TECHNISCHE UNIVERSITÄT MÜNCHEN
Lehrstuhl für Mensch-Maschine-Kommunikation

Audio-Visual Event Recognition with Graphical Models

Benedikt Hörnler

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender:		Univ.-Prof. Dr.-Ing. K. Diepold
Prüfer der Dissertation:	1.	Univ.-Prof. Dr.-Ing. habil. G. Rigoll
	2.	Univ.-Prof. Dr.rer.nat. M. Kranz

Die Dissertation wurde am 15.09.2010 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 10.12.2010 angenommen.

Abstract

In this work, different applications for the automated detection of events have been investigated utilizing audio-visual pattern recognition methods. The recorded data has been taken both from video surveillance or video conferences. Acoustic, visual and semantic features are extracted from the available data and are subsequently analysed with the help of graphical models. These are particularly suitable for modeling multi-modal feature sequences and provide an efficient way for automatic feature fusion. All models are first described in detail theoretically and then the necessary structure for both the learning of required parameters and the classification process are presented. Finally a conclusion is drawn by describing the results and further possible research approaches. Graphical models are suitable for these tasks, but the results are strongly depending on the kind of problem.

Zusammenfassung

In dieser Arbeit wurden unterschiedliche Aufgabenstellungen zur Erkennung von Events in Aufzeichnungen aus Videoüberwachung oder Videokonferenzen mit Hilfe von audio-visuellen Mustererkennungsverfahren analysiert. Aus den vorliegenden Daten werden hierfür akustische, visuelle und semantische Merkmale extrahiert und mit Hilfe von Graphischen Modellen verarbeitet. Diese eignen sich besonders für die Modellierung von multimodalen Merkmalssequenzen und bieten eine effiziente Möglichkeit für die automatische Datenfusion. Alle Modelle werden zunächst ausführlich theoretisch beschrieben und anschließend werden die notwendigen Strukturen für das Lernen der benötigten Parameter und die Erkennung dargestellt. Abschließend werden die Ergebnisse und weitere mögliche Forschungsansätze präsentiert. Graphische Modelle eignen sich für die vorliegende Aufgabenstellung, allerdings hängen die Ergebnisse relativ stark von der Art der Aufgabe ab.

Contents

1 Introduction

Many unsolved problems exist in real applications in the field of information processing. For some of them it is possible to find a description which can be transformed into a pattern recognition problem. Here, modern and popular pattern recognition methods, as for example support vector machines or graphical models, can be applied to solve the problem or at least come up with a solution which is useful for the mankind. These solutions help people by running their daily business more efficiently, for example observing public areas or selling something.

In this thesis, graphical models (GM) are used to come up with solutions. A GM describes a complex problem very intuitively [Bil06] and to calculate the outcome of the model often less computational effort is needed, as when the complex problem is solved directly [JLO90]. Moreover, a GM is adapted to the problem and not the other way around, as it happens with many other approaches, for example support vector machines, where no adaptation is possible. For this reasons, GMs are commonly used for the description of problems and for rapid-prototyping of solutions.

A GM, as it is used in this thesis, consists of vertices and directed edges. A directed edge describes a relation between two vertices. Many algorithms are described in the graph theory [BM76], which for example shows if two vertices are depending on each other or if exchanging of information between two sub graphs is possible. All these allow fast and efficient calculations performed on the graphs. Since the vertices and edges in a GM applied to pattern recognition problems describe random variables and dependences between them, not only the graph theory is important, but also the probability theory [Jor01]. Therefore, GMs are a combination of graph- and probability theory with benefits from both sides. Easy calculations are performed directly via probability theory, but if it is getting more complex the graph theory is used to reduce the complexity first. As mentioned above it often happens that the calculations are getting more efficient, if they are performed on the graph than direct calculation of the probability.

GMs are applied in three different domains in this thesis: surveillance, meeting analysis and interest detection. The surveillance scenario contains a setting, where passengers of an aircraft are analysed, if they are acting normal or suspicious. For the meeting analysis, two tasks are chosen to evaluate: automatic video editing and activity recognition. Activity recognition detects the activity and dominance of each participant during the meeting. The most relevant view of the ongoing meeting is automatically selected by the video editing system. The interest detection scenario is about the interaction of a subject with an intelligent machine. This machine recognises the level of interest and accordingly selects the topic.

All the described problems have access to multi-modal recordings of the scenarios, therefore it is only the consequent next step to evaluate multi-modal features for all approaches. Most GMs developed and evaluated are using multi-modal feature as an input, but some do not. They are used to see if an improvement of the performance exists between single- and multi-modality. The use of different modalities at the same time has shown its performance in various works in the field of human-machine-communication [KPI04, PR03, CDCT+01]. Furthermore, the robustness and reliability of the models are improved by multi-modal data [PR03, SAR+07, WSA+08].

Starting with the results of the analysis of the chosen domains, it has been possible to formulate pattern recognition problems. Various graphical models have been developed theoretically and the needed equations for the calculations have been derived to solve these problems. Afterwards, the models have been trained and evaluated on different data sets. In the following paragraphs, an outline of the chapters which are presented is given:

Chapter 2 describes all the used data sets. It contains a data set recorded at the Technische Universität München (TUM) which simulates an aircraft scenario. In this setting, the behaviour of passengers is analysed, which means the current status in terms of aggressiveness and threat is detected. The second data set is from the AMI project [CAB+06] and contains recorded meetings with four participants. The last corpus was once again recorded at TUM and is used for the analysis of a dialogue, especially for the emotions of the subject. For all these data sets the annotations are described and statistics about the distributions of the classes are given.

Chapter 3 presents information about the extracted features from the data sets. The work uses multi-modal features, which are derived from the acoustic, visual and semantic domain. From the acoustic recordings, low level descriptors and according functionals of short segments are extracted. Two different visual features are derived: skin blobs and global motions. Both are calculated from all available camera views. Furthermore, the subject's face is detected and tracked. The last group of features is the semantic information which exists only for the meeting corpus. It contains data about what the group is doing, what the participants are doing and where they are moving, who is speaking and when a foil is changed during a presentation. Overall more than 3000 features are extracted, thus it is necessary to reduce the feature space. This is done by principal component analysis or sequential forward selection.

Chapter 4 contains a short introduction to graphical models and gives a nomenclature for all the models following in all the other chapters. The introduction is kept short and provides a brief overview, as a wide range of books and tutorials are available which go into more detail. Moreover, all the important model structures of the this thesis are described in detail.

Chapter 5 presents a surveillance system which is based on graphical models. The passengers of an aircraft are audio-visually captured and the recordings are analysed. Various model structures are developed and evaluated. It contains models which automatically segment the data and hierarchical models which classify the features in two steps in order to achieve better results. The chapter is closed with a comparison of the achieved results with performances found in other works.

Chapter 6 discusses models which can learn segment boundaries from training data. This means, not only the sequence of classes, as it is common for automatic speech recognition, is learned but the point of time of a class boundary. This is important for video editing, since a cut between two perspectives has to take place at the correct time frame, otherwise information would be lost or the output video is very disturbing for the watcher. Again in the end the results are compared and an outlook is given.

Chapter 7 is about the activity and dominance recognition in meetings. For the first time, it is evaluated if low level descriptors are applicable for the detection of activity levels during a meeting. Graphical models are used for the classification of the low level features, which are capable of segmenting the meeting into short segments of the same level of activity. Moreover, related work is presented and the results are compared.

Chapter 8 shows an approach for the combination of support vector machines and graphical models to perform a detection of the level of interest during a human-machine dialogue. Support vector machines are used for a classification of each frame on feature level. Functionals of the acoustic features used for that are extracted from a window of 25 frames and therefore should contain information about the development of a feature, which is very important for a good performance. The performance achieved by this approach is evaluated against previous work and a discussion about it is given.

Overall the work describes and discusses different applications of pattern recognition methods in the field of human-machine-communication which can be implemented as a graphical model. All the developed models are theoretically analysed and practically evaluated. Furthermore, different modalities, such as acoustic, visual and semantic, and various types of features are evaluated. For all the problems, related work is presented and the results are discussed.

2 The Used Data Sets

In this chapter the three data sets which are used for this thesis are described. The first one is a subset of the AMI corpus, the second is known as the Aircraft Behaviour Corpus and the third is the Audiovisual Interest Corpus.

2.1 Meeting corpus

The meeting data for this work has been collected within the AMI project [CAB^{+}06] and is publicly available[1]. The recording of the AMI corpus took place in three different smart meeting rooms and the total duration of freely available recordings is approximately 100 hours. The meeting rooms were located at the IDIAP Research Institute in Switzerland, at the University of Edinburgh in Scotland, and at University of Twente in the Netherlands. The setting of each room is slightly different from each other but all meetings are discussing the development of a new remote control. Four participants (Per.1 - Per.4) are attending the meeting in a single room and they are for example discussing, giving presentations, taking notes, or writing on the whiteboard. The same four subjects are participating in four meetings and each of them has a predefined role, such as being the project manager, the industrial designer, the interface designer or the marketing expert. Due to the fact that each subject is participating in four meetings and is not attending any other meeting, it is possible to create subject disjoint training- and test-sets for a cross-validation.

In this thesis, a subset of the data form the IDIAP smart meeting room [Moo02] is used. Hence the cameras in this room are mostly facing the faces of the participants. Moreover, the hands are visible as well, thus it is possible to extract information from them. In figure 2.1 the schematic of the IDIAP smart meeting room is shown. The room is equipped with a table, a whiteboard, a projector with a screen and various different recording devices. A device captures for example the (x, y)-coordinates and the pressure from the pen which is used on the whiteboard and four Logitech I/O

[1] `http://www.amiproject.org`

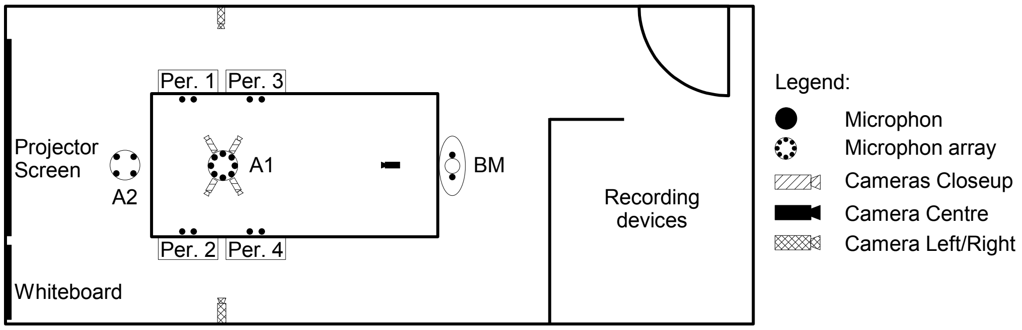


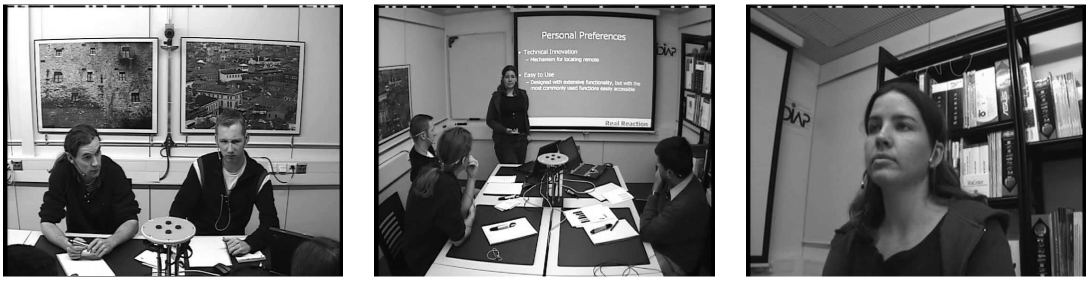
Figure 2.1: Schematic of the IDIAP smart meeting room. 22 microphones and seven cameras are installed. Each participant wears two close-talking microphones and additionally two microphone arrays are located in the room. Four close-up cameras, a centre camera and one camera at each of the long sides of the room are capturing the ongoing meeting.

digital pen devices collect the notes from each participant. In order to create time synchronized recordings, it is necessary to add timestamps to each captured frame in each data stream. Therefore, it is later possible to align all the streams for the multi-modal analysis which is especially necessary for low level feature fusion.

The duration of a meeting in the AMI corpus is between 30 minutes and one hour. The subset used in this thesis consists of 36 five-minute meetings, recorded at the IDIAP smart meeting room, and contains audio and video streams. Each chunk is taken from different parts of different meetings and is not overlapping, thus the data is disjoint. In total the subset has a length of 180 minutes.

2.1.1 Audio recordings

The IDIAP smart meeting room, shown in figure 2.1, is equipped with 22 microphones. Far field recordings are performed by two microphone arrays and a binaural manikin. One, containing eight microphones, is placed in the middle of the table (A1) and the second is mounted between the table and the projector screen on the ceiling (A2) with four microphones. The binaural manikin (BM) is placed at the end of the table for two further audio recordings by replicating a human head and ears. Furthermore, two different types of close-talking microphones, an omni-directional lapel microphone and a headset condenser microphone, have been worn by every participant for the recordings. The lapel microphone has been attached to the user's collar. Only the four headset condenser microphones are used for the feature extraction, which is described in section 3.1 and a mixed signal of them is streamed into the created output videos.



(a) Camera Left (b) Camera Centre (c) Camera Close-up

Figure 2.2: Samples output of three available cameras from the IDIAP smart meeting room.

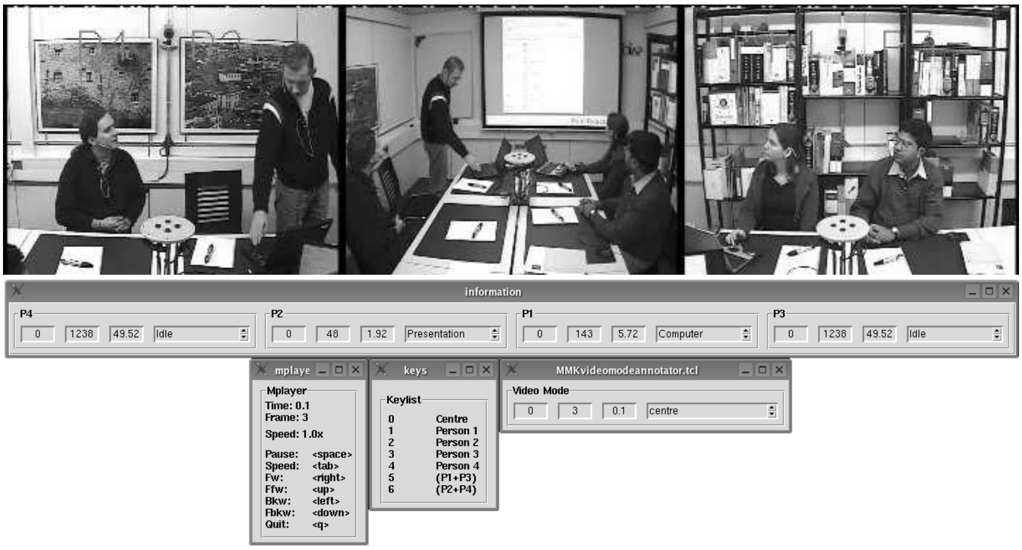
2.1.2 Video recordings

Seven cameras are located in the room. Four close-up cameras record the faces of the meeting participants, if they are sitting at the table. Two cameras (left resp. right camera) are mounted on each of the sides of the room. They capture the table and two participants of the opposite side. These cameras can be used for hand tracking, because the hands are visible in case the persons are located at the side of the table. An overview of the room, which contains the table, the whiteboard, the projector screen, and all participants, records the centre camera. Figure 2.2 shows three example views from cameras which have been recorded in one of the meetings. All available camera views are used for the video editing task in section 6. During the activity detection task, described in section 7, features are only extracted from the close-up cameras.

2.1.3 Annotation

This subset has been annotated for various applications and pattern recognition tasks. Some annotations are publicly available with the AMI corpus, for example the person movements. These movements describe what the person is doing in conjunction with her location in the smart meeting room. Further annotations have been conducted through out this work and are described in more detail here. Annotations can be done via tools like ANVIL [Kip01]. In the case of this work, the annotations, which are needed, are performed by applying tools especially created for the specific annotation.

For example the tool used for the video editing annotation is shown in figure 2.3. This tool consists of one window which displays a video and four other windows are presenting information about the annotation process to the labeler. One window takes the annotation of the person actions and displays the current status of the four participants. The *mplayer* window shows the hot keys which are used for



P4: 0 | 1238 | 49.52 | Idle

P2: 0 | 48 | 1.92 | Presentation

P1: 0 | 143 | 5.72 | Computer

P3: 0 | 1238 | 49.52 | Idle

Mplayer
Time: 0.1
Frame: 3
Speed: 1.0x

Pause:	<space>
Speed:	<tab>
Fw:	<right>
Ffw:	<up>
Bkw:	<left>
Fbkw:	<down>
Quit:	<q>

Keylist

0	Centre
1	Person 1
2	Person 2
3	Person 3
4	Person 4
5	(P1+P3)
6	(P2+P4)

Video Mode: 0 | 3 | 0.1 | centre

Figure 2.3: A screen shot of the annotation tool which is used during the annotation task for the video editing system. The tool consists of five different windows. One which shows a combination of the three cameras centre, left and right. Two windows display information about the mplayer, which is used for playing the video including the audio recordings, and the available labels for the annotation. The information window presents the current status for each of the participants, which is taken from the annotation of the person actions. The last windows displays the information about the position within the video and the selected video mode.

controlling of the video and audio player. The window with the label *keys* displays all available labels for the annotation and the according hot key for selection of the label. The last window gives the user feedback about the annotation which has been performed and shows information about the start and end frame of the current selected label. The tools for the other labeling tasks are similar with some changes since other annotations are used for the information bar or the labels which are available are different. The tool is easy to use, since the user only needs to press the button according to the label, which should be selected, once an action occurs.

