ENABLING A MULTI-PARTY CONVERSATIONAL VIRTUAL AGENT THROUGH HEAD AND MOUTH MOTION TRACKING

by

QIAN MA

(Under the Direction of Suchendra Bhandarkar and Kyle Johnsen)

ABSTRACT

Conversational gaze behavior is an important component of an embodied conversational agent (ECA). Without proper conversational gaze, conversational agents may be less persuasive, emotive, and ultimately less believable or usable. While many conversational agent systems have been created for one-on-one type interactions, there is a noticeable lack of multi-party-capable systems, i.e., systems capable of dealing with more than one user simultaneously. We present a conversational agent system capable of sensing and reacting to the conversational state of multiple users using computer vision algorithms for head and mouth motion tracking.

INDEX WORDS: Embodied conversational agent; Human-computer interaction; Head tracking; Mouth tracking; Eye gaze

ENABLING A MULTI-PARTY CONVERSATIONAL VIRTUAL AGENT THROUGH HEAD AND MOUTH MOTION TRACKING

by

QIAN MA

B.S., Beihang University, China, 2008

A Thesis Submitted to the Graduate Faculty of The University of Georgia in Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE

ATHENS, GEORGIA

2010

ENABLING A MULTI-PARTY CONVERSATIONAL VIRTUAL AGENT THROUGH HEAD AND MOUTH MOTION TRACKING

by

QIAN MA

Major Professor:	Suchendra Bhandarkar
Committee:	Kyle Johnsen
	Khaled Rasheed

Electronic Version Approved:

Maureen Grasso
Dean of the Graduate School
The University of Georgia
December 2010

DEDICATION

I am deeply grateful to Dr. Bhandarkar and Dr Johnsen, for their efforts, guidance and support, which made my research work much better.

I would like to thank Dr. Rasheed for his kind and valuable help.

I also want to thank all of the faculty, staff and my friends in the Department of Computer Science for discussing with them and learning a lot from them.

My final dedication goes to my parents for their unconditional love and motivation.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Page

CHAPTER 1

INTRODUCTION

Multi-party conversations, i.e., situations where three or more people involved in a conversation, are commonplace, particularly in face-to-face meetings. This is because face-to-face meetings provide non-verbal information sources, comprising typically of eye gaze, body posture and gestural cues. These cues allow conversational partners to interact more effectively. Such non-verbal information sources are typically less important in one-on-one conversations. Whereas in one-on-one conversations, hearing is the dominant sense; in multi-party conversations, the visual sense provides the necessary conversational cues to manage the complexity of the situation. This is illustrated by the difficulty of holding a multi-party conference call in the absence of video. To be effective in multi-party situations, conversational agents must be able to also detect and leverage visual and non-verbal signals. Robust detection and interpretation of visual and non-verbal information for multiple users simultaneously presents a challenging problem in visual sensing, computer vision and artificial intelligence. This paper presents an approach to enable an embodied conversational agent (ECA) to detect and respond effectively to the conversational states of multiple conversational partners.

The approach uses a set of visual and non-verbal cues to determine the conversational states of participants in a multi-party conversation.   Using a camera-based, ECA-mounted, hierarchical head and mouth motion tracking system, visual information such as the face locations and mouth motions are obtained in real time for each participant. From this information, the ECA derives the conversational state of the multi-party conversation, i.e.,

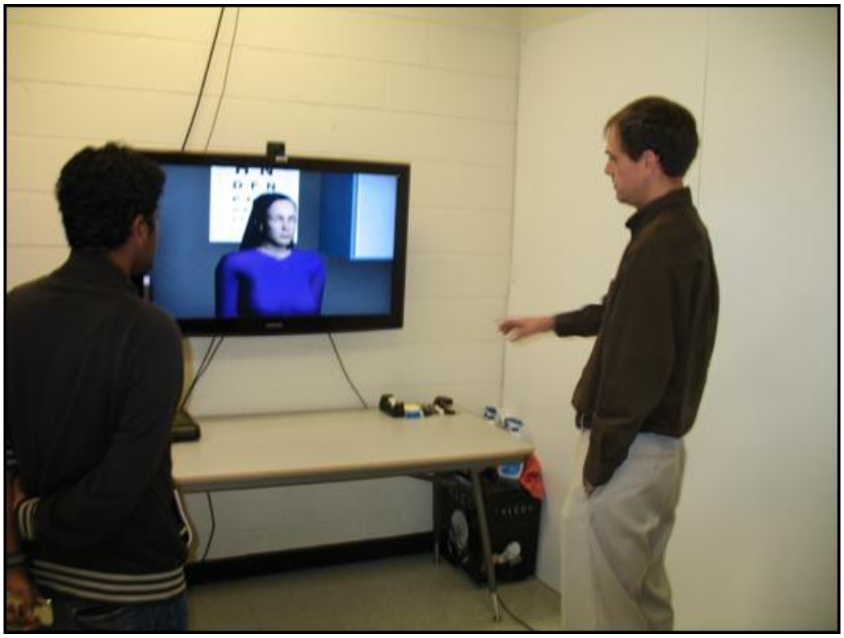

Figure 1.1: Two participants interacting with a virtual patient in the examination room

determines who is speaking and listening to whom. Finally, from the conversational state, the ECA selects an appropriate Markov model to drive its own non-verbal behavior.

The primary benefit of the approach is that it can produce reasonable non-verbal behavior of an ECA with relatively little infrastructure support. It requires only a single ECA head-mounted-camera and commodity PC. Furthermore, it does not require extensive setup beyond camera calibration and is largely user-independent. Thus, the approach can be readily used to drive the behavior of a large variety of ECAs, from robotic humans to computer generated virtual humans. As shown in Figure 1.1, a prototype has been constructed that consists of a single camera mounted directly above an LCD monitor that displays a virtual human. The camera senses the visual states of multiple participants (users) directly in front of the virtual human. The non-verbal gaze behavior of the virtual human is then driven by a set of Markov models chosen by an ECA.

CHAPTER 2

PREVIOUS WORK

2.1 GAZE BEHAVIOR OF EMBODIED CONVERSATIONAL AGENTS

Many non-verbal aspects of ECAs have been explored in the literature, including facial expressions, gestures, and posture. The non-verbal channel is deemed vital for social dialogue, since it can be used to provide such social cues as attentiveness, affect, attraction, and to mark shifts into and out of social activities [1]. Amongst the various non-verbal behavioral aspects of ECAs, the current work focuses on eye gaze, which may account for the majority of the non-verbal information conveyed in face-to-face communication. Eye gaze conveys the level of participation and interest of the ECA to the other participants and is used as a mechanism for turn-taking [2].

Many researchers have proposed models for ECA gaze patterns and studied the impact of the models on users. Colburn et al. have found that avatars that use a natural gaze model elicit changes in viewers' eye gaze patterns [3]. Garau et al. have investigated the impact of avatar gaze on participants' perception of communication quality by comparing a random-gaze avatar with an inferred-gaze avatar whose eye gaze adapts to the conversation context [4]. The results show that the inferred-gaze avatar significantly outperforms its random-gaze counterpart in terms of participants' subjective responses. Lee et al. proposed an eye movement model based on empirical studies of saccades and statistical models of eye tracking data [5]. Their results are consistent with those presented in [8] in that inferred gaze is seen to significantly outperform random gaze. Fukayama et al. use a two-state Markov model which outputs gaze direction in a place derived from three gaze parameters, i.e., amount of gaze, mean duration of gaze and gaze points while averted [6]. They find that the emotional impressions of the ECA formed by the users could be manipulated by adjusting the Markov model to fit known conversational behaviors. Pelachaud et al. propose an eye gaze model for an ECA that embeds information on communicative functions accompanied by statistical information on gaze patterns [7]. Note that the aforementioned ECA gaze models do not perform user tracking, and only allow for one-on-one interaction. In addition, the gaze models described above and the accompanying user studies are difficult to generalize to multi-party situations, thus motivating our current work.

2.2 USER TRACKING FOR EMBODIED CONVERSATIONAL AGENTS

More current research has employed tracking systems to incorporate the users' non-verbal response into the ECA's non-verbal behavior. Bee et al. present an interactive eye gaze model for an ECA by using an eye tracker [8]. They have found that the interactive gaze model leads to a better user experience compared to the non-interactive gaze model. Traum et al. have used real-time visual processing to enhance the ECA dialogue model for multi-party communication [9]. They focus on visual cues such as head orientation, head nods and head shakes, and examine how these behaviors influence various aspects of a multi-layer dialogue model. However, their study is limited to situations in which there is one human participant and two virtual humans involved in a conversation. Vertegaal et al. have developed a multi-agent conversational system called FRED which determines the subject gaze by means of an eye tracking system mounted below the computer screen [10]. Their experiments show that, on average, subjects looked about 7 times more at the individual they listened to and 3 times more at the individual they spoke to than at others. They conclude that gaze directional cues could be used to indicate conversational attention in multi-party conversations. Nakano et al. analyzed eye gaze and head nods to explain how people use non-verbal signals in the process of grounding communication [11]. In experiments with their ECA-based system called MACK, they found that subjects were aware of the lack of the conversational grounding ability of MACK. To overcome this limitation, they developed a stereo-camera-based 6-degree-of-freedom head-pose tracker to recognize the head-nod and eye gaze of subjects.

Greater emphasis has been placed on user tracking for robotic ECAs than for virtual ECAs. Dillmann et al. have built a robot assistant, ARMAR II, capable of articulated body tracking and recognition of human gestures [12]. Sidner et al. have designed an engaging robot capable of tracking the user's face and adjusting its gaze accordingly using algorithms for face detection, sound location, speech detection, and object recognition [13]. They found that users are typically sensitive to the appropriateness of gestures and respond to changes in head direction and eye gaze by changing their own head direction and eye gaze. Matsusaka et al. have developed a robot named ROBITA, who can join in the multi-party conversations with two people [14]. They enabled the robot with multimodal information recognition abilities and non-verbal body expression abilities using algorithms for face detection, face recognition, gesture recognition and speech recognition. The robot was designed to answer questions about itself while adapting its gaze between the conversational participants.

CHAPTER 3

OVERVIEW

Our current work fuses a multi-party conversational gaze model with a multi-party hierarchical head and mouth motion tracking system to drive an ECA's conversational gaze behavior. Particular emphasis is placed on the detection of who is speaking with whom at any particular time in a conversation and adjusting the ECA's gaze model appropriately. The general system architecture is depicted in Figure 3.1.

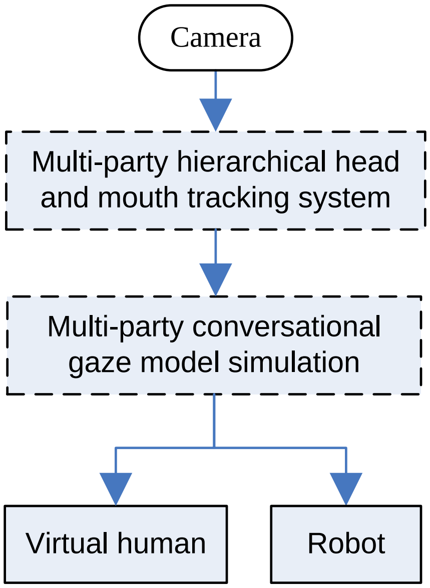
Camera
Multi-party hierarchical head and mouth tracking system
Multi-party conversational gaze model simulation
Virtual human
Robot

Figure 3.1: High-level architectural view of the multi-party-capable ECA system.

A single camera mounted directly above the ECA feeds an image stream to a real-time multi-party hierarchical head and mouth tracking system. The tracking system determines the location of each head, whether or not the head is frontal facing, and whether or not the lips are moving. This information is fed to the conversational gaze model simulation program, which in turn selects a Markov model for the ECA gaze behavior. The state of the Markov model drives the gaze of the ECA that is being presented on a monitor.

CHAPTER 4

HIERARCHICAL HEAD AND MOUTH TRACKING SYSTEM

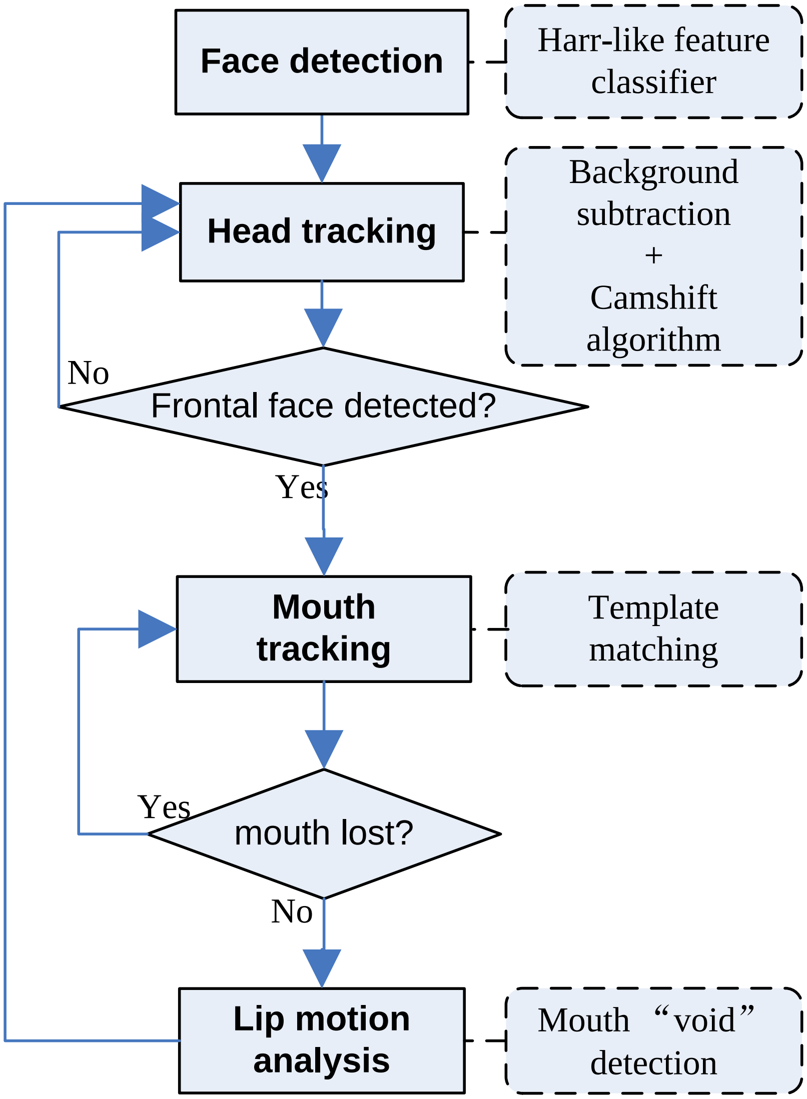

Figure 4.1: Flowchart of the proposed hierarchical head and mouth tracking system

The hierarchical head and mouth motion tracking system is divided into 4 phases: 1) face detection, 2) head tracking, 3) mouth tracking, and 4) lip motion analysis. The flowchart of the overall system is shown in Figure 4.1.

