IDIAP RESEARCH REPORT

BROWSING RECORDED MEETINGS WITH FERRET

Pierre Wellner, Mike Flynn & Maël Guillemot

flynn@idiap.ch, wellner@idiap.ch, guillemot@idiap.ch

IDIAP-RR-04-32

JUNE 1 2004

Dalle Molle Institute for Perceptual Artificial Intelligence • P.O.Box 592 • Martigny• Valais • Switzerland

phone +41 – 27 – 721 77 11
fax +41 – 27 – 721 77 12
e-mail secretariat@idiap.ch
internet http://www.idiap.ch

BROWSING RECORDED MEETINGS WITH FERRET

Pierre Wellner, Mike Flynn & Maël Guillemot

IDIAP Research Institute, Martigny, Switzerland

ABSTRACT

Browsing for elements of interest within a recorded meeting is time-consuming. We describe work in progress on a meeting browser, which aims to support this process by displaying many types of data. These include media, transcripts and processing results, such as speaker segmentations. Users interact with these visualizations to observe and control synchronized playback of the recorded meeting.

1. INTRODUCTION

Imagine that you missed a two-hour meeting with your colleagues, but that this meeting was recorded. You want to know what you missed, but you do not want to replay the entire meeting. Instead, you want to find quickly just the parts that most interest you.

A number of research projects are developing systems to capture, store, process, and retrieve recorded meeting data [2][6][7]. These recordings include not only audio from multiple microphones, but also video from multiple cameras and additional data streams such as hand-written notes and projected displays. The amount of data captured can be enormous, and making this data easily accessible and usable by humans is a challenging problem which has been approached in a number of different ways [10]. The Ferret Meeting Browser aims to provide the ability to quickly find and play back segments of interest from a meeting recording.

This paper begins by illustrating some of the observations people find of interest within meeting recordings; it describes our meeting room and some of the data streams captured from it; the Ferret meeting browser is described in detail, along with its software architecture, and we conclude with discussion on future work.

2. OBSERVATIONS OF INTEREST

A first step towards the design and evaluation of a meeting browser is to determine what users are interested in finding within recording meetings. Understanding this motivates the features we choose to implement, and will help us to evaluate browser performance. The precise information that people find interesting obviously depends on the individuals, and on the specific meeting in question. So rather than ask people in general what they might look for in a meeting recording, we ask people to make specific "observations of interest" about particular meeting recordings. We can then see how much people agree or differ on events of significance within these meetings, and we can design and test features of our meeting browser to help locate the observations that people find interesting. An initial pilot collection of observations indicates that:

- a large majority of observations are associated with a particular meeting participant;
- most observations of interest are about spoken content;
- substantial agreement between observers on their selection of significant events and where they occur.

We do not yet have measures for these patterns, but we can use them to illustrate how a browser can help find the typical observations made in a meeting. Two example observations, on a meeting that discussed movies were:

- "Alice knew the actors well."
- "Bob prefers the movie Terminator 2 over Terminator 1."

In the sections below, we will see how features of our meeting room and browser can help to find observations like these.

3. THE IDIAP SMART MEETING ROOM

The IDIAP smart meeting room is an ordinary conference room with a table, chairs, whiteboard, and computer projection screen. In addition, it is also equipped with 24 microphones, which are configured as lapel microphones, as the ears of a binaural manikin, and as a pair of 8-channel tabletop microphone arrays (see ***Figure 1***).

The room also has three video cameras and equipment for capturing time-stamped whiteboard strokes, Anoto pen strokes [3], and the computer-projected images. All recorded data is precisely synchronized so that every microphone, pen-stroke, and video sample can be associated with simultaneously captured samples from other media streams. Further details about this room and its equipment are described in [9].

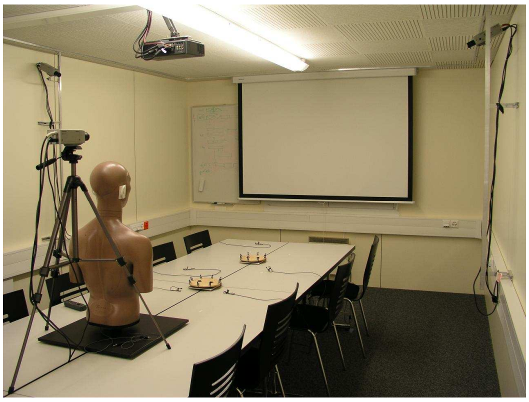


Figure 1 – IDIAP smart meeting room.

