Optimal Gaze-based Attention System for Multi-human Multi-robot Interaction

by

Lingkang Zhang

B.E., Beijing Jiaotong University, 2014

Dissertation Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

in the
Department of Computing Science
Faculty of Computing Science

SIMON FRASER UNIVERSITY
Spring 2017

Approval

Name:	**Lingkang Zhang**
Degree:	**Master of Science (Computing Science)**
Title:	***Optimal Gaze-based Attention System for Multi-human Multi-robot Interaction***
Examining Committee:	**Chair:** Anoop Sarkar Professor

Richard Vaughan
Senior Supervisor
Associate Professor

Greg Mori
Supervisor
Professor

Date Defended: April 3rd, 2017

Abstract

This thesis presents a computer vision based attention system for interaction between multiple humans and multiple robots. The study contains three parts. In the first part each human can "select" (obtain the undivided attention of) a ground robot and interact with it by simply gazing (looking directly) at it. This extends previous work whereby a single human can select one or more robots from a population.

In the second part, in addition to the face direction, the position of the face can also be estimated. And Vicon motion capture system is used to get the position of the ground robots. Since information about detected user locations and gaze directions is shared among the robots via a centralized server. 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.

In the third part, we demonstrate the attention system with flying robots. A new method of "micro-feedback" using LED lights and robot movement feedback to allow users to pre-select, select and de-select robots is introduced. This is the first demonstration of multiple UAV interacting with multiple human using face engagement.

Keywords: Human robot interaction, gaze, pose estimation, identity recognition

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

This thesis presents a computer vision based attention system for interaction between multiple humans and multiple robots using gaze direction. The work has been published in international conferences [34] and [35].

1.2 Motivation

Human robot interaction has become a growing research topic in recent years, which has a huge demand in fields including search and rescue, assistive robotics, military and police, education, space, home and industry [12].

Social skills and capabilities benefit the interaction between humans and robots[4]. To enable robots to interact with humans, it is important for the robots to recognize the attention from the human when a human is paying attention to them [6], especially in a multi-robot environment [19].

To help the robot to recognize the attention of a human, we need an interaction interface. In humans and other social animals, gaze-direction serves an important role in regulating the communication between individuals [15]. Humans are exquisitely perceptive of gaze direction and it is an important aspect of how humans feel engaged [5].

1.3 Contribution

The study contains three parts.

In the first part, each human can "select" (obtain the undivided attention of) a ground robot and interact with it by simply gazing (looking directly) at it. Each robot optimally assigns human identities to tracked faces in its camera view using a local Hungarian algorithm. The gaze-direction of the faces is 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. This extends existing work whereby a single human can select one or more robots from a population. This part has been published as [34].

In the second part, in addition to the direction of the face, the position of the face can also be estimated. And Vicon motion capture system is used to get the position of the ground robots. Since information about detected user locations and gaze directions is shared among the robots via a centralized server. 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. This part has been published as [35]

In the third part, we demoed the attention system with flying robots. A new method of "micro-feedback" using LED light feedback to allow users to pre-select, select and de-select robots is introduced. This is the first demonstration of multiple UAV interacting with multiple human using face engagement.

Chapter 2

Related Work

2.1 Overview

In this chapter, the vision based face detection and face recognition methods related to this study are described. The existing human robot interaction systems are also described.

2.2 Vision based Face Detection

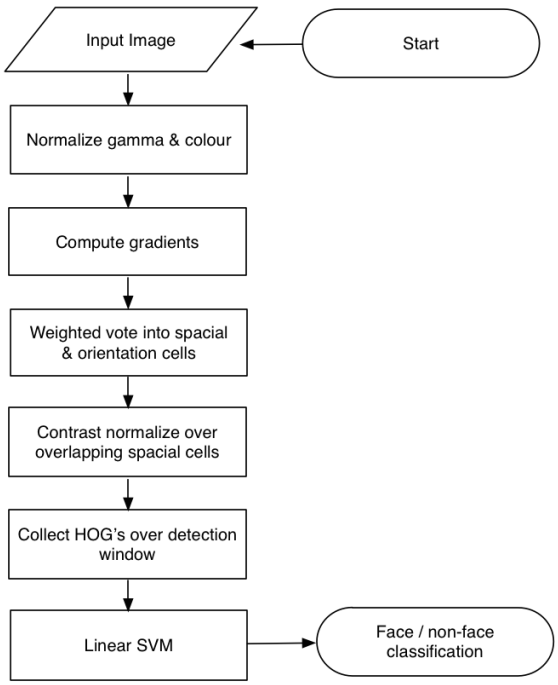


Figure 2.1: HOG method feature extraction and object detection chain

There is a huge number of literature on face detection since the early 1970s [13], but list only the most relevant work here. Viola and Jones proposed an effective object detection

method using Harr feature-based cascaded classifiers [31] in 2001 and the method was applied to a real-time face detection framework [32] in 2004. The method is machine learning based and positive and negative images are used to train a cascaded function. Naveneet and Bill showed that grids of Histogram of Oriented Gradient (HOG) descriptors significantly outperformed existing feature sets for human detection in 2006 [10]. The approach is SVM based and tested with the MIT pedestrian database.

We used the implementation of the HOG method in Dlib [16] for face detection in our system because of the good performance of HOG feature set on shape-based object classification. The feature extraction and object detection chain of the HOG method [10] is shown in Figure 2.1: The HOG feature vectors are extracted in a grid of overlapping blocks and the detector window is tiled with these blocks. A linear SVM is used to classify face / non-face. A HOG face detector looks like Figure 2.2 [16].

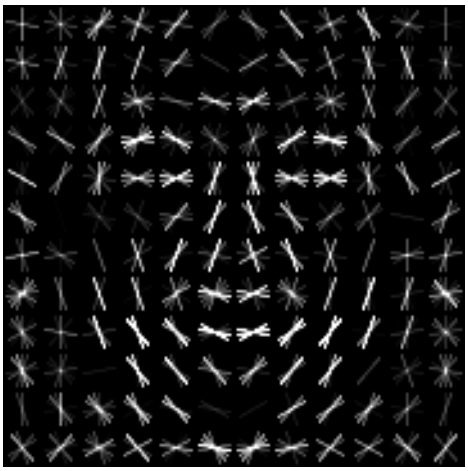


Figure 2.2: HOG face detector

2.3 Vision based Face Recognition

While we have good results for face detection, face recognition is still far from solved [1]. Matthew and Alex presented an approach called "Eigenfaces" for face recognition in 1991 [30]. The faces are projected onto a feature space defined by "Eigenfaces", which are the eigenvectors of the covariance matrix of the faces, or the principle components of the distribution of the faces. It has three main steps:

1. To project training face image into a relative smaller image space than the whole image space, the eigenvectors of the covariance matrix corresponding to the original face images need to be calculated. These eigenvectors represents the most significant difference between input face images, and they are called "Eigenfaces" since they look like faces.

2. Calculate the Eigenface components of the new face image which needs to be recognized.

3. Compare the Eigenface components of the new face image to that of the training face images can find the most similar one.

However, the Eigenface method uses principal components analysis which retains unwanted variations such as facial expression and lighting situation [2]. A new method called "Fisherface" was proposed by Peter et al. in 1997 [2]. The method is based on Fisher's Linear Discriminant and produces well separated classes in a low-dimensional subspace even with severe facial expression and illumination variation.

Recently, a deep convolutional network based method called "FaceNet" was proposed by Florian et al [28] in 2015. The approach has great representational efficiency and achieves state-of-the-art face recognition performance using only 128 bytes per face. As shown in Figure 2.3, the structure of FaceNet consists of a batch input layer, a deep CNN and L_2 normalization. The resulting face embedding is fed into the triplet loss during training. The triplet loss in Figure 2.4 minimizes the distance between an anchor and a positive with the same face identity and maximizes the distance between the anchor and a negative with different face identity.

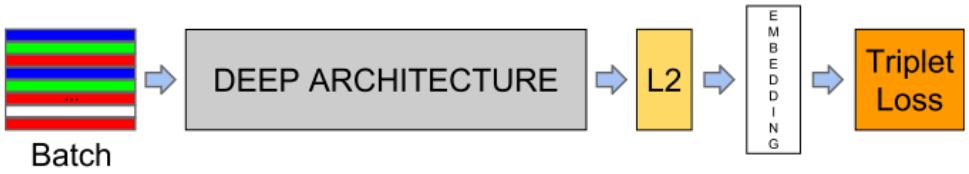


Figure 2.3: FaceNet model structure

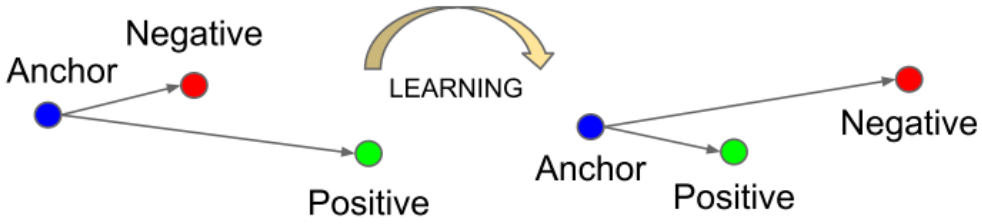


Figure 2.4: FaceNet triplet loss

We chose the Eigenface method for face recognition in ground robot interaction described in Chapter 3 and 4 because it requires reasonable amounts of data for training and works in real time. However, when the resolution of video stream gets lower and long distance interaction is required, the Eigenface method suffers from low accuracy. Thus we chose the FaceNet method in flying robots interaction described in Chapter 5.

