Modeling Focus of Attention for Meeting Indexing

Rainer Stiefelhagen, Jie Yang, Alex Waibel
stiefel@ira.uka.de, yang+@cs.cmu.edu, ahw@cs.cmu.edu
Interactive Systems Laboratories
University of Karlsruhe — Germany, Carnegie Mellon University — USA

Abstract

Visual cues, such as gesturing, looking at each other or monitoring each others facial expressions, play an important role in meetings. Such information can be used for indexing of multimedia meeting recordings. In this paper, we present an approach to detect who is looking at whom during a meeting. Our proposal is to employ Hidden Markov Models to characterize participants' focus of attention by using gaze information as well as knowledge about the number and positions of people present in a meeting. The number and positions of the participants faces are detected in the field of view of a panoramic camera. We use neural networks to estimate the directions of participants' gaze from camera images. We discuss the implementation of the approach in detail including system architecture, data collection, and evaluation. The system has achieved an accuracy rate of up to 93 % in detecting focus of attention on test sequences taken from meetings. We have used focus of attention as an index in a multimedia meeting browser.

1 Introduction

Having meetings is one of the most common activities in business. However, it is impossible for people to attend all relevant meetings or to retain all the salient points raised in meetings they do attend. Meeting records are intended to overcome these problems and extend human memories. At the Interactive Systems Labs of Carnegie Mellon University, we are developing a multimedia meeting browser to transcribe and summarize meetings [15]. The objective of this project is to provide a multimedia meeting record without using constraining devices such as headsets, helmets, suits and buttons. The research issues include to identify: 1) who/what is the source of the message, 2) who or what is the target and object of the message (focus of attention), 3) what is the content of the message in the presence of jamming noise. The main components of the Meeting Browser are: a speech recognizer, a summarization module, a discourse component that attempts to identify the speech acts, a module for audio-visual identification of participants [17] and a module for tracking the participants' focus of attention.

In order to quickly retrieve information from such a multimedia meeting browser, we can use various indexing methods. It is well known that visual communication cues, such as gesturing, looking at each other or monitoring each others facial expressions, play an important role during face-to-face communication. Therefore, to fully understand an ongoing conversation, it is necessary to capture and analyze these visual cues in addition to spoken content. Once such visual cues can be tracked, they can be used to index and retrieve recorded meetings. Queries, such as "show me all parts of the meeting, where John was telling Mary something about the multimedia project" become possible. In addition, during playback of parts of a meeting, we could indicate at whom the speaker was looking.

In this paper we describe our approach to model and track the focus of attention of participants in a meeting. Objects which draw a person's attention can be external stimuli such as pictures, sounds, etc. or internal stimuli such as thoughts and attempts to retrieve information from memory [4]. Gaze is a good indicator of a person's attention on objects of an external nature. When humans pay attention to an (external) object, they usually orient themselves towards the object of interest so as to have it in the center of their visual field. Hence, the first step in determining a person's focus of attention is to track his/her gaze. To map the person's gaze onto the focussed object in the scene, a model of the scene and the interesting objects in it is furthermore needed. In the case of a meeting scenario, clearly the participants around the table are such likely targets of interest. Therefore, our approach to tracking at whom a participant is looking is the following: 1) detect all participants in the scene, 2) estimate each participants gaze and 3) map each estimated gaze to its likely targets using a probabilistic framework.

We propose to employ Hidden Markov Models to characterize attention focus of participants based on gaze information as well as knowledge about the number and positions of people present in a meeting. In our approach, the number and positions of participants' faces are detected within the viewing range of a panoramic camera and we use neural networks to estimate the participants' gazes from camera images.

Tracking a person's focus of attention is useful in several application areas: Intelligent supportive computer applications could use information about a user's focus of attention to get an understanding of the user's internal state, his goals and cognitive load and adjust their own responses to the user

To appear at ACM Multimedia'99, Orlando, Florida, Oct. 30th to Nov. 5th 1999

accordingly.

