Portable Meeting Recorder

Dar-Shyang Lee, Berna Erol, Jamey Graham,
Jonathan J. Hull and Norihiko Murata*

Ricoh Innovations, Inc.
2882 Sand Hill Road, Menlo Park, CA 94025
+1-650-496-5716
{dsl, berna, jamey, hull}@rii.ricoh.com

*Ricoh Office System R&D Center
16-1 Shinei-cho, Tsuzuki-ku, Yokohama, Japan
+81-45-590-1819
nmurata@rdc.ricoh.co.jp

ABSTRACT

The design and implementation of a portable meeting recorder is presented. Composed of an omni-directional video camera with four-channel audio capture, the system saves a view of all the activity in a meeting and the directions from which people spoke. Subsequent analysis computes metadata that includes video activity analysis of the compressed data stream and audio processing that helps locate events that occurred during the meeting. Automatic calculation of the room in which the meeting occurred allows for efficient navigation of a collection of recorded meetings. A user interface is populated from the metadata description to allow for simple browsing and location of significant events.

Categories and Subject Descriptors

H.3.4 [**Information Systems**]: Information Storage and Retrieval – *systems and software.*

General Terms

Algorithms, Performance, Design, Experimentation.

Keywords

Omni-directional video, audio processing, appliance, MPEG-2 compressed domain analysis, meeting recorder.

1. INTRODUCTION

The typical mobile worker visits remote locations and participates in meetings with different people on a regular basis. A common task that must be performed at some subsequent time is the creation of a summary of what happened during a meeting, including who said what, the ideas that were conceived, the events that occurred, and the conclusions that were reached. Oftentimes, it's not just the specific conclusions but also the reasons they were reached and the points of view expressed by various participants that are important.

As most of us know, making an accurate summary is an error-prone process, especially if the only record we have is our own memory, perhaps supplemented with handwritten notes. A commonly used portable memory aid is the audiocassette recorder. It can be effective, but lacks the ability to capture important events that could be helpful later such as gestures, images of participants, body language, drawings, and so on. An easy-to-use method for incorporating video data would help solve this problem.

There have been several meeting recorder systems based on capturing panoramic videos proposed in recent years [1]-[4]. These systems provide a non-intrusive recording technique and use subsequent analysis to generate a user-oriented view for playback. In [3], the user-oriented view is determined based on speaker motion. A perhaps more intuitive solution is to compute the speaker direction as suggested in [4]. The user interface can be made more effective by combining audio and video analysis. A multimodal approach to creating meeting records based on speech recognition, face detection and people tracking has been reported in CMU's Meeting Room System [2]. Furthermore, techniques such as summarization and dialog analysis aimed at providing a higher level of understanding of the meetings to facilitate searching and retrieval have been explored [6].

The proposed solution is a portable meeting recorder that captures an omni-directional audio/video recording for a meeting. We assume that only limited data can be computed in real-time but that this is sufficient to produce a recording that can be replayed on the spot, if required. Subsequent analysis of the recorded data enables various output formats that improve the production of a meeting summary and allow for efficient browsing and navigation of the meeting video. Output formats include a *viewable* representation that shows an image of the person speaking so that the video can be played like a TV program showing a sequence of talking heads. A *searchable* representation is also produced that provides efficient techniques for navigating the multimedia data.

Our meeting recorder system is designed with portability and compatibility with commercial hardware in mind. Although the resolution of our system is lower than other panoramic systems such as FlyCam and RingCam, the advantage is its simplicity

ACM Multimedia 2002, Dec. 1-6, 2002, Juan Les Pins, France, 493-502.

and compatibility with existing commercial hardware, making it suitable for a portable system.

Even though the technology for video capture and storage commonly available today requires a bulky PC-based implementation of a prototype, in the near future the technical device capabilities assumed in this work will be available in a handheld device. This makes it essential for us to solve the problems inherent in a portable system now so that solutions are available when these systems reach the market.

Technical issues addressed in this work include the combination of audio and video data to locate the person speaking. We have developed a novel method of four-channel sound localization that accurately computes the angle and elevation of speakers from the capture hardware. Combined with a face detection algorithm, this technique effectively calculates a view of people speaking in a meeting.

Searching a recorded meeting for specific information can be a tedious and time-consuming process. We've also developed a novel user interface that represents speaker transitions and shows when events happened during a meeting and the context in which they occurred. This lets users easily navigate to those points in the video. A novel technique for compressed domain analysis of the MPEG-2 stream finds localized motion indicative of people moving. An algorithm for audio analysis measures the intensity of a conversation and the speed of participant interaction. These are both represented in the UI in a way that improves navigation of the recorded video.

An additional useful feature that was developed for the portable meeting recorder is the automatic detection of the room in which a meeting occurred. This can significantly improve the speed with which a large collection of meeting videos is searched. We describe a unique algorithm that clusters meeting videos and provides such a room-based search capability.

2. SYSTEM DESCRIPTION

The system architecture for the meeting recorder is shown in Figure 1. The hardware configuration consists of a special capture device, a touch screen monitor, as shown in Figure 2, and a PC. The capture device is composed of an omni-directional camera in the center and 4 microphones positioned at the corners. The camera has a parabolic mirror that captures a panoramic view of the meeting in a single *doughnut* video stream. Audio signals are fed into a multi-channel sound card and processed in real-time to determine the direction of speakers. The results are post-processed to produce a meta file that controls playback. The video, along with digitally mixed stereo audio, is sent to a video capture card and recorded as an MPEG-2 file. Encoding is done at 640x480 at 30fps.

Recording is controlled via a simple VCR-like interface on a 6.5-inch color touch screen panel (see Figure 2). When recording is started, the interface shows the amount of time recorded, the time left on the hard-drive, and a video preview window. Every recording session is automatically assigned an ID number. When recording is stopped, the results of sound localization and the video are post-processed to produce a meta file.