4. CAPTURED MEDIA STREAMS

Initial observations make clear that knowing *who* is talking in a meeting is very important to viewers, and to support this, a tabletop microphone array has been built (see ***Figure 2***). This microphone array is capable of tracking the directions from which speech is coming around the table, and can detect which participants are speaking at any time.

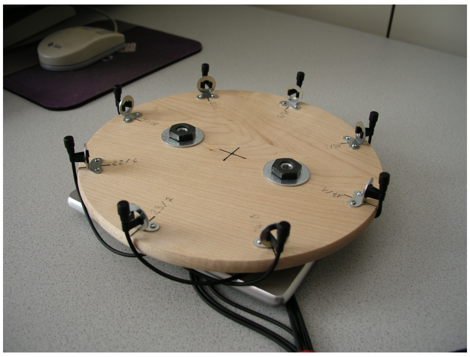


Figure 2 – Tabletop Microphone Array.

Processed output from the microphone array is illustrated in the lower part of ***Figure 3*** and shows the direction of sound sources and energy levels from four different speaker locations [4]. This output can be used in our browser to show when each participant speaks, and also some aspects of global meeting structure, such as when discussions occurred, or when a monologue was interrupted by a question.

Some of these meeting actions can be detected automatically. In experiments described in [8], audio and visual features for each participant are extracted from the raw data, and the interaction of participants is modeled using Hidden Markov Model based approaches. Testing these models on an initial corpus demonstrated the ability of the system to recognize a set of simple meeting actions such as presentations, monologues and discussion.

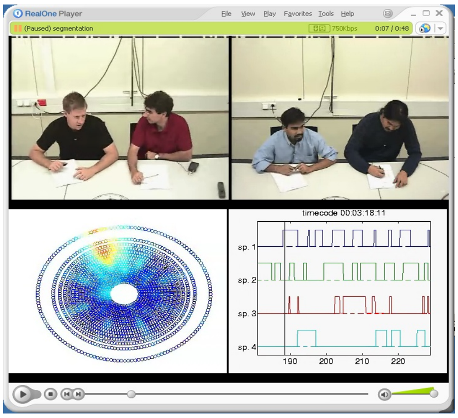


Figure 3 – Microphone array processing output.

5. FERRET FROM USER’S PERSPECTIVE

The Ferret browser is typically used in four stages:

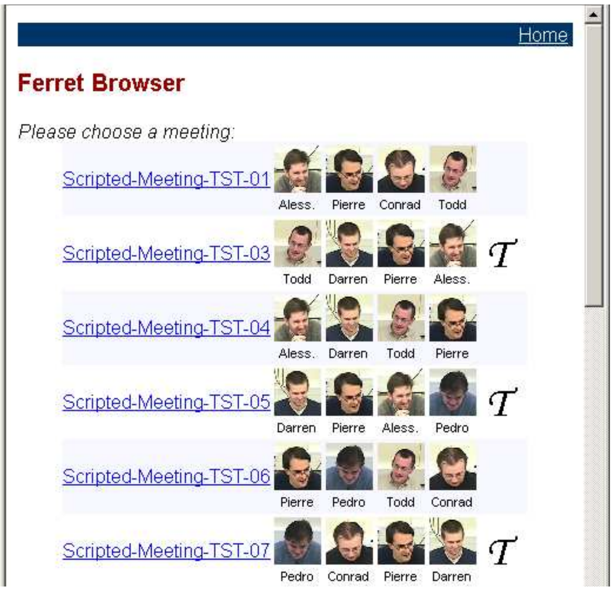
1. Meeting selection, where one meeting is chosen from many;
2. Initial choice of interval data streams to be visualized.
3. Interactive browsing and playback of the recording.
4. Addition and removal of data visualizations.

These are discussed in the following sub-sections.

5.1. Meeting selection

The user first chooses a meeting to browse. ***Figure 4*** below illustrates an example of a browser to navigate within a set of meetings, showing summary information and pictures of which people participated at a particular meeting, as well as a selection of additional data available to browse.

At this time, Ferret does not provide an inter-meeting search facility for querying across a large set of meeting recordings. It focuses primarily on browsing within a single meeting recording, assuming that the meeting was selected through some other means, such as this simple hierarchical list.