For multimodal human computer interaction, the user's focus of attention can be used to determine his/her message target. For example in interactive intelligent rooms or houses [7, 2], focus of attention could be used to determine whether the user is speaking a command to the refrigerator, his TV set, or whether he is talking to another person in the room. In other words, the user's attention focus can be used to guide the environment's "focus" to the right application and to prevent responses generated from applications that have not been addressed. During social interaction gaze serves for several functions which are not easily transmitted by auditory cues alone [1]. In computer mediated communication systems, such as virtual collaborative workspaces, detecting and conveying participants' gazes have several advantages: it can help the participants to determine who is talking or listening to whom, it can serve to establish joint attention during cooperative work and it can facilitate turn taking among participants [14, 5].

The remainder of the paper is organized as follows: In section 2, we introduce the idea of modeling a persons' focus of attention by integrating knowledge about likely targets in the room as well as observable gaze estimates of a person into a Hidden Markov Model (HMM) framework. To track a participant's gaze and obtain the necessary gaze observations for our attention model, we have trained neural nets to estimate head pan and tilt from facial images. Details about architecture, training and results of these nets are given in section 3. In section 4, we describe the use of a panoramic camera to locate and track participants around the table. In section 5 we evaluate the proposed attention model, discuss details of model initialization and present experimental results on video sequences that we recorded during some meetings. In section 6, we present an application of our model to the meeting browser. Information about the participants' focus of attention is tracked and is integrated as a component in the meeting browser. The meeting browser can then be used to index meeting transcriptions and summaries with visual cues. We summarize the paper in section 7.

2 Modeling Focus of Attention

The idea of this research is to track the participants' focus of attention in a meeting. Since a person's gaze direction is closely related to the person's attention, the first step is to track the person's gaze. However, attention does not necessarily coincide with gaze, since it is a perceptual variable, as opposed to a physical one (head or eye positioning). Our approach to modeling focus of attention attempts to model both, a person's head movements as well as the relative locations of probable targets of interest in a room. In a meeting, as depicted in Figure 1, clearly the participants around the table are such likely targets. Other likely targets can be: documents on the table, a whiteboard or slide projections on a wall, or people entering the room.

Therefore, our approach to determine all participants' focus of attention is the following:

1. Detect and track all participants around the table
2. Estimate each participants' gaze direction
3. Map the participants' observed gazes to their likely target (the other participants) using a probablistic framework

Hidden Markov Models can provide such an integrated framework for probabilistically interpreting observed signals

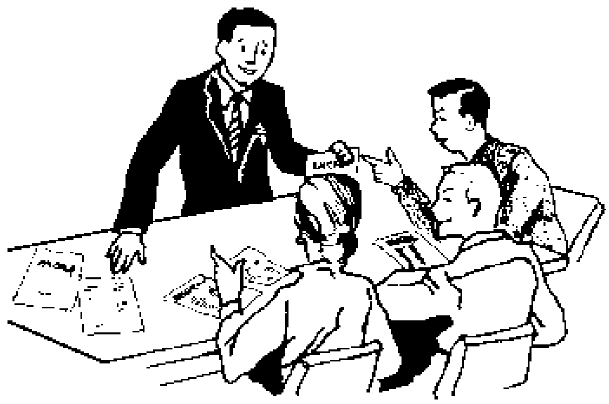

Figure 1: An example of interaction between people in a meeting

over time. In our model, looking at a certain target is modeled as being in a certain state of the HMM and the observed gaze estimates are considered as being probabilistic functions of the different states. Given this model and an observation sequence of gaze directions, it is then possible to find the most likely sequence of HMM states that produced the observations. By interpreting being in a certain state as looking at a certain target, it is now possible to estimate a person's focus of attention in each frame.