The results of sound localization are processed to produce viewing parameters for a virtual camera. A special viewer, shown in Figure 3, uses these parameters to automatically center on the speaker during playback. However, users can manually control the view using pan, tilt, and zoom operations. A compass at the bottom of the display shows the orientation of the current view with respect to the entire panorama.

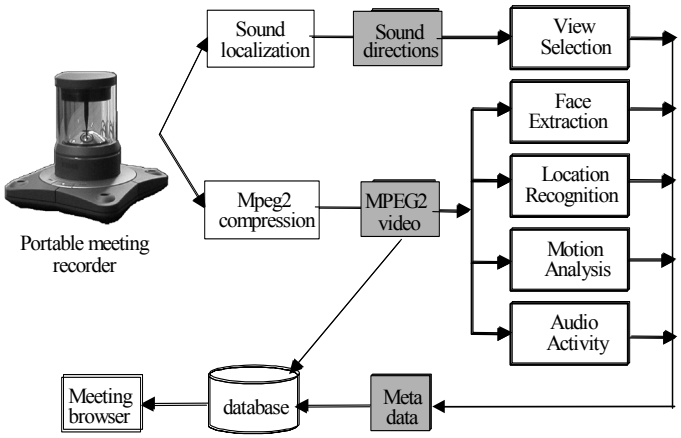

Figure 1. An overview of the meeting recorder system.

Figure 2. A touch screen controlled meeting recorder.

The data on speaker directions is also used in combination with skin detection to extract face images of meeting participants. Background images are extracted from the video to identify the meeting location. This information is displayed in a meeting description document in HTML format along with user added annotations. The audio is further analyzed to detect significant events. Based on motion analysis performed on the compressed data stream, events involving large spatial activities are identified. All the information associated with the meeting is written to a meta file. The video and meta file are archived and made available on a database server.

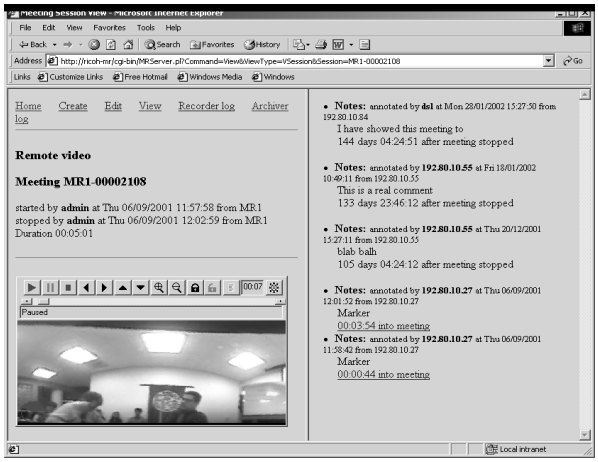

Figure 3. Meeting viewer interface.

3. META DATA CREATION

Efficient access to a recorded meeting is essential for users searching for specific information. Frequently, meetings are boring, unstructured affairs that are not amenable to a hit-or-miss search strategy. After fast-forwarding a few times in a meeting video while looking for something, most people will give up unless what they are seeking is important enough to spend the required time.

Our goal is to augment the audio and video information with meta data that enables a goal-directed search strategy in which users can easily navigate to the specific point in the recording that provides the information they are looking for. In addition, a user interface presents the meta data on a time-line and provides an easy means for browsing and selectively replaying the audio and video.

The use of meta data to help navigate video has been investigated by others. The Informedia project automatically applied a variety of analyses, including speech recognition and natural language processing, on TV footage [6]. The Broadcast News Navigator [7] derived information from the audio, video, and closed caption streams and performed linguistic information that improved the accessibility of the data. Such multi-track information was also used for video editing [8] and has been applied to browsing of recorded meetings [9].

3.1 Real-Time Sound Localization

To avoid the need of handling and saving multiple channels of audio data, sound localization is performed in real-time. The audio signal is processed in segments of 25msec. Since we are interested only in human speech, segments that do not contain speech in at least one of the channels are ignored.

Following speech detection, 360-degree sound localization is calculated as follows. For each pair of microphones on the diagonal, an angle between 0 and 180 degrees is calculated based on phase difference. This angle defines a cone of confusion centered at the midpoint of the diagonal. In theory, the intersection of two cones computed from both diagonal pairs defines the azimuth and elevation of the sound source. Unfortunately, the angle computed by each pair is not perfect. Moreover, phase difference measured on a finite sampling rate over a small baseline is discrete, and the angular resolution over all directions is non-uniform. Higher resolution is obtained near the center, and lower towards both ends. Therefore, we need to compute the intersection of two cones of unequal thickness, if they intersect at all. Furthermore, we want to take into consideration the confidence associated with each angle estimate.