In a hierarchical tracking system, the tracking is a sequence of stages or phases, and results from earlier stages ensure the validity of the result and constrain the search regions for next stage. If one stage fails, all stages after that stage will fail, however, earlier stages will continue to provide tracking information. For example, if the face is not detected at the beginning, then all the later tracking phases will not be initialized. For example, if head tracking fails, which means the user is not present, then all the later tracking phases, i.e., mouth tracking and lip motion analysis, should also fail. Likewise, if mouth tracking fails, the later tracking phase, i.e., lip motion analysis fails because it will not generate quality results. Therefore, the hierarchical structure will ensure a valid result phase by phase. In addition, previous phases will constrain the search regions for the next phase. For example, face detection and head tracking will constrain the search region for mouth tracking and mouth tracking will constrain the region for lip motion analysis. The benefit of this approach, as opposed to a sensor fusion approach, is an increase in overall speed of tracking (by constraining later stage search regions), and a reduction in false positives by increasing the possibility that later stage search regions and tracking results are valid.

4.1 FACE DETECTION

The face detection phase is the starting point for the multi-party hierarchical head and mouth motion tracking system. It is used to establish the number of conversational partners, as it is assumed that those participants who are proximal to the ECA and looking directly at it are interested in a conversation. It also used to initialize all the later tracking phases and constrain the search region for head tracking and mouth tracking.

Numerous approaches have been proposed for face detection. Simple features such as color, motion, and texture have been used for face detection in earlier works. However, these approaches break down easily because of the complexity of real-world situations. The face detection algorithm proposed by Viola and Jones is the most popular amongst the statistical face detection approaches [15]. Viola and Jones have proposed a face detection framework based on the AdaBoost learning algorithm, which uses Haar-like features to achieve rapid and accurate results [16]. The Haar-like feature classifier is trained with a few hundred sample views of various faces under different angles and lighting conditions before it can be applied to an input image. Given a particular image region, the classifier labels it as either "face" or "not face". The face detection algorithm scans through the image multiple times at various scales to detect faces of varying sizes.

For the face detection phase of the current work, the Haar-cascade classifier for frontal face detection, which is distributed freely with the open-source OpenCV library, is seen to provide robust multi-person face detection in real-time. To initiate the tracking process, the face detection algorithm is run on each video frame until a face is found. For subsequent phases, the face detection procedure continues to run as a background thread, providing face detection information as needed to the other phases.

4.2 HEAD TRACKING

Head tracking is used to determine each conversational participant’s position over time. It is also used to judge whether the participant is still in the room, otherwise mouth tracking and lip motion analysis have no meaning. The face detection phase only partially provides the needed information for this phase as participants' faces may not always be directed towards the camera. Furthermore, robust face detection is computationally intensive. On an Intel Core-2 2.5 GHz, quad-core processor and given a 640 x 480 frame resolution, the face detection phase runs at approximately 13 frames per second. Therefore, head tracking could be used to locate each user's position no matter where the user is gazing, so that the ECA would know whether the person is still there and where she should drive her gaze.

Tracking algorithms are classified mainly into two major categories: state space-based and kernel-based. Kernel-based tracking algorithms have seen greater applicability in real-time video surveillance systems on account of their lower computational overhead compared to their state space-based counterparts [17]. In this paper, the Continuously Adaptive Mean Shift (Camshift) tracking algorithm, which belongs to the kernel-based category, is chosen for the head tracking since it is accurate, robust, fast and computationally efficient [18].

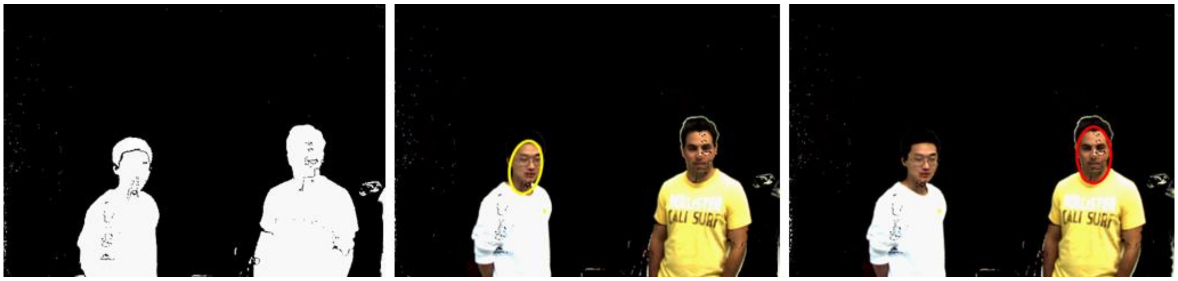

Figure 4.2: Head tracking results after backgroud subtraction

The Camshift tracking algorithm is designed primarily to perform efficient face tracking in a perceptual user interface. It is based on an adaptation of the Mean Shift algorithm that, given a probability density image, finds the mean (mode) of the distribution by iterating in the direction of maximum increase in probability density [18]. The primary difference between the Camshift algorithm and the Mean Shift algorithm is that the Camshift algorithm can handle continuously adaptive probability distributions whereas the Mean Shift algorithm is limited to static distributions. In the Camshift algorithm, the kernel is a simple step function applied to a skin-probability map. The skin probability of each image pixel is based on a color feature computed using histogram back projection. The color is represented using the Hue component from the HSV color model.

The Camshift algorithm is initialized by the distribution of colors in the facial region detected in the face detection phase. As an additional step, we make the process more robust by eliminating background pixels via simple background subtraction when the system is initialized without any users present in the scene. More complex background-subtraction approaches, e.g. [19, 20, 21] could be used if backgrounds are more complex or slightly moving. With the Camshift algorithm and background subtraction, the system can track each user's head position in real time. Figure 4.2 depicts the tracking results after applying the background subtraction and Camshift algorithms.

4.3 MOUTH TRACKING

To track each user's mouth, a different approach from those used in the face detection and head tracking phases was needed. Face detection does not apply here because it cannot run fast enough to capture high frequency lip motion. The head tracking algorithm, Camshift, would also be inappropriate because mouth region color is often similar to the skin color. Thus, in most cases, it fails to differentiate the specific mouth region from the whole face and track this region particularly. However, the mouth region often contains distinct static features (e.g. the nostrils, lip corners, mustaches, etc) that can be used for template tracking.

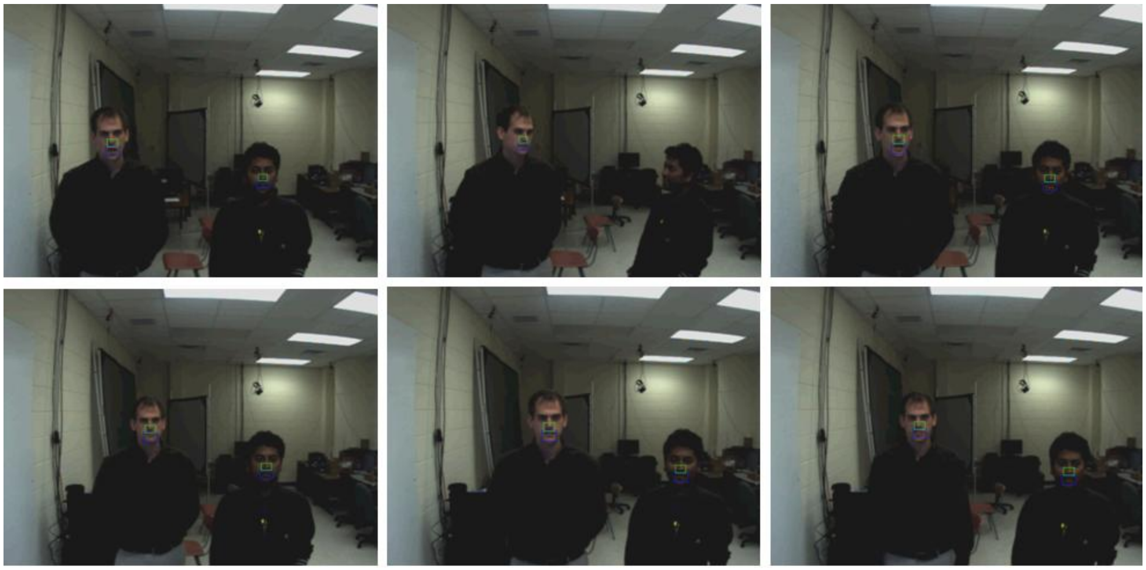

Figure 4.3: Template tracking results. The images are taken from video sequence at frame numbers 25, 50, 75, 100, 125 and 150 (13 frames per second). The green rectangle represents the nose tracking result, whereas the blue rectangle is the desired mouth region

Template tracking works best when there are significant static features moving throughout an image [22]. Given a selected template image region (typically a small region within a starting image), a search region is defined based on anticipated motion from frame to frame. For a particular search region, the template is matched by sliding it, pixel by pixel from the top left to the bottom right of the search region, and finding the overall best match.   An assumption is made that the features have not changed significantly from frame to frame.

Mouth tracking begins when a frontal face is detected. There are two reasons for this. First, the mouth is always within the face, so the virtual human has to find the face first. Second, when a user wants to talk to the virtual human, he/she is expected to face towards the virtual human, and as a result, the frontal face is detected. If this frontal face is not detected mouth tracking will cease for that particular participant.

While the intent is to track the mouth, a particular image of the mouth region is not suitable for use as a template. The features of the mouth region change frequently over the duration of spoken speech (for example, when the mouth is open or closed, when the teeth are visible or invisible etc.) making it difficult to match a static template of the mouth region across multiple frames. Consequently, we use the nose region as a template to localize the mouth region. There are three reasons for this choice. First, the nose region is relatively stable. Whether one is moving or speaking, the features of the nose region are seen to change very slightly. Second, the nose has two well defined features (i.e., the nostrils) making it easier to locate. Third, among all the features of the face, the nose region is the closest to mouth region, thus minimizing the search window required for localizing the actual mouth region.

The template matching is performed based on minimization of the sum of squared differences (SSD) distance measure between the template and the input video frame. For a displacement (*u*, *v*) the distance *D(u,v)* between the reference template and input video frame is defined as:

$$D(u,v) = \sum_{x,y}(f(x,y,t) - T(x-u, y-v, t-1))^2 \qquad (4.3.1)$$

In (4.3.1), *f* (*x*, *y*, *t*) denotes the grayscale value at position (*x*, *y*) at time *t* in image *f*, and *T* represents the template. The best match position is determined by finding the minimum difference value *D*(*u*, *v*) that is greater than a predefined threshold. A threshold of 0.3 is used for the current work, although this must currently be determined empirically.

To track the nose template from frame to frame, a suitable search region must be chosen. The search window is the region where the nose would possibly appear in the current frame relative to the previous frame. It should be neither too big (low efficiency) nor too small (low accuracy). The size must be empirically determined based on the particular application. Given that the intended application would not have rapid head motion, the search region can be small. As an example, a search window that is three times the width and height of the nose template has been sufficient for robust tracking in the multi-party system. Thus the original nose could move as far as one-nose-size distance in all directions. Figure 4.3 shows the mouth tracking result after applying the template matching algorithm.

The search region for mouth tracking is defined and constrained by the face detection result. The head tracking result could reduce the search region for the mouth, however, as the head tracking will incorporate the user's neck with the face together, it's not accurate to estimate the search region for mouth due to this displacement. Therefore, results from face detection will use to establish the search region for mouth tracking.

4.4 LIP MOTION ANALYSIS

To identify the speaking state of each person in the scene, the current system uses a heuristic that frequent lip motion indicates speaking whereas non-moving lips indicate silence.

Methods for efficient lip contour tracking based on active contours and optical flow have been studied in the literature [27,28,29]. These approaches require high resolution images and some even require the participant to stand at a fixed position. However, in our system, the video images captured are in low resolution because participants stand some distance away from the camera and from each other when engaged in the multi-party conversations. In addition, since our goal is to build a more comfortable atmosphere to enable multiple users to interact with the ECA, there should be minimal restrictions on where he/she stands, how he/she acts, etc. Therefore, a more robust approach is proposed to determine the speaking state of the user based on the presence or absence of a "void" in the lip region even when the user is walking around in the room. If the lips are closed, the mouth region is largely skin and lip tone. When the lips are opened, the mouth region contains the teeth and inner mouth. The inner mouth is generally darker that the surrounding scene. Thus, when the lips are opened there is a central region of pixels that is highly dissimilar to the mouth region when the lips are closed.

Figure 4.4: The change of the inner “void” region during speech. The images are taken from 6 consecutive video frames (13 frames per second)

When users speak, the area of the void will increase and decrease with the opening and closing of the mouth. Assuming the user is forming words (not just making a single sound), the area will change rapidly. To detect this phenomenon, the system calculates the change in the number of pixels classified as “void” pixels from frame to frame. If the change is above an empirically determined threshold, the user is assumed to be speaking. The user is determined to be silent when the change is below the threshold for more than 2 seconds. Figure 4.4 shows how the inner void changes when the user is speaking from frame to frame.

The lip motion analysis algorithm is robust to different users and is computationally efficient. It also works well in the presence of facial hair, where a contour-based lip tracking method may be less robust. However, lip motion is at best only an approximation of speaking. To obtain a complete picture, the audio channel must be correlated with the visual channel. This is a potential direction for future work.

CHAPTER 5

EMBODIED CONVERSATIONAL AGENT GAZE MODEL SIMULATION

The intended purpose of the multi-party tracking system is to provide enough information to an ECA, such that the gaze behavior of the ECA is believable. Believability is influenced by many aspects, but the ECA is ultimately limited by the information it can sense about the real world. In multi-party situations, this limitation is particularly noticeable, particularly if the ECA does shift its gaze between people with emphasis on the person or people who are speaking. Furthermore, when a person is speaking, the ECA gaze will be directed at the mouth region more often, as lip motion is typically used to assist a listener in determining the words that are spoken.

The following information is provided by the tracking system (over a Virtual Reality Peripheral Network (VRPN) interface [23]):

1) the total number of people detected
2) whether each person is frontal facing
3) whether each person is speaking

With respect to ECA gaze behavior, simulations may model the user as a single point in space. The effect achieved by this, unfortunately, is interpreted as staring, as though one point on the user's face is more interesting than any other. A more natural gaze model would have the ECA's gaze shift between various regions of interest on the face. For example, listeners tend to shift gaze between the speaker's eyes and mouth. As discussed earlier, previous work suggests that non-random gaze models improve conversational flow and ECA believability [8, 10].