Home

Ferret Browser

Please choose a meeting:

Scripted-Meeting-TST-01 Aless. Pierre Conrad Todd

Scripted-Meeting-TST-03 Todd Darren Pierre Aless. T

Scripted-Meeting-TST-04 Aless. Darren Todd Pierre

Scripted-Meeting-TST-05 Darren Pierre Aless. Pedro T

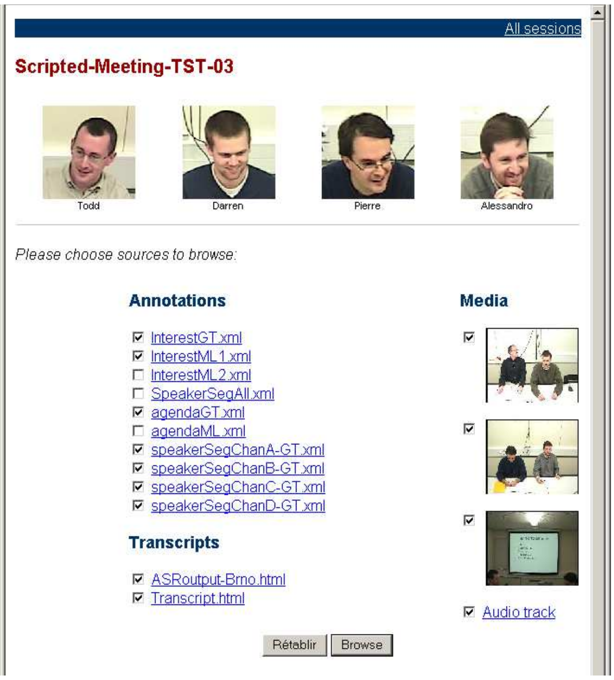
Scripted-Meeting-TST-06 Pierre Pedro Todd Conrad

Scripted-Meeting-TST-07 Pedro Conrad Pierre Darren T

Figure 4 – Inter-meeting browser prototype.

5.2. Initial choice of data

Once users have selected a meeting to browse, they select which particular data streams are of interest by clicking on checkboxes of available interval data streams. ***Figure 5*** below shows an example for one meeting.



All sessions

Scripted-Meeting-TST-03

Todd Darren Pierre Alessandro

Please choose sources to browse:

Annotations

☑ InterestGT.xml
☑ InterestML1.xml
☐ InterestML2.xml
☐ SpeakerSegAll.xml
☑ agendaGT.xml
☐ agendaML.xml
☑ speakerSegChanA-GT.xml
☑ speakerSegChanB-GT.xml
☑ speakerSegChanC-GT.xml
☑ speakerSegChanD-GT.xml

Transcripts

☑ ASRoutput-Brno.html
☑ Transcript.html

Media

☑
☑
☑
☑ Audio track

Rétablir Browse

Figure 5 – Meeting data page.

This web page displays the available annotation files and media files. Once a user has selected which XML streams, HTML transcripts or ASR output is desired, pushing the “Browse” button brings the user to the meeting browser.

In order to verify our example observations, a user might select the speaker segmentations, to see when each person speaks, and the transcript (if available). Note that transcripts produced by automatic speech recognition contain many errors, and may not be directly applicable.

5.3. Interactive browsing and playback

The upper part of the graphical user interface illustrated in ***Figure 6*** below contains a media player for synchronized playback of audio and video. This allows viewing and listening what happened at particular moment of the meeting from multiple angles as with a multi-TV system. The VCR controls (play, pause, sound adjustment, time display) are on the left pane.