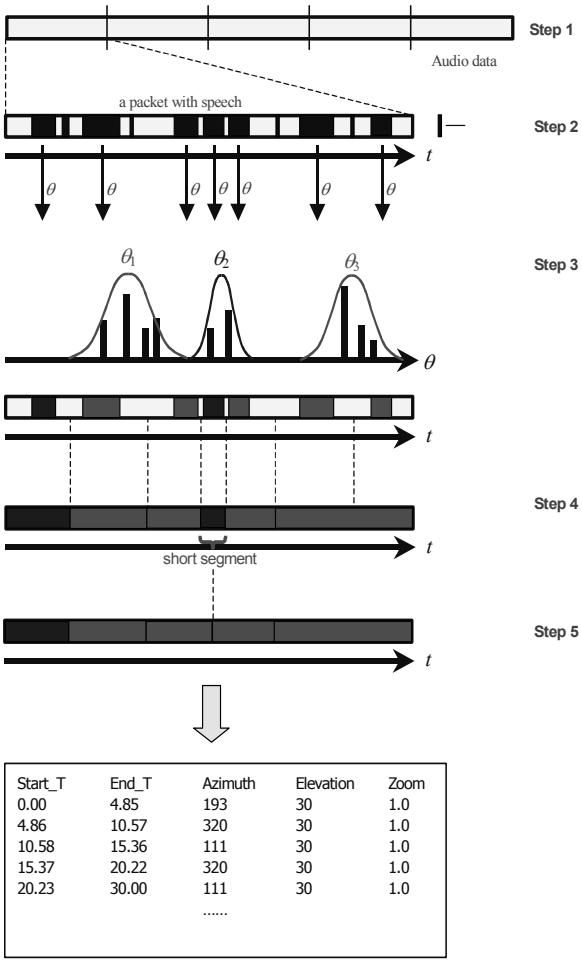

Figure 4. Illustration of the steps in generating view selection metadata. Audio data are divided into blocks of 30 seconds in step 1. Audio segments are grouped in Step 2, followed by clustering in Step 3. Afterward, groups are assigned to clusters.

To resolve these issues, we use an accumulator over the parameter space of azimuth by elevation. Azimuth varies from 0 to 360 degrees and elevation varies from 0 to 90 degrees. For each possible (azimuth,elevation) covered by each cone, its entry is incremented by the confidence associated with the cone. The highest scoring entry in the accumulator corresponds to the best parameter estimate. All entries in the accumulator are decayed by a factor at the end of each segment. However, in trying to estimate both azimuth and elevation, we found the solution unstable and sensitive to the quantization chosen.

Furthermore, it does not account for the fact that sound sources close to the middle are detected more accurately than those close to either end. Therefore, the scores at all elevations are summed up for each azimuth, and the best azimuth is returned if its score exceeds a threshold. Then, for each segment where speech is found, a triplet of time-stamp, angle and score, denoted as (t, θ_i, w_i), is written to a file. We observed this process is capable of performing in real-time, consuming approximately 25% to 40% CPU load on a 933MHz PC.

3.2 Automatic View Selection

At the end of a recording, the result of sound localization is further processed to produce a sequence of viewing instructions for the virtual camera, as illustrated in Figure 4, to generate a normal perspective view during playback. The objective of this view selection process is to create natural looking shots like those produced by a cameraperson. The steps for generating this sequence of instructions are described below.

Raw sound localization results are filtered, grouped, clustered, and smoothed to form viewing instructions. Since the initial analysis is performed on short audio segments (25msec), the results of speech detection and sound localization are sporadic. To find real speech utterances, we use only data where speech is detected in at least 5 consecutive segments. For each group of contiguous segments, a direction is calculated as a weighted average of the azimuth and weight. The total weight of the segment is assigned as the weight for the group. Consequently, groups containing more segments and more reliable sound direction estimates have larger weights.

In the next step, clustering is performed on the directions of these groups to find general speaker location using the ISODATA algorithm. We use a modified version where cluster means are calculated using the weighted average to take group weights into consideration, and an angular distance that wraps around at 360 is used. New clusters are formed for points more than a threshold away from the center. Clusters are merged if their centers are closer than a threshold, currently 30 degrees.

After clustering, every group is assigned to a cluster. This is roughly equivalent to unsupervised speaker clustering based on their (angular) location. To allow for speaker movement, this operation is performed on audio data of a chosen block size. Currently, we use a block size of 30 seconds in our system. It should be pointed out that the real goal of clustering is to identify distinct shot directions. Therefore, it is acceptable to use a single shot centered between two speakers if they are sitting nearby rather than centering exactly on the speaker.

Having obtained clusters of shot directions, we perform the final step to generate the viewing instruction. First of all, neighboring groups that belong to the same cluster are merged to cover any silent period between them. Neighboring groups that belong to different clusters are extended to meet half way, each covering half of the silent period. This allows the virtual camera to focus on the speaker before the actual speech starts. The result of this process is a sequence of view angles corresponding to shots. To avoid rapid switching of camera angles, shots shorter than 2 seconds are removed and considered as silence. The same algorithm is used to find coverage for that period.

In contrast to the work of [1] where regions containing the largest motion are selected, our system focuses on the speaker. The information we obtain at the end of clustering can be improved by speaker identification and displayed as speaker segments, as in [2]. Compared to the virtual director of [4], speaker directions are detected automatically instead of annotated manually.

3.3 Meeting Location Recognition

Unlike systems that are based on instrumentation of a conference room where most meetings are carried out in one place, meetings recorded with a portable meeting recorder can take place in different locations. Identifying the meeting location can provide a very useful retrieval cue. However, this could be a challenging task. One possible solution is to incorporate a GPS device into the meeting recorder. However, the accuracy of today's GPS technology may not be sufficient to accurately identify the meeting locations, especially considering that they take place indoors.

Our solution is based on recognizing the meeting room from visual clues. We first perform background/foreground extraction as the recorder is manually operated and therefore it is unreasonable to assume that a clean shot of the background can be obtained with no person in the room. We use adaptive background modeling to extract the background [10]. Our algorithm is based on an extension of the method of [11]. A Gaussian mixture approximates the distribution of values at every pixel over time. For each Gaussian constituent, its likelihood of being background is estimated based on its variance, frequency of occurrence, color and neighborhood constraints. Therefore, an image of the background can be constructed based on the most likely background Gaussian at every pixel. Since this background estimate changes over time, for example due to the movement of objects in the room, we extract a new image every time a significant change in the background model is detected. These images are dewarped into a panoramic cylindrical projection as shown in Figure 5.

To identify the location, the background images are matched against room templates in the database. Since the number of placements for the recorder in a particular room is usually limited, they are categorically organized and stored as separate templates. In our case, one template is obtained from each end of a table in a conference room. We match the templates with the backgrounds of the meeting recordings by comparing their color histograms. The histograms are formed in the HSV color space because distance values in this space approximate human perception. The color space represented with 256 bins, where Hue is quantized into 16 bins, Saturation and Value are quantized into 4 bins each.

