Using Audio, Visual, and Lexical Features in a Multi-Modal Virtual Meeting Director

Marc Al-Hames, Benedikt Hörnler,
Christoph Scheuermann, and Gerhard Rigoll **

Institute for Human-Machine-Communication, Technische Universität München
Arcisstr. 21, 80290 Munich, Germany
{alh, hbe, chr, rigoll}@mmk.ei.tum.de

Abstract. Multi-modal recordings of meetings provide the basis for meeting browsing and for remote meetings. However it is often not useful to store or transmit all visual channels. In this work we show how a virtual meeting director selects one of seven possible video modes. We then present several audio, visual, and lexical features for a virtual director. In an experimental section we evaluate the features, their influence on the camera selection, and the properties of the generated video stream. The chosen features all allow a real- or near real-time processing and can therefore not only be applied to offline browsing, but also for a remote meeting assistant.

1 Introduction

Projects like Augmented Multi-Party Interaction (AMI) [2, 3], Computers in the Human Interaction Loop (CHIL) [8], or the ICSI meeting project [4] investigate how computers and machine learning techniques can be used to make meetings, lectures, and conferences more efficient, and how to automatically record, transcribe, analyse, and summarise them. One goal of the AMI project is the development of a meeting browser [10] that allows to recapitulate a meeting from its audio-visual recordings and automatically generated transcripts. Furthermore in an international world and with the required technology (like web-cams) now cheaply available, remote meetings are of emerging importance. These meetings allow people to connect audio-visual from their own office to other meeting rooms or participants. This allows regular meetings, while saving travel costs and especially the time of the meeting participants.

Both a meeting browser and remote meetings require to select one video stream, that is shown either to the user of the meeting browser, or to a remote participant. We can of course always show a merged visual stream with all meeting participants. Yet this is not desirable, because watching all persons at the same time is not convenient. Furthermore with an increasing number of

** This work was partly supported by the European Union 6th FWP IST Integrated Project AMI (Augmented Multi-party Interaction, FP6-506811).

participants all subtle details get lost, but they can contain important information, like disagreement [2]. Thus for both a meeting browser and remote meeting transmissions, a selection which camera should be shown is required.

One could possibly argue, that the task of a virtual director is in fact a speaker diarisation task: simply always select the camera that shows the current speaker. However, imagine a person presenting a novel idea in a meeting with the project board. The presentation lasts for about five minutes. If the virtual director follows a speaker diarisation rule, it will show the presenter all the time. Unfortunately this way the system has lost the most important aspect of the idea: during the presentation the project leader has continuously shook his head, indicating he is not very satisfied. This important moment is lost if only the acoustic channel is considered. Meetings are truly multi-modal in nature [1], important information can be in a camera view that doesn't correspondent to the current speaker. This concept is followed by directors of TV talk-shows: They often show persons not currently talking. They wait for their reactions like gestures, or facial expressions. A virtual director has to take care of this as well. Thus selecting a camera is not a speaker diarisation, but a multi-modal task.

In this work we investigate different audio, visual, and lexical features for a virtual director. We introduce seven video modes and explain how they can be derived from the recordings in the smart meeting room. We'll discuss their advantages and when they are best shown in the meeting. We will then show how simple, yet very useful features for meeting analysis can be derived. While simple, they can be derived in real-time, which is important for online analysis. They are not only useful for a virtual director, but also for other kinds of meeting analysis (points of interest, individual and group actions). We will give measures how good the features are, and what the individual strengths, or weaknesses are.

2 Meeting Data

The meeting data for this work has been collected in the AMI and the M4 project [3]. The AMI project uses three different meeting rooms. In this work we concentrate only on the meetings recorded in the IDIAP smart meeting room. The room is equipped with various recording devices (as described below), a table, a whiteboard, and a projector with screen. A schematic of this room and three sample camera outputs are shown in Fig. 2. Each meeting has four participants (P1 - P4).

Close-talking audio is recorded with an omni-directional lapel and a headset condenser microphones for each participant (in Fig. 2 the microphones are indicated by black dots). Far-field recordings are performed with two microphone arrays: A1 is placed on the table in the middle of the participants and consists of eight miniature omni-directional electret microphones. The second array A2 with four microphones is mounted on the ceiling. Furthermore the room is equipped with a binaural manikin (BM) for two further recordings.

