Multimodal Meeting Tracker

Michael Bett, Ralph Gross, Hua Yu, Xiaojin Zhu, Yue Pan, Jie Yang, and Alex Waibel
{mbett, rgross, zhuxj, ypan, yang+, waibel}@cs.cmu.edu

Interactive Systems Laboratories
Carnegie Mellon University
Pittsburgh, PA 15213, USA

Abstract

Face-to-face meetings usually encompass several modalities including speech, gesture, handwriting, and person identification. Recognition and integration of each of these modalities is important to create an accurate record of a meeting. However, each of these modalities presents recognition difficulties. Speech recognition must be speaker and domain independent, have low word error rates, and be close to real time to be useful. Gesture and handwriting recognition must be writer independent and support a wide variety of writing styles. Person identification has difficulty with segmentation in a crowded room. Furthermore, in order to produce the record automatically, we have to solve the assignment problem (who is saying what), which involves people identification and speech recognition. We follow a multimodal approach for people identification to increase the robustness (with the modules: color appearance id, face id and speaker id). This paper will examine a meeting room system under development at Carnegie Mellon University that enables us to track, capture and integrate the important aspects of a meeting from people identification to meeting transcription. Once a multimedia meeting record is created, it can be archived for later retrieval. This paper will review our meeting browser that we have developed which facilitates tracking and reviewing meetings.

1. Introduction

Meetings play a critical role in the everyday life of organizations. In order to retain the salient points for later reference, meeting minutes are usually taken. The process of manual note taking bears a number of problems. It is time consuming and requires the undivided attention of the minute taker. Even under the best of circumstances, the resulting transcript is often incomplete; limited to a single level of detail or compression. Minutes lack drill-down capability, as well as the ability to explore a topic of interest in detail. Minutes fail to capture the nuances of a meeting while reflecting the bias of the minute taker. These inaccuracies often require further review, clarification and discussion about the original meeting.

In addition, there are several questions about meetings that we have that minutes are not suited to answer. We want to know who said what and to whom? Were they angry? Were they sad? Was this a question, an opinion or a statement? Who was paying attention? This information is encoded in various verbal and visual cues delivered by speech, facial expressions, gestures or body language. More often than not, these details are not recorded as part of the meeting minutes. Ideally, we would like to review and explore at our discretion a video, audio or text transcript of the meeting. The manual production of such a multimedia meeting record is highly labor intensive, which makes it unsuitable for constant deployment.

An automated meeting recorder has to solve what we call the “assignment problem”. That is, at any given time the system must know who said what in order to correctly assign utterances to the respective speakers. A straightforward approach to this problem is to equip each meeting participant with a lapel microphone. This simple solution has several drawbacks. First there is the annoyance of preparation: prior to every meeting each participant has to be wired up and the names have to be linked to the different audio sources. Second, unless the microphones are expensive wireless devices, people are tethered down to their seats, unable to move about the room, e.g. to the whiteboard.

An ideal setup would be completely non-intrusive, so that people can just "walk in and talk". In order to meet this requirement the meeting recorder must continuously track and identify the participants in a room. If people are given full freedom of movement then any identification technique will falter in certain situations. For example face recognition will fail if the persons head is turned away from the camera. Equally a speaker identification module will not work well if the speaker is too far away from the nearest microphone. In order to overcome the limitations of single modal recognizers, we explore a multimodal approach for people identification in the context of a multimedia meeting recorder and browser. The system under development at the Interactive Systems Labs at Carnegie Mellon University automatically tracks meetings held in a specially equipped conference room. Furthermore it offers a convenient interface for meeting archiving, retrieval and reviewing.

The paper is organized as follows. In Section 2 we give an overview of the system and introduce the various components. Sections 3 to 5 describe the major parts of the system, speech recognizer, multimodal people identification, and the meeting browser in detail. Finally, Section 6 offers concluding remarks and a preview of future work.

2. System Overview

Figure 1 shows the multimodal meeting room that we are developing at Carnegie Mellon University's Interactive Systems Laboratories. It is comprised of three separate components: a multimodal people identifier, a speech recognizer, and a meeting browser. The meeting room requires a minimal amount of manual input. The system is able to automatically identify up to six distinct speakers in a meeting and automatically creates a transcript for read or conversational speech. Eventually the system will identify when a meeting begins and automatically start creating a complete meeting record. Once a meeting record is complete, the meeting browser allows us to archive and review previously recorded meetings

Figure 1 Multimodal meeting room in use.

In designing the system, we are striving to make:

- The interface as natural as possible
- Use of multiple modalities (speech, handwriting, and vision) when appropriate.
- All tasks automated as much as possible in order to minimize the burden on the user.

In the existing system architecture as shown in Figure 2, audio and video streams feed into the multimodal people id system. The identification is sent to both our Janus speech recognizer and to the

meeting browser. Hypotheses flow from Janus to the Meeting Browser and appear in a transcript window. The transcripts are then summarized by the summary server or archived via the dialogue server.