While a person's gaze is determined by the person's head orientation as well as his/her eye-gaze, we only consider head gaze as the main indicator of a person's gaze. The reason for doing this, is that we want to build a system with minimum intrusion. Without the use of head mounted cameras, infrared eye-trackers or other expensive equipment for each participant and with users that are allowed to move freely, it would be very difficult to track eye-gaze of all users. To obtain the gaze observations needed for our model, we have trained neural networks to estimate a person's head pose from facial images, which are automatically extracted from camera images using a color- and motion based face tracker.

To determine the number of HMM states necessary for each person's attention model, i.e. the number of other participants at the table, we use a face tracker to locate all faces in the field of view of a panoramic camera that is put on top of the conference table. The relative position of the found faces is later used to assign each of the HMM states to a specific participant of the meeting.

3 Estimating Head Pose Using Neural Nets

In this section we describe how we have designed and trained a neural network to estimate a person's head pan and tilt from facial images.

The main advantage of using neural networks to estimate head pose as compared to using a model based approach is its robustness: With model based approaches to head pose estimation [3, 13, 6], head pose is computed by finding correspondences between facial landmarks points (such as eyes, nostrils, lip corners) in the image and their respective locations in a head model. Therefore these approaches rely on tracking a minimum number of facial landmark points in the image correctly, which is a difficult task and is likely to fail. On the other hand, the neural network-based approach does not require tracking detailed facial features. Instead, the whole facial region is used for estimating the user's head pose.

In our approach we are using neural networks to estimate pan and tilt of a person's head, given automatically extracted and preprocessed facial images as input to the neural net. This approach is similar to the approach described by Schiele et. al. [10]. However, Schiele et. al.'s system estimated only head rotation in pan direction. In this research we use neural network to estimate head rotation in both pan and tilt directions. In addition, we have studied two different image preprocessing approaches. Rae et. al. [9] describe a user dependent neural network based system to estimate the pan and tilt of a person. In their approach, color segmentation, ellipse fitting, and Gabor-filtering on a segmented face are used for preprocessing. They reported an average accuracy of 9 degrees for pan and 7 degrees for tilt for one user with a user dependent system.

The work presented in this section extends our previously published work on neural net based head pose estimation [12] in the following ways: where we have only used training data that was collected in one room for our previous system, we have used data that was collected in two rooms and under significantly different lighting conditions here. Also we have changed the network architecture here. Where we have used separate nets with Gaussian output representation to estimate pan and tilt previously, we have now used one net to estimate both, pan and tilt. Only two output units for pan and tilt are used.

3.1 Data Collection

We collected training data from nineteen persons in two different rooms with different lighting conditions. During data collection, users had to wear a head band with a sensor of a Polhemus pose tracker attached to it. Using the pose tracker, the head pose with respect to a magnetic transmitter could be collected in real-time. A camera was positioned approximately 1.5 meters in front of the users head. The user was asked to randomly look around in the room and the images together with the pose sensor readings were recorded. Figure 2 shows two sample images of the same user taken under different lighting conditions during data collection.

Figure 2: Two images of the same person taken in two rooms during data collection

3.2 Preprocessing of Images

To locate and extract the faces from the collected images, we use a statistical skin color model [16]. The largest skin colored region in the input image is selected as the face.

We have investigated two different image preprocessing methods as input to the neural nets for pose estimation [12]: 1) Using normalized grayscale images of the user's face as input and 2) applying edge detection to the images before feeding them into the nets.

In the first preprocessing approach, histogram normalization is applied to the grayscale face images as a means towards normalizing against different lighting conditions. No additional feature extraction is performed. The normalized grayscale images are downsampled to a fixed size of 20x30 pixels and are then used as input to the nets.

In the second approach, a horizontal and a vertical edge operator plus thresholding is applied to the facial grayscale images. The resulting edge images are downsampled to 20x30 pixels and are both used as input to the neural nets.