2.1.3.1 Video Editing

For the annotation of the meeting corpus for the video editing task video modes are defined. This is necessary because several views can be projected out of each available camera. A video mode is the selected view for a single frame. This view can be a camera view, a selected regions of a camera view, an image or a picture, slides or text in the case a description of somebody or something is needed. Moreover, combinations of cameras or pictures are possible. The video mode is always selected for a single frame and therefore it can be changed each frame. These video modes can be stored to a database for a later browsing of the meeting, or in the case of a video conference only the stream of the chosen video mode can be transmitted to the remote participants. Restricted to the recordings from the smart meeting room, the possible definitions of video modes are all available cameras, subregion of camera views, and combinations of cameras and/or subregions. A set of video modes has been defined, based on the user requirements, the available views and additional information, which are recorded and stored with the meeting data. The currently used video modes are shown in figure 2.4 and shall be described shortly:

Video modes 1 - 4 show the close-up camera of one of the four participants Person 1 - Person 4. These modes are used when a person is talking, shaking its head or nodding. The modes 1 - 4 are normally used most frequently during a meeting, because the participants are talking most of the time.

Video mode 5 shows the left camera view and thus Person 1 and Person 3. This mode is perfectly suitable for a discussion of these two persons or as an additional view if Person 1 or Person 3 are talking for a long time and it is necessary to switch to a different shot as video mode 1 or 3. This is used for example if Person 1 or Person 3 talks for a long time and no other actions are worth to make a cut of and to show it. It may also be applied if Person 1 or Person 3 are talking and the other participant on the same side shows a facial expression. Moreover, it shows when Person 1 or Person 3 are pointing at something, are handling an item such as a prototype of a remote control, or are taking notes.

Video mode 6 presents the right camera and shows Person 2 and Person 4. The camera is used in the same way as in video mode 5 and has the same properties.

Video mode 7 shows the total view of the meeting room from the position of the centre camera. The view contains the projector screen, the whiteboard, the table, and all four participants. Therefore, this video mode can be applied, for example for group discussions, showing an item, or the giving a presentation or writing on the whiteboard. However, the participants are too small in the view to recognise important facial expressions and often the persons are looking at the projector screen, thus no face is visible in this perspective.

Video mode 8 inserts a defined image which is annotated into the video output instead of a video frame taken from a camera. Therefore, this mode is ideal if a slide change occurs during a presentation. Furthermore, it is very suitable for adding relevant information at the beginning of a video, such as the recording date or names of the participants.

Video mode 9 is a combination of video modes 5 and 6. A predefined region of the left camera view is shown at the top of a cutout of the right camera, so that

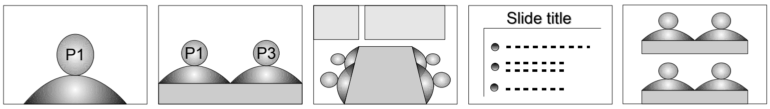


Figure 2.4: Different video modes for the video editing system: from left to right video mode 1, 5, 7, 8, and 9 are presented in an abstract way. The modes 2 to 4, and 6 are left out because the are similar to 1, respectively to 5.

all participants are shown from the front. Therefore, this mode can be used for group discussions, note-taking and group interactions. On the other hand the selecting of predefined regions in the camera streams contains some risk of cutting out interesting parts of the body, for example head or arms.

Additional video modes can be created very easily. For example, a picture in picture mode, as known from news shows on several TV-channels, can be added. This mode is one where the speaker is presented in the front and a second person, who is interacting with him, is placed as a small portrait in the upper right corner. It is also possible to create additional video modes similar to video mode 5 or 6 by combining predefined regions from the close-up cameras from participant Person 1 or Person 3 with Person 2 or Person 4, which can be used for a discussion between to persons seated at opposite sides of the table. A problem with this video mode is, that it causes confusion, because participants are suddenly located next to each other, which are seated on different sides of the table.

For the pattern recognition approach, which is pursued in this thesis, it is necessary to know which video mode should be shown at each frame. Therefore, a data set of three hours has been annotated in order to train pattern recognition models on a training set and to evaluate the results on a predefined test set. For the annotation task, a small set of rules has been defined which contains some basic guidelines for creating a good, watchable movie [Bel05]. Most important for a good video is the duration of a shot. Therefore, the rules prevent the annotators from switching too fast or too slow. For example the close-up has to be shown for at least two seconds and the longest shot should be no more than 20 seconds. The time is different for each of the video modes, depending on how much information is visible in the scene. Another important issue is, that an establishing shot should be added at the beginning of the meeting and when a new scene starts. For example this is the case if a participant gets up and starts walking to the front to give a presentation. The annotators did not get any information about which camera is preferred for the different situations. Therefore, the degree of freedom for the annotators is rather high which leads to a low inter annotator agreement on the data set, the average is $\kappa = 0.3$, which is a fair agreement [Gwe01]. This result shows that the task of annotating the video mode is very subjective, because it depends on the own taste of each annotator. If one annotator does the same meeting twice, with a couple of

Table 2.1: Distribution of the video modes, if the seven cameras are used, based on frames in the meeting corpus with a duration of three hours. The shortcut c-up stands for the close-up cameras.

	centre	left	right	c-up 1	c-up 2	c-up 3	c-up 4
Absolute	77135	12685	13511	53616	35218	37816	40019
Relative in [%]	28.6	4.7	5.0	19.9	13.0	14.0	14.8

days in between, the average inter annotator agreement goes up to $\kappa = 0.6$, thus one and the same annotator is very consistent. Even though the shot boundaries are on a frame base and no gray array is allowed around the shot change, a substantial agreement is achieved. It can be said, that the annotation is a very subjective task, but it is done very consistently by one and the same person. Therefore, we decided only two annotators are performing the annotation of the three hours data set. By doing that a consistent data set is available for the training of the pattern recognition models and an evaluation can be performed. In table 2.1 the distribution of the video modes, for the case that only the seven cameras are used, is presented.

2.1.3.2 Activity Detection

As for the video editing, it is also necessary to label the video and audio recordings for the activity detection system, because novel pattern recognition techniques should be applied to the meeting data. Therefore, the whole meeting data set is annotated by using the labels: absent, not active, little active, active and most active. An additional label called decision making is added at special points of the meeting, when one participant made a decision. This label contains compared to the others more semantic information of the ongoing meeting. Before the labels have been assigned to the meeting data, the meeting is split into short segments. Moreover, the short segments always contain only one label but a five minute meeting consists of 30 of these short segments in average. The distribution of these segments is shown in table 2.2. The inter-annotator agreement of two annotators is $\kappa = 0.6$, which is a moderate agreement and therefore the annotation seems to be quite robust and consistent. Due to the fact that the label absent was not used by the annotators throughout the 36 meetings it was removed from the annotations and the detection process.

2.1.3.3 Group Actions

Group actions have been used in the M4-Project [Ren02a] for the first time. Further research has been conducted during the AMI- and AMIDA-Project [RSR05]. Each meeting has been segmented by visual and acoustic clues before the labels are defined.

Table 2.2: Distribution of the four activity and the decision making labels based on frames in the meeting corpus. Listed for each role of the participants separately and the average of the labels. Active is abbreviated by act. The label absent is not shown as it was not annotated throughout the whole corpus.

in [%]	not act.	little act.	act.	most act.	decision
Project manager	29.9	17.8	21.2	30.6	0.6
Marketing expert	40.8	19.4	24.4	14.8	0.5
Industrial designer	39.4	17.7	18.4	24.4	0.1
Interface designer	38.5	14.2	20.8	26.5	0.0
Average	37.2	17.3	21.2	24.1	0.3

At the beginning only few labels, as idle, monologue, discussion and presentation, have been used. The label presentation is being assigned to the participant who is giving the talk. If two or more persons are talking the segment is labeled as a discussion. The label idle is introduced since it can occur that no action is taking place for short intervals during the recorded meetings. Due to the use of acoustic and visual clues, the annotation also contains high level context information which can not be directly extracted from the recordings. During this work, the set of labels has been extended by splitting the discussion and adding speaker information to the monologue label. The label discussion is separated into dialog, where two persons are speaking to each other, and discussion, where three or more participants are talking. To each of these labels the information about the current speaking participants is added.

2.1.3.4 Person Actions

The last annotation on this corpus is about what the persons are doing during the meeting. For each participant individually a segmentation has been performed. For each segment one of the labels, which are described below, is assigned. The whole procedure should only be depending on the visual channel and therefore, the acoustic channel is not played during the annotation task.

Idle: It is used for a participant talking, listening or moving slightly during the meeting. Additionally it is applied as the initial state for all of the participants.

Taking notes: This is applied if a person is writing notes on the notepad, which does not include a person only holding a pen in his hands.

Computer: Every time a participant is using the computer located in front of him. It is not labeled when he is holding a mouse or lays down his hand on the hand rest of a computer.

Table 2.3: Distribution of the six emotions over the number of segments in the Airplane Behavior Corpus.

	aggressive	cheerful	intoxicated	nervous	neutral	tired
Absolute	94	104	33	93	75	23
Relative in [%]	22.3	7.8	17.6	22.0	17.8	5.5

Pointing: If it is clearly visible that the hand is pointing at something.

Agree: A person is nodding and it is clearly observable in one of the camera views.

Disagree: Disagreement is shown when a participant is shaking his head or doing other gestures which clearly indicate disagreement.

Manipulating: If a person is holding something in his hands, for example a remote control or a prototype. It is not assigned if a subject is handling a mouse.

Presentation: This label is used when a person is standing in front of the projector board or the whiteboard. It is not necessary that he gives a presentation.

Whiteboard: Only if the person is in contact with the whiteboard during the process of writing.

Stand up: It is applied only during the movement of a person who is getting up from the seat.

Sit down: This label is used when a person is actually sitting down on the seat.

Other: A participant is walking from the seat to the area in front of the projector screen or vice versa, then the label other is used. Movements of the hands and arms which can not be assigned to pointing or manipulating.

2.2 Surveillance corpus

In this work, the "Airplane Behavior Corpus" (ABC) [SWA$^+$07, ASR07] is used as a database for the detection of potential threats in planes. Six activities are defined by experts and these are classified as important clues for the identification of threats. These activities are aggressive, cheerful, intoxicated, nervous, neutral and tired.

The corpus has been recorded throughout the SAFEE [Gau04, CF08] project, due to the lack of a publicly available corpus. It contains 11.5h of video material which is recored in front of a blue screen. In total 422 video clips have been derived from the recordings. These segments have an average length of 8.4s and three experienced

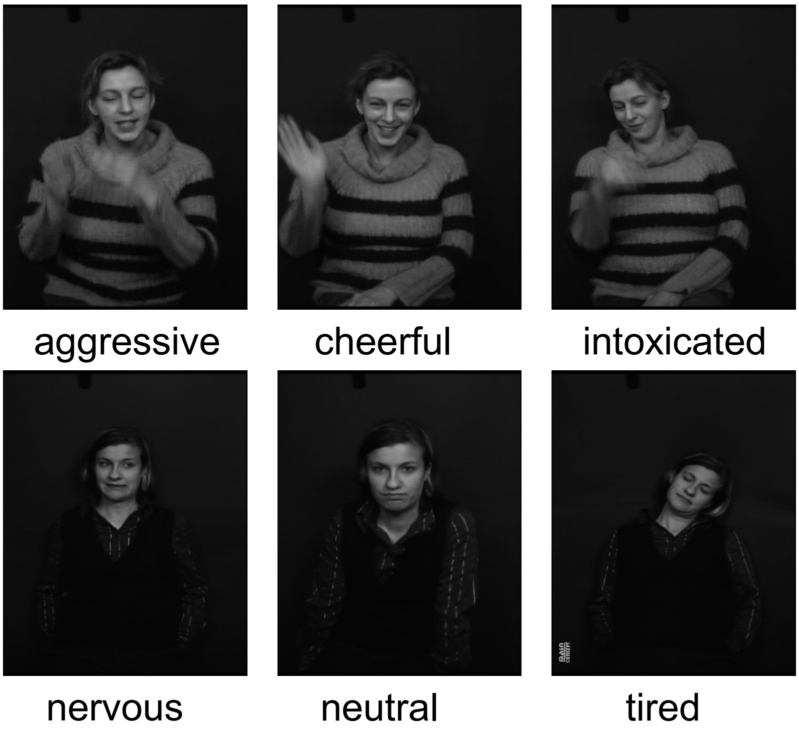


Figure 2.5: Examples for all six emotions in the Airplane Behavior Corpus.

annotators labeled these. For the recordings, a condenser microphone and a DV-camera were located in front of the subject, similar to a position in a seat's back rest of an airplane. Thus, the camera captures the upper body of the subjects, as shown in figure 2.5. The hidden test-conductor leads the subjects through a scenario which consists of a vacation flight with different scenes as start, serving wrong food, turbulences, conversation with a neighbor or falling asleep. By using these scenarios, more realistic reactions are created by the subjects. Each segment was assigned to one of the following labels: cheerful, intoxicated, nervous, neutral, tired, and aggressive. The distribution of these segments is shown in table 2.3. The labels can be grouped to suspicious and normal behaviour. The suspicious group consists of aggressice, intoxicated and nervous and contains 220 segments which represents 52.1% of the total segements. The labels cheerful, neutral and tired are assigned to the normal behaviour group. 202 segments are within this group and therefore 47.9% of the 422 segments are of normal behaviour.

2.3 Interest Corpus

The "Audiovisual Interest Corpus" (AVIC) [SMH$^+$07, SME$^+$09] has been recored due to a lack of a large publicly available audiovisual set dealing with interest. Further-

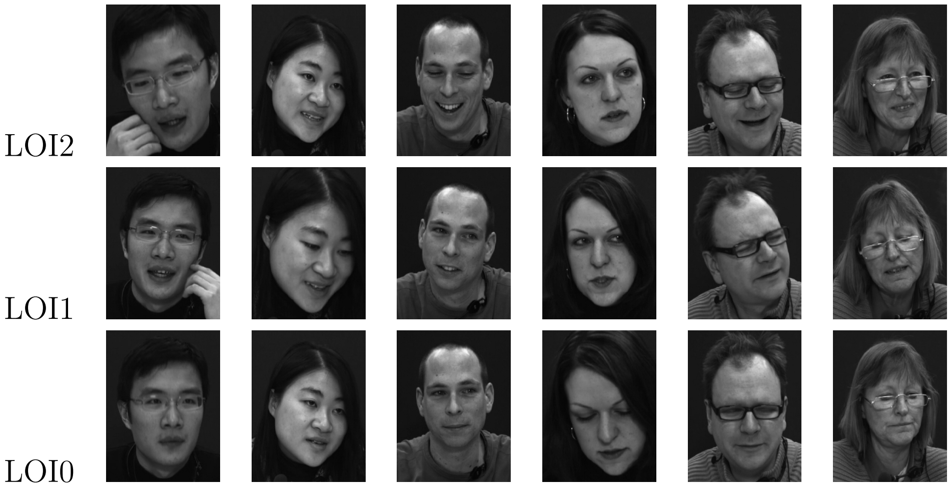


Figure 2.6: Example video frames (for better illustration limited to the facial region here) for ”master Level of Interest 0-2” taken from the AVIC database. Two subjects in gender balance were chosen from each of the three age groups.

more, to overcome the limitation of acted audiovisual databases [SSB+07, ZPRH09]. In the scenario setup, an experimenter and a subject are sitting on both sides of a desk. The experimenter plays the role of a product presenter and leads the subject through a commercial presentation. The subject's role is to listen to explanations and presentations of a so called experimenter, ask several questions of her/his interest, and actively interact with the experimenter considering his/her interest to the addressed topics without respect to politeness. Visual and voice data is recorded by a camera, two microphones, one headset and one far-field microphone.

After the final recording, the AVIC database consists of twenty-one subjects in gender balance. Three subjects are Asian and the others are European. The language throughout experiments is English and all subjects are experienced English speakers. Three age categories were defined during specification phase (< 30 years, $30 - 40$ years, > 40 years) for balancing. The mean age of the subjects resembles 29.9. The total recording time is approximately 10.5h.