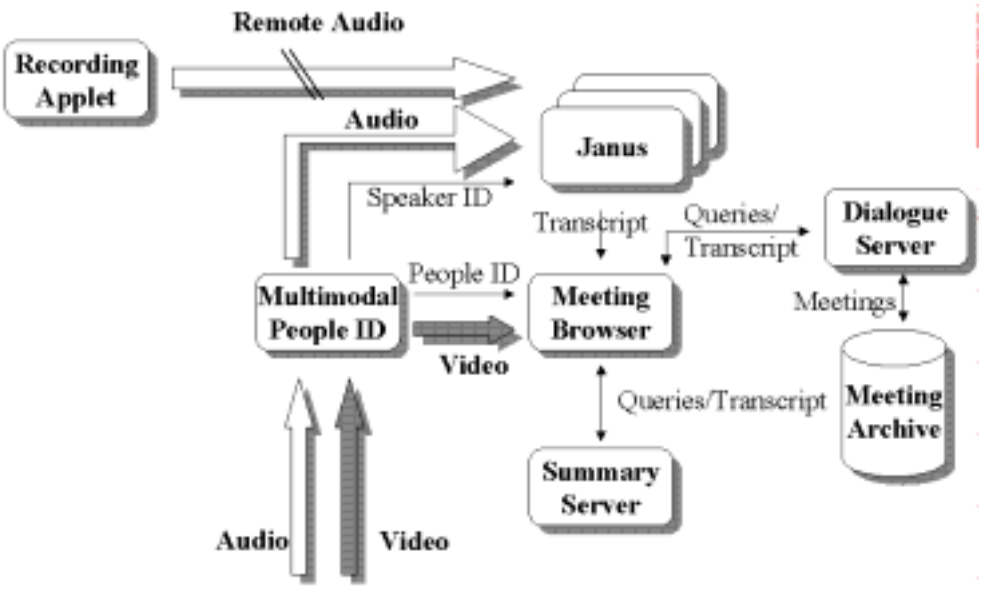


Figure 2: System overview of the multimodal meeting room

The speech recognizer used in the multimodal meeting room is based on the Janus Switchboard recognizer trained for the 1997 NIST Hub-5E evaluation. Janus is gender independent, vocal tract length normalized, large vocabulary recognizer that features dynamic speaking mode, adaptive acoustic and pronunciation models [2]. This allows for robust recognition of conversational speech as observed in human to human dialogs. The recognizer is described in [Waibel98]. We have extended the system to support multiple recognizers simultaneously in order to improve on the speed of the speech recognition.

The people identification module continuously tracks and identifies meeting participants. We currently combine the outputs of three subsystems, namely speaker identification, sound source position estimation, and color appearance identification in a multimodal fusion framework. Using the results we are able to answer the question of "Who said what?" during a meeting. Our preliminary experiments show that the performance of the combined multimodal system is superior to the performance of the individual recognizers

The meeting browser interface records meetings and displays meeting transcripts, time-aligned to the corresponding audio and video files. Included in the meeting transcripts are discourse features and emotions. The user can select all or a portion of these files for playback; text highlighting occurs in sync with the sound and video playback. As software design, the meeting browser is built around information streams. Transcribed meeting text is just one such stream; the interface can accept streams from virtually any source that produces text output. These streams are fully editable and searchable, allowing humans to annotate and correct recognition output as well as adding new informative streams manually. Once a meeting is complete, the meeting room automatically archives the meeting for future use. Users are able to query this archive to create audio, video, and text dialogue summaries of the meetings which can then be mailed to other individuals for playback and review.

The next three sections describe each of these modules in more detail.

3. Meeting Room Speech Recognition

Meeting recognition is a challenging large vocabulary conversational speech recognition task parallel to Hub5 (Switchboard) [Hub96] and Hub4 (Broadcast News) [Hub95]. The difficulty mostly comes from the highly conversational style of meetings, and a lack of training data. Since we are dealing with uninterrupted continuous recording with multiple speakers (possibly using multiple microphones), our task requires three steps. First, we carefully partition the data into homogeneous segments and assign each segment a "speaker" label. Second, we perform a first pass recognition which generates both a hypothesis and a confidence score. Finally, we do some unsupervised adaptation, and re-decode the utterances with the adapted model. ([Yu98], [Yu99])

3.1 Recognizer

Unlike many typical speech recognition tasks, there is not enough data available to train a domain-specific recognizer for the meeting recognition task. We experimented with several systems that we developed at the Interactive Systems Laboratories for different tasks. It is interesting to note how each system, tuned for maximal performance on its respective task, compares with each other on the meeting data. A summary of our results is presented in Table 1.