Since we obtained the best results when combining the normalized histogram and the edge images as inputs to the neural nets [12], we are only presenting results using this combination of differently preprocessed images fed to the neural net here.

Figure 3 shows the corresponding preprocessed facial images of a user. From left to right, the normalized grayscale image, the horizontal and vertical edge images of a user's face are depicted.

Figure 3: Preprocessed images: normalized grayscale, horizontal edge and vertical edge image (from left to right)

3.3 Neural Net Architecture, Training and Results

We have trained one net to estimate both, pan and tilt of the head. We have used a multilayer perceptron architecture with two output units (for pan and tilt), one hidden layer with thirty units and an input retina of 20x90 units for the three input images of size 20x30 pixels. Output activations for pan and tilt were normalized to vary between zero and one. Training of the neural net was done using standard backpropagation.

3.3.1 Results with a Multi-User System

To train a multi-user neural network, we divided the data set of the nineteen users into a training set consisting of 11.500 images, a cross-evaluation set of size 1.500 images and a test set with a size of 1.500 images. After training, we achieved a mean error of 8.8 degrees for pan and 5.7 degrees for tilt on the test set.

3.3.2 User Independent Results

To determine how well the neural net based system can generalize to new users, we have also trained one net on seventeen users and evaluated it on the remaining two users, that have not been in the training set. Table 1 shows the results that we obtained for the two new users. On average we received an error of 11 degrees for pan and 10 degrees for tilt on the new users.

	E_{pan}	E_{tilt}
subject A	11.5	11.3
subject B	9.6	8.5
Average	10.6	9.9

Table 1: Person independent results (mean error in degrees) for two new users

3.3.3 Evaluating the Effect of Different Lighting Conditions

To accurately evaluate the effect of images taken under different lighting conditions, we trained and evaluated neural nets that were trained with images from one room only. Table 2 shows the results that we obtained using these "room-dependent" nets when testing on images from the same room versus testing with images from another room.

Training Data	Test Data	E_{pan}	E_{tilt}
Room 1	Room 1	8.0	5.1
Room 2	Room 2	9.2	5.3
Room 1	Room 2	21.4	18.2
Room 2	Room 1	20.1	18.7
Room 1,2	Room 1,2	8.8	5.7

Table 2: Results obtained when training and testing on images taken under different lighting conditions

It can be seen, that the accuracy of pose estimation dramatically decreases when testing the nets on images that were taken under different lighting conditions than during training. However, when using images from both rooms during training, the pose estimation results remain stable.

4 Detecting and Tracking All Participants Using a Panoramic Camera

In order to assign one HMM state to each participant at the table in our focus of attention model as described in section 2, it is necessary to determine the number and relative positions of participants present around the conference table.

We are using a panoramic camera with a 360 degree field of view that we put on top of the conference table to capture the whole scene around the table. Figure 5 shows a picture of the panoramic camera system that we are using. The camera is located in the top cylinder and is focusing on a parabolic mirror on the bottom plate. Through this mirror almost the entire hemisphere of the surrounding scene is visible. Figure 6 shows the view of a meeting scene as it is seen in the parabolic mirror and as it is captured with this camera. Since the topology of the mirror and the optical system are known, it is possible to compute rectified panoramic views of the scene as well as perspective views in different viewing directions. This can easily be done in real time. Figure 4 shows the rectified panoramic image (with faces marked) of the camera view depicted in Figure 6.

[1] Image courtesy of CycloVision Technologies, Inc.

Figure 5: The panoramic camera used to capture the scene[1]

Figure 6: Meeting scene as captured with the panoramic camera

4.1 Using Color and Motion for Face Detection

To detect and track faces in the panoramic camera view, a statistical skin color model consisting of a two-dimensional Gaussian distribution of normalized skin colors is used. The color distribution is initialized so as to find a variety of face colors and is gradually adapted to the faces actually found. The interested reader is referred to [16]. To detect faces, the input image is searched for pixels with skin colors. Connected regions of skin-colored pixels in the camera image are considered as possible faces.