2.3.1 Annotation

The “Level of Interest” (LOI) is annotated for speaker turns of the data base. It reaches from disinterest and indifference, over neutrality to interest and curiosity. Thus the LOI describes the status of the subject's interaction with the experimenter. To acquire reliable labels of a subject's LOI, the entire video material was segmented in speaker and sub-speaker turns and subsequently labeled by four independent male annotators. A speaker turn is either when the subject is talking or listening to the experimenter. Moreover, such a turn can last for several seconds and the LOI can

Table 2.4: Distribution of the “master Level of Interest” (LOI) of the segments with an inter labeler agreement of 100% and a length of more than ten frames. In total 925 segments are evaluated during the thesis.

	LOI0	LOI1	LOI2
Absolute	268	494	163
Relative in [%]	28.8	53.4	17.6

change during it. Therefore, the LOI is annotated for every sub-speaker turn. A sub-speaker turn is defined as follows: a speaker turn lasting longer than two seconds is split by punctuation and rules until each segment lasts shorter than two seconds. It is done by syntactical and grammatical rules according to [BSS$^+$06]. A speaker turn with a duration of less than two seconds is automatically a sub-speaker turn. Sub-speaker turns are frequently also referred as segments and for the reasons of consistence within this thesis it is done here too. In order to get an impression of a subject’s character and behaviour before the annotation of a person starts, the annotators had to watch approximately five minutes of a subject’s video. This helps to find out the range of intensity, to which the subject expresses her/his curiosity.

These five LOIs were distinguished in the first place by each of the four annotators which labeled each segment of the whole data base:

LOI1 - *Disinterest*: subject is bored of listening and talking about the topic, very passive, does not follow the discourse.

LOI2 - *Indifference*: subject is passive, does not give much feedback to the experimenter’s explanations, unmotivated questions if any.

LOI3 - *Neutrality*: subject follows and participates in the discourse, it is difficult to tell, if she/he is interested or indifferent to the topic.

LOI4 - *Interest*: subject wants to discuss the topic, closely follows the explanations, asks some questions.

LOI5 - *Curiosity*: strong wish of the subject to talk and learn more about the topic.

For an automatic processing during the training and evaluation, a fusion of these four LOIs, one from each annotator, to a new “master LOI” was automatically performed. A scheme has been introduced, which gives a score to each segment depending on the inter labeler agreement. Since for LOI1 and LOI2 there have been too few items, they are clustered together with LOI3, and thus the LOI scale has been shifted to the new LOI0 to LOI2. This means that, the new LOI0 is the cluster of the old LOI1, LOI2 and LOI3. The LOI4 and LOI5 are shifted to new LOI1 respectively LOI2. Thereby, values of $\kappa = 0.66$ with $\sigma = 0.20$ are observed for all segments when the new LOIs are used for the calculation. For this work the inter labeler agreement has to be 100%, which is equal to the case that all annotators give the same rating to one segment. This is the case for in total 996 segments. Since 71 segments are shorter than ten frames these segments are removed before the evaluation takes place. The exact LOI distribution for the new shifted LOI scale is shown in table 2.4 for all 925 segments with an inter labeler agreement of 100% and a length of more than ten frames. Example video frames for LOI0 - LOI2 after clustering and inter labeler agreement based reduction are depicted in figure 2.6.

3 Feature Extraction and Preprocessing

In this chapter the features which are applied to the different tasks in this thesis are presented. Three modalities of features are used: acoustic, visual and semantic. The first two are directly derived from the recording devices. The semantic ones contain more high level information and therefore these are created by different detection systems or have been hand annotated during this work.

3.1 Acoustic features

Information about the mood, the feelings, or the emotions can be gathered [SRL03] from the acoustic channel. To achieve good results in analyzing the people, different features have to be evaluated. Therefore, a large amount of features is extracted from the recorded acoustic channels.

For the meeting corpus only the Mel Frequency Cepstral Coefficients [YEH^{+}02, FGZ01] (MFCC) have been independently extracted from each of the participant's close-talking microphones. The first and the second derivations are calculated from the MFCCs, and this results in a 39 dimensional acoustic feature space for each participant's microphone. These 39 features are derived from not overlapping windows with a size of 40ms, which fits to the framerate of the video recordings in the meeting corpus. In total, for each time window in the meetings, a 156 dimensional acoustic vector is extracted, because of the four participants' microphones which are analysed.

For the behaviour and the emotion corpuses not only the MFCCs are extracted, but also other features which have been developed for interpretation of speech and music. Due to the fact that a large number of features has to be extracted and evaluated, the novel feature extractor *openSMILE*[1] [EWS09] is used. It is a real-time

[1] The open source project *openSMILE* is available at `http://sourceforge.net/projects/`

Table 3.1: 33 low level features extracted from the behaviour and emotion corpuses using the *openSMILE* feature extractor.

Feature Group	**Features in Group**	#
Spectral	Centroid	1
	Frequency-Band Energy (0-250Hz, 0-650Hz, 250-650Hz, 1000-4000Hz)	4
	Roll-off (25%, 50%, 75%, 90%)	4
	Flux	1
	Position of Maximum	1
	Position of Minimum	1
Signal Energy (frame-wise)	Root Mean-Square (RMS-E)	1
	Logarithmic Energy (log-E)	1
Fundamental Frequency (F0) based on Autocorrelation (ACF)	F0-Frequency (Hz)	1
	Voice Probability	1
	Voice Quality	1
	F0-Envelope Curve	1
Mel Frequency Cepstral Coefficients (MFCC)	Coefficients 0-12	13
Time Signal Features	Zero-Crossing Rate (ZCR)	1
	Mean-Crossing Rate (MCR)	1
Number of all low level features		33

and on-line acoustic feature extractor and the acronym stands for *Speech and Music Interpretation by Large-space Extraction*. The extractor was developed at the Technische Universitï¿½t Mï¿½nchen in the scope of the EU-project SEMAINE[2] [SCH$^+$08]. Overall these groups of low level descriptors are extracted: spectral, signal energy, fundamental frequency (FO), MFCC, and time signal features. In total, 33 low level features are extracted from each microphone in the corpuses. In table 3.1 all the extracted features are listed and more details are given. In [EWS09], more descriptions about the different feature types can be found. As the videos are recorded with 25 frames per second, the window size for the acoustic features and functionals is set to 40ms. Thus, it is possible to perform a feature fusion without any up- or down-sampling of the extracted audio and visual feature spaces.

Since functionals are commonly used in emotion recognition [SME$^+$09, SSB09], behaviour detection [KMR$^+$07, KMH$^+$07, SWM$^+$08, WSA$^+$08] and music information retrieval [SER08a, SER08b, SHAR09], these are also applied in this work. Func-

`opensmile`

[2] The web page of the SEMAINE project can be found `http://www.semaine-project.eu`.

Table 3.2: 56 functionals are extracted by the *openSMILE* feature extractor.

Functionals Group	**Functionals in Group**	#
Min/Max	Maximum/Minimum Value	2
	Relative Position of Maximum/Minimum Value	2
	Maximum/Minimum Value - Arithmetic Mean	2
	Range	1
Linear Regression	2 Coefficients (m,t)	2
	Linear and Quadratic Regression Error	2
Quadratic Regression	3 Coefficients (a,b,c)	3
	Linear and Quadratic Regression Error	2
Centroid	Centroid: Centre of Gravity	1
Moments	Variance, Skewness, and Kurtosis	3
	Standard Deviation	1
Discrete Cosine Transformation	Coefficients 0-5	6
Quartiles	25%, 50% (Median), and 75% Quartile	3
	Inter-Quartile Range (IQR): 2-1, 3-2, 3-1	3
Percentiles	95%, 98% Percentile	2
Threshold Crossing Rates	Zero-Crossing Rate	1
	Mean-Crossing Rate	1
Mean	Arithmetic Mean	1
Segments	Number of Segments, based on Delta Thresholding	1
	Mean Segment Length	1
	Maximum/Minimum Segment Length	2
Peaks	Number of Peaks (Maxima)	1
	Mean Distance between Peaks	1
	Mean Value of all Peaks	1
	Mean Value of all Peaks - Arithmetic Mean	1
Times	Values are above 25% of the total Range	1
	Values are below 25% of the total Range	1
	Values are above 50% of the total Range	1
	Rise Time and Fall Time	2
	Curve is convex/concave	2
	Values are below 50% of the total Range	1
	Values are above 90% of the total Range	1
	Values are below 90% of the total Range	1
Number of all Functionals		56

tionals combine information from previous frames, thus the current frame contains more information, especially about changes over time in the extracted features. In *openSMILE*, not only low level feature extractors are implemented, but also various filters, functionals and transformations. Therefore, not only low level descriptors but also functionals [EWS09], such as maximum, minimum, range, different types of means, quartiles, standard deviation, or variance, are calculated. Additional to those, linear and quadratic regression coefficients, discrete cosine transformation coefficients, autocorrelation functions and cross-correlation functions are extracted form the audio sources. Table 3.2 shows the derived 56 functionals from the feature extractor. Furthermore, the first derivation of each functional is calculated. Overall 3729 acoustical features are extracted from each frame. The list of all available low level features and the functionals, which *openSMILE* can extract in real-time and on-line from an audio source is available on the sourceforge project page[3].

3.2 Visual features

In this section, the extracted features from video recordings of all available cameras are described. In human to human communication, not only the speech is important also the visual clues which are exchanged between dialog partners are containing information[JR64, BB73]. Visual clues are already used by infants to communicate with her/his caretaker before they can even speak [Hal77, Bul79]. Therefore, the investigation of the visual recordings is of interest for the meeting analysis. Moreover, it is true for the surveillance task, too. First, the used visual features are characterized in detail and then the regions are pictured where the features are extracted from.

3.2.1 Skin blobs

The first features give the opportunity to derive certain information from the hand and head movements of the participants. In [YKA02], various face detection algorithms are described, one of them is a skin colour detector. An adaptive skin colour detector is presented in [WR05]. This approach can be applied both for the face and hands of participants. In [PSS04], face and hand movements are suggested to be used for video editing. Face movements are also important for the behaviour recognition, which has been shown in [AHSR09]. Therefore, skin blobs are added as the first visual feature. The first step to extract the face and hand movements, is to find the skin colour in the frame. It is performed by a comparing each pixel with a skin colour look-up table, which has been trained on 5.7 million pixels from pictures of people from all over the world. For more details about the training see [Köh06]. The look-up table is a 16 bit rg-table thus the the recorded image is transformed

[3] `http://sourceforge.net/projects/opensmile`

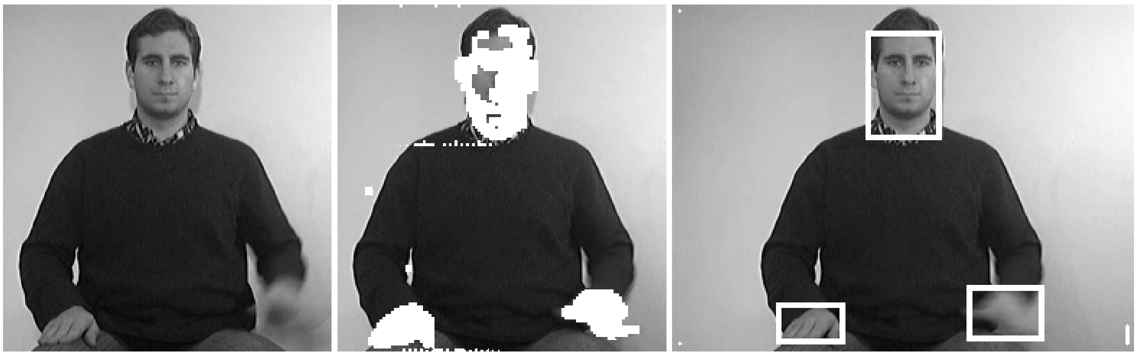


Figure 3.1: The first picture shows a single frame taken from a sample video for illustration of the skin blob detection. In the second picture all detected skin colour pixels are marked by white dots. The frame has been split for the detection task into a head region and a hand region. Especially at the boundaries of the image and of the regions of interest, detection errors are visible. The last image shows the bounding boxes, which are found by the algorithm for the face and the hands. Additionally, the norm of the motion vector for each of these bounding boxes is drawn as a white bar in the corners of the image. The image shows that only the left arm is moved therefore the lower right bar is longer than the other two bars.

from the RGB colour space to the rg colour space. After the comparison, a binary image of the same size as the colour image, is available where each possible skin pixel is marked as one. To fill gaps in the possible skin areas in the binary image, a 5x5 dilation filter [Pra01] is applied to it. The located skin areas are then analyzed for their shape, the relation of their eigenvalues, and context knowledge about possible positions of the head and the hands. Before the analysis is performed several possibilities are available for the searched body parts. Thereafter, the most accurate blobs are selected, which then become the face and the hands. Each of them are marked by a rectangle, as shown in Figure 3.1 for a single frame of a sample video. The feature vector for each bounding box contains the coordinates of the lower left and the upper right corner, the size in x- and y-direction and the movement of the center of the bounding box between two boxes extracted from consecutive frames. The movement is described by the shift along the x- and y-axis and the square-distance. In total 15 parameters are extracted for each bounding box.

3.2.2 Global Motion Features

The second visual feature group estimates the motions from the recorded meeting participants. In [RK97], these features, which represent the motion in a defined region of interest, are introduced for gesture recognition. In [EKR98], the so called global motion features have been used for gesture recognition in acted videos. These features have been successfully applied in the meeting domain in [WZR04] and [ZWR03]. For behaviour recognition the same features have been extracted in [ASR07]. The global motion features can be extracted in real-time with only one frame latency therefore these features are applied to all the tasks in this work.

The extraction of the features is outlined in this paragraph here. First, a difference image $I_d(x, y)$ is calculated from the video stream by subtracting the pixel values of two subsequent frames. Seven features are extracted from the sequence of difference images, by calculating equation 3.1 to 3.7 and concatenated for each time step t into the motion vector $\vec{b}(t)$. By applying the feature extraction, the high dimensional video is reduced to a seven dimensional vector, but it preserves the major characteristics of the motion. The following seven global motion features are derived from the sequence of difference images for the whole video. The center of motion is calculated for the x- and y-direction according to:

$$m_x^L(t) = \frac{\sum_{(x,y)} x \cdot |I_d^L(x,y,t)|}{\sum_{(x,y)} |I_d^L(x,y,t)|} \tag{3.1}$$

and

$$m_y^L(t) = \frac{\sum_{(x,y)} y \cdot |I_d^L(x,y,t)|}{\sum_{(x,y)} |I_d^L(x,y,t)|}. \tag{3.2}$$

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

$$\Delta m_x^L(t) = m_x^L(t) - m_x^L(t-1) \tag{3.3}$$

and

$$\Delta m_y^L(t) = m_y^L(t) - m_y^L(t-1). \tag{3.4}$$

Furthermore, the mean absolute deviation of the difference pixels relative to the center of motion is computed:

$$\sigma_x^L(t) = \frac{\sum_{(x,y)} |I_d^L(x,y,t)| \cdot \big(x - m_x^L(t)\big)}{\sum_{(x,y)} |I_d^L(x,y,t)|} \tag{3.5}$$

and

$$\sigma_y^L(t) = \frac{\sum_{(x,y)} |I_d^L(x,y,t)| \cdot \big(y - m_y^L(t)\big)}{\sum_{(x,y)} |I_d^L(x,y,t)|}. \tag{3.6}$$

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

$$i^L(t) = \frac{\sum_{(x,y)} |I_d^L(x,y,t)|}{\sum_x \sum_y 1}. \tag{3.7}$$

Figure 3.2 pictures the procedure of the extraction of the global motion features from a sample video.

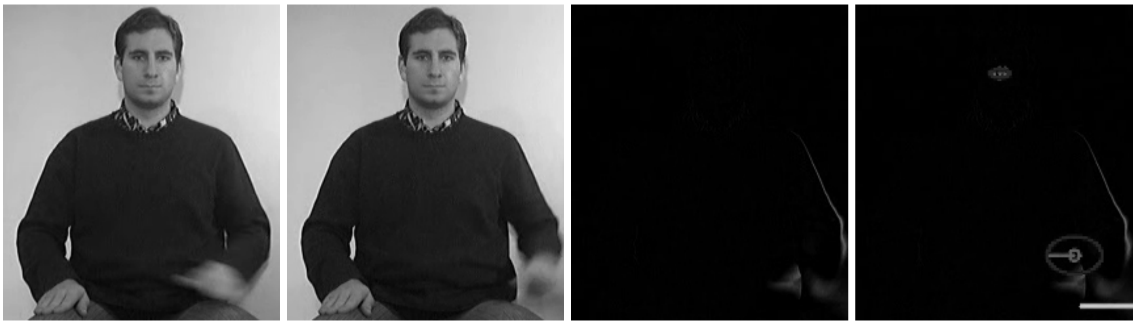
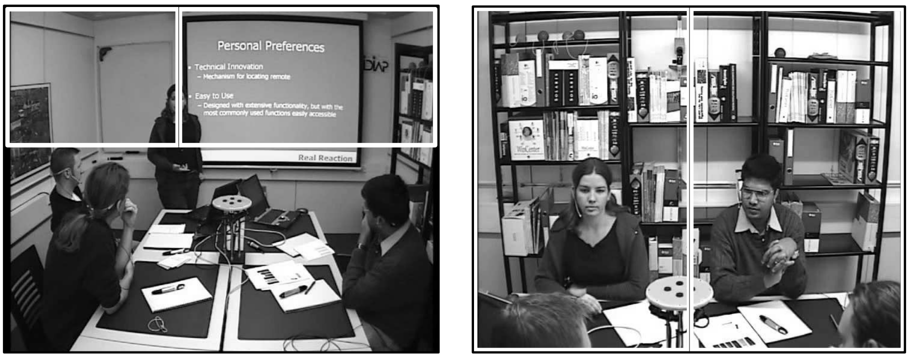


Figure 3.2: The first two pictures show two consecutive frames taken from an example video. The third image shows the difference image of these two frames. The frames are split into two regions, one for the head and one for the hands, and for each of them the difference image is calculated separately. In the last picture, the difference image is presented, but additionally the values of the global motion vectors are drawn into it. There is no visible movement in the head region and therefore only a small light gray circle and a gray ellipsis are shown in the upper half. The light gray line connects the current center of motion (small light gray circle) with the last one and the white bar shows the intensity of motion for the hand region. The gray ellipsis represents the deviation of the motion in x- and y-direction.