Show type	WER (1st pass)	WER (after adaptation)
Newshour	26.9%	26.3%
Crossfire	36.0%	34.6%

Table 1. Word error rates (WER) in percent on the group meeting data (internal group meeting recorded with lapel microphones)

Each of the systems was built upon the Janus Recognition Toolkit (JRTk), which is summarized in [Finke97]. Incorporated into our continuous HMM (Hidden Markov Model) system are techniques like linear discriminant analysis (LDA) for feature space dimension reduction, vocal tract length normalization (VTLN) for speaker normalization, cepstral mean normalization (CMN) for channel normalization, and wide-context phone modeling (Polyphone modeling). See [Rabiner93] for a technical description of each of these.

Recently, we are leveraging the large amount of data in Broadcast News (BN) domain to build a robust BN recognizer. BN data includes a wide range of background conditions (clean/noise/music), planned / spontaneous speech, field speech /telephone interview, etc. While our results are very promising, we still see a diverse picture on different types of data: for the 1998 BN testset, WER on clean planned speech was 13.4% and 25.5% for spontaneous speech (1st pass number, unadapted). Results vary widely for conditions such as background noise.

The BN system is also used to recognize some discussion-style TV news shows. The data is recorded directly from a TV set. Our observation is that Newshour data is fairly well behaved while Crossfire, as its name suggests, involved more heated discussions, crosstalk, and shorter turns.

3.2 Partitioning Strategy

Partitioning means both segmentation and classification. The idea is to make each segment long enough so that it only contains speech from a single speaker and under a single environment condition. Both over-segmenting and under-segmenting can cause difficulties. Over-segmenting produces sentence fragments,

while the recognizer is always expecting a full sentence; under-segmenting yields segments with multiple speakers, invalidating the single-speaker per utterance assumption of the recognizer.

We employ the following techniques which require a single speaker per utterance:

- VTLN (Vocal Tract Length Normalization): we estimate a single warping factor for each speaker, and train the system in a gender/speaker independent fashion
- Speaker adaptation: we pool all data from a single speaker together to adapt the acoustic model

If more than one speaker appears in an utterance, the estimated warping factor would be incorrect, and the adaptation would likely be suboptimal. Since we also do cepstral mean normalization, having the same background condition in one segment ensures proper estimate of channel noise.

Following [Gauvain98], we implemented the LIMSI-style partitioning scheme. We first classify incoming data into speech, music, and silence categories, and discard the non-speech data. Then we do an initial segmentation, with the parameters set to over-generate segments. Assuming each segment is a cluster on its own, we estimate a Gaussian mixture model for each cluster. Next, we iteratively (viterbi) reestimate and cluster these mixture models, until the likelihood penalized by number of clusters and number of segments no longer increases. The result is a segmentation with "speaker" labeling.

Unlike its ad hoc counterparts, the LIMSI approach is quite elegant in that it uses a couple of global parameters to control the process. Each of them has a clear interpretation. This partitioning scheme works well for the Newshour data (audio recorded from television news discussion shows such as Crossfire and Newshour).

4. Multimodal People Identification

The purpose of the people identification module is to continuously track and identify meeting participants within a room. In order to increase the robustness and efficiency of the identification process we have taken a multimodal approach and integrated a number of recognizers that use audio and video information. As shown in Figure 3, the system is comprised of five components: people segmentation, color appearance ID, speaker ID, face ID and multimodal information fusion. The face identification module is not currently incorporated in the system. Figure 4 shows a screenshot of the system output. The following paragraphs describe each module in more detail (see [Yang99] for a technical description).

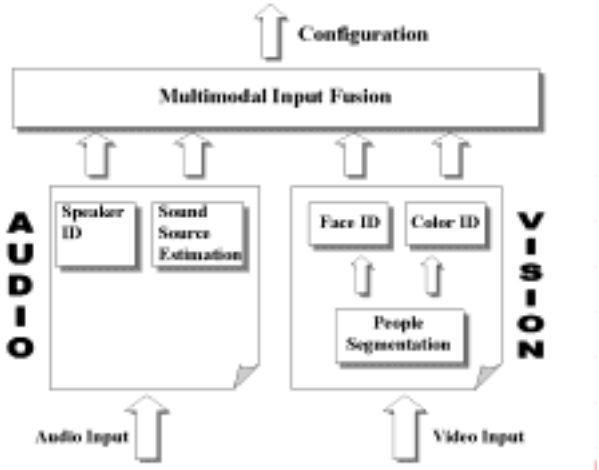


Figure 3: Overview over the multimodal people identification

Figure 4: Example of people identification running in the meeting room

4.1 People Segmentation

The ability to identify an object in a given image or image sequence requires the availability of an internal representation of said object. Assuming that such a model is given, it could be utilized to locate and identify objects in one unified step. Unfortunately the search space the recognizer would have to tackle in each run is too large to meet the real time requirements of an interactive system. We therefore use a motion-based preprocessing step to segment people from the background before we try to identify them. Our approach uses four different stages, namely background subtraction, noise removal, region growing, and background update.