2.3.1 Vision based face pose estimation

Recently a face pose estimation method based on face landmarks was introduced in [17], which provides a direct estimate of gaze direction. In Figure 2.5, the 6D head pose is estimated by fitting a 3D model of an adult head (left) onto the detected 2D features of the face (right) using an iterative Perspective-n-Point (PnP) algorithm implemented in OpenCV **??** using 8 corresponding pairs of key points.

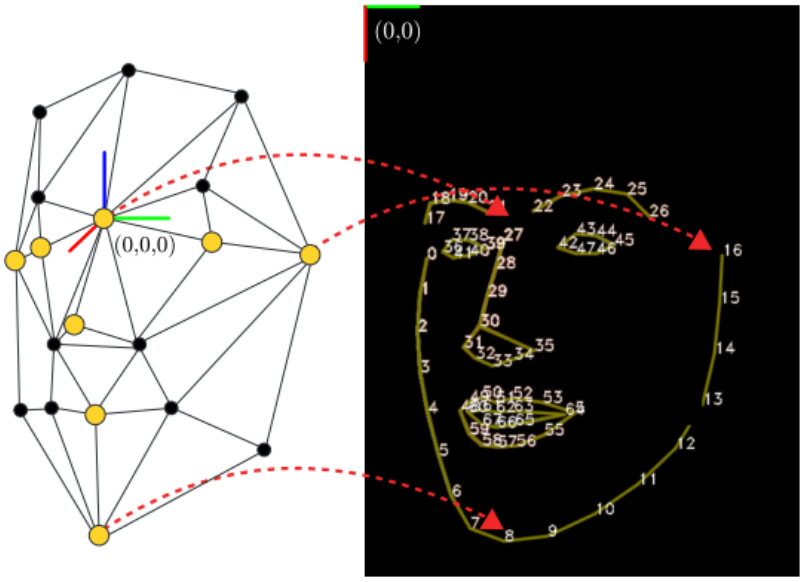


Figure 2.5: Head pose estimation

2.4 Human Robot Interaction

Previous HRI work on attracting the attention of robots has examined single-human multi-robot interaction [9, 20, 27, 26] and multi-human single-robot interaction [25].

2.4.1 Single Robot

In [26] and [25], Pourmehr uses sensors such as RGBD cameras, laser scanners and microphones to enable the robot to detect the human's attention. Figure 2.6 shows the experiment setting of [25] where five uninstrumented humans are at arbitrary poses and one of them is trying to get the attention of the robot. The leg, torso and sound detection data are integrated to determine the target human's position as shown in Figure 2.7.

Bruce presents a method for robots detecting waving humans at long range outdoor environment in [7]. A waving human can be detected up to 35 meters away where the

Figure 2.6: Sensor fusion

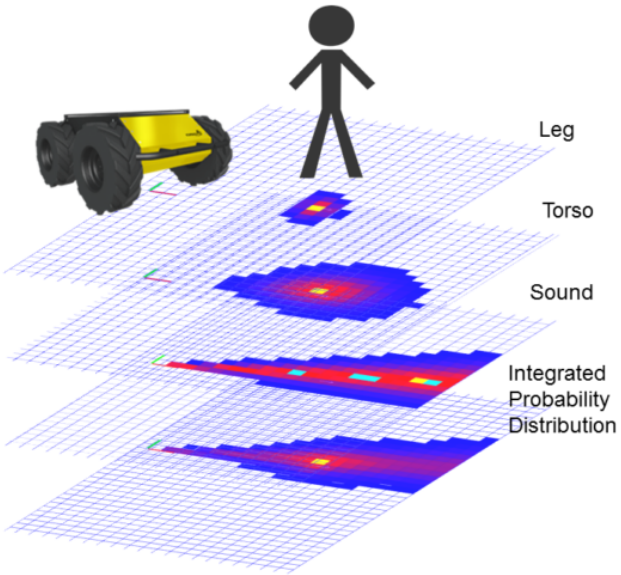


Figure 2.7: Sensor fusion

human is roughly 20 pixels high. Periodic motion at human waving detection with filter is used to identify the target human.

2.4.2 Multiple Robots

In [9, 20] and [27], 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. Both methods only require a monocular camera for a robot, which is low-cost, simple to install and easy to use.

In Couture-Beil's work [9], the face engagement is demonstrated as a method of selecting a particular robot from a multi-robot system. And motion based gestures are used for commanding selected robots individually as shown in Figure 2.9.

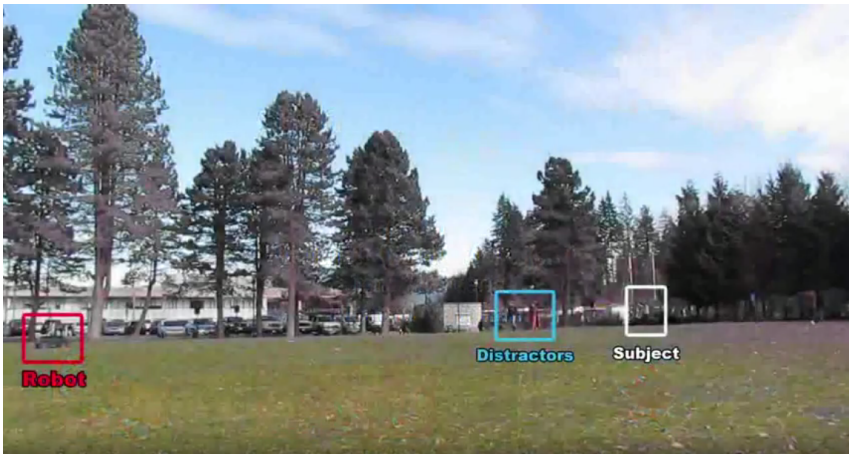


Figure 2.8: Tiny people finder

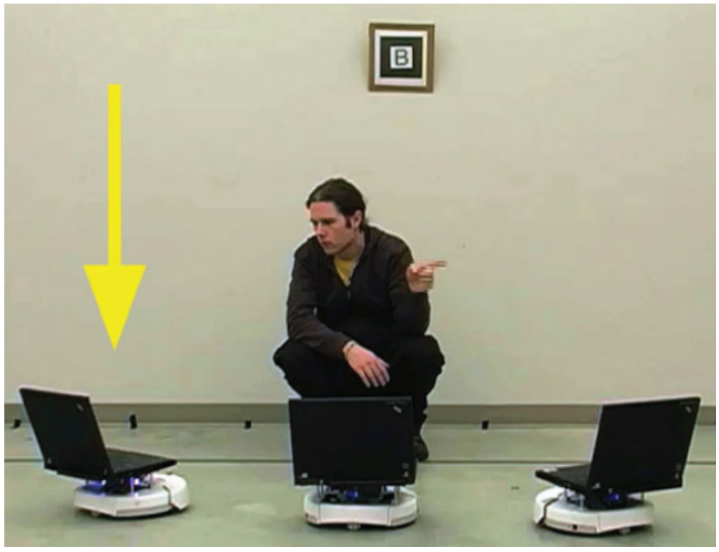


Figure 2.9: Selecting and commanding individual robots

Milligan presents a robot selection method in [20] to allow users simply draws a circle in the air around the desired robots to select them as shown in Figure 2.10.

Pourmehr presents a multimodal system to create, modify and command groups of robots from a population by using voice command and face engagement. The "Face score" is introduced as a measure of the degree of the user's engagement.

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 effectly during human robot interaction is introduced by Nagi in [24], where the multiple viewpoints of the robots are used to disambiguate observed gestures.

Gerkey [11] shows that the Multi-Robot Task Allocation problem (MRTA) can be optimally solved by the Hungarian algorithm. One of the MRTA sub-problem, the Single-task

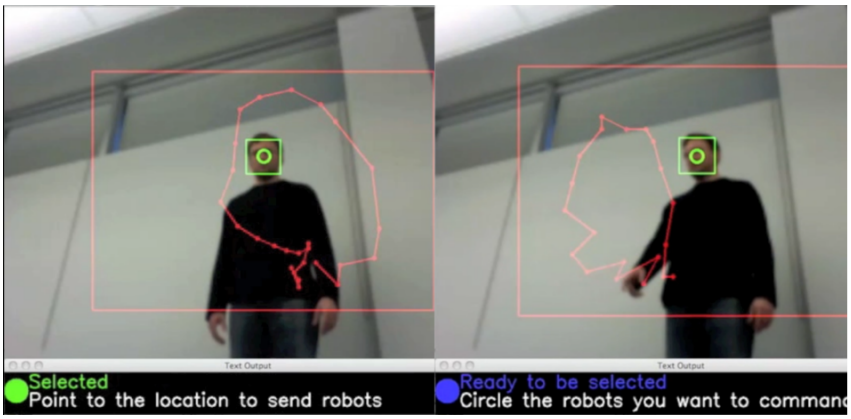


Figure 2.10: Selecting robots by drawing a circle in the air around the desired robots

Figure 2.11: Selecting and commanding multiple robots out of a group

robots, multi-robot tasks, time extended assignment (ST-MR-TE), is related to the allocation we have.