3.2.3 Selected Regions of Interest

The selection of different regions of interest is important to get the most relevant information from the recordings. It can be done either manually or automatically with constraints to the scenarios. In this section several regions, within the views of the cameras, are presented where they visual feature extractors are applied to. First the regions of interest are set manually, which is shown in figure 3.3 and 3.4, and later in this section the regions are detected automatically. The left image of figure 3.3 shows the center camera and two selected areas. These areas mark the whiteboard and the projection screen which are important because, if a participant is located there, she normally gives a speech or is writing on the whiteboard. These are two interesting actions for the video editing and for the activity detection. The right or left camera is vertically cut into two pieces to separate the two participants located in the view. This is presented for the left camera view in the right image of figure 3.3. The resulting half images of the left and right camera are analyzed in total, as well as defined sub regions of it. This sub regions are marked in figure 3.4. In the left image the upper body of the sitting person is the region of interest. This is sometimes necessary because other participants are passing by behind the chair and the skin colour of them interferes with the head detector applied to the seated person, if the whole image is analyzed. The movement of the hands often gives a good impression what the person is currently doing. Therefore, the right image



(a) Center

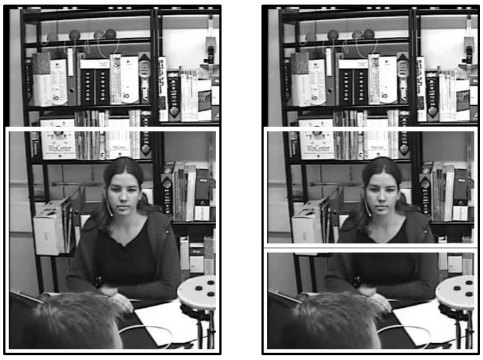
(b) Linke Kamera

Figure 3.3: The selected regions for the extraction of the global motions and the skin blobs are presented here. The left image shows the center view, where the whiteboard and the projector screen are separated into two different regions. The right image shows the recordings of the left camera of the smart meeting room. Furthermore, two regions are selected from this view for the feature extraction. This also applies to the right camera.

shows two selected regions: one for the head and one for the hands. In the lower part only hand movements are detected and in the upper part the head is localized.

3.2.3.1 Automatic Search for Regions of Interest

At first impression, a meeting room or a plane seem to be a static environment, but at the second look there is a lot of movement, even if the persons are located in the chairs. Therefore, dynamic selection of interesting regions is applied to both scenarios. This is especially important, if small movements in the head regions are analyzed. The applied approach can be split into three stages, which is shown in figure 3.5. The first one extracts the global motions from the entire image as described in section 3.2.2. The second stage locates the face in the image, firstly by applying a Rowley approach [RBK98] combined with the condensation algorithm [IB98]. Secondly the face is divided into three pieces as shown in the middle of figure 3.5. The third step calculates the global motions from these three regions. In the third stage the skin colour is detected in the entire image by using a skin locus technique [SHML00]. From the retrieved skin colour regions the face region is removed and the remaining image is vertically split into two pieces. The global motions are extracted from these two pieces, especially for gathering the movements of the left and right hand. This approach is described in more detail in [ASR07]. In total, the global motions are derived from six regions of a camera view. Given the fact that a detected face is necessary for the approach, it is only applied to the ABC corpus and to the half frames of the left and right camera from the meeting corpus.



(a) Global-Motion (b) Skinblobs

Figure 3.4: Additional to the half images of the left and right camera smaller parts of the images are analyzed. The left image shows the selected region for the additional extraction of global motion features. In the right image two regions for extra skin blob extraction are presented. The upper region is for the detection of the head and the lower one for finding the hands. Only the half frame recorded by the left camera is shown, it also applies to the other half frame and the right camera.

3.3 Semantic features

Not only low level features, also higher semantic features are used as an input for meeting analysis. In [Rei08], it is shown that semantic features help to segment meetings into smaller parts. Therefore, such features are used in this thesis. The group action and the slide change features are depending on the current status of the group and both help to segment the meeting into a rough structure. The others are more depending on the various participants than on the group, thus they help to split the meeting in smaller parts. Short descriptions for all applied semantic features are given here.

3.3.1 Group Actions

Group actions in meetings have been deeply investigated [AHDGP$^+$06, RSR07, RSR05]. The meeting is segmented by various approaches and the parts are classified as one of the following classes: monologues of participants Person 1 to Person 4, discussion, presentation, white-board-writing and note-taking. In some publications [ZGPB$^+$04, MGPB$^+$05], it has been suggested to combine the group actions, for example note-taking with either monologue, presentation, or white-board, in order to model the interaction of the group and the participants in a better way. Others split the segment discussion into disagreement and consensus. Both extensions are not used, because they are not necessary for the meeting analysis in this work. This

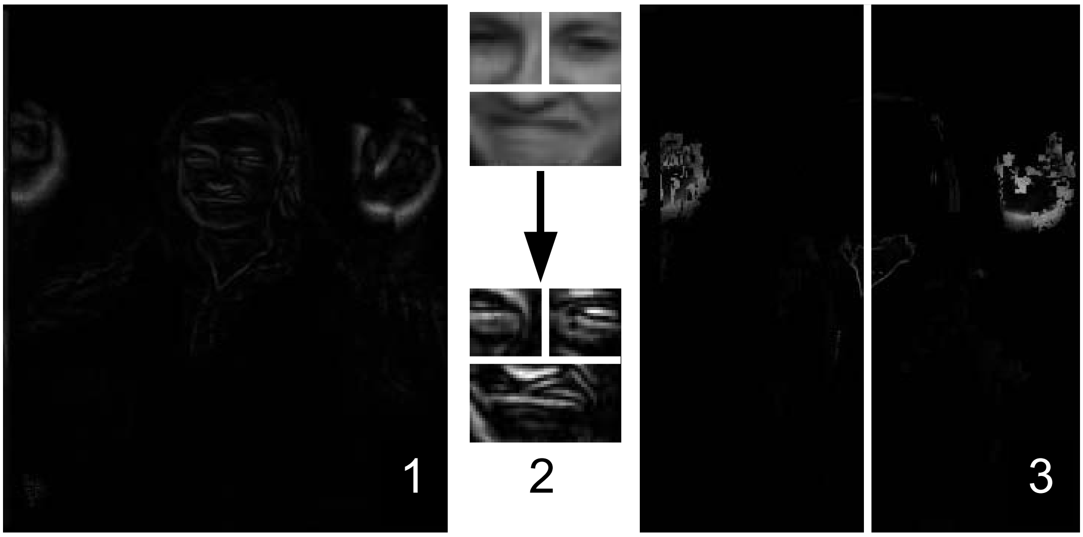


Figure 3.5: From each recorded camera the global motions have been extracted, additionally to the predefined regions, from automatically detected regions of interest. The first region is the entire frame which is shown in the left part of the figure. The second are three parts of the face which are found by a combination of the Rowley approach and the condensation algorithm. The right part of the figure shows the third region, where only the skin colour of the left and right hand is analyzed. More detail are presented in [ASR07]

semantic feature can be easily used to segment the meetings into parts, which helps to find important segments in the meeting. They are very reliably detected directly from the audio and video data streams. However, the current recognition models are not working in real-time, therefore the feature can only be used in off-line systems.

3.3.2 Person Actions

Person actions have been investigated in [ZWR03] and [WZR04] and can be used for the recognition of group actions, as well as a direct input for the meeting analysis in this work. The actions have been limited in previous publications to sitting down, standing up, nodding, shaking the head, writing and pointing. We extended the list with using a computer, giving a presentation, writing on the white-board, manipulation of an important item and being idle, which is used if the person is only listening or talking. Not all of these actions have direct influence on the meeting analysis, but for example nodding or shaking the head are interesting for the output, because they give a nonverbal clue about a thought of a participant. Also writing on the white-board affects directly the video editing and the dominance detection. All of them assist in the process of finding the right time for a boundary between two different labels. The first drawback of this feature is, that the reliability is worse because of occlusions and of the perspectives at the persons. The fact that the recognition models are not working in real-time is the second disadvantage, however with emerging computer power it will become possible in the future. Therefore, this feature currently only applies to the browsing of past meetings.

3.3.3 Person Speaking

Person speaking is a feature which contains information whether a person is currently speaking or not. As all meetings used in this thesis have four participants, a four dimensional vector for each frame is extracted. The information about who is currently speaking is derived from a speaker diarization system, which automatically detects the participants and assigns the spoken words to them. More details about speaker diarization can be found in [MMF$^+$06, WH09]. For this feature it is not necessary to know what the person is speaking.

3.3.4 Person Movements

Movements of hands and the face of a person, which are used in [PSS04] as an input for the video editing are normally too disturbing during a meeting to be used directly as a feature. Therefore, a set of different movements and positions which are common in meetings are defined as follows: off camera, sit, other, move, stand whiteboard, stand screen and take notes. These describe some important tasks which each person can perform in an ongoing meeting. Again the movements of a person are not directly connected with the video modes or the level of activity, but the boundaries are helpful for estimating the point in time of a change. This feature could be used for the recognition of the person action, as well as vice versa. The group action is also closely related.

3.3.5 Slide Changes

Another important information which helps to segment the meeting is the change of the slide, which is projected on the screen. The knowledge about it helps to integrate a picture of the slide into the output video of the system. The feature can be extracted in real-time with a latency of only one frame. Therefore, the difference image of two consecutive frames is calculated and the intensity of the changes is measured. In the case of a peak above a certain threshold a slide change is detected. In [Zak07], this easy approach works well for the detection of the slide changes, as a person, which is moving in front of screen, is creating a high intensity for a long time compared to a slide change. The extraction of images from the projector screen is more critical, because of occlusions which happen often at the moment of the slide change. For a video conference we need a different approach of capturing the slides as described in [VO06]. Another possibility is to have direct file access to the slides.

3.4 Preprocessing of the Extracted Features

Two types of preprocessing are performed during this thesis: normalisation and feature selection. As the range of the different features varies, a normalisation of the mean and the variance of the whole databases is done. Due to the huge amount of features and the dependences among the features it is reasonable to perform a feature selection on the other side. In this theses a principal component analysis, described in section 3.4.2.1, and a sequential forward feature selection (see section 3.4.2.2) are conducted.

3.4.1 Feature Normalisation

Two possibilities for a feature normalisation exist. In the first one features are normalized in a way that their values are located within a predefined range. The second method is one where all features have the same mean and the same variance after the normalisation. During this thesis, the second approach is used and the following values are used: mean $\mu = 0$ and variance $\sigma^2 = 1$. This leads to no limitation of the value range.

Normalisation is performed for each feature separately over all available data files. As each feature is evaluated separately, the dependencies among features are not changed during the process of normalisation. Depending on the three data sets used in this work, the number of files are 36 for the meeting corpus, 422 for the ABC corpus and 925 for the AVIC corpus. In each of these data files, all frames are analysed during the normalisation. In total 270000 frames are normalized in the meeting corpus, 109533 frames in the ABC corpus and 49350 frames in the AVIC corpus. To perform the normalisation it is necessary to concatenate all frames of one data set to a global feature matrix. The number of columns of the global feature matrix is equal to the total number of frames and the number of rows matches the total number of features. The number of frames is according to the length of the recorded data set. The concatenation process is shown in figure 3.6, where the global features matrix for the ABC corpus is depicted.

3.4.2 Reduction of the Feature Dimension

A feature reduction is performed if a huge number of different features are available. The applied graphical model implementation, the so called Graphical Model Toolkit [BZ02], needs a huge amount of training material to train statistical models. The amount rises with a rising number of features. This is particularly difficult, as data is usually sparse. Furthermore, computation is becoming rather hard, as both computation time and memory consumption rise. 3810 audio-visual features are extracted from the ABC and the AVIC corpus and therefore it is important to reduce

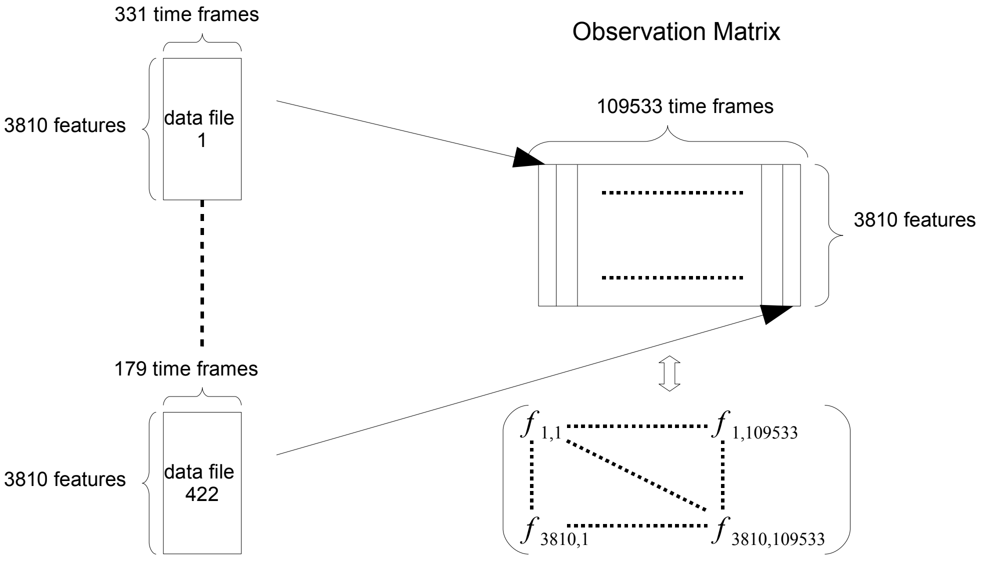


Figure 3.6: The process of the concatenation of all the features of the ABC corpus to the global feature matrix is shown here. All the 422 different files are merged into the final matrix which has 3810 rows and 109533 columns. The normalisation is performed in the next step for each of the 3810 rows independently.

the feature space. For the meeting data no feature reduction by any algorithm is performed, thus various different combinations of features have been evaluated which is described in section 6.4 and 7.4. On the other hand, a feature space reduction often helps to increase the performance of the classification system and at the same time it reduces the computation time and power [KS00]. All these benefits are achieved by using, for example, a principal component analysis or a sequential forward selection. Both of them are described in more detail in the following sections.

3.4.2.1 Principal Component Analysis

Principal component analysis (PCA) [Fuk90, Jol02, MYLB08] is a mathematical function that transforms a number of correlated variables into new uncorrelated variables. The first principal component accounts for as much of the variability in the old variables as possible. Each of the succeeding components also account for as much as possible of the remaining variability. It is also named Karhunen-Loève transform [Kar47, Loè78], the Hotelling transform or the proper orthogonal decomposition. The PCA is applied in order to receive uncorrelated features from the highly correlated feature space which is extracted from the audio and video sources. Moreover, the features stored at the beginning of the new feature set contains the combined information and thus it can be used to reduce the size of the feature space by using only features from the beginning of the new set. In these features most of the relevant information from the original data is preserved. Therefore, it breaks down the complexity of the classification problem by transforming the originally large feature space into a low dimensional new one. A short description of the calculation which has to be performed for the PCA are given in the next paragraph.

For the PCA first the mean centred features are extracted and then an eigenvalue decomposition of the covariance matrix $\mathbf{\Phi}$ is calculated:

$$\mathbf{\Phi} \cdot \mathbf{U} = \mathbf{U} \cdot \mathbf{\Lambda}. \tag{3.8}$$

$\mathbf{U}$ is thereby a matrix containing the eigenvectors $\mathbf{U} = [\mathbf{u_1}, ..., \mathbf{u_D}]$ and $\mathbf{\Lambda}$ is the diagonal matrix of the eigenvalues of the covariance matrix $\mathbf{\Phi}$. The eigenvectors $\mathbf{u_d}$ with $1 \leq d \leq D = 3810$ are the axes of the new and decorrelated coordinate system. The following equation shows the structure of the covariance matrix which consists of mean centered features

$$\mathbf{\Phi} = [\mathbf{f_1} - \overline{\mathbf{f}}, ..., \mathbf{f_T} - \overline{\mathbf{f}}] \cdot [\mathbf{f_1} - \overline{\mathbf{f}}, ..., \mathbf{f_T} - \overline{\mathbf{f}}]^T, \tag{3.9}$$

where $\overline{\mathbf{f}} = (\mu_1, ..., \mu_D)^T$.