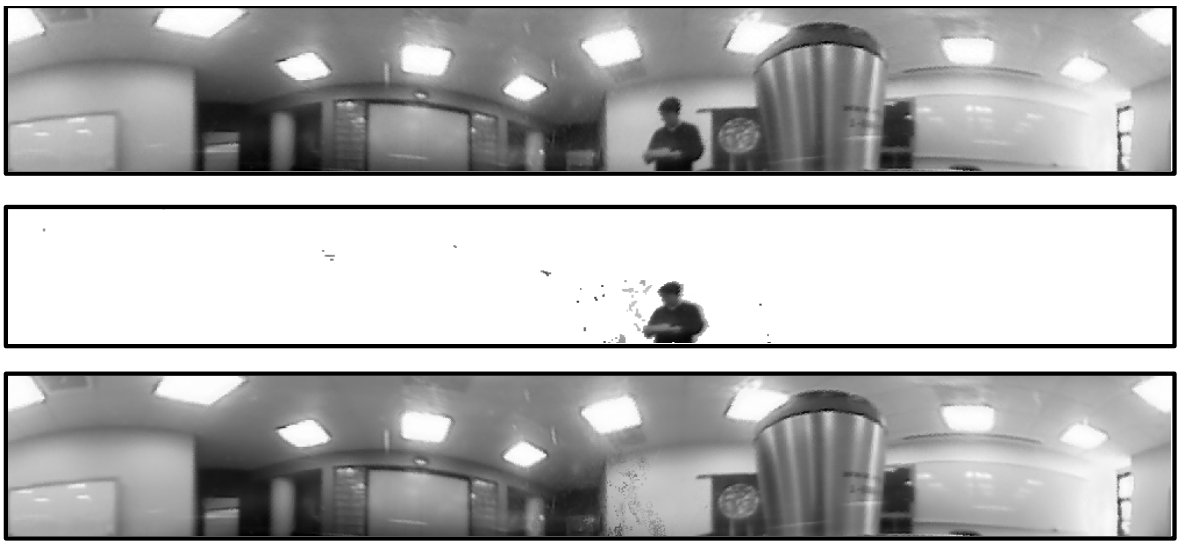

Figure 5. From top to bottom, examples of a panoramic video frame, the extracted foreground and the background image.

Several background images are extracted for each meeting and an intersection histogram is computed using the histograms of these images. The intersection histogram is compared using Euclidian distance with each template in the database to find the closest matching meeting room. Employing an intersection histogram allows us to further eliminate the non-stationary objects in the meeting room and smooth out any background extraction errors. The use of multiple templates for each room provides a robust method for location identification. In our experiments, we successfully identified the 5 meeting rooms that we have in our research facility. We are currently investigating improvements to the algorithm by using the size and the layout of the meeting room to address the issue of distinguishing rooms with similar colors.

3.4 Meeting Description Document

Inarguably, having the time, location, main topic, and participant list of a multimedia meeting document, such as the one shown in Figure 6, helps users to easily browse, search and access a large collection of meeting documents. In our system, users generate meeting content description documents semi-automatically. This is accomplished by extracting metadata automatically where possible and giving users the opportunity to either confirm the accuracy of the data or re-enter it.

The most basic meeting document metadata includes a description of the meeting, its time, date, and location. The date and time of the meeting are automatically obtained from the time stamp of the meeting document. To improve the start and end time accuracy, we also employ image and audio processing to detect exactly when the meeting started (e.g. when a speaker or motion is detected for the first time) and stopped. The location of the meeting is found automatically by using the method described in Section 3.3.

Automatic extraction of meeting title and description is more difficult. This can be accomplished by comparing the participant list and the time/location information with those of previous meetings and suggesting meeting descriptions, such as "regular group meeting." Moreover, if a presentation is detected in the meeting or there is a scheduled talk, this information can also be used to suggest meeting descriptions.

Figure 6. Meeting description document.

3.4.1 Localization of meeting participants

Locating meeting participants is a non-trivial problem especially considering that a clean shot of the background is not available and participants are likely to have minimal motion. We address this problem by using sound localization to find the approximate location of each meeting participant. Then the precise location of each face is found by identifying the skin regions in this approximate location.

Skin pixels are detected in the normalized RG-space [5]. Small holes in skin-colored regions are removed by a morphological closing and then connected component analysis is used to identify face region candidates. In environments with complex backgrounds, many objects, such as wood, clothes, and walls, may have colors similar to skin. Therefore, further analysis of the skin-colored regions, using techniques such as luminance variation [13] and geometric feature analysis [14]- [16], is performed to further eliminate non-face regions. Some example face localization results are shown in Figure 7.

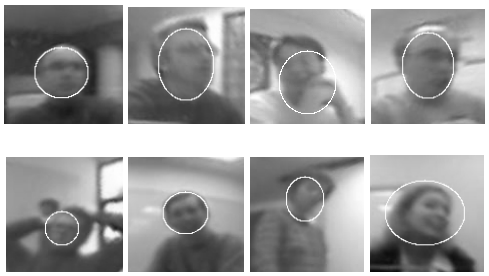

Figure 7. Face localization in various meetings.

3.4.2 Best-shot Selection

One of our goals is to find representative shots of the meeting attendees that can be included in the meeting description document. It is possible to extract many shots of a participant from the video sequence. However, generally not all of these shots are presentable. It is desirable to obtain frames where the individual is not occluded and facing the camera. We find such frames by first extracting several still shots of the speaker, one when she/he first starts speaking, one from when she/he finishes speaking (for the first time) and one between these two times. These shots are then evaluated to pick the best shot of a participant.