Video is recorded with seven cameras: four cameras record closeup views (1 - 4) of the meeting participants. Two cameras record a left (L), resp. right (R)

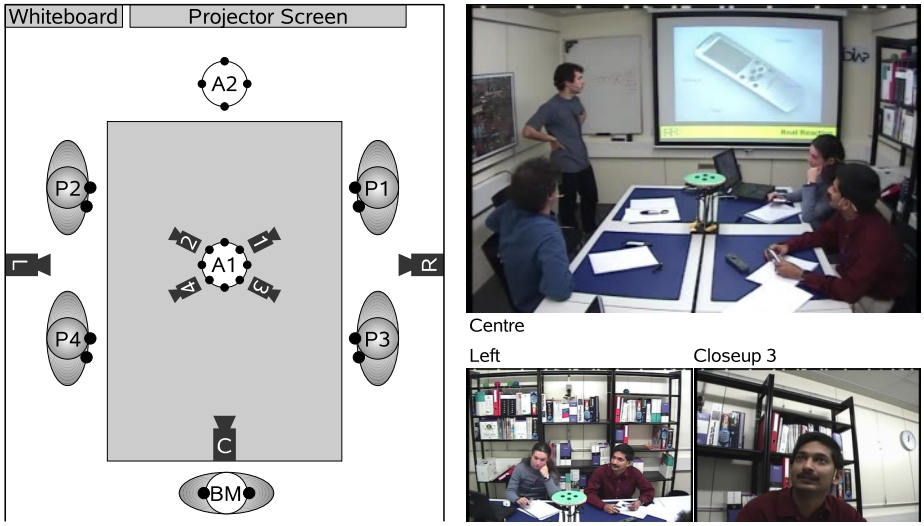

Fig. 1. Schematic of the AMI IDIAP smart meeting room (not drawn to scale).

view of the room; each showing two participants and the table in front of them. The last camera (C) captures a total of the room with all four participants, the table, as well as the whiteboard, and the projector screen.

The content of the projection board is recorded with time-stamps as a series of static images. Furthermore the whiteboard is captured as x-y-coordinates of the pen and individual notes with Logitech I/O digital pens.

All recordings in the room are time-synchronised with a central timecode. In this work we use the the lapel and headset microphones and all visual recordings from 41 videos with different lengths.

3 Video Modes

The task of a virtual meeting director is to select for each frame one camera or one view from the available cameras. In the case of a remote meeting, this view is then transmitted, for later browsing the selected view is stored. Based on the available seven cameras in the meeting room and the possible user requirement we defined seven different video modes. They are shown in Fig. 3 and shall be described shortly:

Mode 1 (P1-P4): Shows the closeup camera of one of the persons P1 - P4. This is the main mode when a person is talking or shows facial expressions.

Mode 2: Shows the left-camera view and thus the persons P1 and P3. This mode is ideal for a discussion between the two, or as a diversification if P1 or P3 talks (a stylistic device that human directors often use in talk shows).

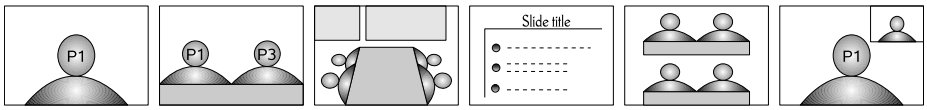

Fig. 2. Available video modes for the virtual director: from left to right 1, 2, 4, 5, 6, and 7. Mode 3 is omitted, as it is analogues to mode 2, but uses the right camera.

It can also be used if P1 or P3 talks, and the other one reacts in some way – e.g. a shaking of the head.

Mode 3: Shows the right-camera and thus the persons P2 and P4, it corresponds to mode 2 and has the same properties.

Mode 4: Shows a total of the room from the central camera. This total involves the whiteboard, the projection board, and all four participants. It is ideal if somebody gives a presentation, or to show group interactions. However the individual persons are rather small in this mode. Furthermore the persons are shown from the side, thus details get lost.

Mode 5: This mode inserts a still image (slides, pictures, etc.) into the video. It is ideal to show up the presentation slides when they are changed.