The PCA is now applied by multiplying the transposed eigenvector matrix $\mathbf{U}^T$ with the mean centered features:

$$\dot{\mathbf{f}}_\mathbf{t} = \mathbf{U}^T \cdot (\mathbf{f_t} - \overline{\mathbf{f}}). \tag{3.10}$$

Due to the large dimensionality of the global observation feature matrix, which consists of $D = 3810$ different features, it is desirable to reduce the dimension. Nevertheless it is important to preserve most of the relevant information of the original data, taking the recognition performance into account. For this purpose, the eigenvectors are sorted from the largest to the smallest corresponding eigenvalues. A new eigenvector matrix $\mathbf{U}_{ord} = [\mathbf{u_1}, ..., \mathbf{u_q}]$ with $q \leq D$ is created using the first q eigenvectors. The subsequent multiplication of the mean centered features with the reduced eigenvector matrix $\mathbf{U}_{ord}$ leads to a reduced feature set consisting of altogether q features:

$$\dot{\mathbf{f}}_\mathbf{t,ord} = \mathbf{U_{ord}}^T \cdot (\mathbf{f_t} - \overline{\mathbf{f}}). \tag{3.11}$$

Additionally, a variance normalisation of the transformed features $\dot{\mathbf{f}}_\mathbf{t}$ and $\dot{\mathbf{f}}_\mathbf{t,ord}$ can be performed and leads to the new normalized and decorrelated features $\ddot{\mathbf{f}}_\mathbf{t}$ and $\ddot{\mathbf{f}}_\mathbf{t,ord}$.

3.4.2.2 Sequential Forward Selection

The second approach for the reduction of the features space is the so called sequential forward selection (SFS) [DK82, PNK94]. The aim of it is to derive R features y_r

out of the complete audiovisual feature set $\mathcal{F}_D = \{f_1, ..., f_D\}$, here $D = 3810$, with $R \leq D$ and concatenate them to a new feature set $\mathcal{Y}_R = \{y_1, ..., y_r\}$ with $\mathcal{Y}_R \in \mathcal{F}_D$. In order to do this, it is necessary to compare two different sets of features $\mathcal{Y}_i$ and $\mathcal{Y}_j$. Therefore, a cost function $J(\cdot)$ is introduced in the form that $J(\mathcal{Y}_i) > J(\mathcal{Y}_j)$. This means that the feature set $\mathcal{Y}_i$ performs better as set $\mathcal{Y}_j$. As cost function the recognition accuracy rate, which is calculated from the number of correct classified segments and the total number of available segments, is used for the ABC corpus. Additionally to the cost function, an individual significance $S_0(f_i)$ of each feature f_i with $1 \leq i \leq D$ and the joint significance $S(f_i, \mathcal{Y})$ with a feature set $\mathcal{Y}$ are introduced.

The first step of the SFS is to evaluate the cost function for each of the 3810 features separately. Consequently, all features f_i are evaluated and the feature with the highest individual significance is selected and will be the first element of the new feature set $\mathcal{Y}_1$.

$$y_1 = \operatorname*{argmax}_{f_i \in \mathcal{F}} J(f_i). \tag{3.12}$$

Then the feature $y_{k+1} = f_i$ which leads to the maximum joint significance is added recursively to the current set $\mathcal{Y}_k$ using the following equation:

$$y_{k+1} = \operatorname*{argmax}_{f_i \in \mathcal{F} \setminus \mathcal{Y}_k} S^+(f_i, \mathcal{Y}_k), \tag{3.13}$$

$$\mathcal{Y}_{k+1} = \mathcal{Y}_k \cup y_{k+1}. \tag{3.14}$$

Thereby $S^+(f_i, \mathcal{Y}_k)$ indicates the difference in recognition accuracy rate between the feature set $\mathcal{Y}_k$ and the feature set $\mathcal{Y}_k \cup f_i$:

$$S^+(f_i, \mathcal{Y}_k) = J(\mathcal{Y}_k \cup f_i) - J(\mathcal{Y}_k), f_i \in \mathcal{F} \setminus \mathcal{Y}_k. \tag{3.15}$$

This concatenation of new features is repeated until the best R features are selected regarding the recognition accuracy. The recognition accuracy is calculated by counting the correct classified instances and dividing it through the total number of instances. Once the recognition accuracy starts to decrease by adding an additional feature, the process is repeated four more times to check that it was not only a local minimum [SKR09]. A disadvantage of the sequential forward selection is the fact that once a feature is added, it cannot be removed from the selected feature set. In order to compensate this disadvantage one can use the sequential forward floating selection [PNK94].

In this work, SFS is performed separately for the 3729 acoustic and 81 visual features, which have been extracted from the ABC. For the evaluation, a single stream model with 20 Gaussian components and 20 class states has been used. The model has been selected because of the computational restrictions. Due to the huge

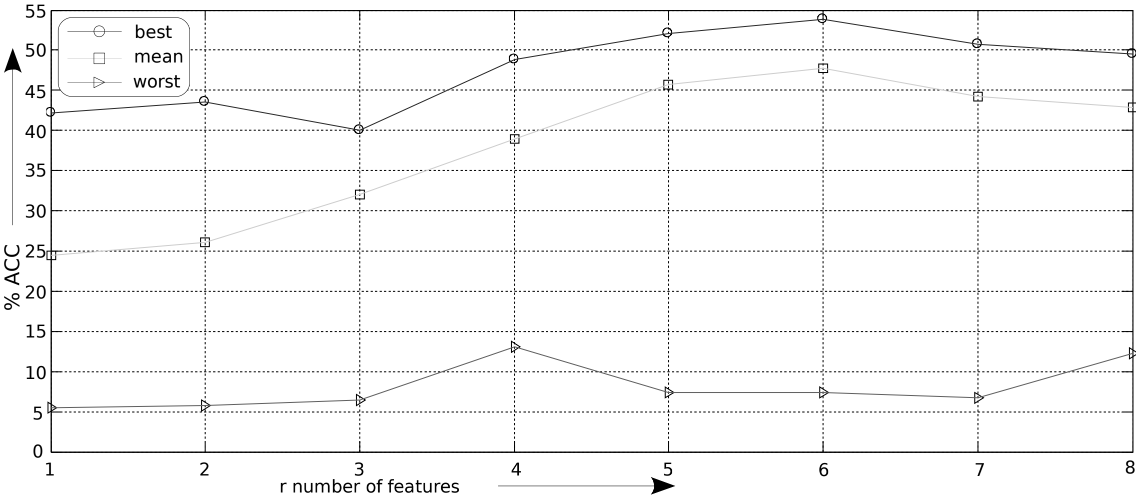


Figure 3.7: This table shows the experimental results derived from the acoustical sequential forward selection. Thereby, three different recognition accuracy rates depending on the number of used audio features are calculated. First of all, the accuracy rate belonging to the most significant feature is presented for each dimension of the feature set $\mathcal{Y}_R$. Moreover, accuracy rates based on the least significant feature of each dimension are illustrated. Finally, the average accuracy rate of the tested graphical models is shown as a function of the number of used features. By the concatenation of six features $f_{1781}, f_{478}, f_{524}, f_{3454}, f_{272}$ and f_{3545} the best result for the acoustic set is achieved.

number of features for the acoustic data, only a single iteration instead of 20, which are used for the visual set, is performed for each feature. In figure 3.7 the evaluation is shown for the acoustic features and in figure 3.8, for the visual features. In both cases the number of features to achieve the best results is dramatically reduced compared to the total number of available ones. For the acoustic features, a set with six features $\mathcal{Y}_6 = (f_{1781}, f_{478}, f_{524}, f_{3454}, f_{272}, f_{3545})$ achieves the best results with an accuracy of nearly 55%. The features which are selected are three functionals of three Mel Frequency Cepstral Coefficients and three functionals of the spectral group. Therefore, no features from the groups signal energy, fundamental frequency based on autocorrelation, and time signal are chosen. Furthermore, no low level feature is selected. The three curves show that some features can not be used to distinguish between the classes. Even after combining several features the accuracy does not get higher than 15%. Furthermore, it is displayed that the average accuracy over all feature combinations for the first six iterations is continuously improved and after that the rate starts to decline.

Compared with the six features from the acoustic data, eight visual features are needed to achieve an accuracy around 50%. It is the best performance measured by using the simple model and the visual data during the feature selection using

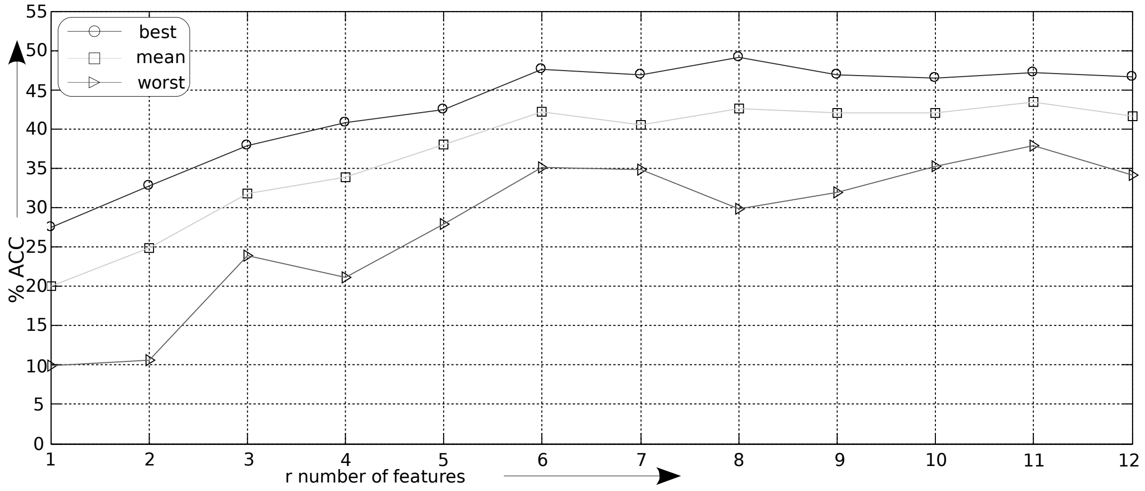


Figure 3.8: It shows the experimental results derived from the visual sequential forward selection. Thereby, three different recognition accuracy rates depending on the number of used video features are illustrated. First of all, the accuracy rate belonging to the most significant features is presented for each dimension of the feature set $\mathcal{Y}_r$. Moreover, accuracy rates based on the least significant feature of each dimension are illustrated. Finally, the average accuracy rate of the tested graphical models is shown as a function of the number of the used features. The best set of features is achieved by combining the eight features $f_{32}, f_{59}, f_{38}, f_5, f_{68}, f_{57}, f_{20}$ and f_{45}.

SFS. $f_{32}, f_{59}, f_{38}, f_5, f_{68}, f_{57}, f_{20}$ and f_{45} are the eigth features which are performing best. The features are selected from all available feature groups: one from the global motion, four from the face motion, and three from the skin motion feature group. In figure 3.8 it is visible that the least significant features of each dimension from the visual data are achieving much better results than the worst features from the acoustic data. On average, rates are achieved which perform nearly three times better. Therefore, the conclusion can be drawn that the visual feature set is already more optimized to the problem than the audio set. This is obvious, because of the huge number of extracted acoustic features.

4

Graphical Model

Graphical models (GM) are a graph-based representation for joint distributions of random variables. Such a distribution is usually very complex and frequently appears to be intractable. A GM provides a compact representation of complex probabilistic problems [KF09] and therefore GMs are often used, for example in expert systems for differential diagnosis [WTVS61, GB68]. Vertices and edges are the two building blocks of a graph [Lau96]. A vertex corresponds to random variables in the probabilistic domain and the edges correspond to direct probabilistic dependencies between these. Therefore, GMs are combining both graph- and probabilistic theory [Jor01]. In [Knö69, Kau71, BM76], more details are provided on the graph theory. The theory of GMs is based on the research in the field of message passing and conditional independence done by Pearl [Pea86, PV87, Pea88]. Later Lauritzen summed up Pearl's work in [Lau96]. In [Smy97, Whi91, Edw95, CDLS99], message passing is analysed in more detail and the efficiency of the algorithms are evaluated and proved. Introductions, especially for GMs can be found for example in [Bil03, Bil06, AH08, KF09].

Today GMs are used in image processing [Win03], in expert systems [Jen96] and for data mining [BK02]. Since the introduction of toolkits, as Bayes Net Toolbox [Mur01], Graphical Model Toolkit (GMTK) [BZ02] or Torch [CBM02], GMs are getting more popular for pattern recognition in the last couple of years.

There is a close relation between different types of graphical models and the probabilistic models [Pea88]. Markov random fields consist of undirected edges and thus two vertices can interact with each other. Causal models, which only allow directed edges and acyclic graphs, are referred to as Baysian Networks (BN) [Cha91, Jen96, Dar09, Jen10]. Probabilistic models which can be described by a BN and a markov random field are named chordal graphs, because such a graph is already triangulated. Not only BNs and markov random fields can be described by graphical models, but also neural networks [JS01, Rig94] and even fuzzy logic [BK02]. Chain graphs [Bun95], which can be considered as general representation of BNs and markov randon fields, are used to model neural networks as a graphical model.

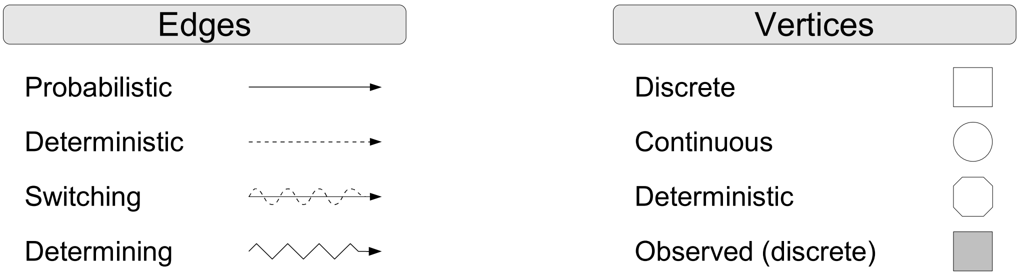


Figure 4.1: Nomenclature, which is used for the graphical models throughout this thesis according to [Mur02]. For the deterministic components the nomenclature is taken from [BB05]. The third edge changes between probabilistic and deterministic depending on a switching parent, which is connected via the zigzag edge.

4.1 Nomenclature

The nomenclature which follows [Mur02] is described in figure 4.1 and is further used for the graphical models during this thesis. Because of the need of deterministic components for some models, additional vertices and edges are utilised according to [BB05]. Directed edges, as the ones used in this thesis, always start at the parent and end at the child vertex [BM76]. A **probabilistic (continuous) edge** is applied if the child is conditionally depending on his parent. This means, depending on the status of the parent, the child's status is defined by a probability distribution. In case the **deterministic (dashed) edge** connects two vertices, the child has a defined status depending on the parent. The **third edge** is a combination of probabilistic and deterministic edges. Which type of edges connects the two vertices is defined by the determining edge. The probabilistic connection of the switching edge is used for example if a class transition occurs, which is modeled in a deterministic vertex. The next class for the new segment has to be defined by the evaluation of the class transition probability. In the case that no class transition occurs, the deterministic edge is selected and the class is inherited from the previous class. The **determining (zigzagged) edge** starts at a deterministic vertex, a so called switching parent, and has to end at the same vertex as the switching edge.

A vertex is either discrete, continuous or deterministic. Moreover, it is either hidden or observed. In case a vertex is observed, then this is characterised by shading it. The status of a hidden vertex is unknown, but it can be calculated by marginalisation over all possible statuses [Jor01] or the maximum can be derived by Viterbi-Decoding [Vit77]. A **square** represents a discrete vertex and is used for a random variable which is limited to N different values. **Circles** characterise continuous vertices and contain probability density functions. An **octagon** is used for deterministic vertices. These perform a predefined action depending on their parents, which is a special case of the discrete vertex. The vertex is important for the automatic segmentation, because it is a starting point of a zigzagged edge and in this case it is a switching parent. The switching parent decides about the type of the connection between other vertices.

4.2 Bayesian Networks

A BN consists of directed edges and vertices as shown in the right part of figure 4.2. It describes conditional dependencies between random variables [Cha91, Jen96, Jen02, Jen10]. Since directed edges always connect a child vertex with its parent vertices, there exists a direct influence from the parents to the child. A BN is defined as a directed acyclic graph (DAG) $\mathcal{G}^D$ that encodes the independence properties of a probability distribution $p(x_{V_1}, \dots, x_{V_N})$ [Jen96]. The definition of a DAG is, that a graph contains only directed edges for the connection of vertices and no cycles. A cycle is a directed path which starts and ends at the same vertex. A positive function exists for each random variable X_{V_i} which is only depending on his own and on a list of his parents $\mathrm{pa}(V_i)$[1]:

$$0 \leq f_{V_i}\big(x_{V_i}, \mathrm{pa}(x_{V_i})\big) \leq 1, \tag{4.1}$$

with

$$\sum_{v \in V} f_v\big(x_v, \mathrm{pa}(x_v)\big) = 1. \tag{4.2}$$

Furthermore, the probability distribution $p(x_{V_1}, \dots, x_{V_N})$ can be recursively factorised, which leads to

$$p(x_{V_1}, \dots, x_{V_N}) = \prod_{v \in V} f_v\big(x_v, \mathrm{pa}(x_v)\big). \tag{4.3}$$

The function is often expressed as conditional probability

$$f_{V_i}\big(x_{V_i}, \mathrm{pa}(x_{V_i})\big) = p(x_{V_i} \,|\, \mathrm{pa}(x_{V_i})), \tag{4.4}$$

but other formulas are possible, as long as equations 4.1 and 4.2 are fulfilled.