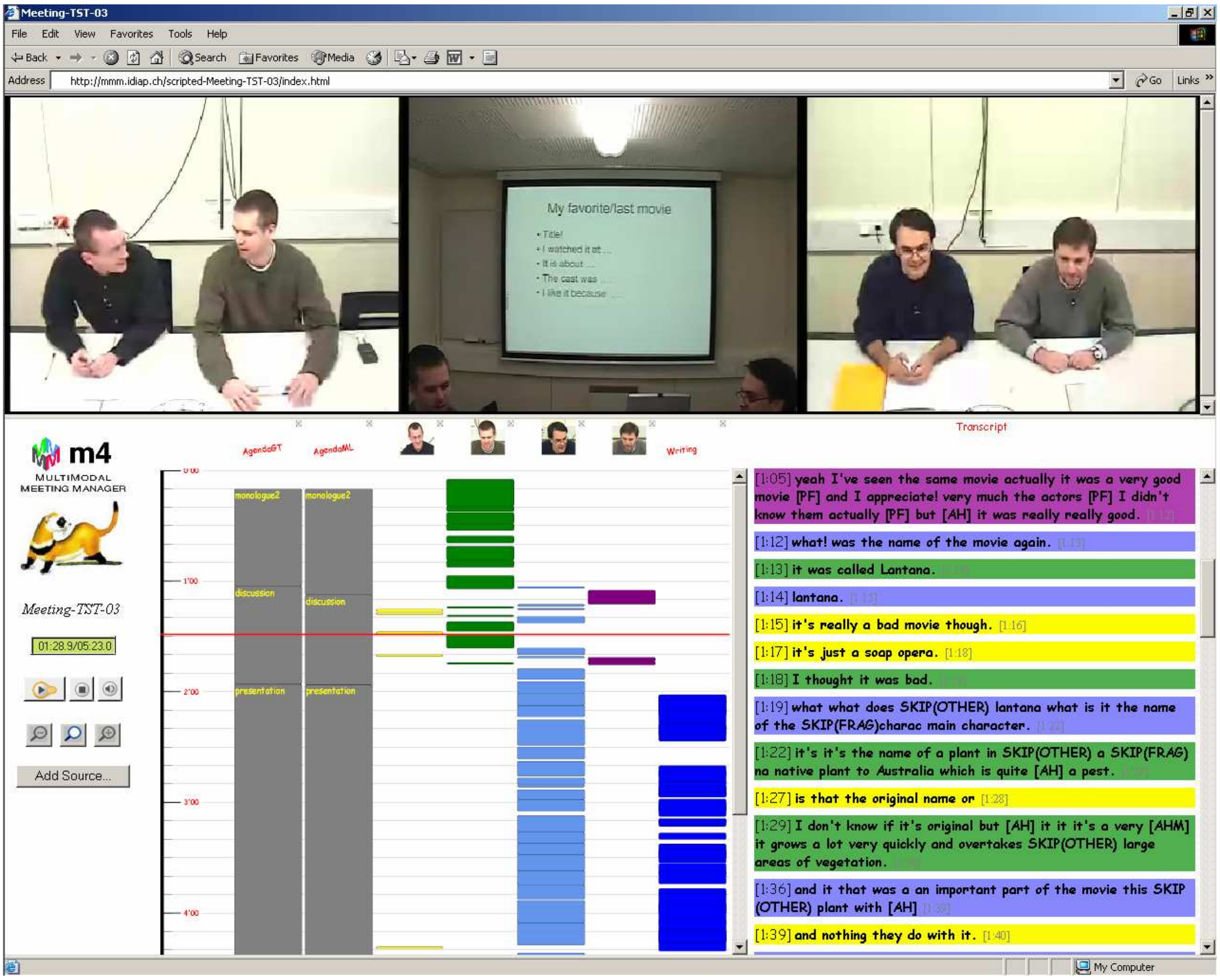
The lower part of the interface shows graphical representations of interval data streams in a vertical, scrollable timeline.

A large number of processed interval data streams can be made available to assist with browsing of the meetings. Ferret allows users to select any combination of available data streams as described in the previous section and display them alongside each other for inspection and comparison.

Interval data can include speaker turn intervals, meeting actions, level of interest representation, or textual transcripts. Clicking on elements in the timeline controls playback in the media player and scrolls the textual transcript. And clicking on the transcript also scrolls the timeline and controls media playback.

The user can zoom into particular parts of interest by means of the zoom buttons on the left. By zooming out, the user gets an overview of the meeting in terms of who talked the most, what meeting actions etc. Crosses at the top of each stream allow for deleting streams, while the add button in the control panel permits adding a new XML or HTML annotation stream to be visualized.

To illustrate how this browser can be used, imagine someone who wants to verify the truth of one of the typical statements shown above, e.g. “Bob prefers the movie Terminator 2 over Terminator 1.”



Meeting-TST-03
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://mmm.idiap.ch/scripted-Meeting-TST-03/index.html
Go Links
My favorite/last movie
• Title!
• I watched it at ...
• It is about ...
• The cast was ...
• I like it because ...
m4
MULTIMODAL MEETING MANAGER
Meeting-TST-03
01:28.9/05:23.0
Add Source...
AgendaGT
AgendaML
Writing
Transcript
0'00
1'00
2'00
3'00
4'00
monologue2
discussion
presentation
My Computer

[1:05] yeah I've seen the same movie actually it was a very good movie [PF] and I appreciate! very much the actors [PF] I didn't know them actually [PF] but [AH] it was really really good.

[1:12] what! was the name of the movie again.

[1:13] it was called Lantana.

[1:14] lantana.

[1:15] it's really a bad movie though. [1:16]

[1:17] it's just a soap opera. [1:18]

[1:18] I thought it was bad.