Inferred situation	P1 frontal facing	P1 speaking	P2 frontal facing	P2 speaking	ECA speaking
P1 speaks to P2 and ECA	1	1	0	0	0
P2 speaks to P1 and ECA	0	0	1	1	0
P1 speaks to ECA	1	1	1	0	0
P2 speaks to ECA	1	0	1	1	0
P1 and P2 speak to ECA	1	1	1	1	0
ECA speak to P1 and P2	1	0	1	0	1
ECA speaks to P1	1	0	0	0	1
ECA speaks to P2	0	0	1	0	1

Table 5.1: Partial map between inferred situation and detected visual features of P1, P2 and ECA

The approach implemented in this work is to select a gaze model that is appropriate for the current situation. The situation is a function of each participant's conversational state.

$$Gaze\ Model = f(S, CS_1, CS_2, \dots CS_n) \qquad (5.0.1)$$

In (5.0.1), S is the speaking state of the ECA (speaking or silent) and CS is the conversational state of each participant, (frontal facing or not, and speaking or silent).

The reason for this function is to select a different gaze model for each combination of visual cues provided by the tracking system. For a 2-person (in addition to the ECA) case, the model is specified by 5 Boolean values: {P1's frontal face detected or not, P1 is speaking or not, P2's frontal face detected or not, P2 is speaking or not, ECA is listening or speaking} where P1 and P2 denote the two persons in question.

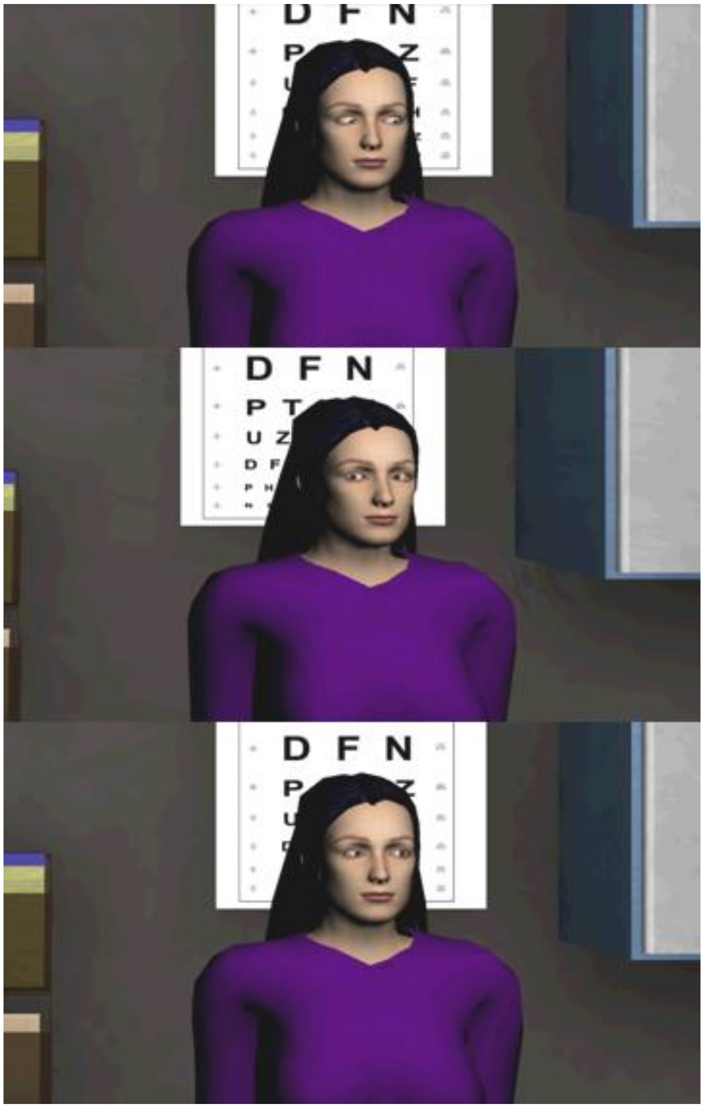

Figure 5.1: A virtual ECA's gaze movement during a multi-party conversation

This information can be used to infer a conversational situation. For example, when P1 is facing the ECA and silent; P2 is facing the ECA and speaking; and the ECA is silent, this implies that the ECA should exhibit behavior indicative of listening to P2. Some other combinations and inferred situations can be found in Table 5.1. Note that not all situations can be inferred from this information, such as one participant talking with another. Such a situation is certainly possible, but audio information would be needed to determine this state.

	P1 eyes	P1 mouth	P2 eyes	P2 mouth	Other	Update time (s)
P1 eyes	0.16	0.08	0.36	0.20	0.20	1.16
P1 mouth	0.16	0.08	0.36	0.20	0.20	0.40
P2 eyes	0.16	0.08	0.36	0.20	0.20	1.93
P2 mouth	0.16	0.08	0.36	0.20	0.20	0.44
Other	0.16	0.08	0.36	0.20	0.20	1.60

Table 5.2: Example transition table of Markov gaze model with gaze transition possibilities and gaze duration time

However, in this case, it is assumed that the majority of the attention is devoted to the speaker, not the ECA, and thus the behavior of the ECA is less important.

The concept for the gaze models used to drive gaze behavior is based on work by Fukayama et al. [6]. The gaze model is a form of a Markov finite state machine. A Markov finite state machine, describes a finite set of states and probabilities of transitioning from one state to another. While the traditional description of a Markov finite state machine does not include a time element, we modify this to also define state update times, i.e., how frequently the model tries to transition from the current state to another state.

Each state in the gaze model determines where the ECA should direct its gaze. There are two states for each conversational partner (face and mouth region) as well as a state for “other”, meaning gaze at something else. This could be a random location or could be used to try to take a conversational turn.

An example gaze model is shown in Table 5.2 that is used for the conversational situation where the ECA listening to the second of two conversational partners. This gaze model would be chosen when person 1 is frontal facing and not speaking, while person 2 is frontal facing and speaking, and the ECA is not speaking. Figure 5.1 depicts the ECA with her gaze movement in a multi-party conversation.

CHAPTER 6

EXPERIMENTAL RESULTS

Seven experiments were designed to evaluate the system accuracy under varying conditions. Each experiment corresponded to a tracking environment variable. During each experiment examining a particular tracking variable, other variables were held constant (unless otherwise indicated) at a default value. Results are reported in terms of the impact on the performance of each of the four phases (i.e. face detection, head tracking, mouth tracking, and lip motion analysis) of the hierarchal head and mouth tracking system. The seven environment variables and their default values are shown in Table 6.1.

	Environment variable	**Default value**
1	Face Occlusion	No occlusion
2	Face Rotation	Frontal facing, no rotation
3	Face Size	100cm to the camera
4	Background	Laboratory background
5	Facial Features	Asian with clean face
6	Light Intensity	Normal laboratory lighting
7	Number of Users	One

Table 6.1: Environment variables and their default values

These variables are chosen because they are considered to be representative for testing the performance of a face tracker.

All experiments were performed on a 2.5 GHz Intel Core 2 Quad CPU with 2GBytes of RAM and 2.0 MBytes of cache. The program was written in the C++ programming language. The camera used for all experiments was the Unibrain Fire-I Pro, 640x480 60fps IEEE 1394a camera.

6.1 EXPERIMENT1– FACE OCCLUSION

6.1.1 FACE OCCLUSION DESIGN

Face occlusion occurs when a face is partially or fully obscured by another environmental object. Two types of occluding objects were considered for the experiment: a sheet of paper and another face. The reason that two occluding object types were chosen is that a human face is treated differently in the face detection phase from arbitrary objects, meaning that performance may be different.

Figure 6.1: Three different users in the test. From left to right are user1, user2, user3.

The experimental design consisted of the percentage of each face covered, the user being occluded, the direction of occlusion, and the occluding object type as the independent variables and the detection and tracking accuracy as the dependent variable. For the test of paper-over-face, a sheet of paper was used to gradually cover the face from left to right and bottom to top. This was tested on three different users, see Figure 6.1. For the test of face-over-face occlusion, two users faced the camera, approached each other, and passed each other, see Figure 6.6. This was designed because it's a typical occlusion situation which could happen in a multi-party conversation, for example a crowd situation.

6.1.2 FACE OCCLUSION RESULTS

For the first test, paper-over-face, three users' face occlusion percentages when the face was first not detected are listed in Table 6.2, broken down by the direction of face occlusion. It was found that if more than approximately 33% of the face was covered from left or right side, the face detection, mouth tracking and lip motion analysis would fail, as shown in Figure 6.2. If more than approximately 20% of the face was covered from top or more than approximately 30% of face was covered from bottom, the face detection, mouth tracking and lip motion analysis failed as shown in Figure 6.4. Head tracking performance varied during the experiment in terms of a location displacement as shown in Figure 6.3 and Figure 6.5. In the second test of face-over-face, the results as shown in Table 6.3 were consistent with the first test, varying insignificantly between users, see Figure 6.6 and Figure 6.7.

	Face covered from left	Face covered from right	Face covered from top	Face covered from bottom
user1	35%	35%	20%	35%
user2	30%	30%	20%	30%
user3	35%	35%	20%	30%
Average	33%	33%	20%	30%

Table 6.2: Failure coverage of face occlusion when the face is no longer detected for paper-over face test

	Occluded by user1	Occluded by user2	Occluded by user3
user1	**/**	35%	35%
user2	30%	/	35%
user3	35%	30%	/
Average	33%	33%	35%

Table 6.3: Failure coverage of face occlusion when face is no longer detected for face-over-face test

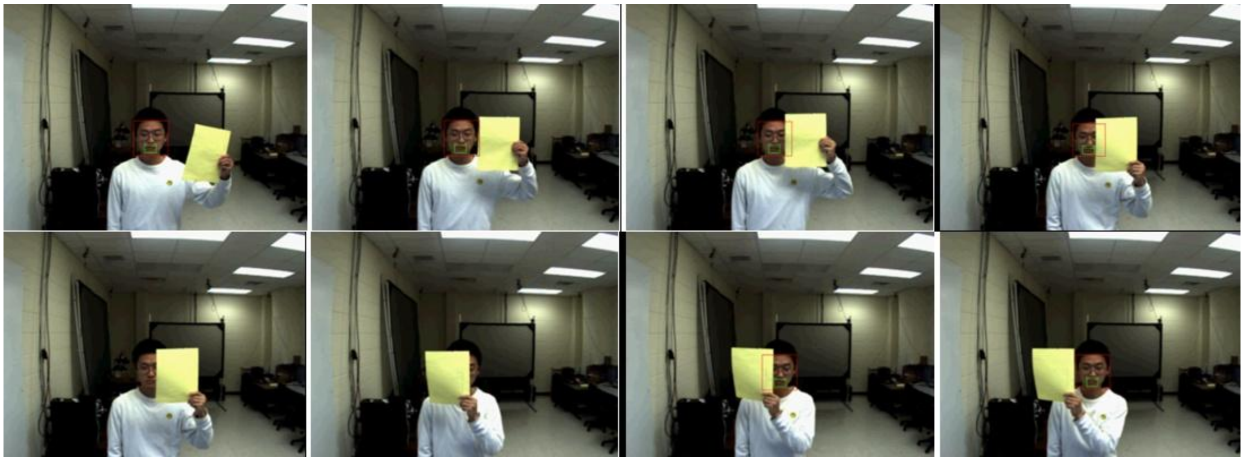

Figure 6.2: A user used a yellow paper to gradually cover his face from left to right. The face coverage of the eight pictures read left to right, top to bottom are 0%, 1%, 15%, 33%, 35%, 60%, 32%, and 1%. The face detection result is in red rectangle and mouth tracking result is in green rectangle.

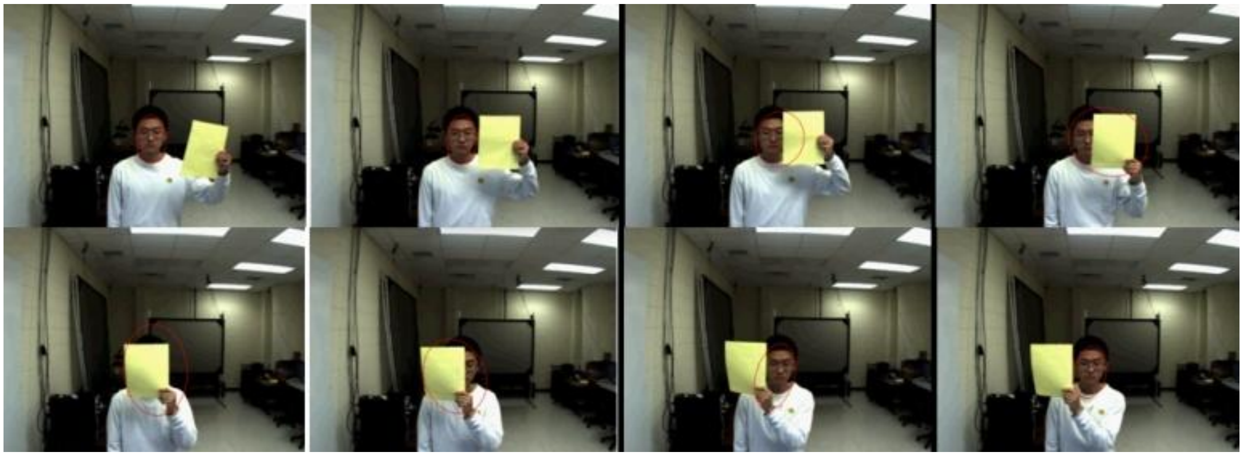

Figure 6.3: The head tracking result (red ellipse) of Figure 6.1.