Since humans rarely sit perfectly still for a long time, motion detection is used to reject outliers that might be caused due to noise in the image or skin-like objects in the background of a scene that are not faces or hands. Only regions with a response from the color-classifier and some motion during a period of time are considered as faces.

Using only this approach however, faces and hands are not yet distinguished sufficiently. h Therefore we are considering skin-colored regions as belonging to the same person if the projection of their centers onto the x-axis are close enough together. Among the candidate regions belonging to one person, we consider the uppermost skin-like region

Figure 4: Panoramic view of the scene around the conference table. Faces are automatically detected and tracked (marked with boxes).

to be the face and consider the lower skin-like region to be hands. Figure 4 shows a sample panoramic image with the four found faces marked with white boxes. Note that the hands present in the panoramic view are not considered to be faces (and therefore not marked here).

5 Experimental Evaluation of the Model

To evaluate our focus of attention model, we have recorded videos during several meetings.

During these meetings we have captured all participants with a panoramic camera as described in section 4. In addition, two cameras were used to capture images from two participants. Since we have not (yet) trained neural nets to estimate head pose from perspective images that can be generated from the panoramic view, the additional cameras are needed to obtain the facial images as input to our neural net based head pose estimation. Figure 7 shows some example images taken with the additional cameras during one of the meetings.

5.1 Initialization of the HMMs

To determine the number of states of each HMM, the number of participants of the meeting is automatically detected in the panoramic image as described in section 4. Since for each person we consider the other participants to be likely focus of attention targets, we assign each of the other participants to one state of the Hidden Markov Model.

We have parameterized the state dependent observation probabilities $B = b_i(\omega)$ for each state i, where $i \in \{Person_1, Person_2, \ldots, Person_n\}$, as two-dimensional Gaussian distributions with diagonal covariance matrices:

$$b_i(\omega) = \frac{1}{2\,\pi\,\sqrt{\sigma_{pan}\,\sigma_{tilt}}}\; e^{-\frac{1}{2}\left[\frac{(\omega_{pan}-\mu_{pan})^2}{\sigma^2_{pan}}+\frac{(\omega_{tilt}-\mu_{tilt})^2}{\sigma^2_{tilt}}\right]}$$

The observable symbols ω are the pose estimation results that we obtain using the neural net based head pose estimation as described in section 3, that is the angles for pan and tilt ω_{pan} and ω_{tilt}.

Using the relative positions of participants that we have found in the panoramic view, we could initialize the observation probability distributions of different states by the means of the Gaussians set to the expected viewing angle, when looking at a corresponding target. However, gaze is not only determined by head pose but also by the direction of eye gaze. People do not always completely turn their heads toward the person at which they are looking. Instead, they also use their eye gaze direction. In our meeting recordings we observed that some people turned their heads more than others, who relied more on eye movements instead and less head turning when looking at other people. Therefore, we are using an unsupervised learning approach to find the head pan of a user when he/she is looking at the other participants. Knowing that the user is likely to look at his participants during the meeting, we can find clusters in the gaze observations of this user. These gaze observations can be clustered to the number of classes corresponding to the known number of other participants. The found means of these classes can then be assigned to each participant based on his relative location at the table.

Table 3 shows the means of each of the three cluster that we found for each participant during a meeting. The cluster were obtained by hierarchically clustering the pan-observations of each participant. The means of these cluster were then used to initialize the HMM for that respective person.

	μ_1	μ_2	μ_3
Person A	-35.1	-7.1	20.9
Person B	-26.3	16.3	36.8
Person C	-26.4	-5.6	13.2
Person D	-19.9	-5.2	12.4

Table 3: Means of clusters found in head pan observations for four different users (in degrees)

The transition matrix $\mathbf{A} = (a_{ij})$ was initialized to have higher transition probabilities in order to remain in the same state ($a_{ii} = 0.5$) and to have uniformly distributed state transition probabilities for all other transitions. The initial state distribution was uniform.