[1:19] what what does SKIP(OTHER) lantana what is it the name of the SKIP(FRAG)charac main character.

[1:22] it's it's the name of a plant in SKIP(OTHER) a SKIP(FRAG) na native plant to Australia which is quite [AH] a pest.

[1:27] is that the original name or [1:28]

[1:29] I don't know if it's original but [AH] it it it's a very [AHM] it grows a lot very quickly and overtakes SKIP(OTHER) large areas of vegetation.

[1:36] and it that was a an important part of the movie this SKIP (OTHER) plant with [AH] [1:39]

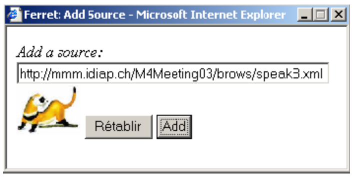
[1:39] and nothing they do with it. [1:40]

Figure 6 – The Ferret browser.

To find where this is discussed in the meeting, a user can invoke the browser's "Find" function in the transcript window, and enter keywords such as "prefer" or "terminator," which would scroll the browser down to the section in the transcript where these words occur. If the words are not visible, e.g. because of speech recognition errors, then the user can see the parts of the meeting where Bob is speaking, and click only on those parts to find the parts where Bob is talking.

5.4. Addition and removal of data

To supplement the data presented initially, users may add data of their own choosing to the display. This data does not need to be on any special server – but it must be accessible by the Internet. This feature is used by researchers who are testing new segmentation or recognition algorithms and want to see how their results look alongside other data, or want to check the accuracy of their results against playback of the media.



Ferret: Add Source - Microsoft Internet Explorer
Rétablir
Add

Add a source: http://mmm.idiap.ch/M4Meeting03/brows/speak3.xml

Figure 7 – The Add Source dialog box.

In order to display a new data source, the user presses the "Add Source" button, and a dialog appears, illustrated in ***Figure 7*** above. Entering the URL of the new data, and pressing the "Add" button, the data is displayed alongside the previously selected data. In effect, the Ferret browser acts as a "shopping basket" into which many disparate sources of data may be loaded from any server on the Internet. Data can also be removed from the main display by pressing the small close button by each source.

6. ARCHITECTURE

6.1. Overview

Ferret works over the Internet, using a normal web browser as the client application. The main meeting corpus also resides on the Internet, along with much processed data – such as transcripts and recognition results.

Figure 8 below illustrates the client-server architecture. The left side of the figure represents the content of the server, while the right side represents the client, an Internet Explorer web browser.

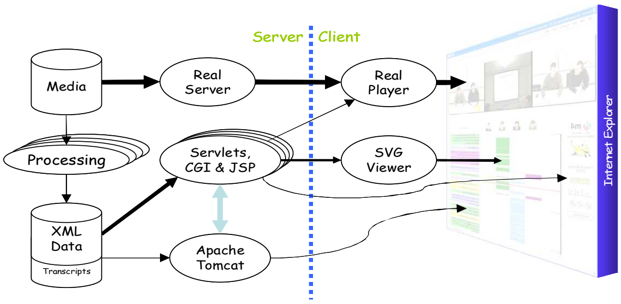


Figure 8 – Ferret Architecture.

6.2. Framework

A set of JSP-generated frames is used to produce the overall frame structure of the Ferret browser. Each JSP-generated page provides one of the many frames in the display, and is used to create the embedding of plug-ins for video and graphics. In turn, each embedded plug-in is given the URL of data to display.

6.3. Graphical display

A free plug-in, the SVG Viewer [1] allows complex graphics to be displayed and dynamically updated. This is used to display the meeting segmentations graphically, with interaction and dynamic updates of the graphics.

The URL passed by the Ferret framework to the graphics plug-in, invokes a Java servlet. The servlet generates the graphical content from the XML sources, dynamically. Running under the Apache Tomcat servlet container, it reads a list of XML data URLs to display. The servlet fetches each in turn, parses the XML content and generates SVG. Finally, the SVG is streamed back to the graphics plug-in in the browser for display.

6.4. Media display

The media resides on the mmm.idiap.ch media server, which contains AVI, RM videos, WAV and RM audio files. It is streamed to the Internet by RealServer and displayed in a browser through the free RealPlayer plug-in (available from www.real.com). RealPlayer facilitates the playback of streamed media, over the Internet, synchronising multiple sources.