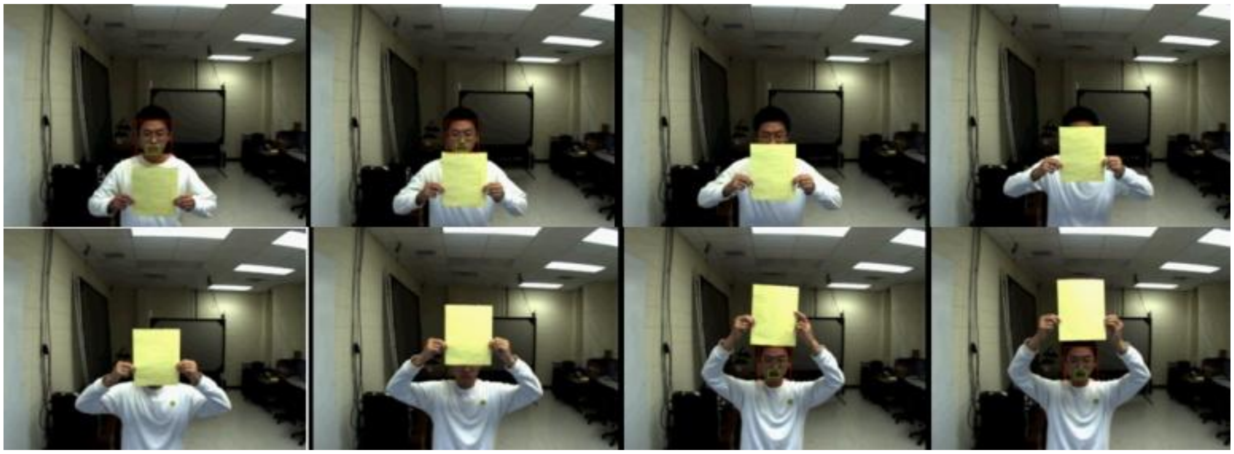

Figure 6.4: A user used a yellow paper to gradually cover his face from bottom to top. Face coverage of eight pictures read left to right, top to bottom are 0%, 1%, 30%, 75%, 100%, 70%, 20%, 5%. The face detection is in red rectangle and mouth tracking is in green rectangle.

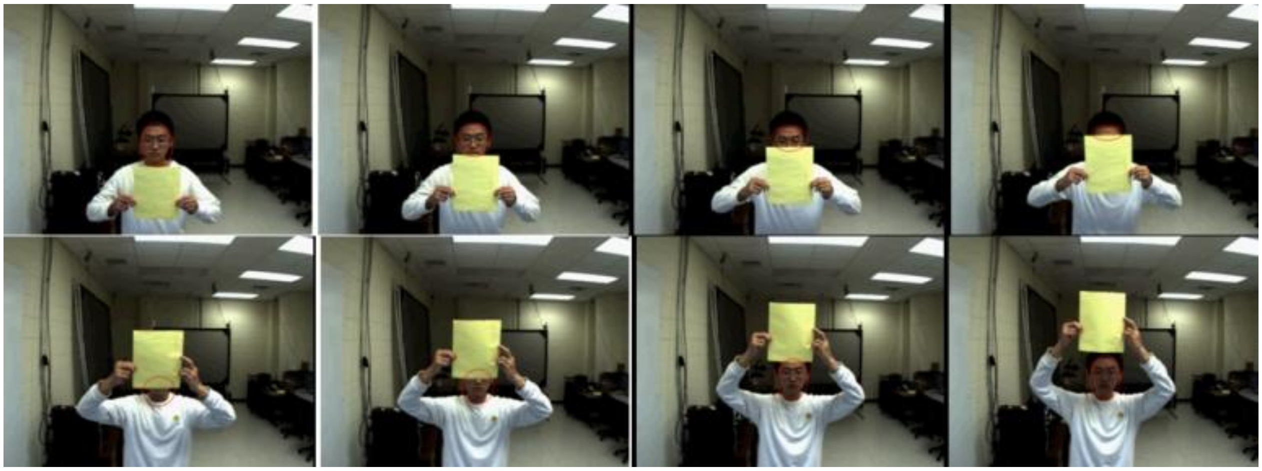

Figure 6.5: The head tracking result (red ellipse) of Figure 6.4.

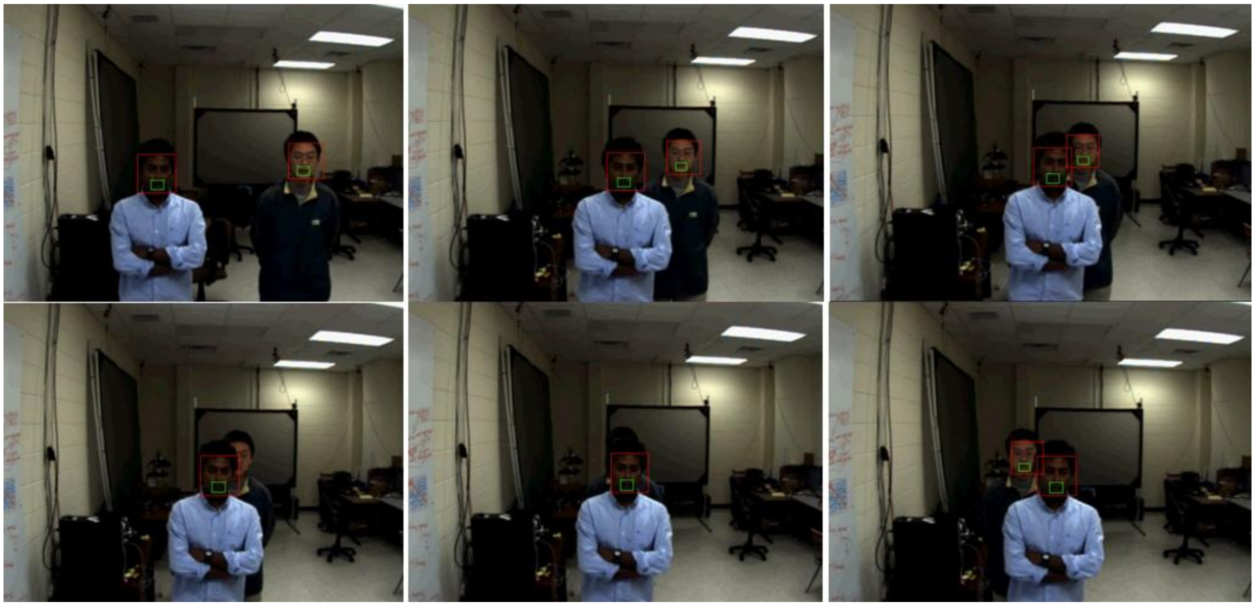

Figure 6.6: Face detection (red rectangle) and mouth tracking results (green rectangle) of face-over-face test, when user1 was occluded by front user2. The coverage of user1's face in pictures read left to right, top to bottom are: 0%, 0%, 20%, 50%, 100%, 25%.

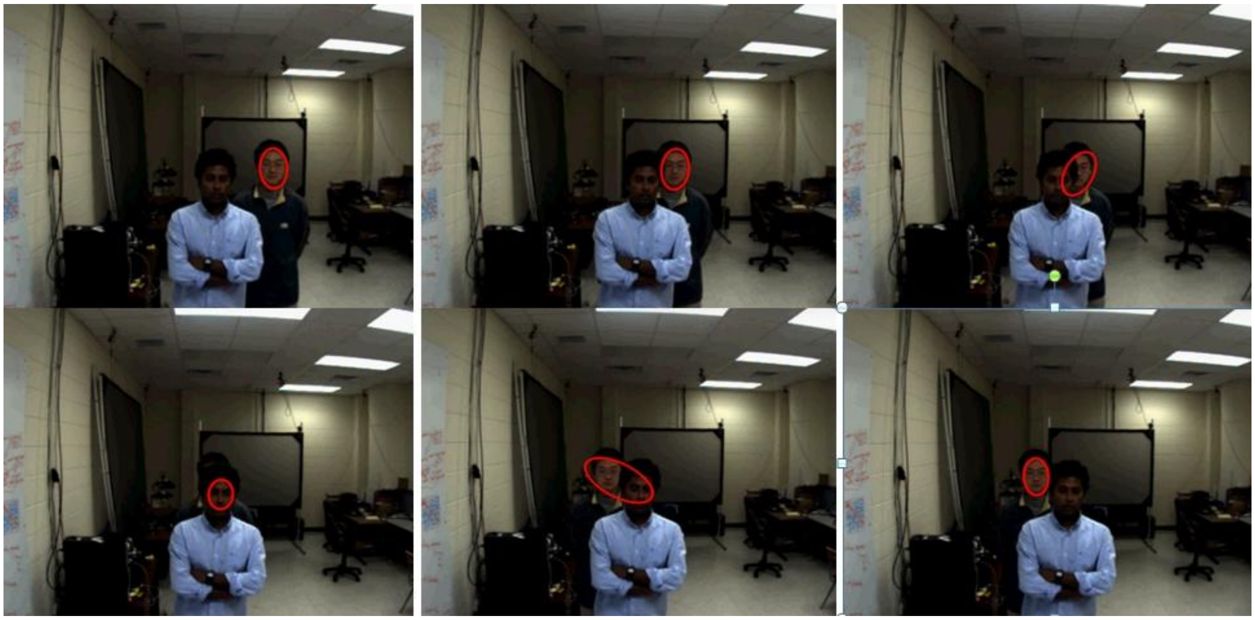

Figure 6.7: Head tracking results (red ellipse) of user1, who was occluded by front user2 for the face-over-face test. The coverage of user1's face in pictures read left to right, top to bottom are: 0%, 5%, 50%, 100%, 10%, 1%.

6.1.3 FACE OCCLUSION DISCUSSION

In the hierarchical head and mouth tracking system, the face detection is based on the Haar-like features. The more face covered, the less face features are available for face detection. Thus the face detection will fail when the face is covered above a threshold limit. According to the percentages of face failure coverage, it was concluded that nose, eye, and mouth are the most important features for face detection.

The head tracking phase was the most robust during the occlusion experiment. This is because the head tracking phase uses color information, and adjusts as the face region color shifts over time. This means that as the face is occluded, the face region color changes. A position displacement occurs, however, when the face is no longer detected, meaning that the facial color histogram remains constant.

Mouth tracking and lip motion analysis were not considered in this experiment because both phases are entirely dependent on face detection. This is because the template matching requires a face detection to be initialized and mouth tracking is based on the template tracking. Lip motion analysis is based on mouth tracking, and thus will also fail when face is not detected. Both will immediately recover when a face is detected. Thus, the tracking results listed in Table 6.2 apply equally to mouth tracking and lip motion analysis.

Through this experiment, it is concluded that head tracking is necessary and more reliable in locating the user's position in terms of head position than face detection under the various occlusion situations. Mouth tracking and lip motion analysis is robust to partial face occlusion (less than 1/3 of face size), which can be used for identifying the conversational state of user.

6.2 EXPERIMENT2 – FACE ROTATION

6.2.1 FACE ROTATION DESIGN

Two types of face rotation were examined independently in this experiment, i.e., in-plane rotation and out-of-plane rotation, because they were widely used for evaluating rotations of face detection in 2D and 3D space.

The first test examined in-plane face rotation. The user frontal faced the camera and rolled his/her face slowly clockwise and counterclockwise in the plane of the camera image. Detection and tracking results were observed for each phase. In addition, the in-plane rotation angle was measured in the captured image when the face detection first failed. This test was also conducted on three different users as in experiment 1.

The second test examined out-of-plane face rotation. Out-of-plane rotation is more difficult to measure, as the angle of rotation is more difficult to measure in the image. To accurately measure this variable, an indirect technique was used. Users were instructed to look at predefined locations on a wall directly behind the camera and parallel to the camera plane. The marked locations were separated by 10cm horizontally and 10cm vertically. Each user stood at two distances (100cm and 200cm) away from the camera, centered with respect to the camera plane and turned their head gradually and slowly in the four out-of-plane rotation directions, i.e. left, right, top, bottom, as shown in Figure 6.8. The reason 100cm and 200cm were chosen is because they represented personal distance and social distance respectively. According to Isa N. Engleberg [24], personal distance starting around 46 cm from our person and ending about 122 cm away is used in conversations with friends, to chat with associates, and in group discussions while social distance that ranges from 120 cm to 240 cm away from you is reserved for strangers, newly formed groups, and new acquaintances. When face detection or tracking first failed, the marked dot that the user was currently looking at was recorded. The arctangent of the distance between the user and camera and the distance between the camera and that dot determined the out-of-plane rotation angle. This angle when the user rotated head to look at a marked dot on the wall can be calculated according to (3), where θ is the out-of-plane rotation degree. d_1 is the distance between the marked dot where the user was looking at and the camera. d_2 is the distance between the user and the camera.

$$\theta = \tan^{-1}(d_1 / d_2) \times 180° / \pi \qquad (6.2.1)$$

6.2.1 FACE ROTATION RESULTS

The failure angles of face detection for the rotation tests for the three users standing at 100cm distance to the camera are shown in Table 6.4 and Table 6.7. The average failure angle for face detection was approximately 25 degrees for in-plane rotation and 42 degrees for out-of plane rotation, being slightly worse for top to bottom rotation than others. These tests are depicted in figures [6.9, 6.10, 6.11, 6.12, 6.13, 6.14]. The results of out-of-plane rotation at

200cm are shown in Table 6.8, as depicted in Figures 6.15. From this data it can be seen that distance does not have a significant effect on the failure angle for typical distances.

During the experiment, head tracking did not fail (although it did degrade) as shown in Figure [6.10, 6.11, 6.12, 6.14]. In addition, according to Table [6.5, 6.6], mouth tracking and lip motion analysis do not fail before face detection in either the in-plane or out-of-plane case. These are depicted in Figure [6.9, 6.11, 6.12, 6.13].

	Face rotates Clockwise	**Face rotates Counterclockwise**
user1	28 °	25 °
user2	25 °	24 °
user3	22 °	25 °
Average	25 °	25 °

Table 6.4: Failure angles of face detection for in-plane rotation test.

	Face rotates Clockwise	**Face rotates Counterclockwise**
user1	28 °	25 °
user2	25 °	24 °
user3	22 °	25 °
Average	25 °	25 °

Table 6.5: Failure angles of mouth tracking for in-plane rotation test.

	Face rotates Clockwise	**Face rotates Counterclockwise**
user1	28 °	25 °
user2	25 °	24 °
user3	22 °	25 °
Average	25 °	25 °

Table 6.6: Failure angles of lip motion analysis for in-plane rotation test.

	Rotate to left	**Rotate to right**	**Rotate to top**	**Rotate to bottom**
user1	42 °	44 °	41 °	38 °
user2	45 °	42 °	45 °	37 °
user3	45 °	46 °	43 °	38 °
Average	44 °	44 °	43 °	38 °

Table 6.7: Failure angles of face detection for out-of-plane rotation test, tested at 100cm distance to the camera.

	Rotate to left	**Rotate to right**	**Rotate to top**	**Rotate to bottom**
user1	44 °	44 °	43 °	37 °
user2	43 °	44 °	45 °	38 °
user3	44 °	45 °	41 °	39 °
Average	44 °	44 °	43 °	38 °

Table 6.8: Failure angles of face detection for out-of-plane rotation test, tested at 200cm distance to the camera.