5.2 Finding the Best Sequence

Let $O = \omega_1 \omega_2 \cdots \omega_T$ be the sequence of gaze direction observations $\omega_t = (\omega_{pan,t}, \omega_{tilt,t})$ as predicted by the neural nets. The probability of the observation sequence given the HMM is given by the sum over all possible state sequences q:

$$\begin{aligned} p(\mathbf{O}) &= \sum_q p(O, q) \\ &= \sum_q p(O|q)\; p(q) \end{aligned}$$

Figure 7: Sample sequence taken during a meeting

$$\approx \sum_q \prod_t p(\omega_t|q_t)\ p(q_t|q_{t-1})$$
$$= \sum_q \prod_t b_{q_t}(\omega)\ a_{q_t,q_{t-1}}.$$

To find the single best state sequence of foci of attention, $q = q_1 \dots q_n$ for a given observation sequence, we need to find

$$max_q(p(O, q)).$$

This can be efficiently computed by the Viterbi algorithm [8]. Thus, given the HMM and the observation sequence of gaze directions, we can efficiently find the sequence of foci of attention using the Viterbi algorithm.

To evaluate the performance of the proposed model, we compared the state-sequence given by the Viterbi-decoding to hand-made labels of where the person was looking at. The evaluated sequences contained 240 frames and lasted for two minutes each. Table 4 shows the results that we obtained on videos from six users. As compared to the hand-labels we obtained an average error of 24 % frames on the six test sequences.

Sequence	Error
A	26 %
B	21 %
C	30 %
D	11 %
E	22 %
F	32 %
Average	24 %

Table 4: Results of focus of attention labeling after Viterbi-decoding on six test sequences

5.3 Unsupervised Adaptation of Model Parameters

It is furthermore possible to adapt the model parameters $\lambda = (\mathbf{A}, \mathbf{B})$ of the HMM so as to maximize $p(O|\lambda)$. This can be done in the EM (Expectation-Maximization) framework by iteratively computing the most likely state sequence and adapting the model parameters as follows:

- means:

$$\hat{\mu}_{pan}(i) = E_i(\omega_{pan}) = \frac{\sum \phi_{i,t}\ \omega_{pan,t}}{\sum \phi_{i,t}}$$
$$\hat{\mu}_{tilt}(i) = E_i(\omega_{tilt}) = \frac{\sum \phi_{i,t}\ \omega_{tilt,t}}{\sum \phi_{i,t}}$$

$$\text{, where } \phi_{i,t} = \begin{cases} 1 & : \quad q_t = i \\ 0 & : \quad otherwise \end{cases}$$

- variances:

$$\sigma^2_{pan}(i) = E_i(\omega^2_{pan}) - (E_i(\omega_{pan}))^2$$
$$\sigma^2_{tilt}(i) = E_i(\omega^2_{tilt}) - (E_i(\omega_{tilt}))^2$$

- transition probabilities:

$$a_{i,j} = \frac{\text{number of transition from state } i \text{ to } j}{\sum_t \phi_{i,t}}$$

Using these formulas, we have automatically adapted the means and variances of the HMM states to the six test sequences. Table 5 shows the results that we obtained after adapting the parameters.

Seq.	λ fixed	λ adapted	error reduct.
A	26 %	16 %	31 %
B	21 %	15 %	29 %
C	30 %	30 %	-
D	11 %	7 %	36 %
E	22 %	19 %	14 %
F	32 %	32 %	-
Avg.	24 %	20 %	17 %

Table 5: Percentage of falsely labeled frames for six users with and without reestimation of means and variances

We can see that the average error we obtained after parameter adaptation is 20 % as compared to 24 % error without parameter adaptation. This corresponds to an error reduction of 17 %.