In its simplest instantiation background subtraction consists of computing the difference between an incoming image and a previously taken background image. We use a more robust but slower variant in which we build a model for each pixel of the background image, classifying foreground pixels by their model distance to the background. The use of point-wise classification can lead to spurious results in two ways. First, there may be holes and eroded outlines due to foreground objects having the same color as the background. Second, small regions may arise caused by noise or object fragmentation. The latter is removed by grouping pixels together into regions and discarding regions with a small pixel count (e.g. 10 – 20). To compensate for eroded outlines noise removal is followed by a stage of region growing, where pixels are added based on their compliance with an object model.

The quality of the segmentation is crucially linked to the accuracy of the background representation. Given a projected runtime on the order of days and a variable environment where items like chairs and tables are likely to be moved, the system must keep its model of the background current. We use a temporal recursive filter, which gradually updates the pixels of the background that are not covered by a region classified as "alive." Liveness is a measure based on motion and size of regions in the segmented image.

4.2 Color Appearance Identification

Based on the segmentation derived by the people segmentation module, we create models for the different meeting participants using color histograms. As noted by other researchers, [Swain91] color histograms provide a stable object representation, which is largely unaffected by occlusions or changes in view. A major obstacle in the use of color for object identification is the fact that colors change with illumination. In order to reduce this sensitivity of the color models we use a perceptually motivated color encoding scheme, the so-called tint-saturation or t-s color space [Terrillon 98]. For each segmented person, we collect (t, s) histogram counts with an M-bin histogram. Based on these counts, we compute a smoothed probability distribution function (pdf). For the t-s space, the pdf is computed as follows:

$$P(t,s) = \begin{cases} \dfrac{count(t,s) - \varepsilon}{\sum_{t,s} count(t,s)}, & \text{count(t,s)} > 0 \\ \dfrac{n \cdot \varepsilon}{(M-n)\sum_{t,s} count(t,s)}, & \text{count(t,s)} = 0 \end{cases}$$

An absolute discounting smoothing scheme is used here to avoid zero probability in the pdf. ε is a small discounting value, e.g., ε = 0.001. *n* is the number of non-empty bins and *M* is the total number of bins. Intuitively we discount histogram bins with non-zero counts by ε, and evenly distribute the discounted probability mass to bins with zero count.

For combination with the other classifiers in our multimodal fusion framework we would like to derive the probability $P(D \mid M_i)$ for an input image region *D* and color appearance models M_i with i = 1, .., k. If

we assume the color histogram pdf's to be Generative Models, i.e. each pixel of the input image is independently sampled from the pdf, then the probability of an input image D being generated by the model can be calculated using the Kullback-Leibler distance between the image pdf and the model pdf, which is defined as:

$$D(P_D \| P_{M_i}) = \sum_{x \in S} P_D(x) \log \frac{P_D(x)}{P_{M_i}(x)}$$

[Bishop95]. Here, *S* stands for all histogram bins in the t-s color space.

We evaluated the color appearance identification on a separate task in which people were walking down a hallway. The recognition rates for different histogram sizes are shown in Table 2. These results were obtained using 16 distinct models tested on approximately 5000 images.

Histogram Size	25x 25	50x 50	75x 75	100x 100	125x 125	150x 150	175x 175	200x 200
Recognition accuracy	63.2	62. 3	67.8	65.8	63.4	59.1	56.4	53.3

Table 2: Recognition rates of the color appearance identification for different histogram sizes

4.3 Speaker Identification and Sound Source Position

The speaker ID module has to solve the problem of finding out which meeting participant is speaking at any given time, independent of what they are saying. This can be seen as a text-independent close-set speaker identification task. We consider both convolution and additive noise as consistent, except for occasional events – phone ringing, door clapping etc. The limited training and test sets are collected in the same noise environment [Bimbot97]. Our experiments show that if training and testing is done on the same noise conditions, the performance is comparable with the performance achieved on clean speech. The major challenge in this task is how to achieve high performance in real-time with a relatively small amount of training data.

Input speech first goes through a segmentation stage. The module roughly detects a possible acoustic event (utterance) and splits the continuous audio data into shorter segments. We use a simple approach based on energy and zero-crossing rate. The speech spectrum reflects a person's vocal tract structure and is used both in speech recognition and speaker identification. We use Mel Frequency Coefficients (MFCs) as feature vectors by applying Mel-scaled filter banks on the FFT spectrum [Rabiner93]. The sampling rate of the speech signal is 16KHz with high-pass pre-emphasis. The frame size is set to 32 ms with a frame shift of 16 ms. The training of the speaker models is done offline. A few utterances of each speaker (roughly 30 seconds) are used to build a Gaussian Mixture Model (GMM). We estimate the parameters of the models using the expectation-maximization (EM) algorithm. Based on an evaluation of systems using between 8 and 32 Gaussians we choose 16 as this configuration achieved the best performance. The results are shown in Table 3 below.