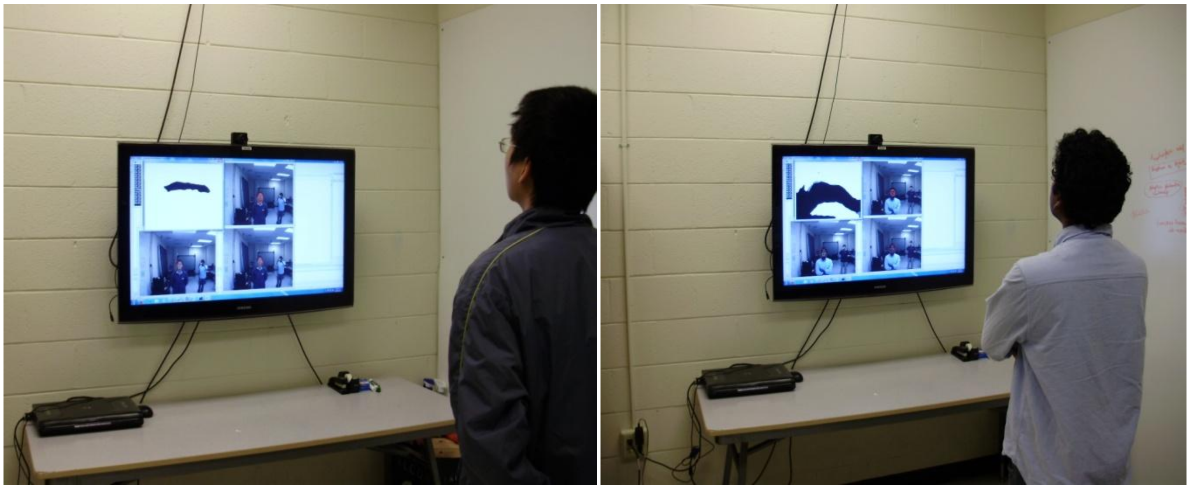

Figure 6.8: Users were looking at marked dots on the wall in the out-of-plane test.

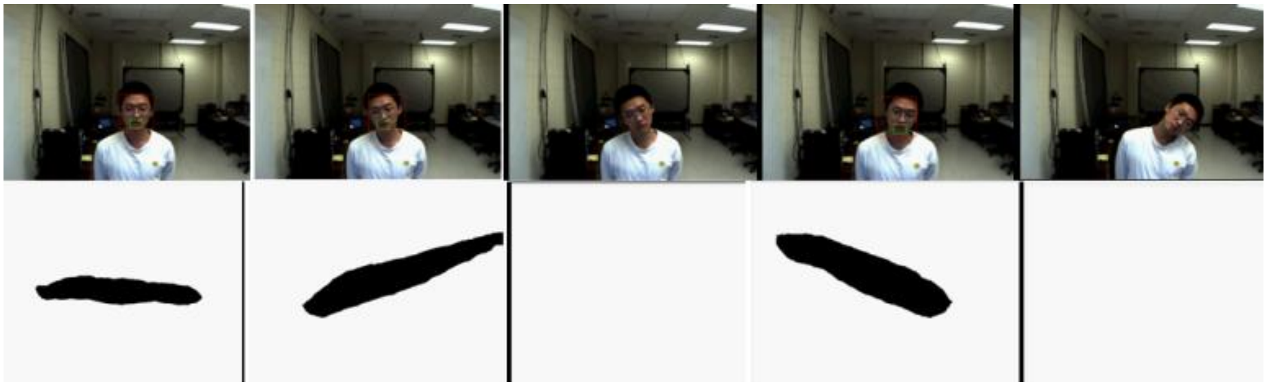

Figure 6.9: A user rotated his face in-plane. Pictures from left to right are face turning clockwise at 0 °, 24 °, 25 °and counterclockwise at 15 °, 45 °. The face detection (red rectangle), mouth tracking (green rectangle) results are in the first row and their corresponding lip motion analysis results are in the second row.

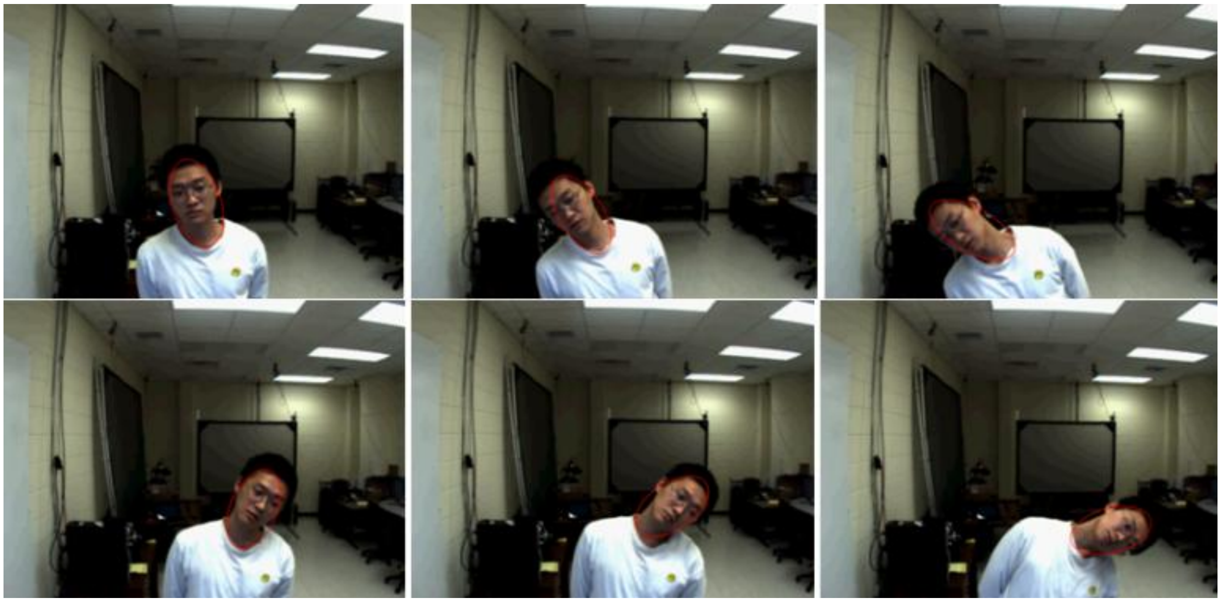

Figure 6.10: A user rotated his face in-plane. Pictures read left to right, top to bottom are face turning clockwise at 15 °, 45 °, 70 ° and counterclockwise at 30 °, 45 °, 90 °. The head tracking is represented by the red ellipse.

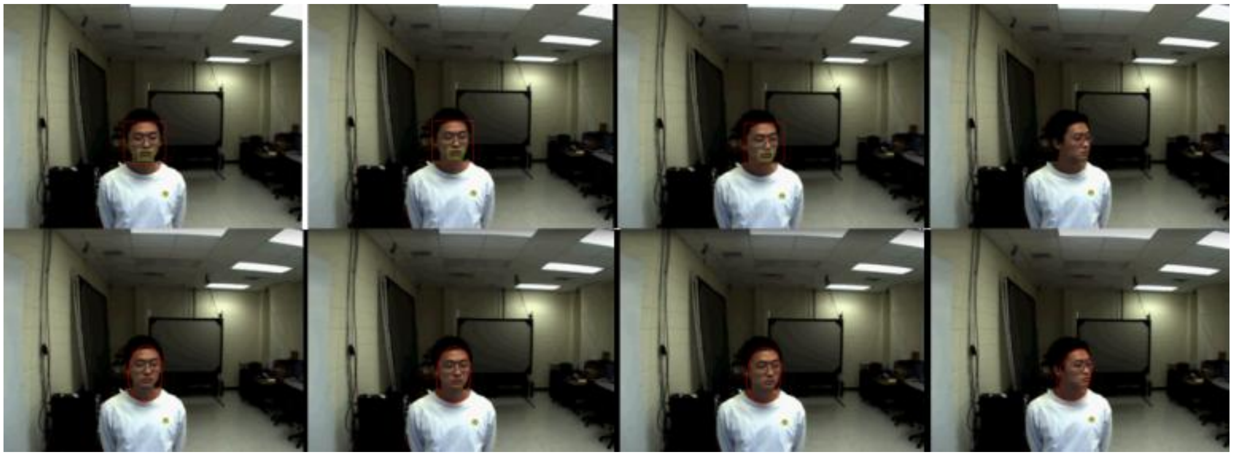

Figure 6.11: A user rotated his face to left out-of-plane. Pictures from left to right are face turning at 10 °, 20 °, 43 °, and 44 °. The face detection (red rectangle), mouth tracking (green rectangle) results are in first row and the corresponding head tracking result, the red ellipse, is in the second row.

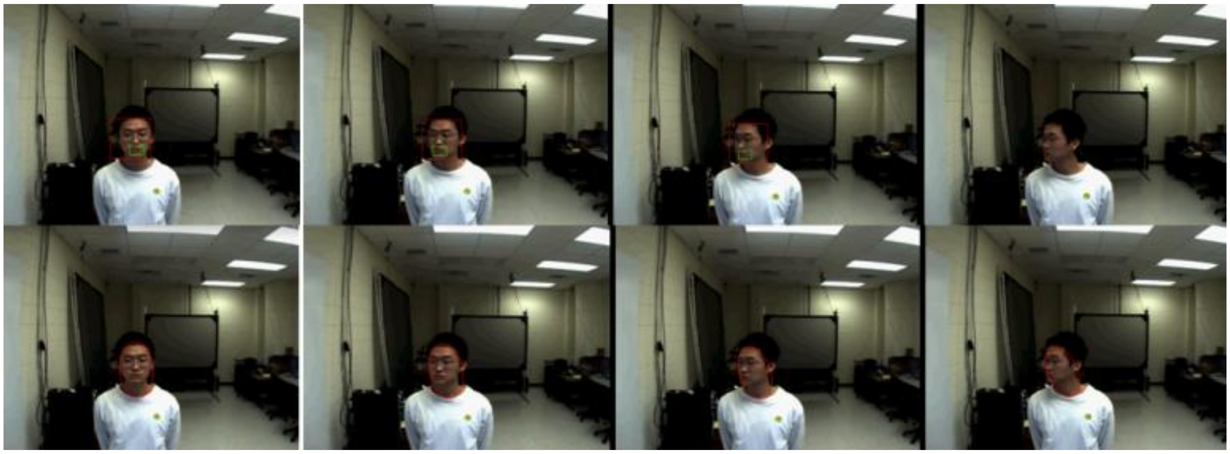

Figure 6.12: A user rotated his face to right out-of-plane. Pictures from left to right are face turning to at 0 °, 20 °, 43 °, 44 °. The face detection (red rectangle), mouth tracking (green rectangle) results are in first row and the corresponding head tracking result, the red ellipse, is in the second row.

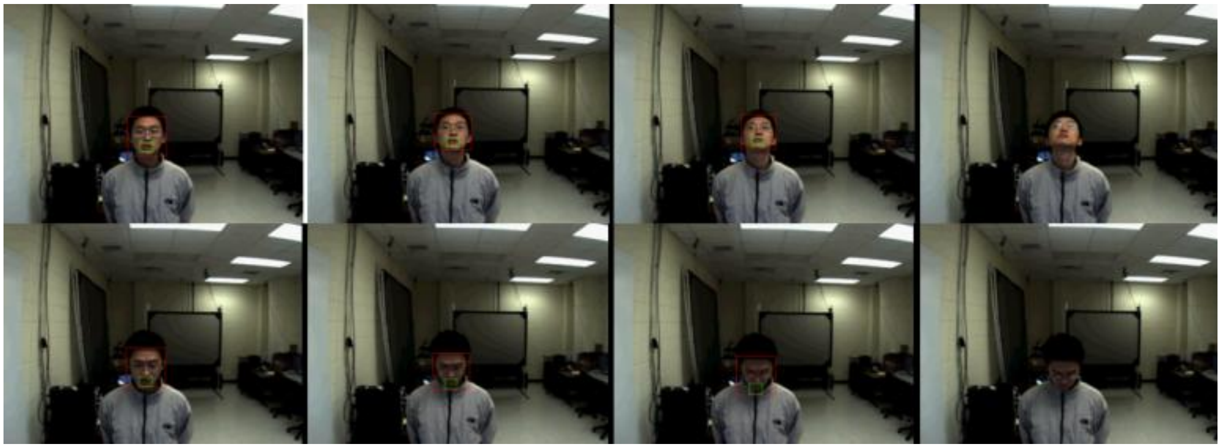

Figure 6.13: A user rotated his face out-of-plane. Pictures read left to right, top to bottom are the face turning at 0 °, 20 °, 42 °, 43 °to top and 10 °, 20 °, 35 °, 40 °to bottom. The face detection result is in red rectangle and mouth tracking result is in green rectangle.

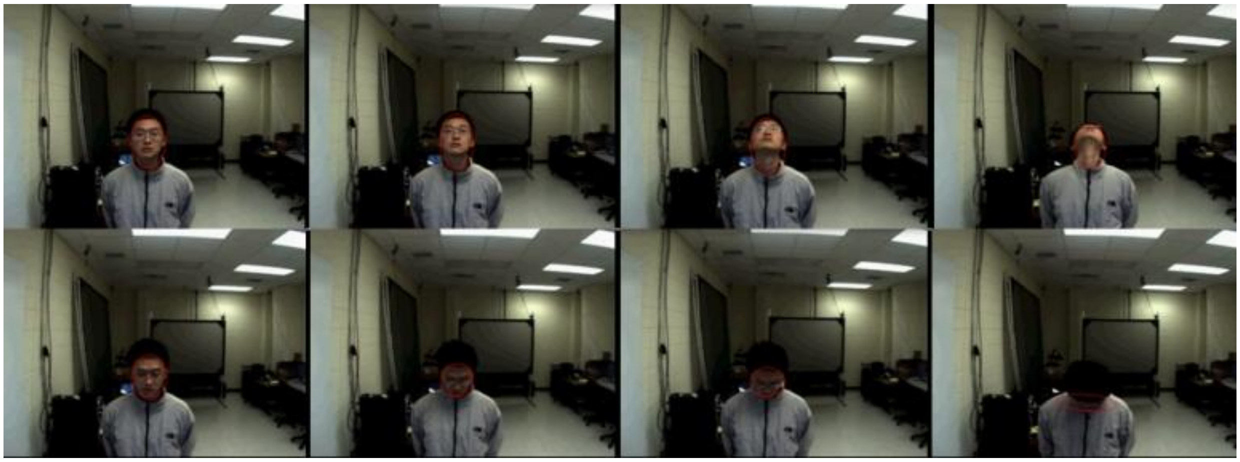

Figure 6.14: A user rotated his face out-of-plane to top at 0 °, 30 °, 50 °, 90 °and to bottom in 10 °, 30 °, 50 °, 90 °, read left to right, top to bottom. The head tracking result is represented by the red ellipse.