2.4.3 Relevance

In this paper we extend Couture-Beil's and Pourmehr's methods for single-human multi-robot selection [9] to multiple humans by incorporating Emami's face recognition to determine identity[1] and Kazemi's face landmark detection [14] together with the EPFL CHILI Lab's attention tracker[2] 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 gaze-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.

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

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

Chapter 3

Direct Gaze based Robot Selecting

3.1 Overview

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. We extend Couture-Beil's method for single-human multi-robot selection [9] to multiple humans by incorporating Emami's face recognition to determine identity [36] and Kazemi's face landmark detection [14] to determine gaze-direction. We find the optimal allocation of human identities to tracked faces, and add a central server to optimally allocate each robot to the human who is looking at it most directly. This is the first demonstration of gaze-based optimal many-to-many robot-selection HRI.

3.2 Hardware and sensors

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. Each of the robots is equipped with a USB camera with $640px \times 480px$ resolution.

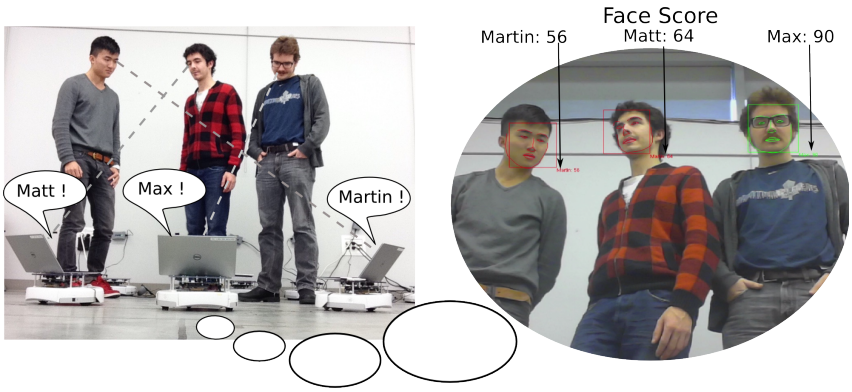


Figure 3.1: Multi Human Multi Robot Interaction

3.3 System Design

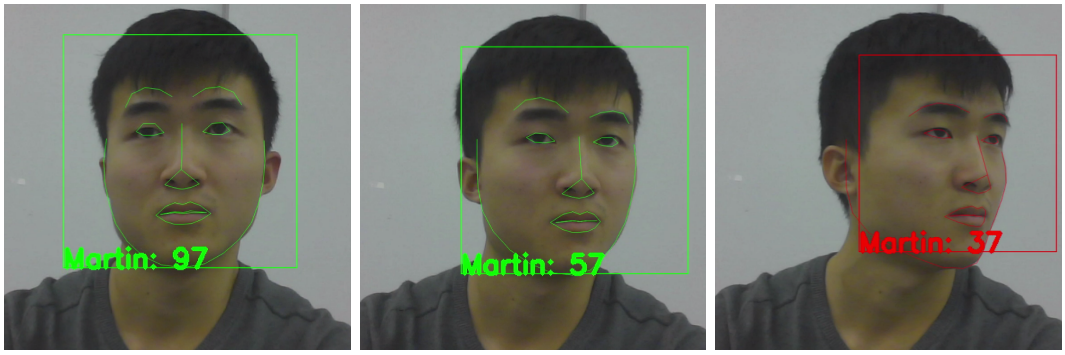


Figure 3.2: Face recognition and face score based on gaze-direction

A multi-human multi-robot allocation process based on face identity and landmark information is shown in Figure 3.3.

1. A robot detects faces in the image stream using the face-landmark method, and the identity of each detected face is recognized using a pre-trained face identity classifier using the ROS face_recognition package. For each face, a confidence is assigned for each identity.

2. Identities are optimally assigned to face-tracks locally using the Hungarian algorithm [33].

3. Face-tracks are also given a "face score" based on the angle of the face as shown in Figure 3.2. This face score is an alternative to the original template-match count used in [9].

4. 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.

5. Each selected robot gives audio and visual feedback indicating which human selected it.

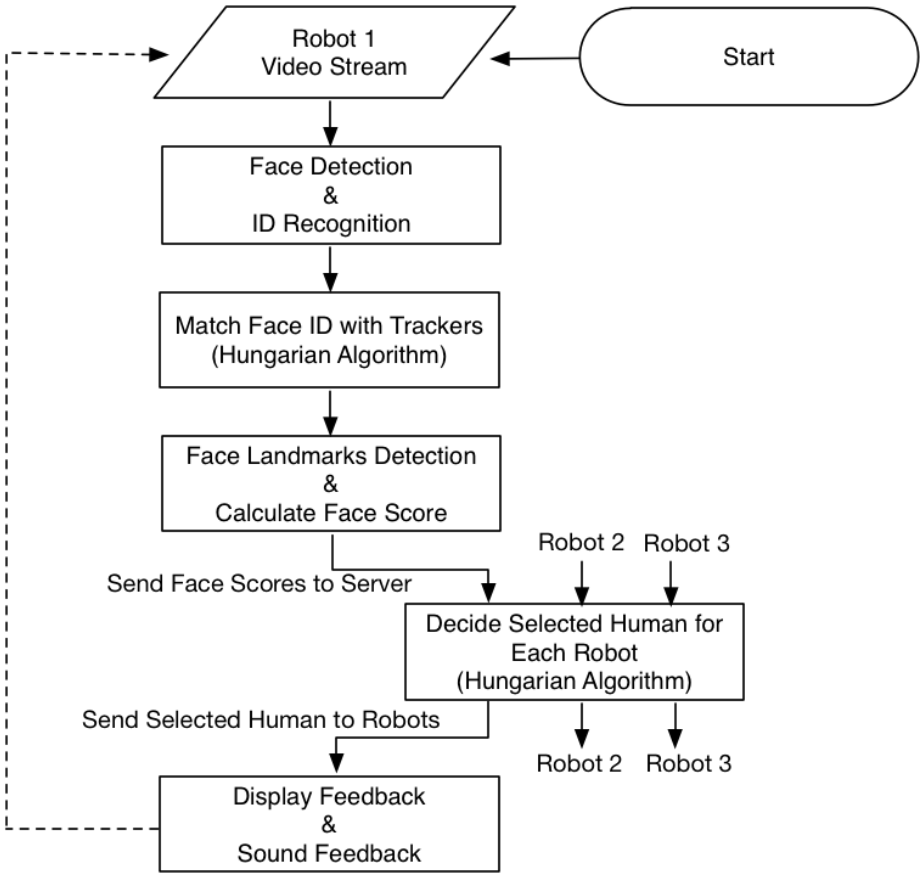


Figure 3.3: Flowchart of the multi-human multi-robot allocation process

The Hungarian algorithm is used to solve the allocation problem in both step 2 and step 4. 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.

3.4 Evaluation

Our test scenarios are shown in Figure 3.4, with 3 humans and 3 robots equipped with low-cost laptops (Intel i5, 4GB RAM). We consider five scenarios:
1) Humans gaze at the robots directly in front of them.
2) Humans gaze at robots other than those in front of them.
3) More available robots than interested humans
4) More interested humans than available robots
5) When one person is in the field of view of robot but the person is not looking at any robot.
In all of the five scenarios above, the robots are reliably allocated to the correct human. A