	Test length	
Recording	3 sec.	6 sec.
Clear	97.8%	100.00%
Noisy	96.6%	100.00%

Table 3. Identification performance on 30 speakers

In order to combine audio and visual information we need an estimation of the sound source position. In our initial system setup this estimation is based on a model of the speech energy pair (e_1, e_2) obtained from two microphones. As for the speaker identification module we use Gaussian Mixture Models for this task. The energy feature is susceptible to the influence of environment noise and it is incapable of distinguishing some symmetric positions. Therefore, we plan to improve the estimation by employing a microphone array.

4. 4 Face Identification

While people identification based on color appearance works reasonably well in most situations, it fails when meeting participants are dressed similarly. To overcome this problem we are developing face identification as part of the system. Just as for all components described so far, the meeting room scenario proves to be a challenging task for a face recognition module. A typical setting includes multiple faces in multiple poses at different sizes in a complex environment. As the subjects are communicating with each other we can also expect the full range of facial expressions to be visible.

Similar to the color appearance identification module described earlier, a face recognizer has to solve the problems of locating and identifying faces in an input image. We use a skin-color based face tracker developed in our lab to locate and extract faces out of the people regions provided by the people segmentation [Yang96]. A face image, if interpreted as a vector, defines a point in a high dimensional space. Different face images share a number of similarities with each other, so that the points representing these images are not randomly distributed in the image space. They all fall into a lower dimensional subspace. The key idea of the recognition process is to map the face images into an appropriately chosen subspace and perform classification by distance computation. If we restrict ourselves to a linear dimensionality reduction, the optimal solution is provided by the principal component analysis [Bishop95]. The basis of the lower dimensional "face space" is formed by the eigenvectors of the covariance matrix of the set of training images corresponding to the largest eigenvalues. In the context of face recognition these eigenvectors are called "eigenfaces". We evaluated the classic eigenface approach as described in [Turk91] and an extension developed in our lab called dynamic space warping (DSW). Here, instead of transforming a face image into one point in the eigenspace, we break down a face image into sub-images using a moving window. The face image is therefore transformed into a sequence of eigenpoints. During the recognition process, the template set of points is compared to the unknown set of points in a procedure similar to dynamic time warping (DTW) used in speech recognition [Rabiner93].

We have tested the proposed approach on a limited database. Table 4 shows results from two different test sets. The first set of data is smaller but with some background. The second set of data contains only face images. The results indicate that the DSW approach performs significantly better than the original eigenface method, especially when the face segmentation is not perfect. Work is under way to integrate the face recognizer into the people identification system.

# of people	DSW	Eigenface
14 (with background)	100%	78.5%
40 (without background)	97.5%	87.5%

Table 4. Face recognition using DSW (dynamic space warping) and eigenface approach

4.5 Multimodal Input Fusion

During input fusion we try to find the most probable configuration of people locations, identities in the room and assignment of a speaker. As explained earlier we simplify the task by using a people segmentation module as preprocessing step to color appearance identification and face identification. This avoids a search over all possible people locations in the input image. Assuming conditional independence of the input signals from the color appearance identification, speaker identification and

sound source position estimation, we can directly combine the probabilities estimated by these modules (see [Yang99] for details).

To demonstrate the feasibility of the framework, we set up a simple meeting. Two video cameras were used to take portions of a scene. The images were merged to create a wide-angle input video image. Two microphones recorded the conversation. The microphones also provided rough directional information of each utterance by the difference in input energies. In our experiment, we collected 2990 audio and video inputs. For both inputs, we found the optimal configuration with information fusion. We also computed the optimal configuration without fusion, i.e. using the models individually: A preliminary result is given in Table 5. We consider a configuration to be erroneous if any of the components failed to give the correct result. In this experiment, the configuration error rate drops by 2% absolute after information fusion. Therefore, we believe applying fusion to people identification is a promising approach.

	Number of Configuration errors	Error rate
Without fusion	374	12.51%
With fusion	319	10.67%

Table 5. Configuration errors without/with information fusion

5. Meeting Browser

In today's time-constrained world, not everyone can attend every meeting. While minutes can serve as a summary or substitute for a meeting, minutes have many inherent problems as discussed in this paper's introduction. Meetings consist of much more than just the dialogue that occurs. In any meeting there are a combination of visual and verbal cues such as handwriting, facial expressions, gestures, body language, and of course speech. All of these facets are important for creating an accurate meeting record.

