Segmentation and Classification of Meetings using Multiple Information Streams

Paul E. Rybski, Satanjeev Banerjee, Fernando de la Torre,
Carlos Vallespi, Alexander I. Rudnicky, Manuela Veloso
Carnegie Mellon University
5000 Forbes Ave.,Pittsburgh, PA 15213
Email: {prybski,banerjee+,ftorre,cvalles,air,veloso}@cs.cmu.edu

ABSTRACT

We present a meeting recorder infrastructure used to record and annotate events that occur in meetings. Multiple data streams are recorded and analyzed in order to infer a higher-level state of the group's activities. We describe the hardware and software systems used to capture people's activities as well as the methods used to characterize them.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

General Terms

Human Factors

Keywords

Multi-modal interfaces, Meeting understanding

1. INTRODUCTION

We are engaged in the design and development of an agent to assist users in everyday office-related tasks. In particular, we are focusing on conversational agents that can participate, in a natural fashion, in multi–participant interactions, such as meetings. In order to address this challenge, we are developing a multi-modal meeting event recording system that attempts to automatically detect the state of the meeting and the roles of the different meeting participants.

2. MULTI-MODAL DATA COLLECTION

Our meeting observation architecture [1] treats each information stream as a sequence of *events*, each with a start and an end time. Events may be instantaneous (such as key presses on a keyboard) in which case the times coincide, or of a finite amount of time long (such as a spoken utterance). To synchronize the information streams recorded by the various sensors during a meeting, the architecture uses the Network Time Protocol (NTP) to time stamp each event. We have currently implemented recording clients that follow the above design for the following information streams.

Figure 1: The instrumented meeting recording environment. Participants wear head-mounted microphones and a CAMEO system is in the center of the table.

2.1 Close Talking Speech

Participants record their speech by wearing head–mounted close–talking microphones that are connected to their individual laptops. All the input sound is broken up into 5–second events, each of which is time stamped using the network time protocol (NTP), stored on the laptop, and later transfered to a central data storage. During recording, a concurrently running process detects the start and end of speech; this information can be used later on to feed the appropriate sound snippets into an automatic speech recognizer.

2.2 Typed Notes

Participants are provided with a GUI interface in which to type notes during the meeting. Every time the user presses the Enter key, a snap–shot is taken of the current note–taking area, constituting an "event".

2.3 Whiteboard Pen–Strokes

We have instrumented the whiteboard with a Mimio [1] device that streams the x–y pen coordinates to a desktop computer. All captured coordinates between pen–down and pen–up constitute a single event, which is processed as above.

2.4 Slide Presentations

[1] www.mimio.com

ICMI'04, October 13–15, 2004, State College, Pennsylvania, USA.
ACM 1-58113-954-3/04/0010.

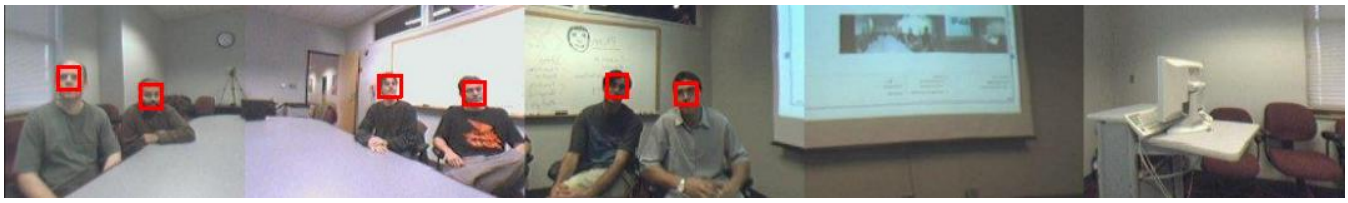

Figure 2: People identified by the CAMEO system's face detector.

We have implemented the PowerPoint Scraper that uses the Microsoft PowerPoint API to detect PowerPoint slide show events like "next slide", "previous slide", etc, and can also capture the contents of the slide on the screen. We have defined an event to occur whenever there is a slide change, and the contents of the event are the contents of the new slide.