The best shot is selected by evaluating the size of the face region and the percentage of skin pixels detected in the best-fitted ellipse around the face region. The larger faces with more skin pixels are selected as better shots. An example is shown in Figure 8. Currently, the resolution of captured video is not sufficient to accurately detect the eye and mouth regions. However, once a higher resolution video is available, the selection of the best attendee shots can be improved by testing the visibility of mouth and eyes. This can also be combined with the geometry of the face to detect whether or not a person is looking straight ahead.

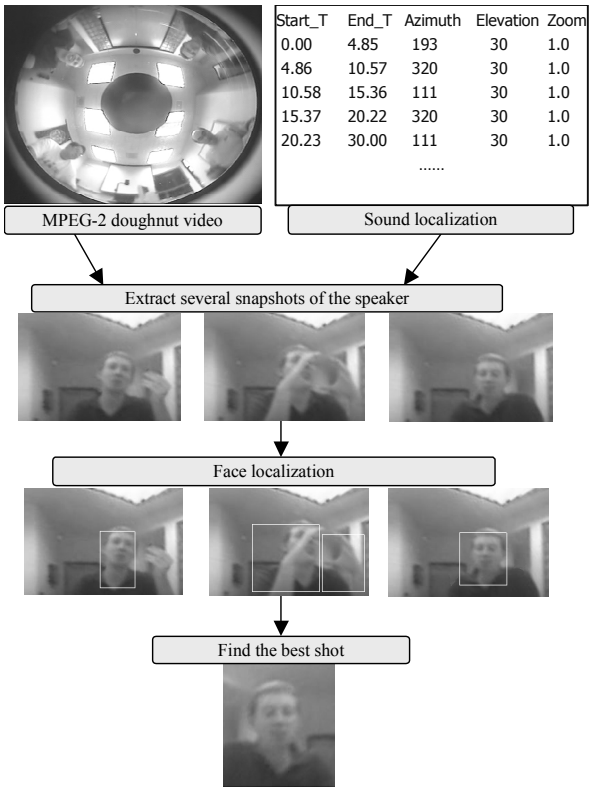

Figure 8. An example of best-shot selection.

3.4.3 Participant Identification

Recognizing people, even more specifically recognizing meeting participants using audiovisual data, has been an active research topic in recent years [13][18]. Nevertheless, face recognition and speaker identification may fail quite often because of poor lighting conditions, poor microphone quality, camera position, low video resolution, or even simply because a particular person looks different that day. The low resolution of our portable meeting recorder makes face recognition from video unreliable. Currently, after obtaining a set of participant shots, we present our best guess to the user and let the user confirm or change the people ID results.

3.5 Searching and Browsing with Visual and Audio Content

Searching and browsing audiovisual information is a time consuming task. In our meeting recorder system, after each meeting is recorded, the audio file can be sent out for transcription. This step can be removed when automatic speech recognition systems become more accurate. The transcription is then used to support text-based search. This is a useful way to access the spoken meeting content. However, it may not be possible to search for visual and audio events, such as a person getting up to write something on the whiteboard or an emotional discussion, in this way. Instead, we provide the user with a representation of the visual and audio activity content that can be easily browsed.

3.5.1 Visual Activity Analysis

Motion content in video can be used to efficiently search and browse particular events in a video sequence as demonstrated in various applications such as sports events and news broadcasts [18]-[20]. In meeting sequences, most of the time there is minimal motion. High motion segments usually correspond to significant events such as a participant getting up to make a presentation, someone joining the meeting, etc. Therefore, providing a visualization of the activity in a meeting potentially enables efficient meeting browsing.

Several motion activity descriptors exist in the literature. Some of these descriptors are based on the magnitudes and directions of the motion vectors in the MPEG bitstream [21]. However, these descriptors have a strong dependence on the bit rate and video encoder parameters. MPEG-7 defines a motion activity descriptor, which describes the amount of motion as well as the number and size of the active regions in a frame [22][23]. Visualization of this descriptor value is not intuitive. The visual activity measure we employ uses the local luminance changes in a video sequence. A large luminance difference between two consecutive frames is generally an indication of a significant content change, such as when somebody gets up to present, leaves the room, etc. However, other events, such as dimming the lights or all the participants moving slightly, may result in a large luminance difference between two frames. In order to eliminate such events, we define the visual activity as the luminance changes in a small window rather than luminance change in a whole frame.

The luminance changes are found by computing the luminance difference between the consecutive intra coded (I) frames. We employ I-frames because the luminance values in I-frames are coded without prediction from the other frames, and they are therefore independently decodable [24][25]. We compute luminance differences on the average values of 8×8 pixel blocks obtained from the DC coefficients. The DC coefficients are extracted from the MPEG bit stream without full decompression. Average values of the 8×8 pixel blocks are found by first compensating for the DC prediction and then scaling by 8.

Because the video in our system is doughnut shaped, the pixels in the outer parts of the video contain less object information (i.e. more pixels per object). Therefore, the pixel values are weighted according to their location to compensate for this when computing the frame differences. The assignment of weights is done considering the parabolic properties of the mirror as follows

$$w(r) = 1/\cos^{-1}\left[\frac{1-4(r/R_{\max})^2}{1+4(r/R_{\max})^2}\right],$$

where r is the radius of the DC coefficient location in frame centered polar coordinates and R_{max} is the maximum radius of the doughnut image. The coefficients that do not contain any information (the location that corresponds to outside of the mirror area) are weighed zero.

We employ a window size of 9×9 DC coefficients, which corresponds to a 72×72 pixel area. The weighted luminance difference is computed for every possible location of this window in a video frame. The local visual activity, ϕ, is defined as the maximum of these differences as follows:

$$\phi = \max\{\sum_{n=-L}^{L}\sum_{m=-L}^{L}(\omega(\sqrt{(xL+n)^2+(yL+m)^2})\ A_{xL+n,yL+m})\},$$
$$\forall x = [L \ldots W-L], \forall y = [L \ldots H-L].$$

where W and H are the width and height of the video frame (in number of DC blocks), L is the size of the small local activity frame (in number of DC blocks), ω(r) is the weight of the DC block at location r (in polar coordinates), and A_{ij} is the luminance difference between two blocks at location (i×8,j×8) in two consecutive I frames.