Mode 6: Shows both the output of the left and the right camera. They are slightly cut on top and the bottom, scaled down, and then merged on top of each other. This mode shows all participants in a frontal view and is therefore good for group discussions, note-taking, or group interactions. The individual persons are larger and better shown as in mode 4, but due to the adding up of two views smaller than in mode 1, 2, and 3. Thus individual reactions are less impressive. Furthermore the cutting contains the risk of cutting out heads or hands.

Mode 7 (P1-P4, P1-P4): Shows the closeup camera of one of the persons P1 - P4. A further closeup of another person is merged into the corner. This view can be used to show reactions of one participant, while another person is talking. However if the persons are sitting next to each other, mode 2 or 3 are preferred, as mode 7 is rather unnatural.

The presented modes are of course adapted to the conditions of the smart meeting room. However similar modes showing the same sets of group, or individual dynamics can easily be derived for other meeting room settings. The presented features in this work are not limited to the here presented modes and can in principle be adapted to different requirements or other meeting rooms.

4 Features

4.1 Visual Features

As a first visual feature we use global motions (GM). They have been successfully applied to various meeting tasks [13, 9] and can be calculated in real-time. We

first split the smart meeting room into six locations L. Each of the four closeup cameras represents one location. From the centre view camera we extract the projection board and the whiteboard location. Then a difference image sequence $I_d^L(x,y)$ is calculated for each of these six locations by subtracting the pixel values of two subsequent frames from the video stream. Then the seven global motion features are derived from the image sequence, again for each location. The centre of motion is calculated for the x- and y-direction according to:

$$m_x^L(t) = \frac{\sum_{(x,y)} x \cdot |I_d^L(x,y,t)|}{\sum_{(x,y)} |I_d^L(x,y,t)|} \quad \text{and} \quad m_y^L(t) = \frac{\sum_{(x,y)} y \cdot |I_d^L(x,y,t)|}{\sum_{(x,y)} |I_d^L(x,y,t)|}. \quad (1)$$

The changes in motion are used to express the dynamics of movements:

$$\Delta m_x^L(t) = m_x^L(t) - m_x^L(t-1) \quad \text{and} \quad \Delta m_y^L(t) = m_y^L(t) - m_y^L(t-1). \quad (2)$$

Furthermore the mean absolute deviation of the pixels relative to the centre of motion is computed:

$$\sigma_x^L(t) = \frac{\sum_{(x,y)} |I_d^L(x,y,t)| \cdot \left(x - m_x^L(t)\right)}{\sum_{(x,y)} |I_d^L(x,y,t)|}$$

and

$$\sigma_y^L(t) = \frac{\sum_{(x,y)} |I_d^L(x,y,t)| \cdot \left(y - m_y^L(t)\right)}{\sum_{(x,y)} |I_d^L(x,y,t)|}. \quad (3)$$

Finally the intensity of motion is calculated from the average absolute value of the motion distribution:

$$i^L(t) = \frac{\sum_{(x,y)} |I_d^L(x,y,t)|}{\sum_x \sum_y 1}. \quad (4)$$

These seven features are concatenated for each time step in the location dependent motion vector

$$\boldsymbol{x}^L(t) = [m_x^L, m_y^L, \Delta m_x^L, \Delta m_y^L, \sigma_x^L, \sigma_y^L, i^L]^T. \quad (5)$$

With this motion vector the high dimensional video stream is reduced to a seven dimensional vector, but it preserves the major characteristics of the currently observed motion. Concatenating the motion vectors from each of the six positions $\boldsymbol{x}^L(t)$ leads to the final motion vector $\boldsymbol{x}_V(t) = [\boldsymbol{x}^{C_1}, \boldsymbol{x}^{C_2}, \boldsymbol{x}^{C_3}, \boldsymbol{x}^{C_4}, \boldsymbol{x}^W, \boldsymbol{x}^P]^T$, that describes the overall motion in the meeting room with 42 features.

4.2 Head and Hand Blobs

While the global motion features of the six locations in the meeting room provide a fast and simple access to the location dependent activity, they only reflect the participants' motions in a very compressed, summarised way. A better way to

access the individual participants activities are hand and head movements. In [5] it was shown how hand and head skin blobs can be used to detect the activity of individual meeting participants. We therefore add skin blobs as a visual feature.