6 Integrating Focus of Attention Modeling into a Meeting Browser

We have integrated a component to track people's focus of attention into the "Meeting Browser" - a system to track and summarize meetings [15]. The Meeting Browser is a system designed to automatically review and search recordings of meetings. The browser is implemented in Java and includes video capture of individuals in the meeting, as pictured in Figure 8. The main components of the Meeting Browser are: 1) a speech recognizer, 2) a summarization module, 3) a discourse component that attempts to identify speech acts 4) a module for audio-visual identification of participants [17] and 5) a module for tracking the participants' focus of attention.

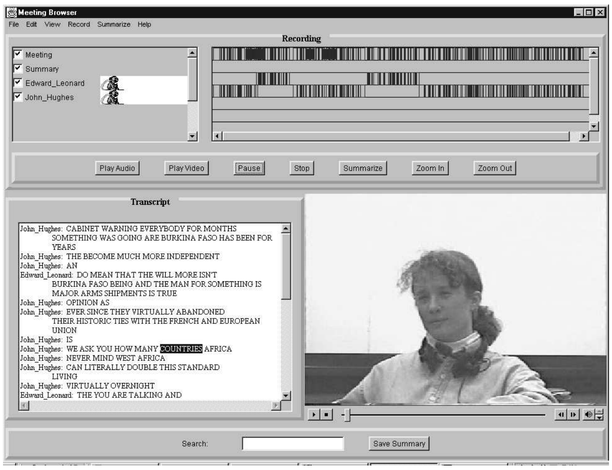

Figure 8: Meeting Browser with video capture

The Meeting Browser is part of a multimodal meeting room. The goal of this project is not only to provide a tool to record and transcribe spoken content of the meetings, but to also detect who participated in the meeting and who was talking when and to whom.

For the data acquisition in the meeting room, we used several microphones, a panoramic camera as described in section 4 and several cameras around the table to capture close-up views of the participants.

With the components described in this paper, it is possible to detect the number and positions of participants in a meeting as well as to track which person at the table each of the participants look at. Together with the components for person and speaker identification, which are described in detail in [17], it is furthermore possible to determine who these participants are and who the speaker of a certain utterance was (speaker ID). Given all these cues for indexing of the meetings, it is then possible to formulate queries such as: "show me all parts, where John was telling Mary something about the multimedia project". In addition, during playback of parts of the meeting, we could indicate at whom the speaker was looking during his speech. For example Figure 9 shows an example where the gaze tracking component detected and indicated that the person was looking at the participant to her left and at the one to her right respectively. Finally, we could even use this data to analyze meetings in many ways. One such usage could be to calculate how much of the time someone was speaking or how much of the time person X was addressing person Y.

7 Conclusion

In this paper we have addressed the problem of tracking focus of attention of the participants in a meeting. We have described how our system automatically locates and tracks the participants in the view of a panoramic camera. We have proposed the use of a HMM framework to detect focus of attention from a trajectory of gaze observations and have evaluated the proposed approach on several video sequences recorded during meetings.

For gaze tracking, we have employed neural networks to estimate head pose from facial images. We have obtained mean error as small as 9 degrees for pan and 6 degrees for tilt with a multi-user neural network that was tested on nineteen users.

Figure 9: Examples in which the attention model indicates that the person is looking to the participant to the left and right, respectively

We have integrated a module to track focus of attention into a meeting browser - a system which automatically produces transcriptions and summaries of meetings. The visual cues given by the attention model can be used for indexing the transcriptions and summaries.

Other application areas of tracking focus of attention include: multimodal human computer interfaces, computer supported collaborative work, and interactive intelligent environments.