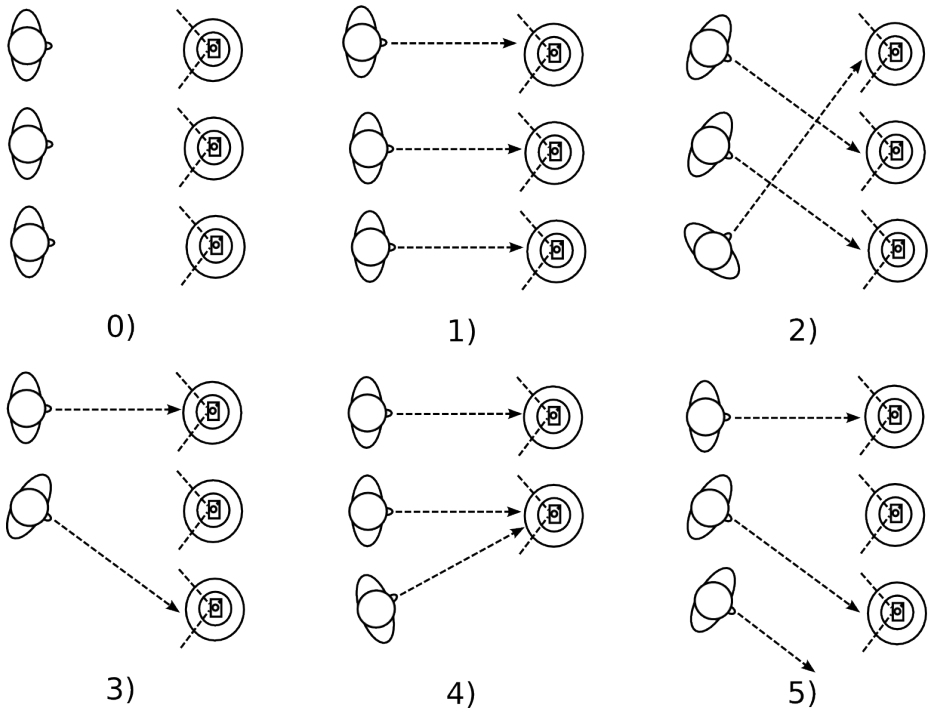


Figure 3.4: Experiments

robot will not be allocated to more than one human at the same time. The computation time for each new video image frame is in 150 msec.

3.5 Discussion

We describe the first demonstration of uninstrumented people selecting a robot for further interaction in a multi-human multi-robot setting. The allocation method is optimal given the available face tracking data.

The system described in this section can be extended to use more sophisticated interaction cues in the multi-human multi-robot interaction. For example, the basic method can be iterated for maximum robustness. Besides the sound and display feedback, the robots can move towards their chosen human, then the humans' reaction can be observed to detect errors in allocation.

Chapter 4

Indirect Gaze based Robot Selecting

4.1 Overview

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 4.1. In a multiple human interaction scenario as shown in Figure 4.2, 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.

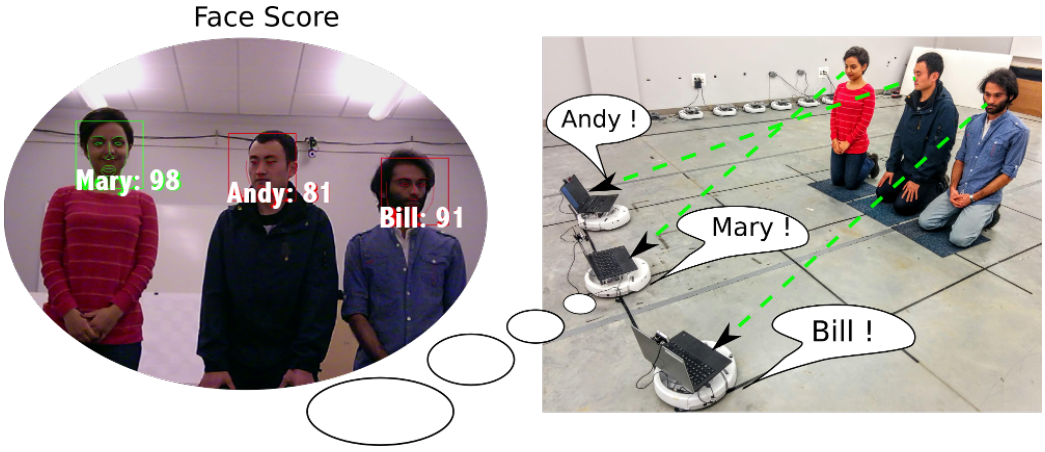


Figure 4.1: Multi-Human Multi-Robot Interaction

Figure 4.2: "Indirect" gaze based interaction

4.2 Hardware and Sensors

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. Each of the robots is equipped with a USB camera with $640px \times 480px$ resolution. Robots are globally localized using an external Vicon motion capture system, though any reasonably accurate localization system could be substituted (Figure 4.3).

4.3 System Design

We describe a multi-human multi-robot allocation process based on face identity and landmark information (Figure 3.3).

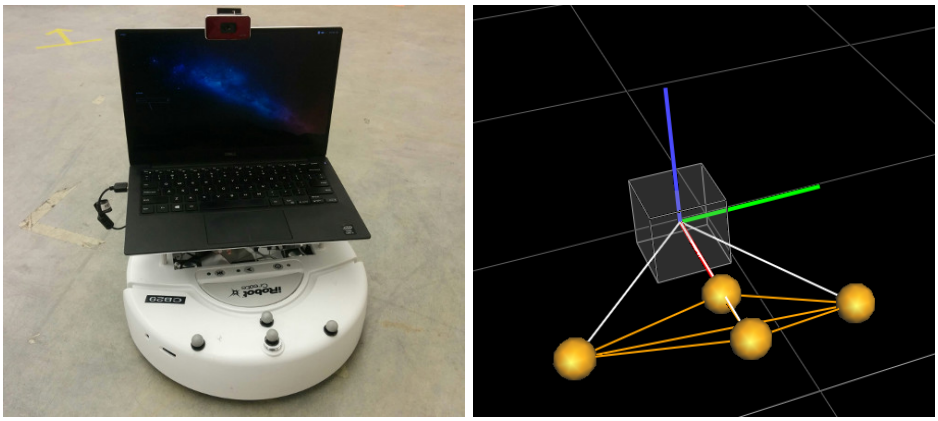


Figure 4.3: Robot platform and Vicon motion capture system

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 package. 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 [18] who applied the Hungarian algorithm to identify the types of vehicles tracked in image sequences.

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.

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 4.3.2).

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 the Hungarian algorithm[3] 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.

4.3.1 Face Identity Recognition and Tracking

Each robot is equipped with an RGB camera in this system. The ROS face_recognition package 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 package 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

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

with the multiple object detector in Dlib[4] 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 package 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 4.4. Thus we can obtain a face confidence matrix as Table 4.1 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 4.2. 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.

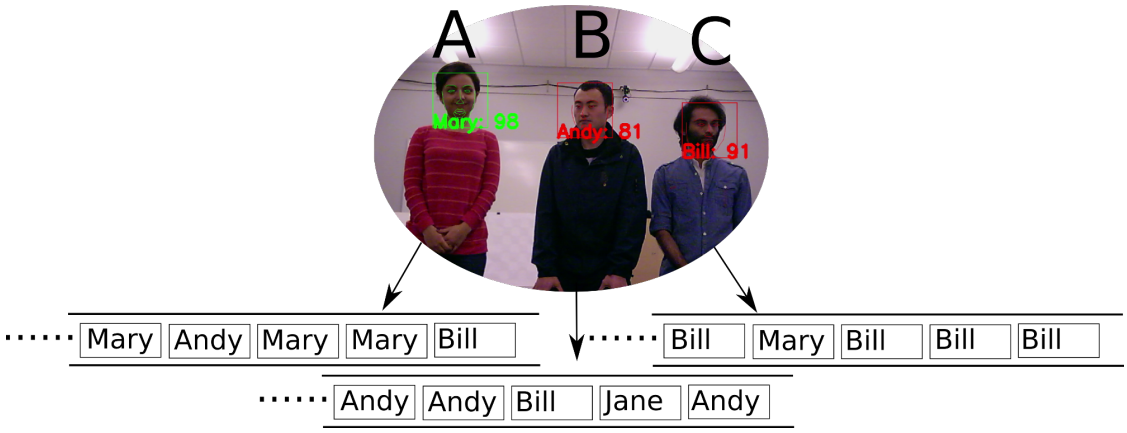


Figure 4.4: Face trackers and identities matching

[4]Dlib, Author: King, DE, Access on: http://dlib.net

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 4.1: Face identities confidence matrix

Identity	Mary	Andy	Bill
Tracker	A	B	C

Table 4.2: Face identities and trackers matching result using Hungarian algorithm

4.3.2 Face Pose Estimation

The face landmark (Figure 4.7) is obtained using Dlib [16] based on the method described in [14]. The face pose is estimated using the method described in [17]: eight interesting feature points (right eye, left eye, right ear, left ear, sellion, menton, nose and stomion) are selected from the 68 face landmark 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 [17] using the *solvePNP* function implemented in OpenCV [3] to obtain a 6-DOF pose (x, y, z, yaw, $pitch$, $roll$) of the detected face relative to the 3D virtual human head.