An important part of meeting recognition is the ability to efficiently capture, manipulate and review all aspects of a meeting. To that end our goal is to develop a meeting browser that supports the following capabilities:

- Create meeting records and transcripts of meetings with participants located throughout the world.
- Create and customize dialogue, audio, and video summaries to the user's particular needs.
- Create a database of corporate knowledge.
- Quickly and accurately create and disseminate a list of conclusions and action items
- Provide rapid access to meeting records to allow browsing and reviewing existing meetings

Using LV-CSR (large vocabulary, continuous speech recognition) described in Section 3, we are able to generate a meeting transcript. We are able to generate automatic summaries according to the user's initial specification, however, we provide the capability to create summaries at anytime. As part of the meeting, we identify for each utterance the speaker properties (type, social relationships, and emotion) as well as the discourse structure and type.

Our meeting browser, shown in Figure 5, is written completely in Java. This is a powerful tool that allows us to create a new meeting, review or summarize an existing meeting or search a set of existing meetings for a particular speaker, topic, or idea. We describe each of the components of the meeting browser in detail in this section.

5.1 User Interface

The meeting browser main window consists of three sections, an upper graphical display which shows the meeting over time, a lower left window that shows a meeting transcript, and a lower right window which displays either a video of the current meeting or a dialogue summary. Figure 5 shows the meeting browser with a video window displayed.

5.2 Meeting Creation

When a meeting is being created, each participant may join either remotely or locally by starting the people id system described in Section 4. We currently support up to six participants either which can be broken into any number of local or remote groups.

Figure 5 Meeting browser shown with a meeting video

The meeting browser is started and all invited participants are listed within the meeting browser. It is necessary to identify the meeting participants for two reasons. 1)We do not want to allow uninvited persons to eavesdrop on or participate in our meeting 2)Multiple meetings may be occurring simultaneously and we do not want the wrong speaker to be inadvertently included in the wrong meeting.

Once the meeting has begun, speech along with the speaker id derived from people id flows to Janus, our speech recognition engine. As the speech is recognized, the hypothesis is sent to the dialogue server where it is assembled into a meeting format. The meeting browser displays the transcript for the current meeting. The meeting transcript can be sent to the summarization server which will create a summary of the current dialogue. Finally, a user may elect to save a meeting from within the meeting browser. The meeting including any summaries is stored in the meeting archive for later recall and review.

At the end of many meetings, it is customary to reiterate a set of action items. Using speech recognition technology, we can recognize the items and mail them out to each of the meeting participants. Likewise, we can mail complete meetings, meeting segments, or summaries including the audio portion directly from within the meeting browser to meeting participants or any other interested parties. Each of these may include annotations, comments or corrections. Corrections can be done by using a keyboard or handwriting recognition [Manke95]. In the future we plan to add speech recognition as an additional error repair modality.

5.3 Summary Server

The meeting browser has the capability to create audio, video, and text summaries. In each of these cases, a summary is created on the basis of the recognized text dialogue, then the appropriate portions of the audio or video are clipped in order to create a summary. A user specifies the summary size as well as the central topic (if any) of the summary. This information along with the meeting dialogue is sent to a summary server which runs remotely. The server analyzes the dialogue and returns a summary to the meeting browser. In this way we can create summaries that allow the user to drill down from a general summary to a very specific topic or area of interest.

Col. M. S. Fisher*: yes sir what i'm hearing is that the iranian delegation is becoming quite belligerent at the conference last night at the annual opec social the iranian delegation asked to be seated with the oman's vice the saudis as the seating chart had indicated*
Adm. Roc Kelly*: if i could interrupt sir scott when i was commander of the middle east force back in the mid eighties the iranians were always spouting rhetoric however this seems a little more unusual*
Susan Ling*: well sir as you know this area of the spratleys is particularly important because of the immense oil reserves discovered last year by the s. s. texaco explorer this was funded by vietnam*
Susan Ling*: sir if i may remind you during your asean visits last year we stopped in to see the sultan of brunei and he offered to broker a peace accord if we would chair the conference*
Col. M. S. Fisher*: yes sir admiral you hit it right on the head about fifteen minutes ago the whole iranian delegation walked out of the abu dhabi princess hotel conference center spouting rhetoric about retaliation against the saudis and closing the straits of hormuz if necessary*