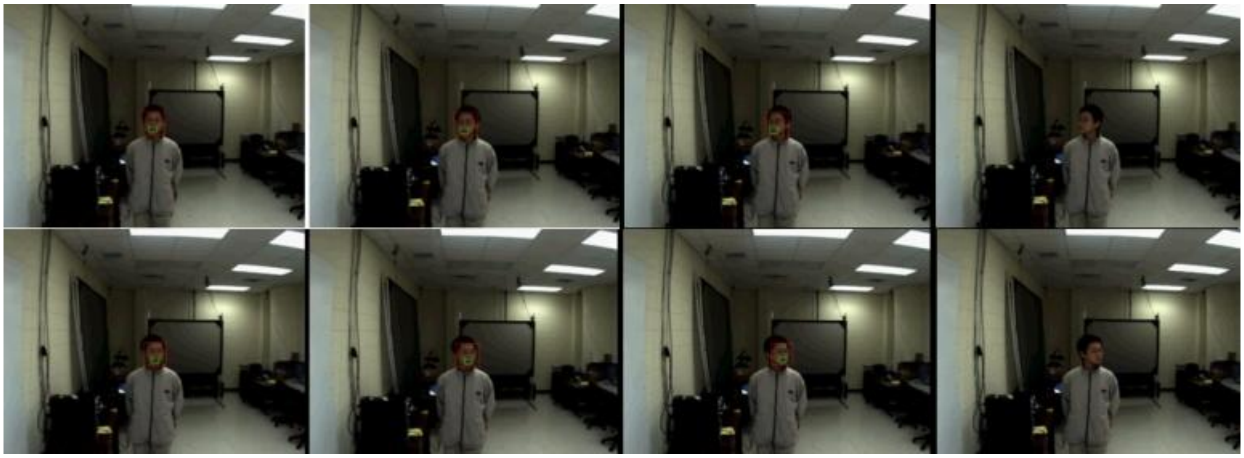

Figure 6.15: A user stood at 200cm to the camera and rotated his face out-of-plane. Pictures read left to right, top to bottom are face turning to right at 0 °, 30 °, 40 °, 45 °and left at 10 °, 30 °, 40 °, 45 °. The face detection result is in red rectangle and mouth tracking result is in green rectangle.

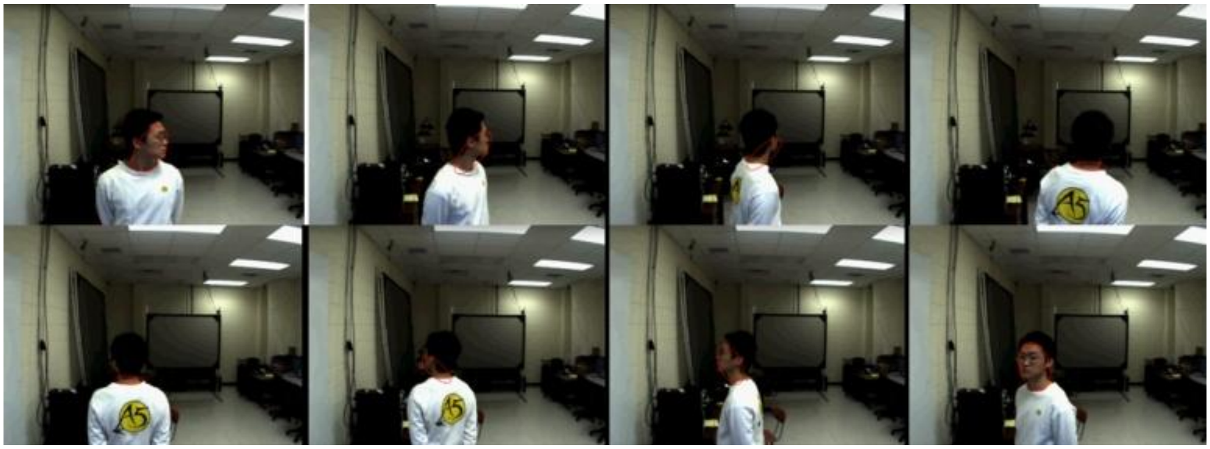

Figure 6.16: A user looked around (head tracking results represented by red ellipse) in 360 °.

6.2.3 FACE ROTATION DISCUSSION

The face detection results can be explained in terms of the Haar-classifier training set. In this system, a frontal-face training set with little face-rotation in-plane or out-of-plane is used. The features used in these sets are not rotation invariant, therefore, when the face rotates out of a certain degree, in-plane or out-of-plane, face detection fails. Similar to experiment 1, when face tracking fails, mouth tracking and lip-motion analysis also fail.

Also similar to experiment 1, head tracking is robust with respect to face rotation because the color histogram of the head region remains largely constant as the face is rotated. Figure 6.16 shows this as the head rotates 360 degrees. This supports the conclusion from Experiment1 that head tracking is more reliable in locating a user’s position in the room than face detection. Furthermore, when the face is not detected it is less important what facial features are present, but still important to know the general location of another person.

Mouth tracking and lip motion analysis did not fail before the face detection failed in the experiment. The reason varies by the type of rotation. For in-plane rotation, the mouth region is near the axis (around neck) of the head plane. Therefore, the location of mouth region varies very little during in-plane head rotation so that mouth region estimation is accurate and mouth tracking works well. For out-of-plane rotation, the mouth region locates in the same vertical line under the nose, which is not changed in out-of-plane rotation. Therefore, during out-of-plane rotation, mouth tracking and lip motion analysis will always work until face detection fails and template is lost. However, they both will immediately recover when a face is detected.

In conclusion, mouth tracking based on template matching and the lip motion analysis based on the mouth tracking are robust to most rotation situations and thus reliable for identifying the conversational states of user.

6.3 EXPERIMENT3 – FACE SIZE

6.3.1 FACE SIZE DESIGN

This experiment was designed to determine how robust detection and tracking results are with respect to face size. The user was instructed to stand at different distances to the camera and speak to the virtual agent at each distance so that face size, in pixels (width x height), would chang accordingly. The pixel dimensions of the face represent a face size measure that is independent of distance, and camera lens/sensor (assuming square pixels). The four phases of the hierarchical head and mouth tracking system were observed and their corresponding results were recorded at each face size. The experiment was conducted on three different users, as in Experiment1.

6.3.2 FACE SIZE RESULTS

The starting and failure face rectangle dimensions of face detection, head tracking, mouth tracking and lip motion analysis are shown in Table 6.9. The face detection, head tracking, mouth tracking and lip motion analysis started to work when the user’s face dimension are about

180x180 in pixel numbers. When the distance became farther and the face dimensions become smaller than 27x27 in pixel numbers, the lip motion analysis failed. Then at approximately 25x25 pixel numbers of face dimension, mouth tracking failed and finally the face detection failed at approximately at dimension of 23x23 in pixel numbers. The accurate failure pixel numbers of face dimension for head tracking could not be established as the laboratory setting was limited to 500cm distance in which the face dimensions could not be smaller than 20x20. The face detection and mouth tracking results are shown in first row in Figure [6.17, 6.18], the corresponding head tracking and lip motion analysis results are shown second and third row in Figure [6.17, 6.18].

	User1	User2	User3	Average
Face detection starts	180 x180	180 x 180	180 x 180	180 x 180
Head tracking starts	180 x 180	180 x 180	180 x 180	180 x 180
Mouth tracking starts	180 x 180	180 x 180	180 x 180	180 x 180
Lip motion analysis starts	180 x 180	180 x 180	180 x 180	180 x 180
Lip motion analysis breaks	27 x 27	27 x 27	27 x 27	27 x 27
Mouth tracking breaks	25 x 25	25 x 25	25 x 25	25 x 25
Face detection breaks	23 x 23	23 x 23	23 x 23	23 x 23
Head tracking breaks	< 20 x 20	< 20 x 20	< 20 x 20	< 20 x 20

Table 6.9: Starting and failure face dimension in pixel numbers for face detection, head tracking, mouth tracking and lip motion analysis when user's face dimensions become smaller

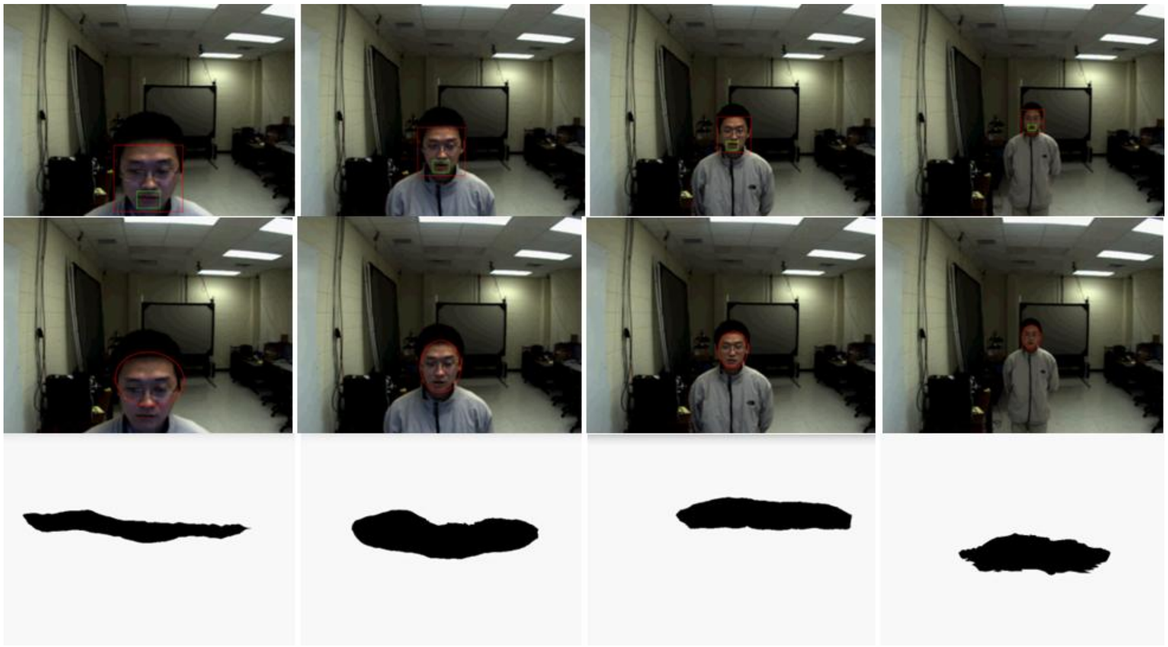

Figure 6.17: A user stood at different distances to the camera with different face dimensions (120x120, 70x70, 50x50, 40x40 in pixel numbers), read left to right. The face detection (red rectangle) and mouth tracking results (green rectangle) are in first row, head tracking results (red ellipse) are in second row and lip motion analysis is in third row.

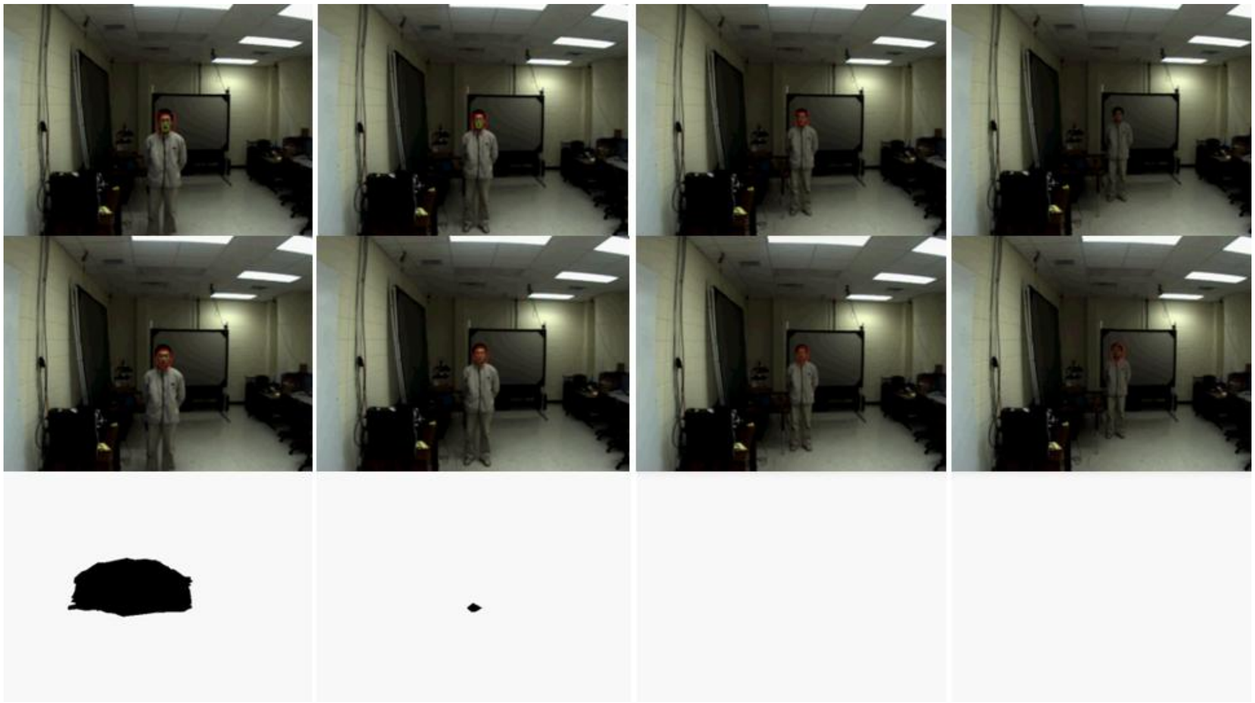

Figure 6.18: A user stood at different distances to the camera, resulting in different face pixel dimensions (30x30, 27x27, 25x25, 23x23 in pixel numbers), read left to right. The face detection (red rectangle) and mouth tracking results (green rectangle) are in first row, corresponding head tracking results (red ellipse) are in second row and corresponding lip motion analysis are in third row. (note: the last three lip tracking results are all white because the void could not be detected).

6.3.3 FACE SIZE DISCUSSION

The face could be detected when it became 180x180 pixels. This is an imposed algorithmic limit to improve performance. A successful face detection initializes head tracking, mouth tracking and lip motion analysis. When the face size decreased, the lip motion analysis failed first as the central mouth region is the smallest area being analyzed and thus becomes harder to detect and differentiate clearly. However, face detection, head tracking and mouth tracking uses larger features and thus features can be differentiated at greater distances from the camera. In a similar vein, mouth tracking was the second to fail because the nose region is the limiting size factor. Finally, the face detection is limited by small feature sizes. The process is shown in Figure 6.17 and Figure 6.18.