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 4.5, where x axis points in the forward direction and y axis points in the left direction, and yaw ranges from $-179°$ to $180°$. All angle calculations are normalized to this range. Figure 4.6 shows the process of face pose estimation. The global pose of the robot is $\boldsymbol{P_0}(x_0, y_0, yaw_0)$, and the global pose of the face is $\boldsymbol{P_1}(x_1, y_1, yaw_1)$. The red vector indicates the facing direction of the robot and the face. The green vector, $\boldsymbol{l}$, indicates the direction from the face to the robot. The angle between the gazing direction of the face and $\boldsymbol{l}$ is θ (θ is a negative value in the case shown in Figure 4.6). 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 [17] 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 $\boldsymbol{R}$ using the *Rodrigues* OpenCV function, then we use

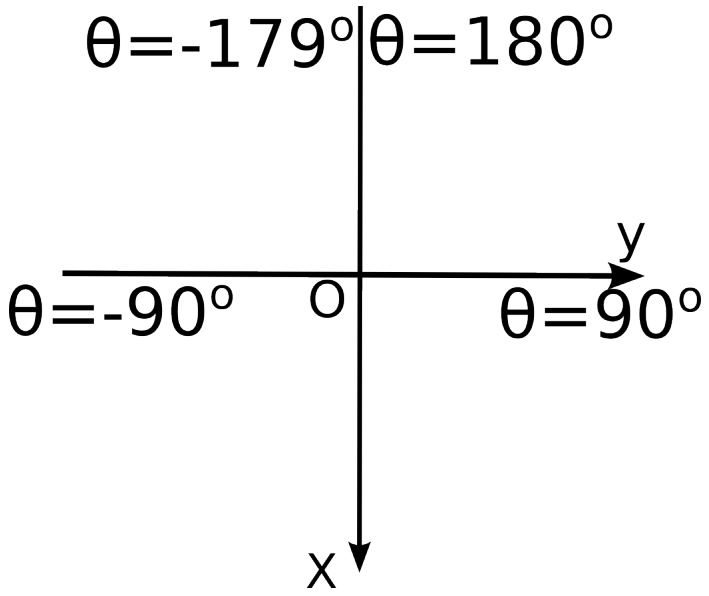


Figure 4.5: Coordinate system

the method in [29] to calculate angle θ. The global pose of the robot $\boldsymbol{P_0}$ is obtained from the Vicon motion capture system. Thus we can calculate out the global pose of the face $\boldsymbol{P_1}$ by using Equation 4.1, 4.2 and 4.3. $\boldsymbol{P_1}$ and θ are also part of the face tracker mentioned in Section 4.3.1.

$$x_1 = x_0 + dx \times cos\ yaw_0 - dy \times sin\ yaw_0 \tag{4.1}$$

$$y_1 = y_0 + dx \times sin\ yaw_0 + dy \times cos\ yaw_0 \tag{4.2}$$

$$yaw_1 = yaw_0 + \theta + atan2(dy, dx) + 180^o \tag{4.3}$$

4.3.3 Face Score and Matching

The proposed face score is a more principled alternative to the original template-match count used in [9], and is used to indicate the directness of a face's gaze to a certain robot. Each robot calculates a a "face score" for all faces it is tracking, based on the angle of the face as shown in Figure 4.7. The face score is calculated using Equation 4.4.

$$score = max(100 - |\theta|, 0) \tag{4.4}$$

where θ is the angle in degree between vector $\boldsymbol{l}$ and the gaze direction of the human as mentioned in Section 4.3.2, 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 4.6. Given a

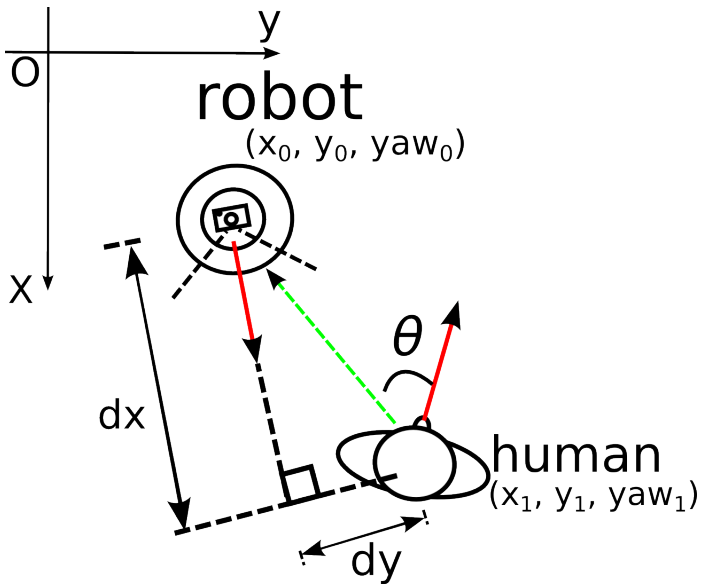


Figure 4.6: Face pose estimation

Figure 4.7: Face landmark and face score based on gaze-direction

robot global pose $\boldsymbol{P_0}(x_0, y_0, yaw_0)$ and a face global pose $\boldsymbol{P_1}(x_1, y_1, yaw_1)$, we use Equation 4.5 and 4.6 to calculate θ which can be used to calculate the face score using Equation 4.4.

$$\theta = yaw_1 - arctan(\frac{y_0 - y_1}{x_0 - x_1}) \tag{4.5}$$

$$\theta = \begin{cases} \theta - 360°, & \text{if } \theta > 180° \\ \theta + 360°, & \text{if } \theta < -180° \\ \theta, & \text{otherwise} \end{cases} \tag{4.6}$$

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 4.8, where robot_0 can see Mary and Andy, robot_1 can see all the three humans, and robot_2 can only see Bill.

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

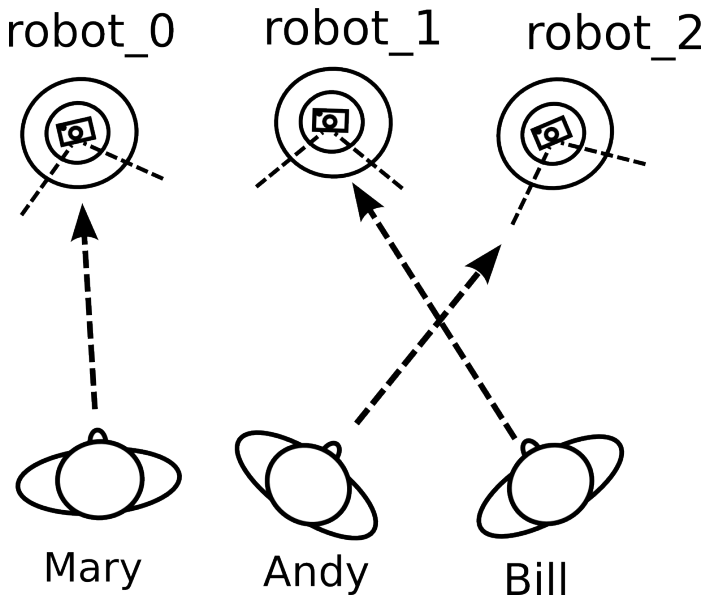


Figure 4.8: Calculate face score in multi-human multi-robot scenario

	Mary	Andy	Bill
robot_0	95	20	-1
robot_1	72	48	-1
robot_2	15	90	-1

	Mary	Andy	Bill
robot_0	-1	-1	72
robot_1	-1	-1	88
robot_2	-1	-1	44

	Mary	Andy	Bill
robot_0	93	24	70
robot_1	70	50	90
robot_2	17	94	40

Table 4.3: 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 4.4: Face score matrix in the centralized server

Face	Mary	Andy	Bill
Robot	robot_0	robot_2	robot_1

Table 4.5: Robots and humans matching result using Hungarian algorithm

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 4.4.3. From Table 4.3 we can calculate the score matrix in the server which is shown in Table 4.4, and the matching results obtained by the Hungarian algorithm are shown in Table 4.5.

4.3.4 Robots' Feedback

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

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 3.1. 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 4.9, the robot needs to calculate the angle α between the vector $\boldsymbol{l}$ 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 $\boldsymbol{l}$ can be calculate using equation 4.7.

$$\alpha = arctan(\frac{y_1 - y_0}{x_1 - x_0}) - yaw_0 \tag{4.7}$$

Since we are using the coordinate system described in 4.3.2, the robot should turn left if α is positive value or right if α is a negative value. 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°. Once the robot can see the matched face, it will use the movement strategy mentioned in the paragraph above.