2.5 CAMEO Vision System

CAMEO (the Camera Assisted Meeting Event Observer [4]) is an omni-directional camera system consisting of four or five firewire cameras (CAMEO supports both configurations) mounted in a circle, as shown in Figure 3. The individual data streams coming from each of the cameras are merged into a single panoramic image of the world. The cameras are connected to a Small Form-Factor 3.0GHz Pentium 4 PC that captures the video data and does the image processing.

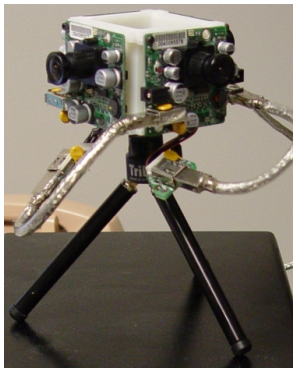

Figure 3: CAMEO is an omnidirectional camera system used to locate people, recognize them, and classify their actions.

Faces are detected using a parts–based method for classification of the image into "face" and "non–face" regions [5]. This method explicitly models and estimates the joint statistics of local appearance and position on the face and the statistics of local appearance in the visual world. Once located, the positions of people's faces are tracked using a combination of template matching and color histograms. This allows the person to be tracked from frame to frame regardless of whether their face is directly visible to CAMEO. Finally, CAMEO is capable of identifying people whose faces it has been trained to recognize. The recognition system uses a new technique, non-linear oriented discriminant analysis, which allows fast recognition and outperform classical linear methods like principal components analysis or linear discriminant analysis.

People's activities are recognized by a hidden Markov model-based classifier. The hidden states represent a finite state machine model of a person's behavior. States for an individual person's actions include: stand, standing, sit, sitting, fidgeting, and walking. The conditional probability distribution for the hidden nodes are learned from collecting statistics from CAMEO's observations. Given a sequence of real-valued state observations from the meeting, a real-time implementation of the Viterbi algorithm [3] is used to infer the state of each person at each timestep.

All of CAMEO's recognition capabilities can be executed in real-time on a live video stream (at 3-5 frames/second), or on an MPEG movie it has previously recorded.

3. INFERRING MEETING STATE

In order to infer the state of the meeting from the low level recorded sensor information, we construct classifiers both from *a priori* models as well as from previously recorded meeting data. The meeting state is tracked by comparing the activities of the people against known behaviors ordered as a first-order (fully observable) Markov model which takes into account a minimum duration for a state transition. The specific topologies of the allowable state can be hand-coded as well as learned from recorded meeting data.

We have also taken a machine learning approach where we first hand–annotate a corpus of recorded meeting data with meeting state and participant role labels, and then train a decision tree classifier from this data. The resulting tree takes as input low–level sensor information over a window of meeting time and outputs a probability distribution over the possible meeting states and participant roles for the current time instant [2].

4. REFERENCES

[1] S. Banerjee, J. Cohen, T. Quisel, A. Chan, Y. Patodia, Z. Al-Bawab, R. Zhang, A. Black, R. Stern, R. Rosenfeld, A. Rudnicky, P. E. Rybski, and M. Veloso. Creating multi-modal, user-centric records of meetings with the carnegie mellon meeting recorder architecture. In *Proceedings of the ICASSP 2004 Meeting Recognition Workshop*, Montreal, Canada, May 2004.

[2] S. Banerjee and A. I. Rudnicky. Using simple speech–based features to detect the state of a meeting and the roles of the meeting participants. In *Proceedings of the 8th International Conference on Spoken Language Processing (Interspeech 2004 – ICSLP)*, Jeju Island, Korea, 2004.

[3] L. R. Rabiner. A tutorial on Hidden Markov Models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[4] P. E. Rybski, F. de la Torre, R. Patil, C. Vallespi, M. M. Veloso, and B. Browning. Cameo: The camera assisted meeting event observer. In *International Conference on Robotics and Automation*, New Orleans, April 2004.

[5] H. Schneiderman. Feature-centric evaluation for cascaded object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004.