Figure 9 shows the plot of local visual activity measure for a meeting video. The large values of visual activity correspond to important visual events. As shown in the figure, most peaks in the visual activity score correspond to significant visual events, for example, a person taking his place at the table (Figure 9.a), another person leaving the meeting room (Figure 9.c), entering the room (Figure 9.d and Figure 9.e), etc. On the other hand, the video segment shown in Figure 9.b does not have a visual significance. This segment has a large activity value because the person moved close to the camera and appeared as a large moving object because of the perspective. Exclusion of such segments from the important visual events is possible only if we compensate for the distance of the objects from the camera via utilizing techniques such as stereovision.

3.5.2 Audio Analysis

Our system enables navigation of meeting content based on the amplitude of audio signal and speaker changes. High speech volume often corresponds to meeting segments involving discussions or high emotion. Capability of browsing meetings using speaker changes allows the user to skim through the audio efficiently and listen only to the speakers he/she is interested in.

There are many techniques that segment audio to obtain speaker segments and acoustics classes [26]-[30]. In [26], Arons gives an overview of audio segmentation. Pfau et al. propose an HMM-based speaker segmentation method using a mixture of Gaussians [27]. Error rates of 20% are reported even in controlled environments. Kimber et al propose an audio browsing tool based on acoustics classes [28]. In [29], Tritschler et al. perform speaker clustering using a Bayesian Information Criterion. It is reported that speaker segmentation is particularly difficult when the speakers are distant from the microphone, the room has many reflective surfaces, training data is not available, and/or multiple speakers talk at the same time. In our system, many of these obstacles are present. Our experiments showed that basing speaker segmentation on the results of sound localization performed much better than using audio features for speaker clustering. Currently we are working on combining sound localization with people tracking to further improve speaker segmentation.

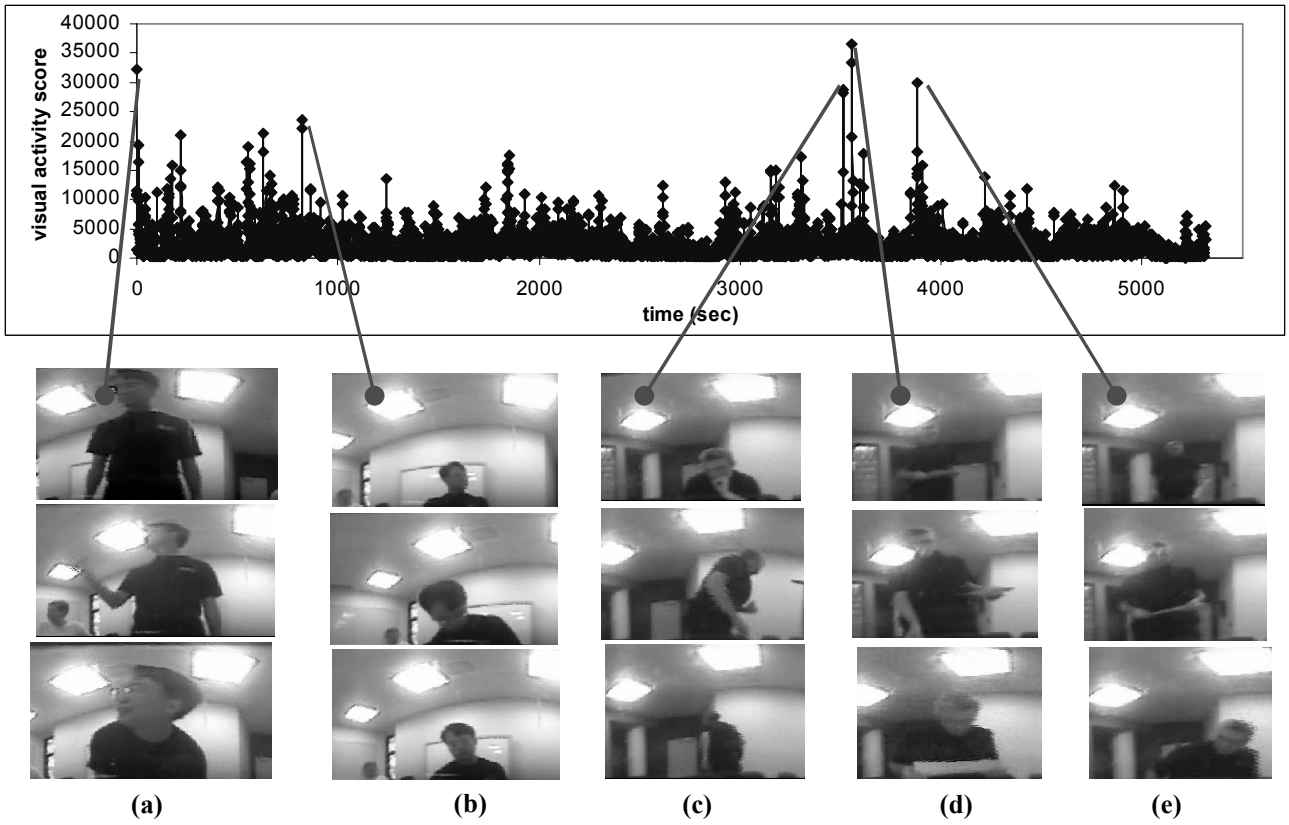

Figure 9. Examples of high visual activity scores corresponding to significant visual events.

4. SYSTEM IMPLEMENTATION AND USER INTERFACE

Using the prototype for the portable meeting recorder, we have been recording meetings since January 2002, which has provided a growing collection of more than 65 meetings, totaling over 50 hours of video and occupying over 135 GB of disk space. Video files are saved in the MPEG-2 format and metadata is saved in XML.