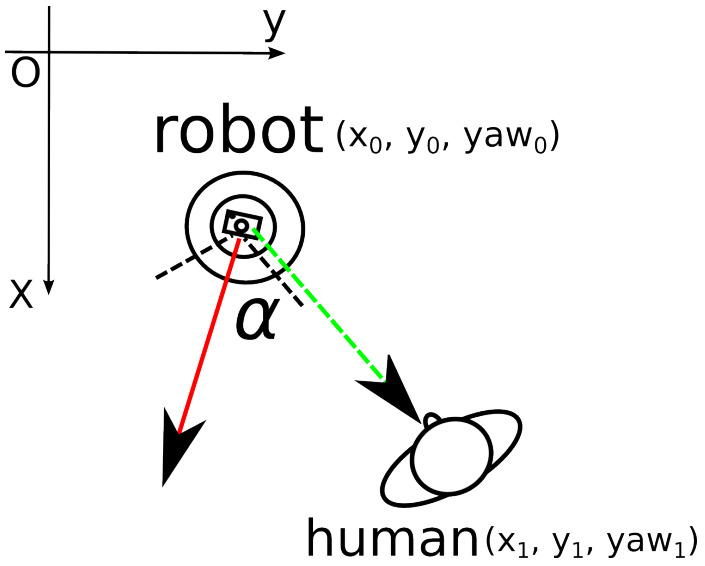


Figure 4.9: Robot's movement feedback when it cannot see the matched face

4.4 Evaluation

4.4.1 Processing Time

Table 4.6 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 4.6: Processing time

4.4.2 Face Pose Estimation Evaluation

The face pose estimation is evaluated with the method shown in Figure 4.10. 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 a Vicon motion capture system to obtain the ground truth pose of the face. The accuracy of the ground truth data should be better than 1 cm. Figure 4.11 shows the distance between the ground truth and estimated location of face over time with an average value of 17.68 cm. Figure 4.12 shows the ground truth face angle comparing with the estimated face angle overtime with an average difference value of 12°. The sampling rate of the ground truth is lower than the estimation rate in Figure 4.12.

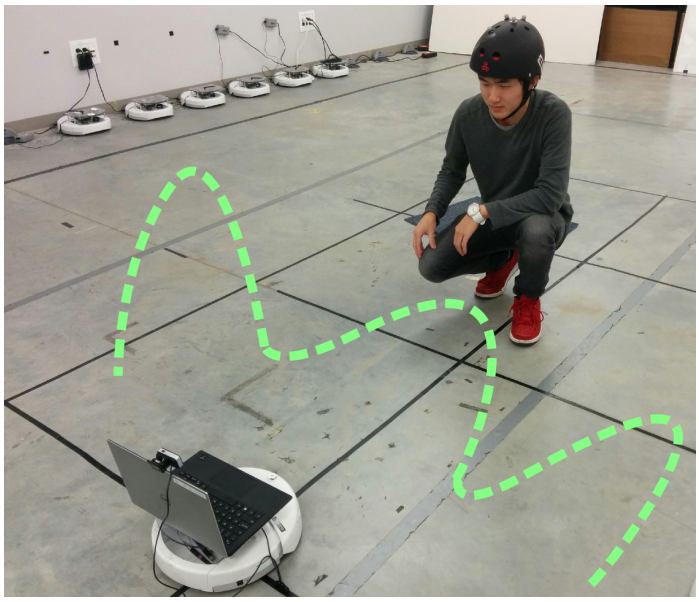


Figure 4.10: Face pose estimation evaluation

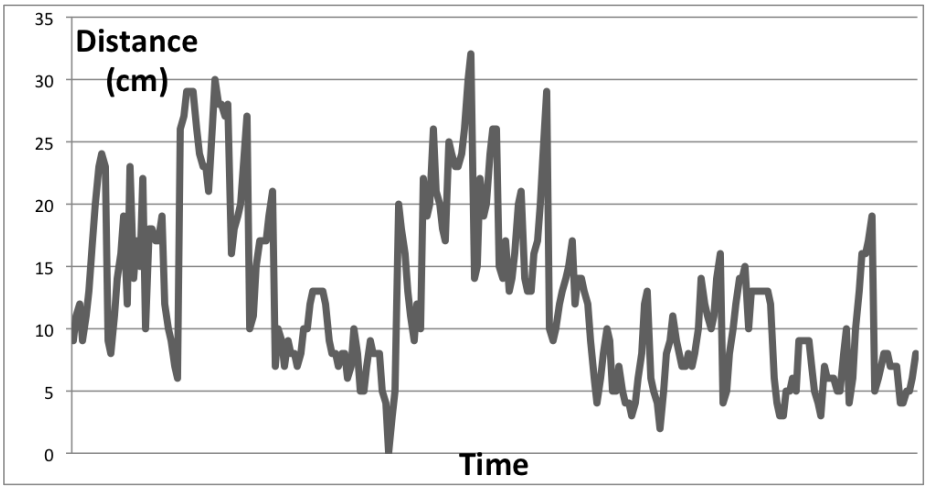


Figure 4.11: Face localization error: distance between estimated face location and ground truth

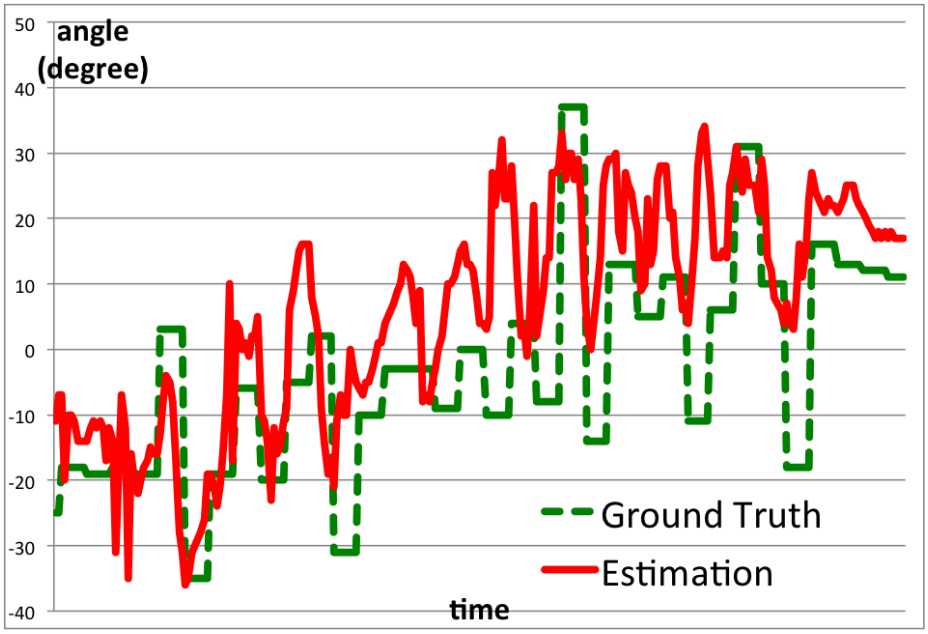


Figure 4.12: Estimated face angle and ground truth

4.4.3 Face Score Estimation Evaluation

The face score is evaluated as shown in Figure 4.13. 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 keeps changing the face direction for 2 minutes, and the score error for each video frame is recorded. The average face score error for each round are shown in Table 4.7.

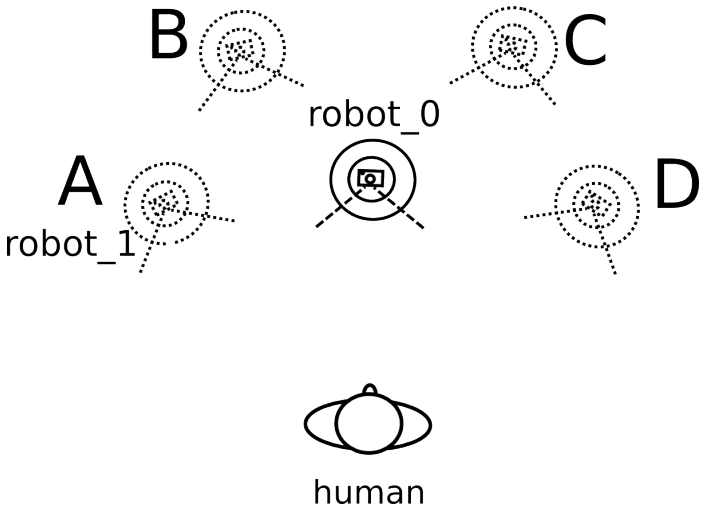


Figure 4.13: Face score estimation evaluation

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

Table 4.7: Face score estimation error

4.4.4 Face Identities Matching Evaluation

The face identity tracker is evaluated as shown in Figure 4.14. 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 4.8.

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

Table 4.8: Face identities matching success rate

4.4.5 Multi-human Multi-robot Experiments

Our test scenarios are shown in Figure 4.15, with three humans and three robots. We consider six scenarios where in a, b and c, all the robots can see all the person, 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.

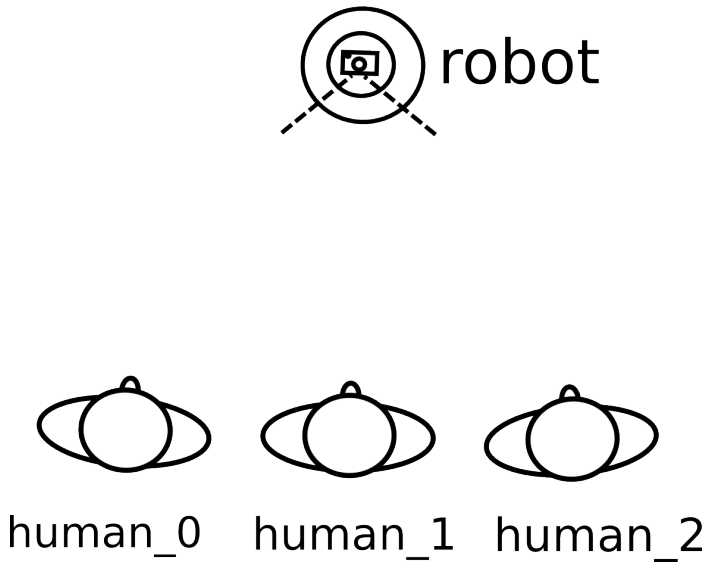


Figure 4.14: Face tracker evaluation

- 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.