Although the others rely on features, which degrade quickly with decreasing size, head tracking relies on color and thus is more robust to changes in face size. It is expected that head tracking could function properly up to a size of only a few pixels, and thus could not be adequately measured within the limited space of the laboratory. Practically, however, one would impose a software limit on the size of the head tracking ellipse, which we have set as 15x15.

From this experiment, we can conclude that within a certain range of face size, from about 30x30 to 180x180 face width in pixel numbers, all the detection and tracking phases work well. According to Engleberg [24], conversational distances larger than 240cm, which is approximately 50x50 pixel number of face width, belong to public distances which are used for speeches, lectures, and theater. Therefore, face dimensions larger than 50x50 in pixel numbers (distance less than 240cm) are adequate for detection and tracking in a multi-party conversation and thus the hierarchical head and mouth tracking system is robust to different locations where users may practically stand.

6.4 EXPERIMENT4 – BACKGROUND

6.4.1 BACKGROUND EXPERIMENT DESIGN

In this experiment, two tests were designed for investigating the influence of background subtraction on the detection and tracking results. In the first test, the user walked in the lab against a complex background without applying background subtraction. The results were observed and investigated to determine if the accuracy would be influenced by the complexity of background. In the second test, detection and tracking results were compared before and after applying background subtraction. The goal was to determine if background subtraction would improve the detection and tracking accuracy over the previous case.

6.4.2 BACKGROUND EXPERIMENT RESULTS

The first test showed that head tracking was not reliable if a user stood against a complex background, especially if there was a region in the background that had the same hue color as human face, see pictures in first row of Figure 6.19. In this situation, head tracking results were inaccurate. However, the face detection, mouth tracking and lip motion analysis were largely robust to background changes as seen in the second row in Figure 6.19.

The second test showed that head tracking results were improved greatly by applying background subtraction compared to without it, as shown in Figure 6.20.

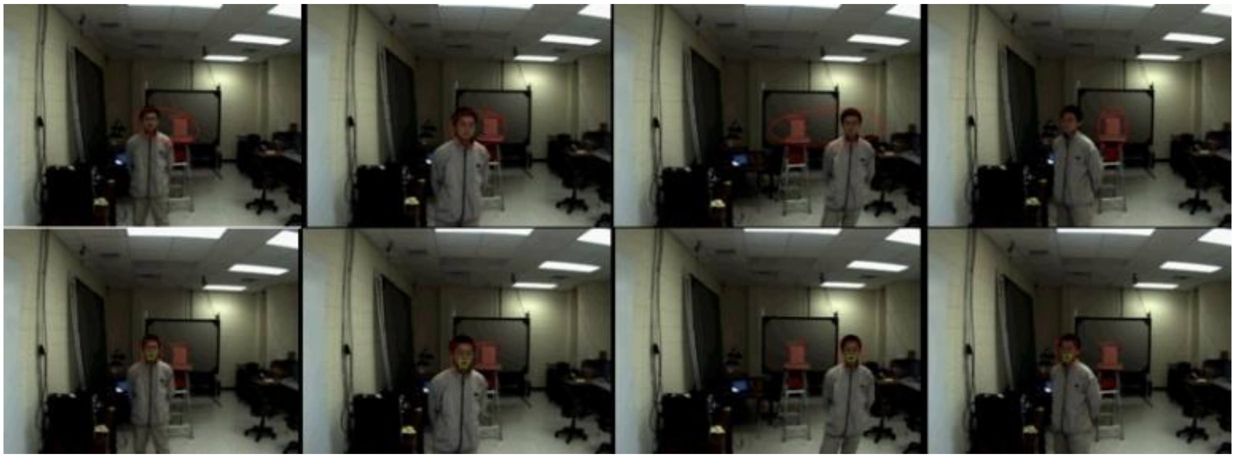

Figure 6.19: A user stood at different positions against a complex background. The head tracking result (red ellipse) is in first row and corresponding face detection (red rectangle) and mouth tracking (green rectangle) results are shown in second row.

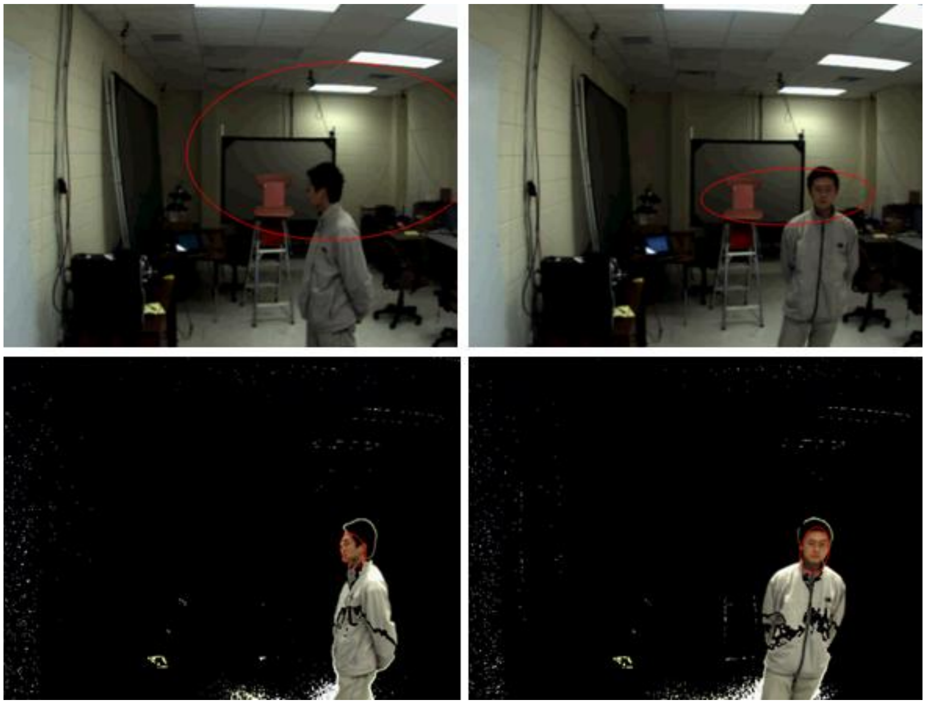

Figure 6.20: Comparison of head tracking results before (first row) and after (second row) applying background subtraction.

6.4.3 BACKGROUND EXPERIMENT DISCUSSION

This experiment showed the necessity of background subtraction for color-based tracking. The first test showed that head tracking was not reliable if the user stood against a complex background. This is because the head tracking uses the hue color of a face which is unique most of the time. However, certain background colors such as orange approximate the hue color of the human face, meaning that the head tracking may become inaccurate.

The second test shows that background subtraction eliminated errors in head tracking caused by the complexity of background. This is because the background is masked so that only colors that are part of the user image are used in tracking. This lets the face color be tracked accurately, as seen in the second row in figure 6.20.

6.5 EXPERIMENT5 – FACIAL FEATURES

6.5.1 FACIAL FEATURES EXPERIMENT DESIGN

This experiment was designed for investigating whether the face color and facial hair would influence the accuracy of tracking results. The system was tested on 20 different users with different face colors, i.e., white, yellow, and black. The users came from different background with different race, gender, age. Figure 6.21 shows three examples of them.

6.5.2 FACIAL FEATURES EXPERIMENT RESULTS

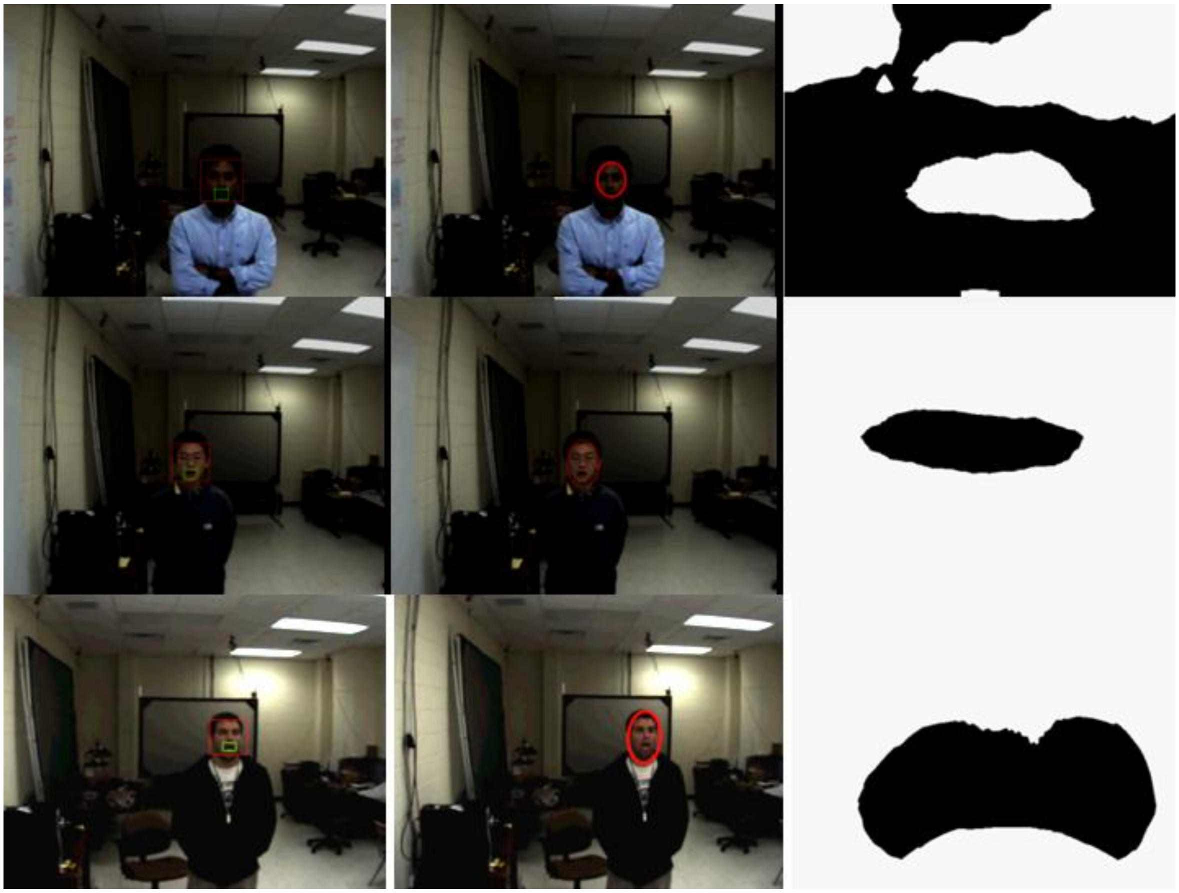

Figure 6.21: face detection (red rectangle in first column), mouth tracking (green rectangle in first column), head tracking (red ellipse in second column) and lip motion analysis (third column) of three users with different skin colors.

The results showed that face detection, head tracking and mouth tracking worked normally on all tested users. However, lip motion analysis sometimes failed on users who had a darker skin tone and facial hair in the mouth region, as seen in the first row in Figure 6.21.

6.5.3 FACIAL FEATURES EXPERIMENT DISCUSSION

For the face detection, the classifier we used was trained with thousands of sample faces for people from different race, gender, facial feature, etc. Therefore, face detection worked well on all the testing users. Similarly, the head tracking and mouth tracking are based on the face template determined from the results of face detection, and thus also performed well, as seen in the first and second column in Figure 6.21. However, the lip motion analysis failed when the user had very dark skin and strong beard at the mouth region. This is because the dark skin and black beard that covered this region made it difficult to differentiate the mouth contour by converting the right pixels and identifying the correct mouth "void".

Through this experiment, it is concluded that face detection, head tracking, and mouth tracking work well for a wide variety of people, while lip motion analysis is highly dependent on the face color and hair features in the mouth region. Lip motion analysis works much better in the mouth region which has clear color contrast and simple hair features. This is understandable because visual information is often insufficient to determine when a person who has strong hair and dark color in the mouth region is speaking, thus audio information must be included to help in this situation. In addition, the threshold algorithm for lip motion analysis could be automatically adjusted depending on the face color.

6.6 EXPERIMENT6– LIGHT INTENSITY

6.6.1 LIGHT INTENSITY EXPERIMENT DESIGN

This experiment was designed to investigate whether detection and tracking results were influenced by the intensity of lights. In this test, a user would talk to the virtual human under different lighting conditions, as adjusted by light switches in the testing laboratory. While not a rigorous test, it typifies variations in lighting that would be experienced by such a tracking system in practice.

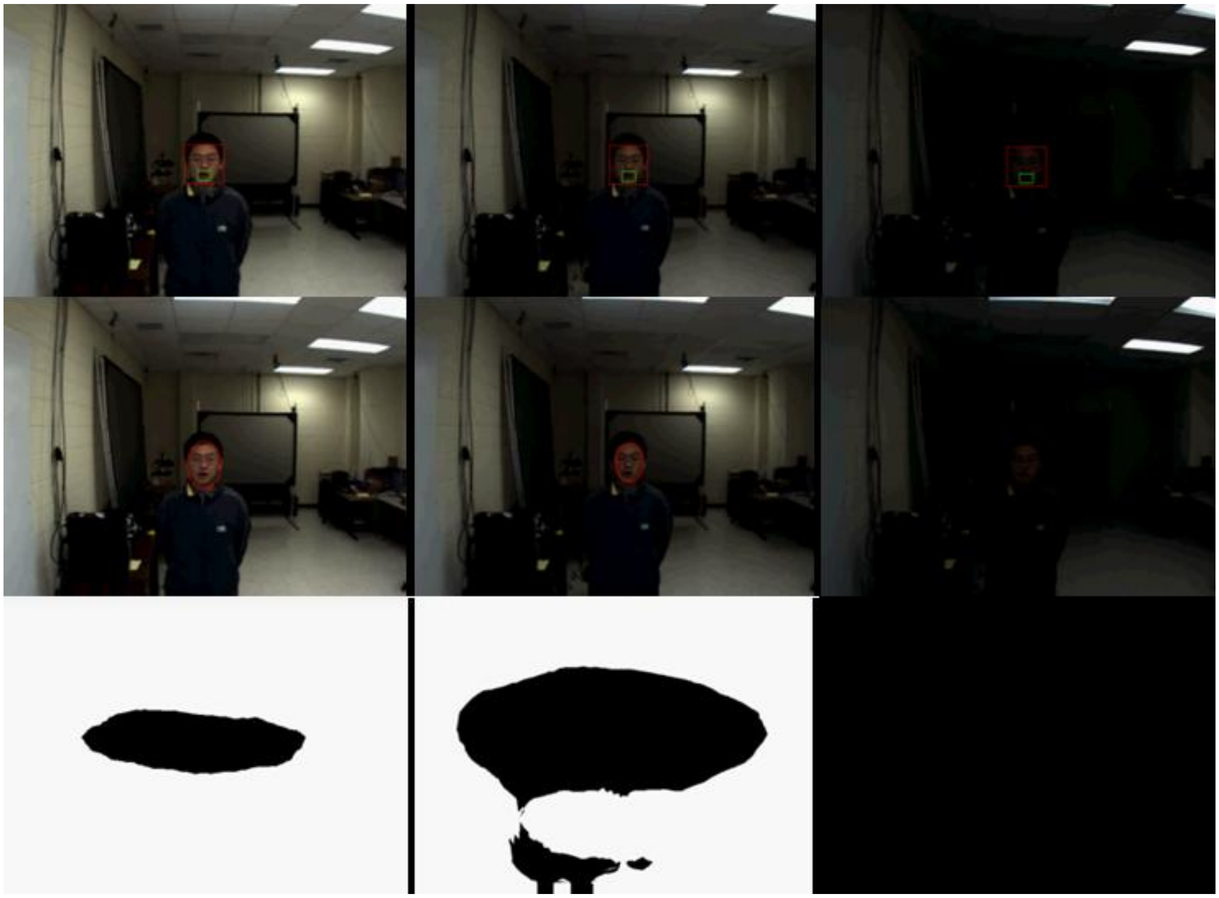

Figure 6.22: A user was tested under different light conditions, from light to dark, read left to right. The face detection (red rectangle) and mouth tracking (green rectangle) results are in first row, the corresponding head tracking results (red ellipse) are in second row and corresponding lip motion analysis results are in third row.

