does not look at any robot.
- c. One human gazes at the robot in front of her, one human gazes at the robot not in front of them, and one human does not look at any robot.
- d. Humans gaze at the robots directly in front of them. Only two robots can see all the humans and one robot can not see anyone.
- e. Humans gaze at the robots directly in front of them. Only one robot can see all the humans and the other two robots can not see anyone.
- f. Two humans gaze at the robots not in front of them, and one human does not look at any robot. Only one robot can see all the humans and the other two robots can not see anyone.

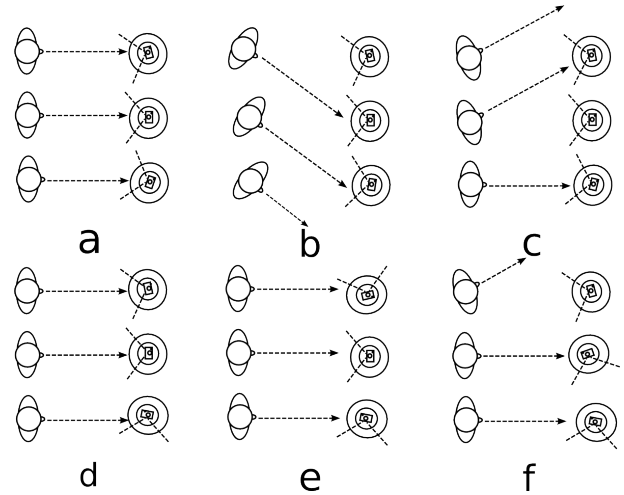


Fig. 16: Multi-human multi-robot experiments

An experiment case is counted as success only when all the robots approach the correctly matched humans. In all of the six scenarios above, the robots are allocated to the correct human in most of the cases as shown in Table IX and Table X.

Exp.	Trial				
	1	2	3	4	5
a	35s	32s	37s	38s	33s
b	36s	fail	45s	40s	43s
c	34s	36s	fail	31s	48s
d	34s	47s	fail	36s	49s
e	56s	45s	44s	55s	60s
f	34s	fail	34s	fail	43s

TABLE IX: Multi-human multi-robot experiments

Experiment	a	b	c	d	e	f
Success rate	5/5	4/5	4/5	4/5	5/5	3/5
Average Time (s)	35s	41s	37s	41s	52s	37s

TABLE X: Result of multi-human multi-robot experiments

V. DISCUSSION

Although this is the first attempt to introduce the gaze-based robot selection method in the multi-human multi-robot interaction, the experiment results are quite encouraging. Section IV-A shows that the system operates at a frequency more than 6 Hz using a low-cost robot platform mentioned in III, which is fast enough in practice for a real-time HRI system if people move slowly.

Within an interaction area of $2m \times 2m$, Section IV-B shows that the average face location (x, y) error is 17.68 cm, which means that the resolution of the face location estimation is better than 1%. And section IV-B shows an average pose estimation error of 12 degrees when the faces rotates in a range of $(-40 \text{ degrees}, 40 \text{ degrees})$, thus the angle estimation has got an accuracy higher than 85%. A good estimation of the face pose of a robot helps its estimation of the face score for other robots. Section IV-C shows an average score estimation error of 11.7, which is 11.7% of the score range $(0, 100)$. And section III-A shows the face tracker method we introduced in III-A has an accuracy of 83% to match the detected faces with identities in the camera view of the robot.

The integrated system has a success rate higher than 80% as shown in Section IV-E. In most of the failed experiments cases, only one of the three robots failed to approach the matched person. The field of view of the camera we use on the robot platform is 60 degrees, thus there exists a situation when one of the humans cannot be seen by any of the robots. Since all the humans are almost stationary during the experiments the system cannot recover. If the humans were allowed to move, this failure mode would be less likely. Cameras with wider FOV could also help with this issue.

VI. CONCLUSION AND FUTURE WORK

We describe the first demonstration of uninstrumented people selecting a robot for further interaction in a multi-human multi-robot setting. A robot can be selected by a human with the help of other robots even when it cannot see the human in its camera view. The allocation method is optimal given the available preprocessed face tracking data.