BNs describe static problems and the number of vertices is known and can not be changed during the training or testing without presegmentation. Therefore, it is impossible to model data with an unknown number of time frames. To overcome this limitation, an extension called a Dynamic Bayesian Network (DBN) is introduced in [Mur02, Bil04, BB05]. It is split into prologue, chunk and epilogue and each of these describes a static BN. The prologue is used for the first time frame (t=1) and the epilogue for the last one (t=T) respectively. The chunk is unrolled as many

[1] The parents $\mathrm{pa}(V_i)$ is a list of all vertices which are connected to vertex X_{V_i} via a directed edge ending at it.

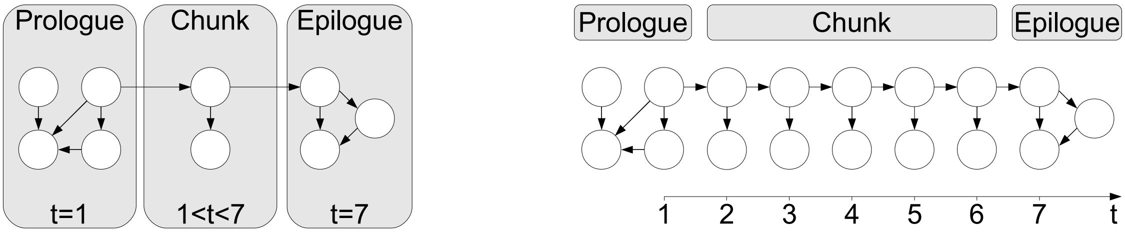


Figure 4.2: A sample DBN containing the prologue, chunk and epilogue is shown in the left part. The unrolled static BN for the observation with seven time frames is shown in the right part. For the first time frame, the prologue is used ($t = 1$). The chunk is repeated five times for the time frames $t = 2$ until $t = 6$. Finally, the epilogue is inserted into last time frame $t = 7$.

times as needed to cover all the available observations, which means that a chunk may not be present at all for very short observations of the length $T <= 2$. The chunk is unrolled $T - 2$ times for an observation of the length T. By repeating the chunk as many times as needed, a new static BN is created by the DBN and the entire algorithm which is necessary for the calculation stays the same as for the commonly known BN. In many models the prologue and the epilogue are similar to the chunk and thus the DBN only consists of a single graph, which is used for each time frame. Additionally to the graph, the connections between two consecutive time frames have to be specified in this case.

In figure 4.2 an example for a DBN is shown including the prologue, chunk and epilogue, as well as the unrolled DBN which is again a static BN. The observation has a length of seven and therefore the chunk is repeated five times for the time frames $t = 2$ until $t = 6$. For $t = 1$ the prologue is used and for $t = 7$ the epilogue.

4.3 Efficient Calculations of Graphical Models

There are two ways for the calculation of graphical models: solving the factorisation in equation 4.3 or using a message passing algorithm. The factorisation has to be solved for each variable separately, which means that for a different graph or a different variable the whole calculation has to be performed from scratch. The calculation is by far more efficient of the BN by using Pearl's message passing [Pea86, Pea88] or other approaches [JLO90, SS90, MA00] which are based on it. These approaches are valid for all BNs and are performed on the junction tree[2] of the

[2] Starting from the BN [Cow01b, Cow01a]: First the parents of a vertex are connected with an undirected edge. Secondly, the directed edges are replaced by undirected ones (moralisation). Thirdly, edges are added to the graph until no more cycle is found with a path longer than three and no chord. The last step is to create the junction tree from the found maxcliques [Mur02, Bil04, Bil06].

BN [LS88]. Therefore, all variables can be calculated from the junction tree by using a message passing algorithm.

It is necessary to calculate the probability of the observation sequence ($p(\vec{o}\,|\,\lambda)$), to perform Viterbi-Decoding and to adjust the model parameters (λ) [Rab89] for an efficient calculation of GMs. Viterbi-Decoding [Vit77] is expressed by

$$\vec{h}^* = \underset{\vec{h}}{\operatorname{argmax}}\, p(\vec{h}, \vec{o}\,|\,\lambda), \tag{4.5}$$

and parameter learning is described by

$$\widehat{\lambda}_k = \underset{\lambda}{\operatorname{argmax}} \sum_{i=1}^{I} \log p(\vec{o}_i\,|\,\lambda), \tag{4.6}$$

where $\vec{o}_i$ is the observation of the i sample from I samples in the data set. The model parameters λ are learned for example either by a gradient descent procedure [Mur02] or by the Expectation-Maximisation algorithm (EM) [DLR77]. $\vec{h}$ is the set of hidden vertices and $\vec{h}^*$ is the configuration of them with the highest probability of a model. The junction tree of the GM is important for a high performance, as all calculations can be traced back to an estimation of the status of one or several vertices. Before the calculations can be performed, the probabilities of the BN have to be assigned to the according junction tree, since it has been created only by using algorithms from graph theory. This assignment is depending on the selected message passing algorithm, in this work the HUGIN[3] procedure [JLO90, AOJJ90] is used. It is currently the most common message passing procedure and is based on [Pea86, Pea88] as many others, too. The HUGIN procedure is split into the initialisation, which assigns probabilities to the junction tree, and the global propagation of information to all maxcliques. The initialisation is performed only once while creating the junction tree. The global propagation is done each time new information is available at any of the vertices of the junction tree, respectively the graph. This means whenever a new observation arrives at a vertex.

4.4 Training of Bayesian Networks

The training of a BN is performed in two steps. Both steps can be done automatically from data or the parameters are defined by an expert.

- The structure of the model has to be found, which includes the number of vertices, the connections between the vertices (edges) and the types of the probability distributions for each vertex.

[3] HUGIN stands for Handling Uncertainty in General Inference Network.

- The estimation of the parameters of the probability distributions, for example mean and variance, of each vertex.

In this thesis the structure of the GM is based on expert knowledge. There are some approaches which perform this step automatically by analysing the data base [Hec01], but these are not used due to the fact that structure learning is very difficult for dynamic data. At the end of this chapter, various structures, which are used during the training and decoding procedure are described.

Due to the fact, that BNs are applied to pattern recognition problems where data for training is available, an automatic learning of the parameters of the probability distributions is used in this thesis. Various supervised and unsupervised approaches are known [Hec01, Nea03, Jor01, Gha98, Mur02] for the learning procedure. Since the data used is not completely observed, the EM-algorithm [DLR77] is applied to learn the model parameters. Incomplete data means, that not for all vertices training data is available or some vertices are hidden. In the next section the learning procedure is described in more detail.

4.4.1 EM-Training for Bayesian Networks

Often not all variables are observed for dynamic problems of different observation length and in this case the EM-algorithm is chosen for the training of models, as it deals with hidden or unknown variables. Overall, the likelihood for a GM with N_V vertices is

$$\mathcal{L}(\lambda \mid \mathcal{O}) = \log \prod_{i=1}^{I} p(\vec{o}_i \mid \lambda) = \sum_{i=1}^{I} \sum_{v=1}^{N_V} \log f_v \left(x_v, \mathrm{pa}(x_v) \mid \vec{o}_i \right) \tag{4.7}$$

with I training data samples $\mathcal{O} = \{\vec{o}_1, \ldots, \vec{o}_I\}$. For a BN with hidden vertices the data-likelihood is described as

$$\mathcal{L}(\lambda \mid \mathcal{O}) = \sum_{i=1}^{I} \log \sum_{h'=1}^{H} p(x^h = h', x^o = \vec{o}_i), \tag{4.8}$$

with x^h being the set of hidden variables and $h' \in \{1, \ldots, H\}$ describing all possible configurations of them. The current observation is $\vec{o}_i$ which is assigned to the observation variable x^o. The optimisation of the function can only be done by an approximation procedure, because of the sum inside the logarithm. The EM-algorithm [DLR77, Moo96, Bil97, AHR06] is commonly used for this approximations. A new parameter set λ is found by the maximisation of

$$\widehat{\lambda} = \operatorname*{argmax}_{\lambda} Q(\lambda, \lambda') \tag{4.9}$$

starting with a known parameter set λ' for the probability distributions of all vertices of the GM. This is the maximisation step of the algorithm, where the initial parameters λ' are improved. In [DLR77], it is shown that in case $Q(\lambda, \lambda')$ is maximised the data-likelihood $\mathcal{L}(\lambda \,|\, \mathcal{O})$ is also maximised. The function $Q(\lambda, \lambda')$ is defined by

$$\begin{aligned} Q(\lambda, \lambda') &= \Big\langle \log p(\mathcal{O}, h' \,|\, \lambda) \Big\rangle_{\mathcal{O}, \lambda'} \\ &= \sum_{i=1}^{I} \sum_{h'=1}^{H} \log p(x^h = h', x^o = \vec{o}_i \,|\, \lambda) \; p(x^h = h' \,|\, x^o = \vec{o}_i, \lambda'). \end{aligned} \tag{4.10}$$

The factorisation of it yields

$$\begin{aligned} Q(\lambda, \lambda') &= \sum_{i=1}^{I} \sum_{h'=1}^{H} \log \prod_{v=1}^{N_V} p(x_v \,|\, \mathrm{pa}(x_v), x^o = \vec{o}_i, x^h = h') \; p(x^h = h' \,|\, x^o = \vec{o}_i, \lambda') \\ &= \sum_{i=1}^{I} \sum_{h'=1}^{H} \sum_{v=1}^{N_V} \log p(x_v \,|\, \mathrm{pa}(x_v), x^o = \vec{o}_i, x^h = h') \; p(x^h = h' \,|\, x^o = \vec{o}_i, \lambda'). \end{aligned} \tag{4.11}$$

$p(x^h = h' \,|\, x^o = \vec{o}_i, \lambda')$ is thereby the expectation step of the EM-algorithm, which is calculated from the junction tree via message passing of any BN. For the expectation step the updated parameters from the maximisation step are used. Only for the first step the parameters are chosen randomly. The expectation and maximisation steps are performed iteratively until the algorithm converges.

4.4.2 Training Structure

In this section, several used training structures of the GMs are described. For all of these the production probability of the models is specified and for the training the EM-algorithm is used.

4.4.2.1 Linear Model

The first model is used during the evaluation of pre-segmented data, as it is the case for both the ABC and AVIC corpuses. It is described in figure 4.3 and selects a single class for the entire sample. This means, it does not perform any segmentation and only the class with the highest probability for the sample is selected during evaluation. During the training the frame counter t, class c_t and the observation $\vec{o}_t$ are known. Every other vertex is hidden and the probability distributions of them have to be learned. Going into more detail of the model structure, it can be seen that each chunk consists of the vertices frame counter t, class c_t, class position q_t^c,

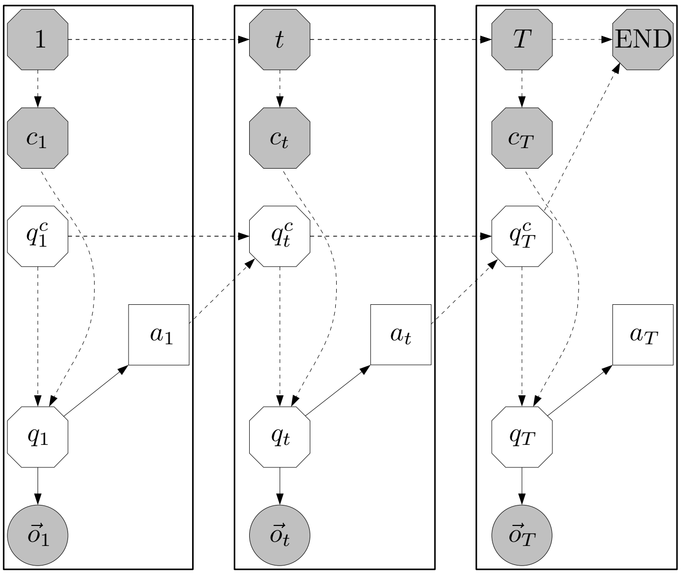


Figure 4.3: The training structure of a simple continuous single stream graphical model. It shows the complete structure as it is used for the implementation in GMTK, but the vertices are labeled in the way that the calculation is getting more clearly.

transition probability a_t, state pool q_t and the observation $\vec{o}_t$. The joint probability, which is factorised by the graph, is

$$\begin{aligned}
&p(\vec{o}_1, \ldots, \vec{o}_T, q_1, \ldots, q_T, a_1, \ldots a_T, q_t^c, \ldots, q_T^c, c_1, \ldots, c_T, t_1, \ldots, t_T) = \\
&p(\vec{o}_1 \,|\, q_1)\; p(a_1 \,|\, q_1)\; f(q_1 \,|\, q_1^c, c_1)\; f(q_1^c)\; f(c_1 \,|\, t = 1)\; f(t = 1) \\
&\prod_{t=2}^{T-1} p(\vec{o}_t \,|\, q_t)\; p(a_t \,|\, q_t)\; f(q_t \,|\, q_t^c, c_t)\; f(q_t^c \,|\, a_{t-1}, q_{t-1}^c)\; f(c_t \,|\, t)\; f(t) \\
&p(\vec{o}_T \,|\, q_T)\; p(a_T \,|\, q_T)\; f(q_T \,|\, q_T^c, c_T)\, f(q_T^c = 1 \,|\, a_{T-1}, q_{T-1}^c)\; f(c_T \,|\, t = T)\; f(t = T).
\end{aligned} \tag{4.12}$$

Deterministic conditions between vertices are described by $f(\cdot)$ and probabilistic condition with $p(\cdot)$. The second line of equation 4.12 shows the prologue, the third line the chunk and the last line the epilogue. The probability can be also written as

$$\begin{aligned}
&p(\vec{o}_1, \ldots, \vec{o}_T, q_1, \ldots, q_T, a_1, \ldots a_T, q_t^c, \ldots, q_T^c, c_1, \ldots, c_T, t_1, \ldots, t_T) = \\
&f(q_1^c) \prod_{t=1}^{T} p(\vec{o}_t \,|\, q_t)\; p(a_t \,|\, q_t)\; f(q_t \,|\, q_t^c, c_t)\; f(c_t \,|\, t)\; f(t) \\
&f(q_T^c = 1 \,|\, a_{T-1}, q_{T-1}^c) \prod_{t=2}^{T-1} f(q_t^c \,|\, a_{t-1}, q_{t-1}^c),
\end{aligned} \tag{4.13}$$

which is a simplification of equation 4.12, but the structure of prologue, chunk and epilogue is not visible by inspection anymore.

For decoding, the frame counter t is removed and an additional deterministic edge is added in between the class vertex c_t. This edge copies the states of the detected class c_t into the next time frame. Via Viterbi decoding the class with the highest probability is selected once the whole sample is evaluated. The class vertices c_t are not observed any more and replaced by a discrete type of vertex.

By replacing the deterministic edge, which only copies the status between two consecutive vertices c_t, by a probabilistic edge which selects for each time frame the class independently from the previous one, it is possible that the decoding model segments a test sample during the evaluation. The model does not learn the boundaries, as no class changes occur in the training data, but it has the capability to switch between classes, which are equally distributed, via Viterbi decoding. This possibility is used for some evaluations of the ABC corpus.