Figure 6 Sample Dialogue Summary

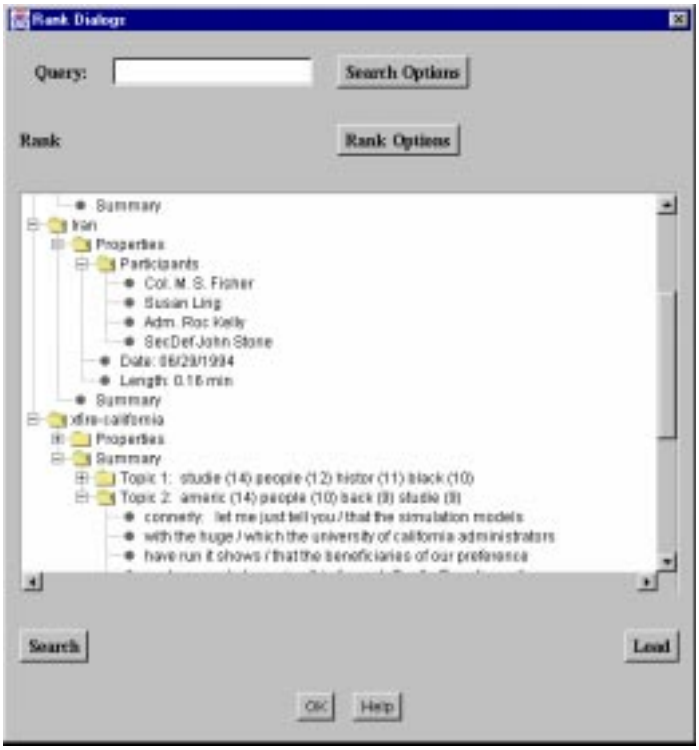
The algorithm for the summarization techniques is based on the MMR (Maximal Marginal Relevance) [Carbonell97]. This is a uniqueness measure that ranks the turns in the dialogue by topic and includes only turns for which topics have not previously been included. The summary server identifies the set of topics and returns a marked dialogue to the meeting browser. The summary server eliminates redundant turns from the dialogue without loss of meaning. Figure 6 shows an example of a summary. See [Waibel98] for a technical description of the algorithm. In the future we plan to incorporate the ability to create summaries across multiple meetings.

5.4 Dialogue Server

The dialogue server parses the input from Janus, or the meeting archive (described below) and sends it to the meeting browser. It processes requests from the meeting browser to retrieve meeting data such as transcripts, audio or video of the meeting. As such the dialogue server is tightly integrated with the meeting archive; it stores meeting transcripts as they are being created.

5.5 Meeting Archive

An important part of meeting tracking is the creation of corporate knowledge that is archived and available for later reference. Given the vast number of meetings individuals attend, it is very difficult if not impossible for one to remember the important events and details of each one. In addition, no means currently exist to identify the topics that occur in each meeting.



The meeting archive presents meetings in a tree format. It allows individuals to search for meetings based on any combination of participants, topics discussed, keywords, meeting length, and meeting date. In addition, if there is a summary for a meeting, the user can review it without loading the entire meeting in the meeting browser. The summary can be topic based, turned based or both. If a user chooses, they can load a meeting into the meeting browser from within the meeting archive. While the meeting archive is currently integrated into the meeting browser, we are planning to componentize it in the future so users can run it remotely.

Figure 7 Meeting Archive shown with dialogue expansion

5.6 Discourse Feature and Emotion Detection

Another important part of a meeting is the ability to detect opinions, statements, and emotions. These are seldom recorded as part of meeting minutes, yet they play a crucial role in determining the meaning and importance of what was said. Queries such as “Was John angry when he was discussing Iran?” or “Show me John’s opinions about Iran” become possible.

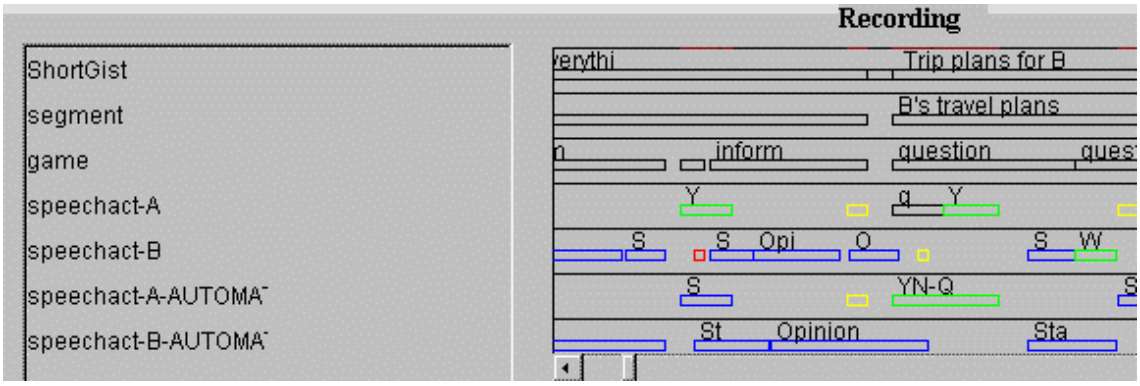


Figure 8 Discourse features displayed in the meeting browser

We interface with two systems, an emotion detection [Polzin98] and a discourse feature detection system [Ries99] which tap this higher level meaning. Both of these systems run offline on meeting data, but the resulting features are incorporated as part of the meeting record and are displayed in the meeting browser as shown in Figure 8.