We plan to increase the robustness of the system, and in particular to extend the range of the face tracking system. Since faces are hard to recognize far away, we plan to examine the ability of the tracker to handle very sparse identity data.

The system could be extended to use human facial expressions, mouth movement and head movement as more sophisticated interaction cues in the multi-human multi-robot interaction. In indoor environments where there are obstacles and humans, we can incorporate navigation and planning methods such as [14] to enable the robots to reach the

selected human faster and improve the performance of the whole system. We can also apply the method to outdoor human robot interaction which involves both the UAV and UGV as an extension of the work described in [6], [15] and [16].

ACKNOWLEDGMENT

This work was supported by the NSERC Canadian Field Robotics Network.

REFERENCES

- [1] A. Kendon, "Some functions of gaze-direction in social interaction," *Acta psychologica*, vol. 26, pp. 22–63, 1967.
- [2] A. Couture-Beil, R. T. Vaughan, and G. Mori, "Selecting and commanding individual robots in a multi-robot system," in *Computer and Robot Vision (CRV), 2010 Canadian Conference on*. IEEE, 2010, pp. 159–166.
- [3] B. Milligan, G. Mori, and R. T. Vaughan, "Selecting and commanding groups in a multi-robot vision based system," in *Proceedings of the 6th international conference on Human-robot interaction*. ACM, 2011, pp. 415–416.
- [4] S. Pourmehri, V. M. Monajjemi, R. Vaughan, and G. Mori, "You two! Take off!": Creating, modifying and commanding groups of robots using face engagement and indirect speech in voice commands," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. IEEE, 2013, pp. 137–142.
- [5] S. Pourmehri, V. Monajjemi, J. Wawerla, R. Vaughan, and G. Mori, "A robust integrated system for selecting and commanding multiple mobile robots," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 2874–2879.
- [6] S. Pourmehri, J. Bruce, J. Wawerla, and R. Vaughan, "A sensor fusion framework for finding an HRI partner in crowd," in *IEEE Int. Conf. on Intelligent Robots and Systems, Workshop on Designing and Evaluating Social Robots for Public Settings*, 2015.
- [7] S. Lemaignan, F. Garcia, A. Jacq, and P. Dillenbourg, "From real-time attention assessment to with-me-ness in human-robot interaction," in *11th ACM/IEEE International Conference on Human-robot Interaction (HRI 2016)*. ACM/IEEE, 2016.
- [8] J. Nagi, H. Ngo, L. Gambardella, and G. A. Di Caro, "Wisdom of the swarm for cooperative decision-making in human-swarm interaction," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1802–1808.
- [9] B. P. Gerkey, "On multi-robot task allocation," Ph.D. dissertation, University of Southern California, 2003.
- [10] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 1867–1874.
- [11] F. Lütteke, X. Zhang, and J. Franke, "Implementation of the hungarian method for object tracking on a camera monitored transportation system," in *Robotics; Proceedings of ROBOTIK 2012; 7th German Conference on*. VDE, 2012, pp. 1–6.
- [12] G. Bradski, "OpenCV," *Dr. Dobbs' Journal of Software Tools*, 2000.
- [13] G. G. Slabaugh, "Computing euler angles from a rotation matrix," *Technical Report*, 1999, <http://www.staff.city.ac.uk/~sbbh653/publications/euler.pdf> (visited: 29-July-2016).
- [14] J. Bruce and M. Veloso, "Real-time randomized path planning for robot navigation," in *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, vol. 3. IEEE, 2002, pp. 2383–2388.
- [15] M. Monajjemi, J. Bruce, S. A. Sadat, J. Wawerla, and R. Vaughan, "UAV, do you see me? establishing mutual attention between an uninstrumented human and an outdoor uav in flight," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 3614–3620.
- [16] V. M. Monajjemi, J. Wawerla, R. Vaughan, and G. Mori, "HRI in the sky: Creating and commanding teams of uavs with a vision-mediated gestural interface," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. IEEE, 2013, pp. 617–623.