6.6.2 LIGHT INTENSITY EXPERIMENT RESULTS

The face detection and mouth tracking worked properly, even under very dark lighting, as seen in the first row in Figure 6.22. However, the head tracking and lip motion analysis failed in dark lighting as seen in second and third row in Figure 6.22.

6.6.3 LIGHT INTENSITY EXPERIMENT DISCUSSION

In this experiment, the color-based approaches are shown to be highly sensitive to lighting conditions. This makes sense, as perceived material color and lighting are linked. The face detection is more robust to the light intensity because it is based on detecting the facial feature, as discussed above which is less sensitive to the intensity of light. Similarly, mouth tracking is less sensitive to the light conditions because nose template is determined by the face detection phase. However, the head tracking and lip motion analysis failed to work under dark lights. Head tracking relies the facial regions being a contrasting color from other image regions. If the light is not strong enough, the hue color of face is harder to differentiate accurately to locate the contour of head. The lip motion analysis, which based on detecting the changes of black "void" in the user's mouth region, also breaks more easily. This is because the threshold algorithm which converts the mouth region into a black "void" and the analysis algorithm which identifies the opening and closing of mouth are more dependent on light conditions. Therefore, once the light intensity changes, the head tracking and lip motion analysis might fail, especially when light is low, as seen in Figure 6.22.

Through this experiment, it is concluded that the head tracking and lip motion analysis failed to work in dark lighting conditions. However, in a comfortable environment for multi-party conversation, lighting can be expected to be light enough without much change during the course of the conversation. Therefore, the hierarchical head and mouth tracking system can be expected to works well under most normal situations. Furthermore, if better lighting independence is desired, the color tracking methods could be automatically adjusted depending on light intensity measurements.

6.7 EXPERIMENT7 – NUMBER OF USERS

6.7.1 USER NUMBER EXPERIMENT DESIGN

In this experiment, the total time cost on each of the four components, i.e., face detection, head tracking, mouth tracking, lip motion analysis of the hierarchical head and mouth tracking system in every frame was measured and compared when the number of user increased. To accurately compute the processing time, all visualization effects that showed the captured images of detected face, mouth, and lip motion were closed.

6.7.2 USER NUMBER EXPERIMENT RESULTS

	Thread1					Thread2
Number of users detected	Background subtraction (ms)	Head tracking (ms)	Mouth tracking (ms)	Lip motion analysis (ms)	Overall system (ms)	Face detection (ms)
1	30	12	2	18	62	122
2	30	24	4	36	94	122
3	30	36	6	54	126	122

Table 6.10: The time spend by each component of the hierarchical head and mouth tracking system when user number increases from one to three

The times that the system spends performing the computation for each of the four tracking phases are listed in Table 6.10. Results are divided into two threads, one for face detection, and one for the remaining operations. The underlying system for the experiment was a multi-core processor, and thus the times can be considered mostly independent. It can be seen that face detection is approximately twice as computationally intensive than the rest of the phases combined. Times for face detection and background subtraction were found to be independent of the number of users, while head tracking, mouth tracking, and lip motion analysis appear to increase linearly with the number of users. Combining the components dependent on the number of users, the algorithm slows by 32 ms per frame for each additional user.

6.7.3 USER NUMBER EXPERIMENT DISCUSSION

This experiment showed that the hierarchical head and mouth tracking system is directly proportional to the number of users present. Performance decreased linearly from 62ms with one user to 126ms with three users.

Face detection is a fixed time cost in the system. It is a separate thread in the algorithm because the other phases only rely on it for initialization, and then operate independently after that time. When a new face is detected, these other phases can be reinitialized. If face detection were incorporated into the main thread, it would cause the entire system to slow considerably, while providing little new information.

Background subtraction can be seen as a fixed, but time expensive operation. However, it vastly improves the performance of the head tracking results. Users of the tracking system would need to weigh the tracking accuracy gain against the frame rate reduction.

The least amount of time is spent on head tracking, mouth tracking, and lip motion analysis. This is because these parts of the algorithm typically operate on smaller image regions.

However, as the number of users increases, more image regions need to be considered, and thus the performance of these phases is linearly proportional to the number of users.

In summary, the performance of the system reduces as the number of users increases. However, the impact for an ECA is likely to be small. The current system is designed for interactions with a few users, but not crowd situations. In crowd situations, the conversational states of each participant are less important than multi-party situations. In the multi-party situations that the system was designed for, the performance will likely be adequate, or if not, would be improved significantly by turning off background subtraction (at the expense of accuracy). One concern is that high frequency lip motion may be lost as a result of a slower frame rate. This is somewhat natural, as when multiple speakers are present, attention of a conversational participant is inherently divided.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

This paper proposes an approach to enable an embodied conversational agent (ECA) to effectively detect and respond to multiple conversational partners through computer vision and conversational agent behavioral simulation techniques. A hierarchical head and mouth tracking system identifies the conversational state of each visible person using only visual information. A multi-party Markov gazing model is proposed that drives an ECA's gaze behavior based on the detected conversational state information. Finally, the tracking system performance is evaluated to demonstrate robustness for typical multi-party conversational situations.

Our research provides a new direction in developing an ECA that can take part in multi-party conversational situations. We mainly focused on visual perception and addressed the problem of detecting the conversational state of participants through tracking algorithms. The purpose of this information was to enable enhanced non-verbal behaviors, such as gaze behavior, of an ECA in multi-party situations; thus, we provided a preliminary gazing model as evidence. Furthermore, the tracking system does not require extensive infrastructure support beyond camera calibration and is largely user-independent. Thus, the approach can be readily transplanted to other ECAs, from robotic humans to computer generated virtual humans.

Our research can be expanded in several directions. First, the impact of an ECA's gaze on the users in a multi-party conversation needs to be investigated; for example, how users feel about an ECA's attention in a multi-party conversation, whether gaze behavior influences the communication quality in multi-party situations, and whether the ECA's gaze which shifts from the user’s eyes and mouth performs better than merely gazing in the user’s eye when the user is speaking to ECA.

Second, for gaze modeling, the major bottleneck in the current approach is the creation of Markov gaze models for each conversational situation. For two conversational participants plus the ECA and five gaze possibilities (face1, face2, mouth1, mouth2, random), a 5 x 5 table of transition probabilities are needed for each detected conversational situation (18 are detectable for the example interaction). With more users, the size of gaze transition probability table will increase exponentially, and the various users’ gaze direction could be much more complicated. However, much of the information in the tables is redundant. Therefore, an automated or greatly simplified process is needed to identify and eliminate redundant information. Moreover some inferred situations should be simplified to a point where a multi-party conversation would become a crowd situation. That is, when more than N users are present, a crowd gaze model is employed. This would reduce the number of tables needed. At what point this should happen, and how to make the transition gracefully is the topic of further research.

Third, for a virtual ECA displayed on a 2D screen, the virtual perspective (an artifact of the 3D rendering process) can be adjusted to match only a single user’s viewing perspective. A matched perspective allows for correct gaze angle judgments. Thus, in multi-party conversations, those participants whose perspective does not match the virtual perspective will judge the gaze of the virtual ECA incorrectly.   To solve this problem, either holographic displays that provide true 3D (like HoloVizio 3D display system [25]), the use of robotic ECAs (like RoboThespian™) or hybrid techniques such as Animatronic Shader Lamps Avatars [26] could be used,

In addition, to better analyze the conversational states in the multi-party situations, the audio perception channel should be incorporated and correlated with the visual perception channel for each person. This hybrid situation further provides the possibility of audio-visual speech recognition.

REFERENCES

1. M. Argyle, Bodily Communication. New York: Methuen & Co. Ltd, 1988.
2. M. Argyle and M. Cook, Gaze and mutual gaze. Cambridge University Press, 1976.
3. A. Colburn, M. Cohen, and S. Drucker, The role of eye gaze in avatar mediated conversational interfaces. Technical Report MSR-TR-2000-81, Microsoft Corporation, 2000.
4. M. Garau, M. Slater, S. Bee, and M. Sasse, The Impact of Eye Gaze on Communication using Humanoid Avatars, in Proceedings of CHI'01: ACM Conference on Human Factors in Computing Systems, 309-316, 2001.
5. S. Lee, J. Badler, and N. Badler, Eyes alive. In ACM Transactions on Graphics, Siggraph, pages 637-644. ACM Press, 2002.
6. A. Fukayama, T. Ohno, N. Mukawaw, M. Sawaki, and N. Hagita, Messages embedded in gaze on interface agents -Impression management with agent's gaze. In CHI, volume 4, pages 1-48, 2002.
7. C. Pelachaud and M. Bilvi, Modelling gaze behavior for conversational agents. In International Working Conference on Intelligent Virtual Agents, pages 15–17, September 2003.
8. N. Bee, J. Wagner, E. Andr é, T. Vogt, F. Charles, D. Pizz. and M. Cavazza, Gaze Behavior during Interaction with a Virtual Character in Interactive Storytelling. International Workshop on Interacting with ECAs as Virtual Characters (AAMAS 2010), Toronto, Canada, May 2010.

9. D. Traum and Louis-Philippe Morency, Integration of Visual Perception in Dialogue Understanding for Virtual Humans in Multi-Party interaction, AAMAS Workshop on Interacting with ECAs as Virtual Characters, Toronto, Canada, 2010.
10. R. Vertegaal, R. Slagter, G.v.d. Veer, and A. Nijholt, Eye Gaze Patterns in Conversations: There is More to Conversational Agents Than Meets the Eyes. Proc. CHI2001, pp. 301–308, 2001
11. Nakano, Reinstein, Stocky, and J. Cassell, Towards a model of face-to-face grounding. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Sapporo, Japan, July 2003.
12. R. Dillmann, R. Becher, and P. Steinhaus, ARMAR II – a learning and cooperative multimodal humanoid robot system. International Journal of Humanoid Robotics, 1(1):143–155, 2004.
13. C. L. Sidner, C. D. Kidd, C. Lee, and N. Lesh, Where to look: a study of human-robot engagement. In IUI'04: Proc. of the 9th international conference on Intelligent user interfaces, pages 78-84. ACM Press, 2004.
14. Y. Matsusaka, T. Tojo, S. Kuota, K. Furukawa, D. Tamiya, K. Hayata, Y. Nakano, and T. Kobayashi, Multiperson conversation via multi-modal interface – a robot who communicates with multi-user. In Proceedings of Eurospeech, pages 1723–1726, 1999.
15. P. Viola and M. Jones, "Robust real-time object detection," International Journal of Computer Vision, 57(2), 137-154, 2004.
16. P. Viola, and M. Jones, Rapid object detection using boosted cascade of simple features. IEEE Conference on Computer Vision and Pattern Recognition, 2001.
17. D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," IEEE Trans. Pattern Anal.Machine Intell., vol. 25, no. 5, pp. 564–577, 2003.

18. G.R. Bradski, Computer Vision Face Tracking For Use in a Perceptual User Interface, Microcomputer Research Lab, Santa Clara, CA, Intel Corporation, 1998.
19. A. Elgammal, D. Harwood, L. Davis, Non-parametric model for background subtraction.In: European Conference of Computer Vision, pp. 751–767 , 2000
20. N.M. Oliver, B. Rosario, A.P. Pentland, A Bayesian computer vision system for modeling human interactions. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 831–843, 2000
21. A. Mittal, M. Paragios, Motion-based background subtraction using adaptive kernel density estimation. In: IEEE International Conference on Computer Vision and Pattern Recogntion, pp. 302–309, 2004
22. R. Brunelli, Template Matching Techniques in Computer Vision: Theory and Practice, Wiley, ISBN 978-0-470-51706-2, 2009
23. Russell M. Taylor II, Thomas C. Hudson, Adam Seeger, Hans Weber, Jeffrey Juliano, Aron T. Helser, "VRPN: A Device-Independent, Network-Transparent VR Peripheral System," Proceedings of the ACM Symposium on Virtual Reality Software & Technology 2001, VRST 2001. Banff Centre, Canada, November 15-17, 2001.
24. Isa N. Engleberg, Working in Groups: Communication Principles and Strategies. My Communication Kit Series, page 140-141, 2006.
25. T. Balogh, P.T. Kovacs, Z. Megyesi, HoloVizio 3D display system, Proceedings of the First International Conference on Immersive Telecommunications, 2007
26. P. Lincoln, G. Welch, A. Nashel, A. Ilie, A. State, H. Fuchs, Animatronic Shader Lamps Avatars, 8th IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2009

27. R. Kaucic, B. Dalton, and A. Blake. “Real-time lip tracking for audio-visual speech recognition applications,” in Proc. European Conf. Computer Vision, Cambridge, UK, 1996, pp. 376-387.

28. Matthews, I., Cootes, T.F., Bangham, J.A., Cox, S., Harvey, R.: Extraction of visual features for lipreading. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2002

29. N. Eveno, A. Caplier, and P.Y. Coulon, “Automatic and Accurate Lip Tracking,” in IEEE Trans. on Circuits and Systems for Video Technology, Vol. 14, No 5, pp. 706-715, 2004.