The URL passed by the Ferret framework to the media plug-in, invokes a CGI program that dynamically generates SMIL to send back to the plug-in. SMIL is a standard language for presentation of multiple media sources. It allows multiple sources to be played in parallel, or in sequence (or any combination) with time offsets. The video plug-in reads the SMIL and then begins to fetch each media source at the appropriate moment.

6.5. Data stream management with cookies

The list of data to display is obtained from cookies kept by the client browser – no session information is kept in Ferret. Every client request to the server is accompanied by the current set of cookies, each giving one source to browse. Cookies are largely managed at the client, by JavaScript functions associated with the various browser forms, including the “Add Source” dialog, and the source-closing buttons.

6.6. Controls

A simple control frame contains player controls, zoom buttons and the “Add Source” button, described below.

The zoom buttons reload the graphics frame with a new zoom parameter. The list of sources to display is not disturbed, since this is stored as cookies. Therefore, exactly the same set of sources is re-displayed at the new zoom setting. The control panel itself is re-loaded, as new settings for the zoom buttons must be supplied.

The “Add Source” button either creates a new “Add Source” dialog window, or brings any existing one to the front. Entering a URL and pressing the “Add” button creates a new cookie (holding the new URL) and forces a reload of the graphics frame. The graphics are re-generated, now with the new source included.

In addition to the data currently displayed by Ferret, we are in the process of adding additional streams such as pen and whiteboard strokes, and representations of continuous data, such as energy levels or head positions.

6.7. Browser Evaluation Test

The evaluation of meeting browsers is typically subjective, making it impossible to compare browsers and browsing techniques objectively. It can also be difficult to know when a browser is truly effective in helping users find the information they are interested in, rather than simply finding the information the experimenter believes is important.

To evaluate Ferret (and other meeting browsers), we are developing a test that aims to be:

a) an objective measure of browser effectiveness based on user performance rather than judgement;

b) independent of experimenter perception of the browsing task;

c) able to produce directly comparable numeric scores, automatically; and

d) replicable, through a publicly accessible web site.

The design of this test is discussed in [5], and depends on the collection of "observations of interest" made by people who view the meetings. These observations are subsequently used to generate test questions about the meeting which subjects must try to answer using a browser, and resulting in a numeric score that can be used to compare alternate browser designs.

7. SUMMARY AND CONCLUSION

We describe work in progress on the Ferret meeting browser, which can display a number of user-selected processed interval data streams, and transcripts generated from meeting recordings. These displays are interactively used to control synchronized media playback of the recorded audio and video.

The aim of this work is to make it easy and quick for users to find information of interest within meeting recordings. Future versions of Ferret (and other meeting browsers we develop) will be objectively compared, in order to learn which of the interaction techniques and data visualization methods we have implemented are of most value to the task of browsing recorded meetings.

A working version of Ferret is available online at http://mmm.idiap.ch along with a collection of short meetings that can be browsed with it. Additional meeting recordings and data are available to project partners.

8. ACKNOWLEDGEMENTS

The Authors thank our colleagues at IDIAP for their collaboration in building the systems used in this paper. This work was partly supported by: the European Union 6th FWP IST Integrated Project AMI (Augmented Multi-party Interaction, FP6-506811, publication AMI-17); the Swiss National Science Foundation, through the National Centre of Competence in Research (NCCR) on "Interactive Multimodal Information Management (IM2)"; the European project "M4: MultiModal Meeting Manager", through the Swiss Federal Office for Education and Science (OFES).

9. REFERENCES

[1] Adobe SVG viewer http://www.adobe.com

[2] AMI project http://www.amiproject.org

[3] Anoto http://www.anoto.com

[4] G. Lathoud and I. McCowan. *Location Based Speaker Segmentation*, in "Proceedings of the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-03)", 2003.

[5] M. Flynn and P. Wellner, In Search of a good BET: A proposal for a Browser Evaluation Test, IDIAP-COM 03-11, September 2003.

[6] IM2 project http://www.im2.ch

[7] M4 project http://www.m4project.org

[8] I. McCowan, S. Bengio, D. Gatica-Perez, G. Lathoud, F. Monay, D. Moore, P. Wellner, and H. Bourlard. *Modeling Human Interaction in Meetings* in "Proceedings of International Conference on Acoustics, Speech and Signal Processing", 2003.

[9] D. Moore. *The IDIAP Smart Meeting Room.* IDIAP-COM 01-07, November 2002.

[10] Simon Tucker, Steve Whittaker, *Accessing Multimodal Meeting Data: Systems, Problems and Possibilities* Multimodal Interaction and Related Machine Learning Algorithms Workshop, June 2004.