6. Conclusions and Future Work

While we have made good progress in developing an integrated meeting room, much work remains to be done. We need to continue to improve our meeting recognition results and to experiment with a variety of microphones in multiple settings. Currently much of our work has been done in our lab. We want to duplicate our efforts at other locations in order to verify that our method is feasible. In addition, we are beginning to incorporate real time mpeg file creation and focus of attention tracking [Stiefel99] into the meeting record. Ultimately we would like to provide a meeting room on a laptop.

7. Acknowledgements

We gratefully acknowledge the support and backing of several individuals. We would like to thank Robert Malkin, Klaus Ries, Thomas Polzin, Klaus Zechner, and Weiyi Yang for their support on this project. We would also like to thank our sponsors at DARPA. 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.

8. References

[Bimbot97] F.Bimbot et al. *Speaker Verification In The Telephone Network: Research Activities In The Cave Project.* Technical report, PTT telecom, ENST, IDIAP, KTH, KUN, and Ubilab, 1997

[Bishop95] C.M.Bishop, *Neural Networks for Pattern Recognition.*Oxford University Press, 1993

[Carbonell97], Jaime Carbonell, G., Geng, Y., and Goldstein, J., *Automated Query-Relevant Summarization and Diversity-Based Reranking*, IJCAI-97 Workshop on AI and Digital Libraries, 1997.

[Finke97a] Michael Finke, et al. *Flexible Transcription Alignment*, Proceedings. ASRU '97, Santa Barbara, USA, Dec. 1997

[Finke97b] Michael Finke and Alex Waibel, *Speaking Mode Dependent Pronunciation Modeling in Large Vocabulary Conversational Speech Recognition*, Eurospeech 97, Rhodes, Greece

[Finke97c] M. Finke and J. Fritsch and P. Geutner and K. Ries and T. Zeppenfeld and A. Waibel, *The JanusRTk Switchboard/Callhome 1997 Evaluation System*, Proceedings of LVCSR Hub 5-e Workshop, May 1997

[Gauvain98] Jean-Luc Gauvain, et al. *Partitioning and Transcription of Broadcast News Data*, ICSLP98, Sydney

[Hub95] Proceedings of ARPA Speech and Natural Language Workshop, 1995, Morgan Kaufman Publishers.

[Hub96] Proceedings of LVCSR Workshop, Oct 1996, Maritime Institute of Technology.

[Manke95] S. Manke, M. Finke, and A. Waibel. *NPen++: A Writer Independent, Large Vocabulary On-Line Cursive Handwriting Recognition System*. Proceedings of the International Conference on Document Analysis and Recognition 1995

[Polzin98] Thomas S. Polzin and Alex Waibel, *Detecting Emotions In Speech*, Proceedings of the CMC 1998.

[Rabiner93] L.R. Rabiner and B-H. Juang. *Fundamentals Of Speech Recognition.* Englewood Cliffs, N.J. : PTR Prentice Hall, 1993.

[Ries99] Klaus Ries. *HMM And Neural Network Based Speech Act Detection*. Proceedings of the IEEE 1999 International Conference on Acoustics, Speech and Signal Processing (ICASSP), Phoenix, Arizona, March 1999.

[Stiefel99] Rainer Stiefelhagen, Michael Finke, Jie Yang and Alex Waibel. *From Gaze To Focus Of Attention.* Lecture Notes in Computer Science, Vol. 1614, pp. 761-768, June 1999, Proc. of Third International Conference on Visual Information Systems, VISUAL99

[Swain91] M. J. Swain and D. H. Ballard. *Color Indexing*. International Journal of Computer Vision, 7(1) pages 11-32, 1991.

[Terrillon98] J.-C. Terrillon, M. David and S. Akamatsu. *Automatic Detection Of Human Faces In Natural Scene Images By Use Of A Skin Color Model And Of Invariant Moments.* Proceedings of the Third International Conference On Automatic Face And Gesture Recognition. Nara Japan, 112-117, 1998.

[Turk91] M.A. Turk and A. Pentland. *Face Recognition Using Eigenfaces*. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pages 586-591, 1991.

[Waibel98] A. Waibel, M. Bett , M. Finke, R. Stiefelhagen, *Meeting Browser: Tracking And Summarizing Meetings*, Broadcast News Transcription and Understanding Workshop, Lansdowne, Virginia, 1998

[Yang96] J. Yang and A. Waibel. *A Real-time Face Tracker*. Proceedings of WACV'96, pages 142-147, 1996.

[Yang99] J.Yang, X.Zhu, R.Gross, J.Kominek, Y.Pue and A.Waibel. *Multimodal People ID for a Multimedia Meeting Browser*. Proceedings of ACM Multimedia 1999

[Yu98] Hua Yu, et al. *Experiments in Automatic Meeting Transcription Using JRTk*, Proc. ICASSP '98, Seattle, USA, May 1998

[Yu99] Hua Yu, et al. *Progress in Automatic Meeting Transcription*, Eurospeech '99, 1999