4.4.2.2 Multi-Stream Model

Figure 4.4 shows a two stream GM. The vertices are similar to the single stream model: frame counter t, class c_t, class position $q_t^{c,n}$, transition probability a_t^n, state pool q_t^n and observation $\vec{o}_t^n$. For the two stream model is $n = 2$. The model can be extended to more streams easily by increasing n and hereby adding more vertices of the types: class position $q_t^{c,n}$, transition probability a_t^n, state pool q_t^n and observation $\vec{o}_t^n$. The streams are processed separately and the combination of them is fulfilled on class level. Moreover, they are statistically independent from each other, but at the same time the features inside a stream are depending on each other, which is visible in equation 4.14. Two vertices, class c_t and frame counter t, are accessed from each stream, since the model only selects a single class for the whole segment. The joint probability of the model is

$$\begin{aligned}
&p(\vec{o}_1^1, \dots, \vec{o}_T^1, q_1^1, \dots, q_T^1, a_1^1, \dots a_T^1, q_1^{c,1}, \dots, q_T^{c,1},\\
&\quad \vec{o}_1^2, \dots, \vec{o}_T^2, q_1^2, \dots, q_T^2, a_1^2, \dots a_T^2, q_1^{c,2}, \dots, q_T^{c,2}, c_1, \dots, c_T, t_1, \dots, t_T) =\\
&\quad \prod_{t=1}^{T} p(\vec{o}_t^2 \,|\, q_t^2)\; p(a_t^2 \,|\, q_t^2)\; f(q_t^2 \,|\, q_t^{c,2}, c_t)\; p(\vec{o}_t^1 \,|\, q_t^1)\; p(a_t^1 \,|\, q_t^1)\; f(q_t^1 \,|\, q_t^{c,1}, c_t)\; f(c_t \,|\, t)\; f(t)\\
&\quad \prod_{t=2}^{T} f(q_t^{c,2} \,|\, a_{t-1}^2, q_{t-1}^{c,2})\; f(q_t^{c,1} \,|\, a_{t-1}^1, q_{t-1}^{c,1})\\
&\quad f(q_1^{c,2})\; f(q_T^{c,2} \,|\, a_{T-1}^2, q_{T-1}^{c,2})\; f(q_1^{c,1})\; f(q_T^{c,1} \,|\, a_{T-1}^1, q_{T-1}^{c,1}).
\end{aligned} \tag{4.14}$$

The decoding model does not contain the frame counter t and the class c_t is not observed. Furthermore, the class vertex c_t is discrete. Two consecutive class vertices are connected via deterministic edge, which is necessary for the propagation of the current states of the class into the next time frame. It is only a deterministic edge

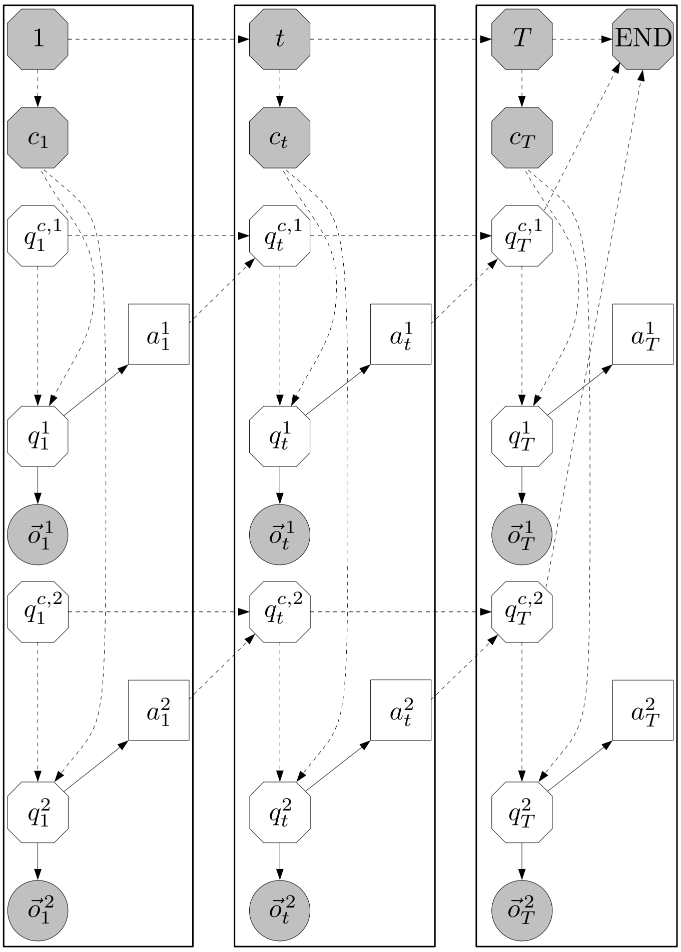


Figure 4.4: The training structure of a continuous two stream graphical model. The model contains nearly twice the structure of the model shown in figure 4.3. All vertices except the frame counter t and the class c_t exist twice. Both streams influence the class vertex and therefore the outcome of the model. It can be extended to more streams by just adding more structures similar to GM1. Here, the complete structure is shown as it is used for the implementation in GMTK, but the vertices are labeled in a way that the calculation is getting more clearly.

which copies the status to the next time frame, since only one class label is needed for the whole segment. A change of the class is prohibited by this type of edge.

4.4.2.3 Learning of Segment Boundaries

A model structure which is capable of switching between classes is presented in figure 4.5. This is necessary to learn the correct points in time, when between two classes has to be switched, which is important for many scenarios, for instance the

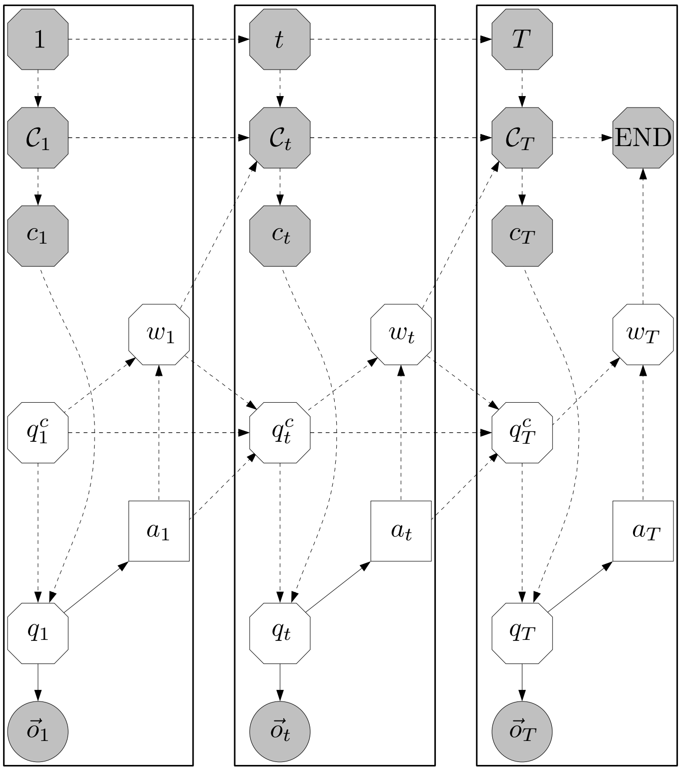


Figure 4.5: The training structure of a continuous single stream graphical model which can train the correct class boundaries. This model learns during the training procedure the correct boundaries of the classes. For this model, the boundaries have to be known exactly, since they are learned on a frame base. Shown is the complete structure as it is used for the implementation in GMTK, but the vertices are labeled in the a way that the calculation is getting more clearly.

automated selection of the camera which shows a threat in a multi-camera video surveillance setting. To train a model like this, it is important that annotations are available on frame level, because otherwise only the sequence of the classes can be learned and not the exact boundaries. It is used for the meeting corpus, since the meetings are not pre-segmented and the correct point of time for a change between the classes is of importance. Especially for event recognition, where the point of time for the segmentation and the correct sequence are substantial. The vertices used are similar to the previous models: frame counter t, class counter $\mathcal{C}_t$, class c_t, class transition w_t, class position q_t^c, transition probability a_t, state pool q_t and observation $\vec{o}_t$. Compared to the models described before, only the class counter $\mathcal{C}_t$ and the class transition w_t are added. Both are important for the learning of a class transition at the correct point of time, which means for example that the video mode or the level of activity changes. The model presented is a single stream version, but it can be extended to a multi stream model easily. The vertices, class position $q_t^{c,n}$, transition probability a_t^n, state pool q_t^n and observation $\vec{o}_t^n$ have to be repeated as many times as streams ($n = N$) are used for an input of the model. The class transition w_t and the class c_t are connected with each of the repeated structures and therefore each stream influences the decision about the class and the class transition. A class transition only occurs if all the streams are in the last class position $q_t^{c,n}$. The next class is defined by the combination of all class transition probabilities of the streams. Again each stream is statistically independent from all remaining other streams, but the features within a stream are depending on each other, which can be seen in equation 4.14. From the figure 4.5, the joint probability of the single stream model can be derived with

$$\begin{aligned}
&p(\vec{o}_1, \ldots, \vec{o}_T, q_1, \ldots, q_T, a_1, \ldots a_T, q_1^c, \ldots, q_T^c, w_1, \ldots w_T, c_1, \ldots, c_T,\\
&\quad \mathcal{C}_1, \ldots, \mathcal{C}_T, t_1, \ldots, t_T) =\\
&\quad \prod_{t=1}^{T} p(\vec{o}_t \,|\, q_t)\; f(q_t \,|\, q_t^c, c_t)\; p(a_t \,|\, q_t)\; f(c_t \,|\, \mathcal{C}_t)\; f(t)\\
&\quad f(q_1^c) \prod_{t=2}^{T} f(q_t^c \,|\, a_{t-1}, q_{t-1}^c, w_{t-1}) \qquad (4.15)\\
&\quad f(w_T = 1 \,|\, q_T^c, a_T) \prod_{t=1}^{T-1} f(w_t \,|\, q_t^c, a_t)\\
&\quad f(\mathcal{C}_1 \,|\, t = 1)\; f(\mathcal{C}_T = \mathrm{END} \,|\, \mathcal{C}_{T-1}, w_{T-1}, t = T) \prod_{t=2}^{T-1} f(\mathcal{C}_t \,|\, \mathcal{C}_{t-1}, w_{t-1}, t).
\end{aligned}$$

The decoding model has a slightly different structure, since the vertices frame counter t and the class counter $\mathcal{C}_t$ are removed. Moreover, the class vertex c_t is not observed and it is a discrete vertex. In between two consecutive class vertices, a switching edge is added, which depends on the class transition w_{t-1} of the previous time frame. The edge, starting at w_{t-1} and ending at c_t, which is added as well, is of the type determining and therefore switches the type of connection between two consecutive class vertices. The decoding structure of this model is shown in the last section of the chapter.

4.4.2.4 Model with GMTK Labels

The same model as illustrated in figure 4.5 is shown in figure 4.6 once more. As can be seen the only difference between the models are the labels of the vertices. These are now adapted to the conventions of GMTK, and will be used for the implementation task. The abbreviations are quite close to the names of the vertices in the last

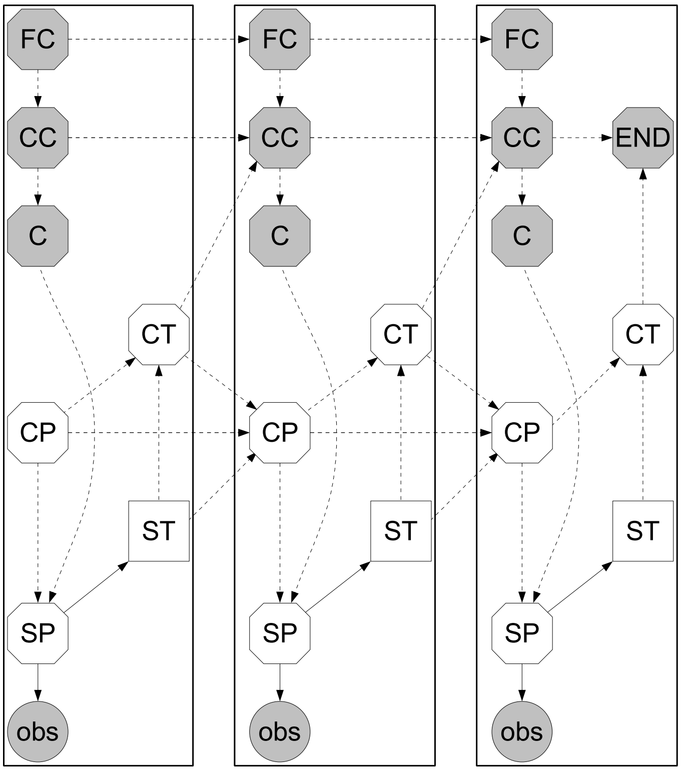


Figure 4.6: The training structure of a continuous single stream graphical model which is able to learn the segment boundaries. It shows the complete structure with the labels as they are used for the implementation in GMTK.

sections: frame counter FC, class counter CC, class C, class transition CT, class position CP, state transition probability ST, state pool SP and observation obs. END describes a vertex which contains information about the final state of the model which is reached once the last time frame T is processed. There are some more labels used in the work, especially for the two stream models $CP2$, $ST2$, $SP2$ and $obs2$. These are similar to the according ones, but these are only required for the second input stream. The abbreviation of the vertex CG means class group, which is used for hierarchical models, where the classes can be grouped together. In this case, the class group CG has to be classified first and in a second step the class C has to be found.

4.5 Decoding of Graphical Models

For decoding, two steps have to be performed: first definition of the decoding structure and second the evaluation of the model for an observation sequence. The decoding structure of all training structures presented in the previous sections can be

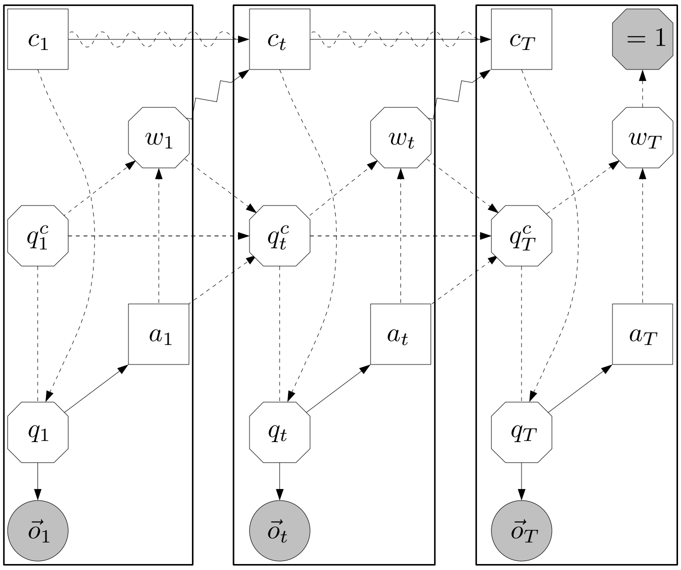


Figure 4.7: The decoding structure of a continuous single stream graphical model which automatically segments the class boundaries. This model learns during the training procedure the correct boundaries of the classes and therefore it is capable of switching more accurately than other models which only train the sequence of the classes. For the training of this model, the boundaries have to be known exactly since they are learned on a frame base. It shows the complete structure as it is used for the implementation in GMTK, but the vertices are labeled in the way that a calculation is getting more clearly.

created easily by removing the frame counter t and, if it exists, the class counter $\mathcal{C}_t$. Furthermore, the deterministic and observed vertex of the class c_t has to be replaced by a discrete and hidden vertex. Depending on the model, whether it should automatically segment or not, a switching or deterministic edge has to be added between two consecutive class vertices. In the case the model is capable of segmentation, an additional determining edge has to be added between the vertex class transition w_{t-1} of the previous time frame and the current class vertex c_t. All these changes are performed and the result is shown in figure 4.7. It is the corresponding structure to the training model with the capability of automatic segmentation from figure 4.5. For all the other structures it is even easier to adapt the training structure for the decoding process.

For evaluation purposes, the configuration of the hidden variables $\vec{h}$ with the highest probability $p(\vec{o}\,|\,\lambda)$ of a known observation $\vec{o}$, given the model parameters λ, has to be found for each time frame, as equation 4.16 describes. This is done efficiently by performing message passing on the junction tree of the decoding structure. The necessary calculation steps are similar to the training process. The only difference is that all sigma signs are replaced by an argument of the maximum over the hidden variables of the graphical model $\vec{h}$. This means, that in each time frame all possible configurations, which are leading to a state of the model, are evaluated and compared. After that, for each state the path with the highest probability is pursued into the next time frame. This procedure is continued until only a single path exists, which is described by the sequence of the best fitting configurations $\vec{h}^*$ and has the highest probability for the evaluated observations $\vec{o}$. This procedure is known as Viterbi-Decoding [Vit77] and is formulateded as follows

$$\vec{h}^* = \underset{\vec{h}}{\operatorname{argmax}}\, p(\vec{h}, \vec{o} \,|\, \lambda). \tag{4.16}$$

4.6 Evaluation of the Different Model Structures

The evaluation always contains both the learning and the testing of a model. Therefore, it is necessary to develop a training structure and the according decoding structure. Different parameters can be adjusted, for example the number of states, Gaussian mixtures and iterations of EM-algorithm. Moreover, the input data stream can be varied, which means different feature sets are evaluated. Except the number of iterations of the EM-algorithm all other parameters influence the model structure, which has to be adapted for the training and decoding structure simultaneously. If the number of states per class is changed, which means the state pool vertex q_t has a different size, also the class position g_t^k and the transition probability a_t have to be adjusted accordingly. More Gaussian mixtures usually mean that the observation vertex $\vec{o}_t$ has to be changed. If other feature sets are evaluated as an input to the structure with a different number of features, the observation vertex $\vec{o}_t$ has to be adapted again. All these adaptations to the structure can be automatically performed in case the number of features used as an input is known and the number of states and the number of Gaussian mixtures are predefined.