In this work we extract the head and hand skin blobs with a simple - yet in the context of recorded meetings successful - skin colour look up table approach[11]. First the RGB-images are transformed into the rg-space

$$r = \frac{R}{R+G+B}, \quad g = \frac{G}{R+G+B}, \quad b = \frac{B}{R+G+B}, \tag{6}$$

which is less crucial to light changes. Each image pixel is then compared to a 16 bit rg-look up table. This table has been learned from 5.7 Million skin colour pixels from different non-meeting images. The comparison with the skin colour table results in a binary image, where each possible skin pixel is marked. Then a 5x5 dilation filter [6] is applied to the binary image. This filtering leads to an extension of skin pixel areas. Especially gaps in large skin pixel areas are filled with this filter. The found skin pixel areas are then analysed for their shape and for the relation of their eigenvalues. Only blobs that are large enough for a face, resp. hands, are selected; these selected areas are further narrowed down by the relation between the blob edges. Furthermore we apply context knowledge about the usual position of heads and hands of meeting participants. This already leads to a quite stable finding of heads and hands in the meeting videos. Then subsequent images are averaged by a recursive approach, that is applied individually to all hand and head blobs in the meeting videos:

$$\boldsymbol{m}(t) = 1 - \frac{1}{T}\boldsymbol{m}(t-1) + \frac{1}{T}\boldsymbol{x}(t), \tag{7}$$

where $\boldsymbol{x}(t)$ is the current measured value, $\boldsymbol{m}(t)$ is the resulting averages vector for the blob position, $\boldsymbol{m}(t-1)$ the position in the last image, and T a constant that determines the relation between previous frames and the current measurement.

This approach is simple, yet fast and for the target application reliable enough. Of course it would be better to apply a face and hand tracking. Once these systems are available in the AMI project we will exchange this module with advanced tracking methods for meeting scenarios, as e.g. compared in [7]. This however only improves the accuracy of the coordinates, the principles suggested in this work and the influence of this feature to virtual editing will not be influenced.

4.3 Acoustic Features

Beside the visual features we derive a range of simple acoustic meeting features. In this work we use the information from each of the four participants lapel microphones. At first we use a frame energy derived directly from the audio stream from each microphone. While simple, this approach is very fast and gives at least a cue for a speaker activity detection.

As second acoustic feature we perform a short time fast Fourier transform of the microphone channels. We then filter this short time spectrum with a

band-path filter to extract only regions relevant for speech. The filtered values are then summed up, resulting in a "frequency frame energy" that highlights the spoken aspects in the microphone channels. Again, this approach is simple and can easily be performed in real-time. In the future we plan to introduce a further speaker diarisation module from the AMI project [2], which should better discriminate between speech and non-speech regions. However the drawback of these methods is that they often can not be applied in real time. The frequency frame energy method is therefore required for remote meetings, where real-time processing is necessary.

As the third acoustic feature we derive 13 Mel frequency cepstral coefficients, and their first and second derivations. They will later be used in the statistic fusion process, but are not applied in this work. In the future we also plan to use the position dependent microphone array information.

4.4 Lexical Features

We also use higher semantic information as input to the virtual meeting director. Group actions in meetings have been deeply investigated [1]. We revert to the eight well-known group action classes:

Monologue (person 1-4): One person speaks without being interrupted.
Discussion: Two or more persons talk alternately.
Presentation: One person gives a presentation in front of the projection board.
Whiteboard: One person writes on the whiteboard and talks.
Note-taking: (All) persons write something down.

Sometimes the discussion class is further split into disagreement and consensus to reflect the kind of discussion. This is however not required for a virtual director. To better model the interaction between the group and individuals, it has also been suggested to combine different meeting group actions [12], note-taking with either monologue, presentation, or whiteboard. This combinations will not be used in this work.

The meeting group actions have two main advantages, making them very well suited for automatic video editing: they can very reliable detected from the raw-audio visual stream (see [1] for a comparison of various automatic recognition models). And compared to other semantic information (like dialogue acts), the meeting group actions can easily be mapped to video modes, e.g. a monologue of a participant to a camera view of this particular person. There is one drawback of this feature: while in principle possible, the automatic recognition models do usually not work in real-time. The use of the feature for remote meetings is therefore currently not possible. However for an offline editing and later browsing, they can be applied. And with emerging computer power, and the models at hand, online detection of meeting group actions will become possible in the future.