Meeting recordings and the relevant metadata are presented to the user with the MuVIE Client interface shown in Figure 10. MuVIE Client is a Java application that supports video editing, navigation of video using key frames, displaying and searching of the transcript, and an embedded web browser for displaying information relevant to the video [31]. Capabilities are also provided for viewing slides, whiteboard images, meeting minutes, both the perspective and panoramic meeting video, and tracks for speaker location, as well as visual and audio activity measures. Using this interface, the user can browse a meeting by reading the description page, listening only to the speakers that he is interested in, looking at the high-motion parts, searching for keywords in the transcription, looking at the presentation slides and whiteboard images, and so on. This way, hours of meetings can be browsed in much less time. The user interface also supports editing of the video, which enables the user to efficiently communicate meeting documents with others.

5. CONCLUSIONS

The design and implementation of a portable meeting recorder was presented. Even though today's prototype requires a small PC and is not easily moved, it is an excellent test bed for the development of the algorithms that will be required when suitably small devices become available in the near future. We also described novel algorithms for meta data extraction, including a four-channel sound localization technique, a view selection method, and a meeting location recognition technique. A meeting viewer interface (MuVIE) was described that displays the meta data, the transcript, as well as views of audio and video activity in a meeting. It allows users to easily find information in a recorded meeting and helps overcome the natural reluctance of people to search for information in a medium that's difficult to navigate.

The prototype system has been in regular use in our lab since January 2002. The reliable capture system, coupled with a web-based retrieval interface, has provided data that's easy to use and applies to common office-related applications.

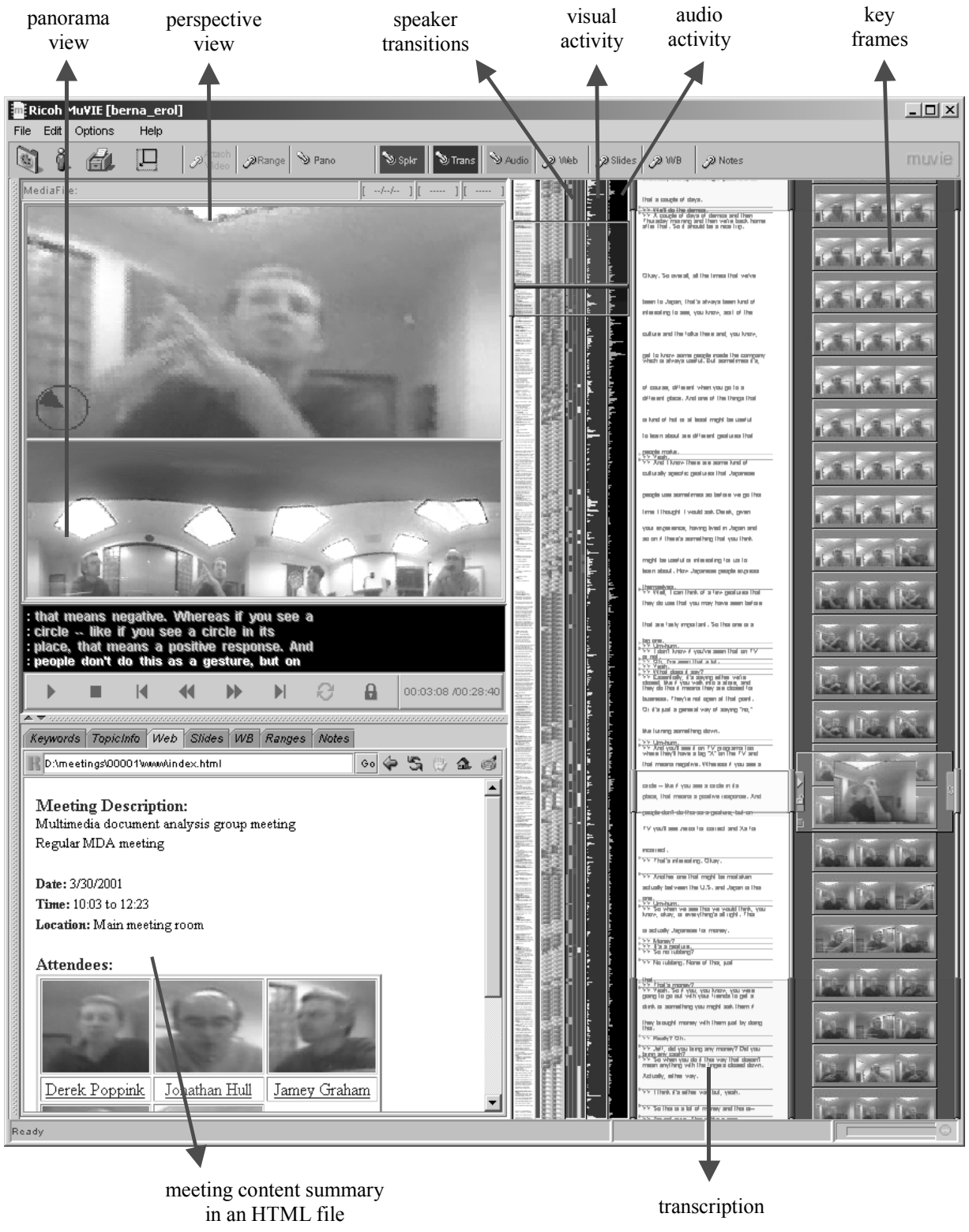

Figure 10. Meeting browsing using the Muvie Client.

6. REFERENCES

[1] Foote, J. and Kimber, D., "FlyCam: Practical panoramic video and automatic camera control," Proceedings of International Conference on Multimedia & Expo, vol.3, pp.1419-1422, 2000.