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 humans in most of the cases as shown in Table 4.9 and Table 4.10. Each experiment scenario has 5 trials and the time for the successful trials is recorded.

4.5 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 4.4.1 shows that the system operates at a frequency more than 6 Hz using a low-cost robot

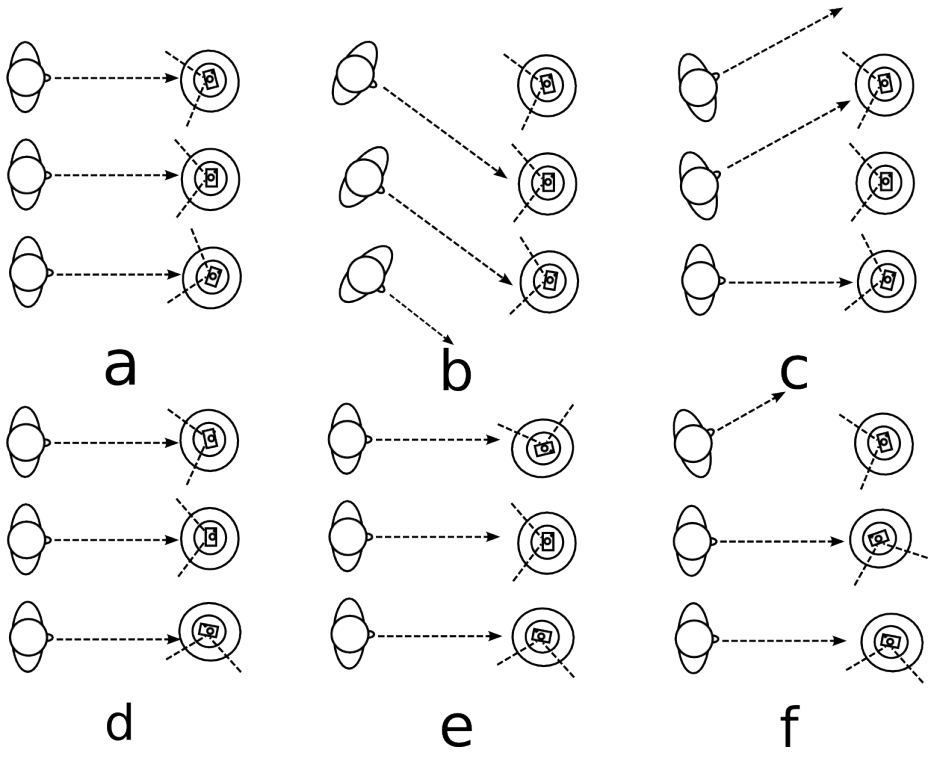


Figure 4.15: Multi-human multi-robot experiments

platform mentioned in 4.3, 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 4.4.2 shows that the average face location (x, y) error is 17.68 cm, which means that the resolution of the face location estimation is $\frac{17.68cm \times 17.68cm}{2m \times 2m}$, which better than 1%. And section 4.4.2 shows and average pose estimation error of 12° when the faces rotates in a range of $(-40°, 40°)$, 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 4.4.3 shows an average score estimation error of 11.7, which is 11.7% of the score range $(0, 100)$. Section 4.3.1 shows the face tracker method we introduced in 4.3.1 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 4.4.5. 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°, 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.

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 4.9: 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 4.10: Result of multi-human multi-robot experiments

We describe the first demonstration of uninstrumented people selecting a robot for further interaction in a multi-human multi-robot setting. An interesting novel property of our system is that 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.

Chapter 5

Gaze based UAV Interaction

5.1 Overview

An attention system with flying robots is demoed in this chapter. This work extends the previous work of gaze-based robot selection [34] [35] by a new method called “micro-feedback” using LED light feedback to allow users to pre-select, select and de-select robots. This is the first demonstration of multiple UAVs interacting with multiple humans using face engagement.

5.2 Hardware and Sensors

We use the method described in [22] to set up our flying robots. The robot platform is the Bebop UAV equipped with Intel Edison Linux computer and RGB LED strip as shown in Figure 5.1. The Bebob UAV has on-board camera and the video of resolution $640px \times 368px$ is streamed to low-cost laptops (Intel i5, 4GB RAM) at 30Hz using WiFi for processing. The Intel Edison Linux computer receives decisions from the laptops via WiFI and controls the RGB LED strip.

5.3 System Design

5.3.1 Face Recognition

We use a deep convolutional network based method called "FaceNet" [28] for face recognition. The approach has great representational efficiency and achieves state-of-the-art face recognition performance using only 128-bytes per face. The challenge we face is that the speed is too slow comparing to the face recognition method used in Section 4.3.1. By using the Python implementation described in [1], it takes more than 500 ms to process a single frame, which does not satisfy the real-time application.

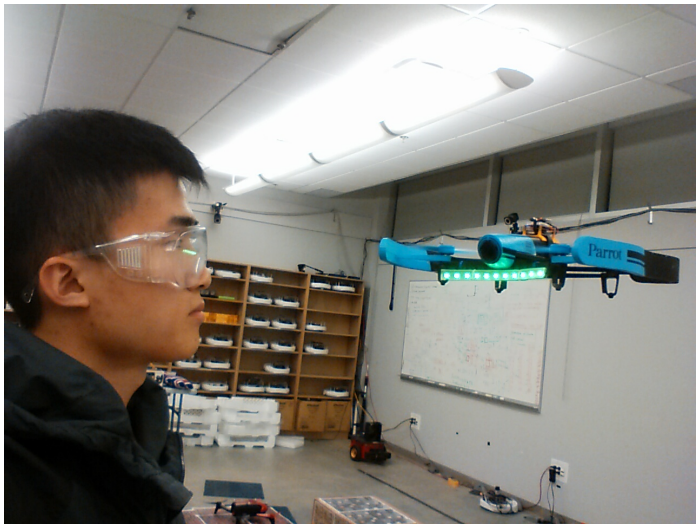


Figure 5.1: UAV experiment setting

To solve the problem, we implemented a ROS action node[5] wrapper on top of the original implementation so that the face recognition runs at 2 Hz in the background while the face detection and trackers as described in Section 4.3.1 runs at 30 Hz in the foreground. The running time is acceptable and the accuracy of this method is higher than 90%, which is better than the previous method described in Section 4.3.1.

5.3.2 Micro-feedback Behaviour

The micro-feedback behaviour is a series of robot movement and LED lighting patterns to help the human understand which state the robot is in at the moment. This allows uninstrumented humans to interact with flying robots in an easy way. The four micro-feedback behaviours of the robot are listed in the following. The video has been provided at https://vimeo.com/172533006.

1) Preselect: The human can preselect a robot by simply gazing at it. The robot will spin to the direction facing the human and LED lights turn to yellow as shown in Figure 5.2a.

2) Lock: The human nods the head so that the robot will be locked to the human and cannot be selected by anyone else. The robot follows and hovers in front of the locked human, and LED lights turn to green as shown in Figure 5.2b.

3) Lost: When the robot is selected by a human but this human is not in the camera view of the robot at the moment, LED lights turn to red as shown in Figure 5.2c.

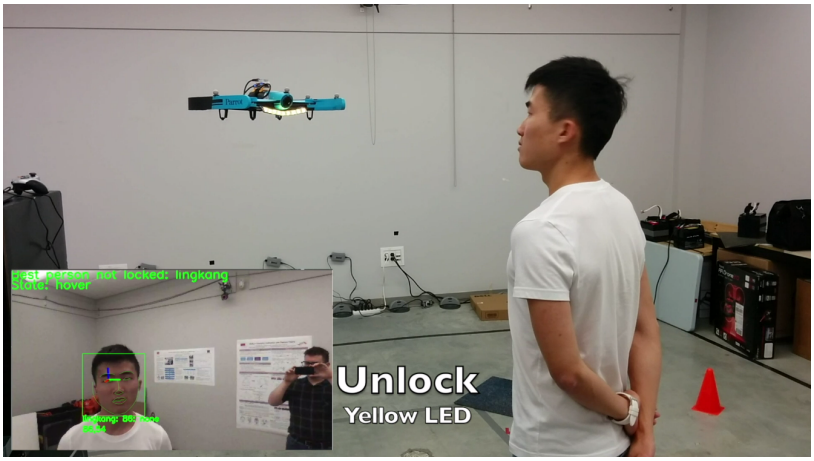
4) Unlock: The human shakes the head and the robot is unlocked as shown in Figure 5.2d.