5 Feature Scopes

In the last section we showed a range of features that can be applied to virtual meeting editing. We have derived these features on a per frame basis. If this output is directly used as an input to a virtual director, this can lead to unintentional twitches. We therefore also evaluate the influence of windowing a range of subsequent frames for the audio and the visual features. We apply three different window functions, where each function represents a different feature scope: a history oriented, a history-future balanced, and a future oriented approach.

In the following let T denote the total length of the meeting, t the current time step, $P \in \{P_1, P_2, P_3, P_4\}$ one of the meeting participants, W the window size (with increasing W the scope of the features is increased), and $F^P(t)$ be the addressed feature for person P at time t. Then the windowed output of the feature is denoted as $D^P(t)$. The *history scope* sums up features from the history:

$$D_h^P(t) = \sum_{\tau=0}^{t} F^P(\tau) \qquad \text{for } t < W, \tag{8}$$

$$D_h^P(t) = \sum_{\tau=t-W}^{t} F^P(\tau) \qquad \text{for } t >= W. \tag{9}$$

The windowed output $D_h^P(t)$ therefore represents what has recently happened. It does not reflect what will happen in the near future. In contradiction, the *balanced scope* sums up features from the history and the near future:

$$D_b^P(t) = \sum_{\tau=0}^{t+W/2} F^P(\tau) \qquad \text{for } t < W/2, \tag{10}$$

$$D_b^P(t) = \sum_{\tau=t-W/2}^{t+W/2} F^P(\tau) \qquad \text{for } W/2 <= t <= T - W/2, \tag{11}$$

$$D_b^P(t) = \sum_{\tau=t-W/2}^{T} F^P(\tau) \qquad \text{for } t > T - W/2. \tag{12}$$

The windowed output $D_b^P(t)$ represents both what has recently happened and what will happen in the near future. Finally the *future scope* sums up features from the future only:

$$D_f^P(t) = \sum_{\tau=t}^{t+W} F^P(\tau) \qquad \text{for } t <= T - W, \tag{13}$$

$$D_f^P(t) = \sum_{\tau=t}^{T} F^P(\tau) \qquad \text{for } t > T - W. \tag{14}$$

The windowed output $D_f^P(t)$ therefore represents only what will happen in the near future, without including past actions.

Only the history oriented approach is causal and can therefore be applied to online processing. The other two concepts can only be applied offline. However in the experimental section we will show that they are indeed useful for browsing.

6 Video Mode Decision

The scope of this work is to investigate the influence of the different features to a virtual meeting director. We therefore do not apply statistical or other machine learning methods for the actual video mode decision. For the acoustic and the visual features we calculate the windowed output $D^P(t)$ for each time and each person. We then choose the "most active" person with

$$V(t) = \underset{P}{\operatorname{argmax}}\, D^P(t). \tag{15}$$

Depending on the desired output this decision $V(t)$ can now be mapped directly to one of the seven video modes (e.g. an activity of person $P2$ to mode 1). For the semantic group action features a decision function is not required, they can directly be mapped to a video mode (e.g. a discussion to mode 2).

This process of course does not optimise the features, nor does it model interactions between the features. This way the influence of the features to the video output stream is not diluted and can therefore best be evaluated. In future the features and the evaluation experience can then be used for statistical models (like HMMs).

7 Experiments

We performed three sets of experiments: In the first we evaluate the single features, both on a per frame basis and windowed. In a second experiment we perform a simple fusion scheme. In the last experiment we investigate the influence of the three different window scopes history, balanced, and future.

The output of a virtual director can not be trivially evaluated with "objective measures". Indeed directing is depending on taste, the difference between the cut of two human directors can be huge. However there are some measures, that indicate the quality of the feature. As a first evaluation measure we use the percentage similarity to the output of the meeting group actions. That is, we compare the output of the audio and visual features with the output of the meeting group action features. As the meeting group action features model the interactions between the participants, this is a meaningful number, how strong the feature represents interactions. However, a high value does not automatically correspond to a good visual stream (or this work would be reduced to the problem of meeting group action recognition). As a second value we use the average frequency of mode changes. This value indicates the number of cuts in the stream and thus corresponds to how fast the system changes to activities in other channels. We also evaluate the maximum segment length per meeting: this

Table 1. Evaluation results for the lexical, acoustic, and visual features: Similarity to the meeting group actions, average mode changes per minute, and maximum length of a sequence without mode changes in seconds. Standard deviations are given in brackets.