Acknowledgements

We would like to thank the many colleagues in Interactive Systems Lab for participating in experiments during data collection. Thanks to Prof. Dillmanns group at the University of Karlsruhe for letting us use their Polhemus tracker several times for training data collection. Also thanks to Thomas Kemp and Frances Ning for proof-reading manuscripts of this paper. The neural networks used in this research were trained using the Stuttgart Neural Net Simulator tool [11]. This research is sponsored in part by the Defense Advanced Research Projects Agency under the Genoa project, subcontracted through the ISX Corporation under Contract No. P097047 and by the Department of Defense (project Clarity). Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA or any other party.

References

[1] M. Argyle. *Social Interaction*. Methuen, London, 1969.

[2] M. H. Coen. Design principles for intelligent environments. In *Intelligent Environments, Papers from the 1998 AAAI Spring Symposium*, number Technical Report SS-98-92, pages 37–43. AAAI, AAAI Press, 1998.

[3] A. H. Gee and R. Cipolla. Non-intrusive gaze tracking for human-computer interaction. In *Proc. Mechatronics and Machine Vision in Practise*, pages 112–117, 1994.

[4] D. Gopher. *The Blackwell dictionary of Cognitive Psychology*, chapter Attention, pages 23–28. Basil Blackwell Inc., 1990.

[5] H. Ishii and M. Kobayashi. Clearboard: A seamless medium for shared drawing and conversation with eye contact. In *Proceedings of ACM CHI'92 Conference on Human Factors in Computing Systems*, pages 525–532. ACM, 1992.

[6] T. Jebara and A. Pentland. Parametrized structure from motion for 3d adaptive feedback tracking of faces. In *Proceedings of Computer Vision and Pattern Recognition*, 1997.

[7] M. Mozer. The neural network house: An environment that adapts to its inhibitants. In *Intelligent Environments, Papers from the 1998 AAAI Spring Symposium*, number Technical Report SS-98-92, pages 110–114. AAAI, AAAI Press, 1998.

[8] L. R. Rabiner. *Readings in Speech Recognition*, chapter A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, pages 267–295. Morgan Kaufmann, 1989.

[9] R. Rae and H. J. Ritter. Recognition of human head orientation based on artificial neural networks. *IEEE Transactions on neural networks*, 9(2):257–265, March 1998.

[10] B. Schiele and A. Waibel. Gaze tracking based on face-color. In *International Workshop on Automatic Face- and Gesture-Recognition*, pages 344–348, 1995.

[11] SNNS. The Stuttgart Neural Net Simulator. http://www.informatik.uni-stuttgart.de/ipvr/bv/projekte/snns/snns.html.

[12] R. Stiefelhagen, M. Finke, J. Yang, and A. Waibel. From gaze to focus of attention. In M. Turk, editor, *Proceedings of Workshop on Perceptual User Interfaces: PUI 98*, pages 25–30, San Francisco, CA, November, 4-6th 1998.

[13] R. Stiefelhagen, J. Yang, and A. Waibel. A model-based gaze tracking system. In *Proceedings of IEEE International Joint Symposia on Intelligence and Systems*, pages 304 – 310, 1996.

[14] R. Vertegaal. The gaze groupware system: Mediating joint attention in multiparty communication and collaboration. In *ACM CHI'99 Conference on Human Factors in Computing Systems*, Pittsburgh, PA, 1999. ACM.

[15] A. Waibel, M. Bett, M. Finke, and R. Stiefelhagen. Meeting browser: Tracking and summarizing meetings. In D. E. M. Penrose, editor, *Proceedings of the Broadcast News Transcription and Understanding Workshop*, pages 281–286, Lansdowne, Virginia, February. 8-11 1998. DARPA, Morgan Kaufmann.

[16] J. Yang and A. Waibel. A real-time face tracker. In *Proceedings of WACV*, pages 142–147, 1996.

[17] J. Yang, X. Zhu, R. Gross, J. Kominek, Y. Pan, and A. Waibel. Multimodal people id for a multimedia meeting browser (?). In *Proceedings of ACM Multimedia '99 (to appear)*. ACM, 1999.