[5]ROS actionlib, Author: Eitan Marder-Eppstein, Vijay Pradeep, Access on: http://wiki.ros.org/actionlib

(a) Pre-select

(b) Lock

(c) Lost



(d) Unlock

Figure 5.2: Micro-feedback behaviours

The head nodding and shaking detection is implemented with face landmark detector in Dlib[6]. By using the method described in Section 4.3.2, the 6-DOF pose of the head can be obtained. The center-crossing frequencies in the yaw and pitch directions are compared to decide whether the head is nodding, shaking or neither.

5.4 Demonstration

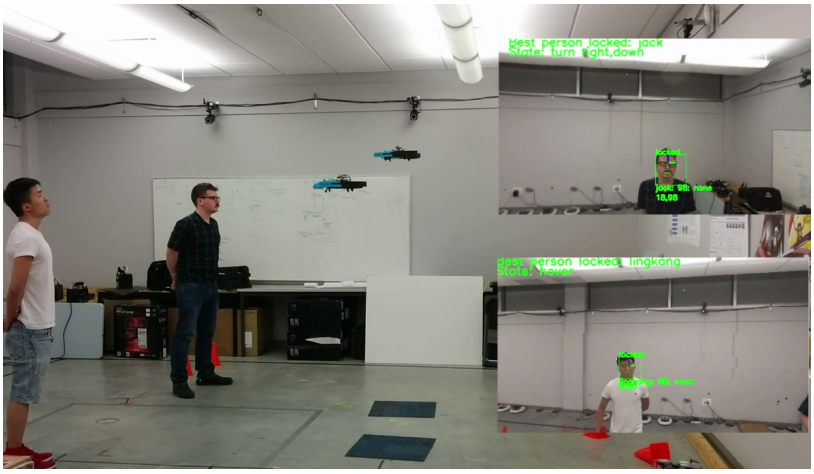


Figure 5.3: 2 Human and 2 UAV demonstration setting

By using the system setting up described in Section 5.3 and the micro-feedback behaviour described in Section 5.3.2, we have demonstrated the capability of the attention system within the multiple UAVs multiple humans interaction scenario. The demonstration has been set in a $4m \times 4m$ area. Two users pre-select the one of the UAVs and nod their

[6]Dlib, Author: King, DE, Access on: http://dlib.net

heads to the UAV so that it will be "locked" to the user. We recorded the time from the beginning of pre-select to lock. A case is counted as success only when both the of the two UAVs are locked to the correct users. The demonstration is repeated 10 times and the time of the successful ones is recorded as shown in Table 5.1. A demonstration is counted as success only when both of the two UAVs are allocated to the right humans. The success rate is 90% and the average time is 33 s. The video of the demonstration has been provided at https://vimeo.com/172533042.

Trial	1	2	3	4	5	6	7	8	9	10
Time (s)	52	29	30	34	34	fail	30	33	27	35

Table 5.1: Results of 2 human and 2 UAV demonstration

Chapter 6

Conclusions

6.1 Direct Gaze based Robot Selecting

We describe the first demonstration of uninstrumented people selecting a robot for further interaction in a multi-human multi-robot setting. The allocation method is optimal given the available face tracking data. A paper based on Chapter 2 was published at 2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI) [34].

6.2 Indirect Gaze based Robot Selecting

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. A paper based on Chapter 3 was published at 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) [35].

6.3 Gaze based UAV Interaction

We demoed the attention system with flying robots. A novel set of "micro-feedback" using the combination of LED light-based animation and robot movement to allow users to pre-select, select and de-select robots is introduced. This is the first demonstration of multiple UAV interacting with multiple human using face engagement.

6.4 Future Work

While our systems were shown to work better than 85% of the time, there is scope to improve robustness considerably. In particular it would very useful to extend the range of

the face tracking system as this would increase the practicality of this methods. 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 [8] 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 [25], [21] and [23].

Bibliography

[1] Brandon Amos, Bartosz Ludwiczuk, and Mahadev Satyanarayanan. Openface: A general-purpose face recognition library with mobile applications. Technical report, CMU-CS-16-118, CMU School of Computer Science, 2016.

[2] Peter N. Belhumeur, João P Hespanha, and David J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on pattern analysis and machine intelligence*, 19(7):711–720, 1997.

[3] G. Bradski. OpenCV. *Dr. Dobb's Journal of Software Tools*, 2000.

[4] Cynthia Breazeal. Toward sociable robots. *Robotics and autonomous systems, Elsevier*, 42(3):167–175, 2003.

[5] Cynthia Breazeal. Social interactions in hri: the robot view. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 34(2):181–186, 2004.

[6] Allison Bruce, Illah Nourbakhsh, and Reid Simmons. The role of expressiveness and attention in human-robot interaction. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, volume 4, pages 4138–4142. IEEE, 2002.

[7] Jake Bruce, Valiallah Monajjemi, Jens Wawerla, and Richard Vaughan. Tiny people finder: Long-range outdoor hri by periodicity detection. In *Computer and Robot Vision (CRV), 2016 13th Conference on*, pages 216–221. IEEE, 2016.

[8] James Bruce and Manuela Veloso. Real-time randomized path planning for robot navigation. In *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, volume 3, pages 2383–2388. IEEE, 2002.

[9] Alex Couture-Beil, Richard T Vaughan, and Greg Mori. Selecting and commanding individual robots in a multi-robot system. In *Computer and Robot Vision (CRV), 2010 Canadian Conference on*, pages 159–166. IEEE, 2010.

[10] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005.

[11] Brian Paul Gerkey. *On multi-robot task allocation.* PhD thesis, University of Southern California, 2003.

[12] Michael A Goodrich and Alan C Schultz. Human-robot interaction: a survey. *Foundations and trends in human-computer interaction*, 1(3):203–275, 2007.

[13] Erik Hjelmås and Boon Kee Low. Face detection: A survey. *Computer vision and image understanding*, 83(3):236–274, 2001.

[14] Vahdat Kazemi and Josephine Sullivan. One millisecond face alignment with an ensemble of regression trees. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1867–1874. IEEE, 2014.

[15] Adam Kendon. Some functions of gaze-direction in social interaction. *Acta psychologica*, 26:22–63, 1967.

[16] DE King. Dlib c++ library. *Access on: http://dlib. net.*

[17] Séverin Lemaignan, Fernando Garcia, Alexis Jacq, and Pierre 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.

[18] Felix Lütteke, Xu Zhang, and Jörg 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*, pages 1–6. VDE, 2012.

[19] James McLurkin, Jennifer Smith, James Frankel, David Sotkowitz, David Blau, and Brian Schmidt. Speaking swarmish: Human-robot interface design for large swarms of autonomous mobile robots. In *AAAI Spring Symposium: To Boldly Go Where No Human-Robot Team Has Gone Before*, pages 72–75, 2006.

[20] Brian Milligan, Greg Mori, and Richard T Vaughan. Selecting and commanding groups in a multi-robot vision based system. In *Proceedings of the 6th international conference on Human-robot interaction*, pages 415–416. ACM, 2011.

[21] Mani Monajjemi, Jake Bruce, Seyed Abbas Sadat, Jens Wawerla, and Richard 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*, pages 3614–3620. IEEE, 2015.

[22] Mani Monajjemi, Sepehr Mohaimenianpour, and Richard Vaughan. Uav, come to me: End-to-end, multi-scale situated hri with an uninstrumented human and a distant uav.

[23] Valiallah Mani Monajjemi, Jens Wawerla, Rodney Vaughan, and Greg 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*, pages 617–623. IEEE, 2013.

[24] J Nagi, H Ngo, LM Gambardella, and Gianni 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*, pages 1802–1808. IEEE, 2015.

[25] Shokoofeh Pourmehr, Jake Bruce, Jens Wawerla, and Richard 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.

[26] Shokoofeh Pourmehr, Valiallah Monajjemi, Jens Wawerla, Rodney Vaughan, and Greg Mori. A robust integrated system for selecting and commanding multiple mobile robots. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 2874–2879. IEEE, 2013.

[27] Shokoofeh Pourmehr, Valiallah Mani Monajjemi, Rodney Vaughan, and Greg 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*, pages 137–142. IEEE, 2013.

[28] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 815–823, 2015.

[29] Gregory 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).

[30] Matthew A Turk and Alex P Pentland. Face recognition using eigenfaces. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, pages 586–591. IEEE, 1991.

[31] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511. IEEE, 2001.

[32] Paul Viola and Michael J Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.

[33] John Weaver. Kuhn-munkres (hungarian) algorithm in c++. *Access on: https://github.com/saebyn/munkres-cpp*.

[34] Lingkang Zhang and Richard Vaughan. Optimal gaze-based robot selection in multi-human multi-robot interaction. In *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 645–646. IEEE, 2016.

[35] Lingkang Zhang and Richard Vaughan. Optimal robot selection by gaze direction in multi-human multi-robot interaction. In *Intelligent Robot s and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 617–623. IEEE, 2016.

[36] Pouyan Ziafati. Ros face_recognition package. *Access on: http://wiki.ros.org/face_recognition*.