Feature	*Group Similar.*	*Changes/Min*	*Max. length/s*	*Score*
Group Actions	100.0 (0.0)	1.1 (0.3)	116.0 (31.3)	4
Audio (per Frame)	21.7 (13.8)	373.3 (83.4)	6.1 (4.3)	0
Audio (History)	25.0 (15.7)	18.7 (8.3)	83.0 (37.0)	5
Frequency (per Frame)	20.5 (12.9)	435.7 (78.0)	3.6 (1.2)	0
Frequency (History)	25.0 (15.7)	16.5 (8.1)	81.6 (37.2)	6
Global Motion (per Frame)	20.8 (9.3)	205.5 (42.9)	7.9 (5.2)	0
Global Motion (History)	19.6 (10.2)	32.1 (5.3)	19.8 (13.8)	4
Skin Blobs (per Frame)	13.5 (6.1)	935.4 (85.6)	0.9 (0.3)	0
Skin Blobs (History)	14.3 (10.4)	63.8 (18.6)	16.1 (9.9)	2

value shows the longest time period without a cut. A very low number indicates that the system is not staying in one view for a sufficient period. On the other hand a very long segment can be rather boring to watch.

Finally we give a user rating score between zero and ten (where ten is the best) for each feature. While this is not an objective measure, it expresses the significance of the individual features from a user's point of view.

7.1 Feature Results

We first evaluated the influence of the features to the virtual editing. The results are presented in Tab. 1. Each audio and visual feature was both evaluated on a per frame and with history basis. The group actions have a very low change frequency. Once a speaker gives a monologue, they do not change at all, thus information in other channels can get lost. However they are very stable to watch. On the other hand all "per frame" features are – if used directly – not watchable at all, they twitch heavily between the modes. In the case of the blobs a mode change happens in average in every second frame, the longest segment has duration of less than a second. Such a cut of a meeting can't be watched.

These results get much better if the history is taken into account, according to (8) and (9). Then the frequency of mode changes reduces for all four features to reasonable numbers, e.g. for the audio feature to 18.7 changes per minute. In general it can be seen that the two visual features result in more changes than the acoustic features. Especially the blobs still have a very high mode change.

In a second experiment we applied a very simple late fusion scheme, where the single feature video modes are merged with a confidence multiplication. The results after fusion for different feature combinations are shown in Tab. 2. It can be seen, that this simple late fusion scheme doesn't lead to better video outputs. The high frequency of mode changes remains unchanged. This directly calls for advanced statistical methods to benefit from the information in all channels, but on the other hand reduce the high mode changes.

Table 2. Evaluation results after late fusion of different feature combinations. If not otherwise indicated, the methods are with history and not on a per frame basis.

Feature	*Group Similar.*	*Changes/Min*	*Max. length/s*	*Score*
All (per Frame)	20.6 (10.9)	641.5 (112.7)	3.2 (1.8)	0
All (History)	25.6 (14.2)	26.6 (9.2)	51.2 (22.1)	5
Audio + Frequency	25.0 (15.7)	16.5 (8.1)	81.6 (37.2)	6
Global Motion + Skin Blobs	14.3 (10.4)	63.8 (18.6)	16.1 (9.9)	2
Global Motion + Audio	26.1 (13.9)	22.2 (6.5)	48.4 (19.9)	4
Skin Blobs + Audio	23.7 (14.0)	38.3 (13.0)	41.6 (26.1)	4

7.2 Feature Scope Evaluation

In the last experiment we investigated the influence of the three different window functions (8) - (14). The *history scope* is the only causal scheme, and therefore the only one that can be applied to a remote meeting, as this requires online processing. The drawback of the history scope is it's lack to react to actions in the future. If, e.g. a participant shakes his head, the history scope can only react after the shaking has started. A human director can of course neither know what will happen in the next seconds, but a human director can use intuition and from time show other participants. Therefore if the history scope is used for online processing a virtual director has to model some "intuition" and from time to time randomly switch to other channels.