[2] Gross, R., Bett, M. Yu, H., Zhu, X., Pan, Y., Yang, J., Waibel, A., "Towards a multimodal meeting record," Proceedings of International Conference on Multimedia and Expo, pp. 1593-1596, New York, 2000.

[3] Sun, X., Foote, J., Kimber, D., and Manjunath, "Panoramic video capturing and compressed domain virtual camera control", ACM Multimedia, pp. 229-238, 2001.

[4] Rui, Y., Gupta, A., and Cadiz, J., "Viewing meetings captured by an omni-directional camera", ACM CHI 2001, pp. 450-457, Seattle, March 31- April 4, 2001.

[5] Waibel, A., Bett, M., Metze, F., Ries, K., Schaaf, T., Schultz, T., Soltau, H., Yu, H., and Zechner, K., "Advances in automatic meeting record creation and access", Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, 597–600, 2001.

[6] Hauptmann, A. G., and Smith, M., "Text speech and vision for video segmentation: The informedia project," Proceedings of the AAAI Fall Symposium on Computational Models for Integrating Language and Vision, 1995.

[7] Maybury, M., Merlino, A., and Rayson, J., "Segmentation, content extraction and visualization of broadcast news video using multistream analysis", AAAI, 1997.

[8] Myers, B. A., Casares, J. P., Stevens, S., Dabbish, L., Yocum, D., and Corbett, A., "A multi-view intelligent editor for digital video libraries", Joint Conference on Digital Libraries, Roanoke, VA, June 24-28, 2001.

[9] Foote, J., Boreczky, J., Girgensohn, A., and Wilcox, L., "An intelligent media browser using automatic multimodal analysis", ACM Multimedia, pp. 375-380, 1998.

[10] Lee, D. "Segmenting People in Meeting Videos Using Mixture Background and Object Models," Proc. of Pacific Rim Conf. on Multimedia, Taiwan, Dec.16-18, 2002.

[11] Stauffer, C. and Grimson, W.E.L, "Adaptive Background Mixture Models for Real-Time Tracking," Proceedings of Computer Vision and Pattern Recognition, pp. 246-252, 1999.

[12] Gross, R., Yang, J., Waibel, A., "Face Recognition in a Meeting Room", IEEE International Conference on Automatic Face and Gesture Recognition, 294-299, 2000.

[13] Hsu, R.L., Abdel-Mottaleb, M., and Jain, A. K., "Face detection in color images", Proc. International Conference on Image Processing , pp. 1046-1049, 2001.

[14] Yang, M.H., Kriegman, D.J., Ahuja, N., "Detecting Faces in Images: A Survey", PAMI(24), No. 1, pp. 34-58, January 2002.

[15] Kapralos, B., Jenkin, M., Milios E., and Tsotsos, J.: "Eyes 'n Ears Face Detection", 2001 International Conference on Image Processing, vol 1, pp. 66-69, 2001.

[16] Abdel-Mottaleb, M. and Elgammal, A., "Face Detection in complex environments from color images," IEEE ICIP, pp. 622-626, Oct. 1999.

[17] Yang, J., Zhu, X., Gross, R., Kominek, J., Y. Pan, Waibel, A., "Multimodal People ID for a Multimedia Meeting Browser," Proceedings of ACM Multimedia, pp. 159-168, 1999.

[18] Pingali, G. S., Opalach, A., Carlbom, I., "Multimedia retrieval through spatio-temporal activity maps", ACM Multimedia, pp. 129-136, 2001.

[19] Divakaran, A., Vetro, A., Asai, K., Nishikawa, H., "Video browsing system based on compressed domain feature extraction", IEEE Transactions on Consumer Electronics, vol. 46, pp. 637 - 644, 2000.

[20] Erol, B., Kossentini, F., "Local motion descriptors", IEEE Workshop on Multimedia Signal Processing, pp. 467-472, 2001.

[21] Dorai, C., Kobla, V., "Perceived visual motion descriptors from MPEG-2 for content-based HDTV annotation and retrieval", IEEE 3rd Workshop on Multimedia Signal Processing, pp. 147-152, 1999.

[22] Sun, X., Divakaran, A., Manjunath, B.S., "A motion activity descriptor and its extraction in compressed domain," Proc. IEEE Pacific-Rim Conference on Multimedia (PCM '01), pp. 450-457, 2001.

[23] ISO/IEC JTC1/SC29/WG11, "Multimedia Content Description Interface - Part 3 Visual". Publicly available at http://mpeg.telecomitalialab.com/ working_documents.htm, March 2001.

[24] Aramvith, S., and Sun, M.T., "MPEG-1 and MPEG-2 video standards", Handbook of Image and Video Processing, pp. 597-610, Academic Publishers, 2000.

[25] ISO/IEC, "Information technology - generic coding of moving pictures and associated audio information: Video," 13818-2, 1995.

[26] Arons, B., "Speech skimmer: A system for interactively skimming recorded speech", ACM Transactions on Computer-Human Interaction, vol 4, pp. 3-38, 1997.

[27] Pfau, T., Ellis, D.P.W., and Stolcke, A., "Multispeaker Speech Activity Detection for the ICSI Meeting Recorder", Proc. IEEE Automatic Speech Recognition and Understanding Workshop, 2001.

[28] Kimber, D., and L. Wilcox, L., "Acoustic segmentation for audio browsers," in Proc. Interface Conference. Sydney, Australia, 1996.

[29] Tritschler, A. and Gopinath, R., "Improved Speaker Segmentation and Segments Clustering using the Bayesian Information Criterion", Proc. of Eurospeech, pp. 679-682, 1999.

[30] Johnson, S.E., "Who Spoke When? - Automatic Segmentation and Clustering for Determining Speaker Turns", Proc. Eurospeech, Vol. 5, pp. 2211-2214, 1999.

[31] Graham, J., "The MuVIE Client System: A Multimedia Visualization and Integration Environment," Ricoh Innovations, March 2002.