In our evaluations the *balanced scope* gave the best results. We suggest to use it for offline meeting editing. It both reacts on past activities, and switches fast enough to observe the start of reactions. The pure *future scope* is interesting to watch, as all user reactions are fully captured. However from time to time the system tends to switch to fast to a different participant and then nothing happens for a while. Thus the balanced scope is preferred.

8 Conclusion

In this work we introduced five features for a virtual meeting director. The presented features are all very simple, but useful and - especially important for remote meetings - can in principle be processed online in real-time. We deeply investigated the influence of the different features on the resulting video stream. We showed that acoustic features alone are not sufficient for this task, but visual features are required. We also investigated a simple late fusion scheme and the influence of different window scopes.

We now have a measurement where the strength and weakness of the individual features are. In the future we plan to model the group and individual interactions with statistical models. This should result in a much more advanced and better virtual meeting director. The features and evaluation results from this work will be of valuable input to such a system.

References

[1] M. Al-Hames, A. Dielmann, D. Gatica-Perez, S. Reiter, S. Renals, G. Rigoll, and D. Zhang. Multimodal integration for meeting group action segmentation and recognition. In S. Renals and S. Bengio, editors, *MLMI 2005, 2nd Joint Workshop on Multimodal Interaction and Related Machine Learning Algorithms*. Springer Verlag, 2006.

[2] M. Al-Hames, T. Hain, J. Cernocky, S. Schreiber, M. Poel, R. Muller, S. Marcel, D. van Leeuwen, J.M. Odobez, S. Ba, H. Bourlard, F. Cardinaux, D. Gatica-Perez, A. Janin, P. Motlicek, S. Reiter, S. Renals, J. van Rest, R. Rienks, G. Rigoll, K. Smith, A. Thean, and P. Zemcik. Audio-visual processing in meetings: Seven questions and current AMI answers. In *MLMI 2006, 3rd Joint Workshop on Multimodal Interaction and Related Machine Learning Algorithms*. Springer Verlag, 2006.

[3] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska, I. McCowan, W. Post, D. Reidsma, and P. Wellner. The AMI meetings corpus. In *Proceedings of the Measuring Behavior 2005 symposium on Annotating and measuring Meeting Behavior*, 2005.

[4] A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters. The ICSI meeting corpus. In *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2003.

[5] I. Potucek, S. Sumec, and M. Spanel. Participant activity detection by hands and face movement tracking in the meeting room. In *Proceedings IEEE Computer Graphics International (CGI)*, pages 632–635, 2004.

[6] W.K. Pratt. *Digital image processing*. John Wiley & Sons, 2001.

[7] K. Smith, S. Schreiber, V. Beran, I. Potúcek, and D. Gatica-Perez. A comparitive study of head tracking methods. In *MLMI 2006, 3rd Joint Workshop on Multimodal Interaction and Related Machine Learning Algorithms*. Springer Verlag, 2006.

[8] A. Waibel, H. Steusloff, R. Stiefelhagen, and the CHIL Project Consortium. CHIL: Computers in the human interaction loop. In *Proceedings of the NIST ICASSP Meeting Recognition Workshop*, 2004.

[9] F. Wallhoff, M. Zobl, and G. Rigoll. Action segmentation and recognition in meeting room scenarios. In *Proceedings IEEE International Conference on Image Processing (ICIP)*, Singapore, October 2004.

[10] P. Wellner, M. Flynn, and M. Guillemot. Browsing recorded meetings with Ferret. In S. Bengio and H. Bourlard, editors, *Machine Learning for Multimodal Interaction: First International Workshop, MLMI 2004*. Springer Verlag, 2005.

[11] M.-H. Yang, D.J. Kriegman, and N. Ahuja. Detecting faces in images: A survey. *IEEE Transasctions on Pattern Analysis and Machine Intelligence*, 24(1):34–58, 2002.

[12] D. Zhang, D. Gatica-Perez, S. Bengio, I. McCowan, and G. Lathoud. Modeling individual and group actions in meetings: a two-layer hmm framework. In *Proceedings IEEE Workshop on Event Mining at the Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004.

[13] M. Zobl, F. Wallhoff, and G. Rigoll. Action recognition in meeting scenarios using global motion features. In J. Ferryman, editor, *Proceedings Fourth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS-ICVS)*, pages 32–36, 2003.