5 Multi-modal Surveillance

The Screening Passengers by Observation Techniques programme (SPOT) is a development of the Transportation Security Administration and the U.S. Department of Homeland Security [Haw07]. The programme aims to guarantee the safety of the passengers in public transport by observing these. Specially trained police officers are watching out for suspicious behaviour among the people which are using the public transportation system [Adm06]. They have studied fleeting and involuntary microexpressions [1] and movements which suggest abnormal stress, fear or deception. At Boston's Logan Airport, the SPOT programme has resulted in the arrested of more than 50 people within several months in 2006. These people have been arrest for having fake IDs, entering the country illegally or drug possession [Don06]. The SPOT is currently only deployed at airports, but the Transportation Security Administration is considering to deploy it to train and bus stations as well. A similar programme, called Suspect Detection System, is running in Isreal [KM06]. Another behaviour detection programme was tested in Ottawa by the Canadian Air Transport Security Authority at the beginning of the year 2010 [Mac09]. All these programmes have in common that the security persons have to be trained to recognise the microexpressions and movements. The training process and the deployment process of such a system are very time consuming and have to be planned years in advance. Furthermore, all the programmes are applied at airports or agencies are planning to deploy them at other transportation systems, but the programmes are not being used it in buses, trains, or planes. This means, that they try to prevent suspicious people to enter the transportation system, but they do not detect what is happening inside the means of transportation.

Firstly, guaranteeing the passengers' safety does not stop at the entrance of transportation. Secondly, it is a complex, difficult and expensive task for the security officers, since they have to watch multiple screens at the same time. Due to these facts,

[1] A microexpression is a facial expression shown by humans according to emotions experienced [MR83]. It is very difficult to fake these compared to regular facial expressions. More details can be found in [MR83].

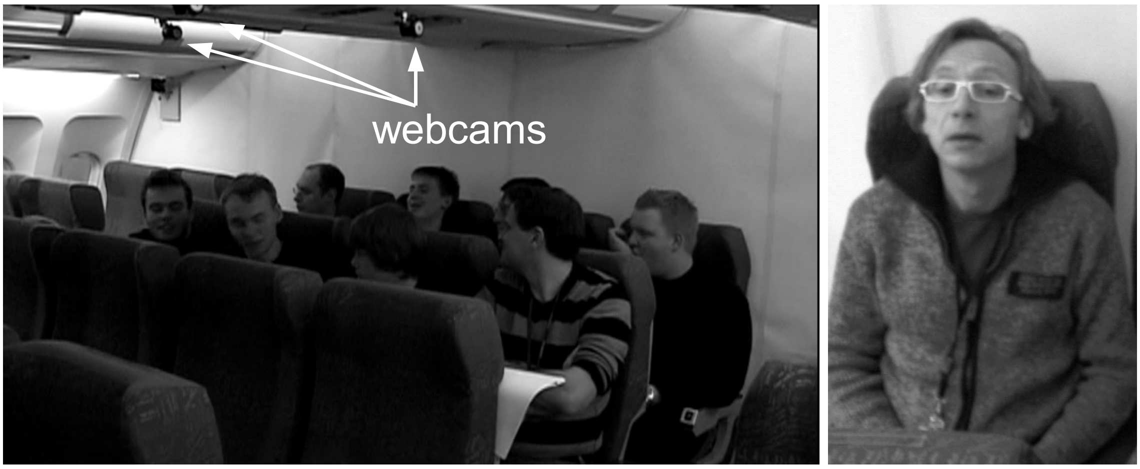


Figure 5.1: The left picture shows the whole scenario within an Airbus mock-up. The cameras which are installed are simple webcams and are placed underneath the overhead lockers and therefore no problems with movements of the passengers seats can occur. In the right image the view of one camera is presented and the nearly perfect frontal view can be seen [Ars10].

it seems desirable to monitor passengers and automatically classify their behaviour by installing recording devices and applying novel pattern recognition techniques. The aim of the system is to detect suspicious behavior of passengers, before they might endanger the security of other passengers, of flight attendees, or the whole plane. Once a suspicious behavior is recognised, the cabin crew will be informed about the location of the passenger and the observed activity. The next step is, that specially trained flight attendees take immediate action depending on the situation. The system and the procedure should help to prevent serious incidents and leads to a more secure public transportation. It could be easily adapted to buses or trains. Furthermore, it could be applied in addition to security personnel at transportation stations, like airports, bus and train stations and would assist the security personnel by detecting suspicious behaviour.

An airplane scenario was chosen to be analysed during this thesis. For this purpose, a camera is installed underneath the overhead bin of the passenger which leads to nearly perfect frontal views most of the time during the flight. This position is stationary compared to the installation in the headrest of the passenger's seat. In the frontal view, it is easy to find the head and extract the visual features described in section 3.2. The camera records the upper part of the body, which means that the arms are visible most of the time. As passengers are nowadays disposed to remain seated during the flight, additional cameras are only needed for the bathrooms [Ars10]. Not only cameras, but also microphones, are spread all over the cabin which makes it possible to analyse the passengers by using audio and visual clues. The acoustic features derived from the recorded data are discussed in 3.1. In

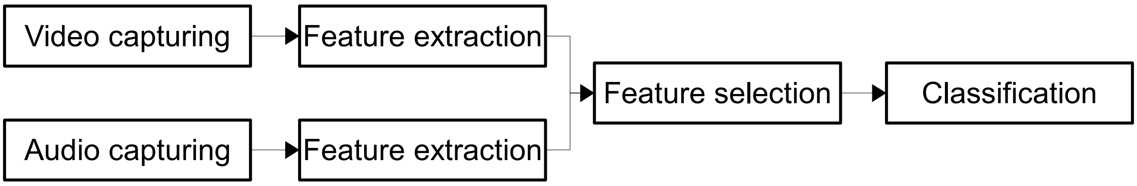


Figure 5.2: The structure of the analysis process with all the components is shown here. First the microphone and the camera capture the ongoing situation inside the plane and then the features are extracted and selected. Finally the classification is performed.

figure 5.1 a picture of the scenario is shown.

For the analysis of these audio-visual features, various GMs are used, which classify the behaviour of the observed passenger into six different classes. These classes are aggressive, cheerful, intoxicated, nervous, neutral, and tired. The whole process of the analysis from capturing to classification is shown in figure 5.2. Before the used graphical models are described in section 5.2, the recognition process is discussed in the next section.

5.1 The Recognition Process

In figure 5.2, the whole process of passengers analysis in an aircraft is shown. The process can be used for most of the audio-visual analysis tasks, as it is kept very general. It starts with microphones and cameras which are used for the observation of the cabin and the recording of the passengers. For each of the cameras and microphones, features are extracted separately on the same time base. This is necessary for the concatenation of visual and acoustic features on feature level. In the next step, the features which are leading to a better recognition than others, are selected and finally the classification is performed by applying various graphical models.

For the classification, a 5-fold cross validation is used because of the small amount of available training and test data. Moreover, all available segments are once in a test set and four times in a training set. The sets are composed in such a way, that a subject is never contained both in the test and training set. Therefore, an adaptation during the training process to a special subject is not possible. For the evaluation not only the recognition accuracy rate (ACC) which is used as a measure for the performance of a model but also recall, precision and f-measure are calculated. These measures are helpful to identify the performance for each of the classes separately and to identify which classes can be distinguished. Precision can be seen as a measure of fidelity, whereas recall is a measure of completeness. The f-measure is the harmonic mean of precision and recall. In the following sections, various approaches using graphical models are described and the results are presented.

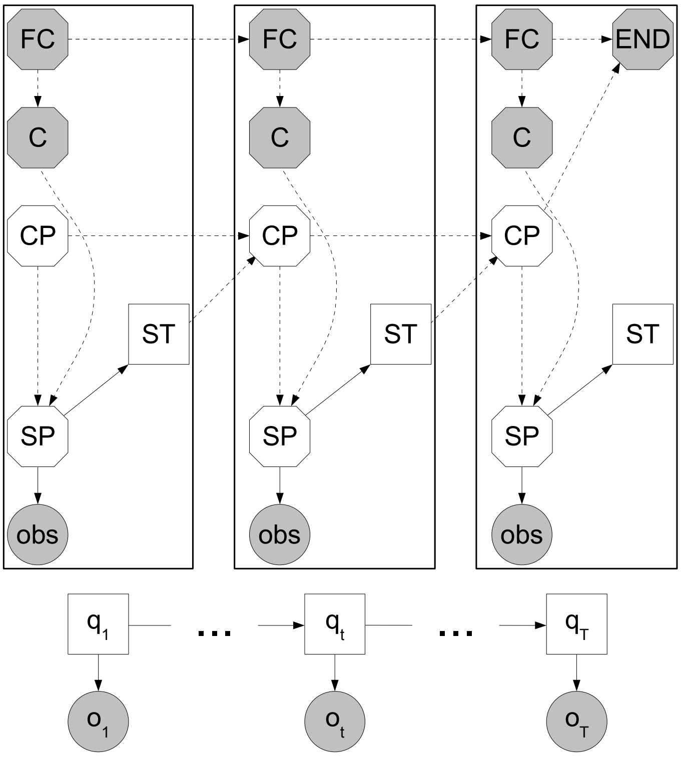


Figure 5.3: Simple continuous single stream graphical model GM1 used in this thesis for the classification of six states of behaviour. The upper part of the figure shows the training structure, as it is implemented in GMTK for the training of the model. A description of all the used vertices can be found in chapter 4. The lower part presents the simplified structure which is used for the calculations.

5.2 Used Graphical Models

Various graphical models which have been developed during the thesis are presented in this section. Several combinations of models, features and training data are theoretically described and an evaluation is performed for each of them. At the end of the section, all combinations are compared with other state of the art approaches and a conclusion is drawn.

5.2.1 Single Stream Classifying Graphical Model (GM1)

Figure 5.3 shows the training structure of a simple continuous single stream model GM1. The complex and unhandy structure is derived from the implementation of the model in GMTK. A description of all the used vertices of figure 5.3 can be found in chapter 4. The model can be simplified for the calculation of the production probability, which is shown in the lower part of figure 5.3. All hidden random variables, the frame counter FC, the class C, the class position CP, the state pool SP and the state transition matrix ST, are combined into one random variable q_t, which represents the state of the model for a certain time frame t. The production probability yields

$$p(\mathbf{O}, \mathbf{q}|\lambda) = p(q_1) \cdot p(\vec{o}_1|q_1) \cdot \prod_{t=2}^{T} p(q_t|q_{t-1}) \cdot p(\vec{o}_t|q_t), \tag{5.1}$$

considering the state sequence $\mathbf{q} = (q_1, \dots, q_T)$. Using the already described substitutions a_{ij}, $b_{s_i}(\vec{o}_t)$ and π_i and by marginalizing, i.e., summing up equation 5.1 over all possible state sequences $\mathbf{q} \in \mathbf{Q}$, the following production probability of the observation $\mathbf{O} = (\vec{o}_1, \dots, \vec{o}_T)$ is derived:

$$p(\mathbf{O}|\lambda) = \sum_{q \in Q} \pi_{q_1} \cdot p(\vec{o}_1|q_1) \cdot \prod_{t=2}^{T} a_{q_{t-1},q_t} \cdot p(\vec{o}_t|q_t), \tag{5.2}$$

while each observation $\vec{o}_t$ is a vector consisting of continuous normalized features. Moreover, equation 5.2 depicts the well-known production probability of a HMM and shows that a HMM can be considered as a simple graphical model [Rab89, RJ93, Sch09].

For decoding, a slightly altered model, as shown in figure 5.4, is used. As no segmentation is necessary, because of the pre-segmented test data, the model can only output one class for each test file. This can be done by copying the current class variable C into the next time frame. Therefore, a deterministic arc is used between to consecutive vertices C.

Throughout the evaluation, various combinations of number of states, number of Gaussian components per Gaussian mixture, number of iterations of the EM training and several feature sets with different dimensions have been tested and analysed. The following features are selected by the sequential forward selection as described in section 3.4.2.2. For the visual features, these eight features are selected $\vec{o}_{vis}^{best} = f_{32}, f_{59}, f_{38}, f_5, f_{68}, f_{57}, f_{20}, f_{45}$. They are containing one feature from the global motion, four from the face motion, and three from the skin motion feature group. From the acoustic features, these six $\vec{o}_{ac}^{best} = f_{1781}, f_{478}, f_{524}, f_{3454}, f_{272}, f_{3545}$ are performing best during the selection. The selected acoustic features contain three functionals of three Mel Frequency Cepstral Coefficients and three functionals of the spectral group. No features are chosen from the groups: signal energy, fundamental frequency based on autocorrelation, and time signal. Furthermore, no low level acoustic feature is selected.

In table 5.1, various adjustments to the training and test data have been evaluated. For all evaluations, the simple graphical model from figure 5.3 is used. A single classification is performed for each test file separately. For the first line of the

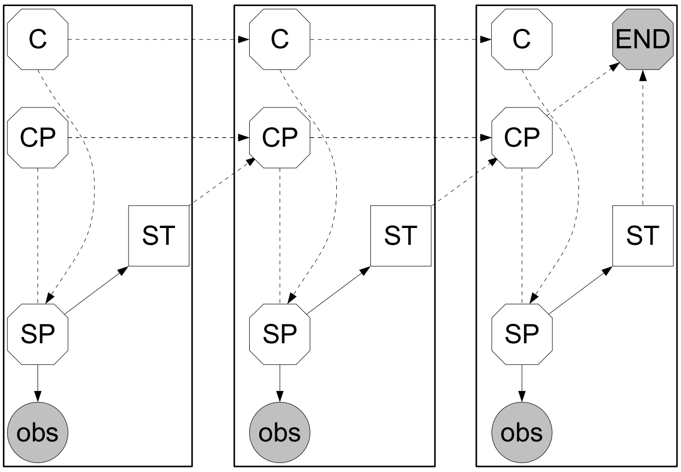


Figure 5.4: The decoding structure of GM1 as implemented in GMTK. A description of all the used vertices can be found in chapter 4.

table, the full length of each available file is tested. Moreover, no adjustments to the training and test data have to be performed. Table 2.3 shows that the number of segments for each class are very different, thus the class distribution is highly unbalanced. In table 5.2, it is visible that the classification of the classes which are under-represented is nearly impossible. Hence, the number of training segments is balanced. Therefore, in the first case, segments of the under-represented classes were copied and in the second case the number of segments of the over represented classes were reduced. The test set is not changed, thus it is possible to compare the results.

The results in the forth row are achieved by removing a tenth at the beginning and at the end of each data segment. It is based on the idea that an actor needs some time to find into his role and to express the acted behaviour pattern. Thus, it is believed that most of the characteristic patterns of a certain behaviour are located in the middle of the data segment. The approach is relevant only in the case that the corpus is acted and pre-segmented, otherwise it is nearly impossible to automatically detect the beginning of the more characteristic patterns.

The majority vote method is taken from the approach which is used in [ASR07] for Support Vector Machines. This takes 25 frames from a data segment and combines it into a non overlapping window. This is done for the whole data segment and for each of these windows, a single classification is performed. Only the last couple of frames are dropped due to the fact that the total number of frames is in most cases not a multiple of a window size. Once all the windows of one data segment are classified with the simple model GM1, the result for the whole data segment is created via a majority vote over all classified windows. By this approach, it is possible to analyse data segments of various length with a static classifier, such as a Support

Table 5.1: The results of various adjustments to the training and test set are shown. All evaluations are conducted for the eight selected visual features, for the six selected acoustic ones and for the combination of these two sets. For more detail about the feature selection see section 3.4.2.2. The best results for all the selected features are achieved with the unchanged data. The three values of the parameters represent first the number of states, second the number of Gaussian components per Gaussian mixture and third the number of iterations of the EM training. All results are measured as recognition accuracy rates.

data	parameters	$\vec{o}_{vis}^{best}$	$\vec{o}_{ac}^{best}$	$(\vec{o}_{vis}^{best}, \vec{o}_{ac}^{best})$
Unchanged	20-20-20	**49.1**	**49.8**	**52.4**
Balanced, copying	20-20-20	47.6	48.6	51.7
Balanced, reducing	20-20-10	25.8	40.8	37.0
Shortened	20-20-20	36.7	25.4	30.2
Majority vote, 15 frames	20-20-10	39.8	45.3	44.8
Majority vote, 25 frames	20-20-10	47.2	44.3	50.5
Majority vote, 35 frames	20-20-10	43.1	44.1	52.1
Majority vote, 50 frames	20-20-10	45.3	43.1	48.6

Vector Machine, without the additional effort of calculating time series features for the whole data segment. In this thesis, the window size was set to 15, 25, 35, and 50 frames and an evaluation for each of these was conducted. The processing of the features and the majority vote is illustrated in figure 5.5.

The results of table 5.1 show that all the variations of the data do not lead to an improvement in the accuracy. The best results are achieved for the acoustic, visual and combined feature set by applying the simple GM1 to the unchanged data. The accuracies are about 49% for the visual features and about 49.8% for the acustic ones. The best rate is achieved by the multi-modal features with a rate of 52.4%. Therefore, an improvement of 2.6% absoulte is achieved compared to the best single modality result. When the two balanced data sets are compared, it is clearly visible that visual features are performing worse then the acoustic ones in the case of little traning data. The problem of little training data occurs most of the time when a dynamic classifier, such as GMs or HMMs, is used and the training data is reduced due the fact of balancing available data between the different classes. Another fact, which can be pointed out is that when more than two third of the training data is removed the result derived with the acoustic features drops about 8% compared to 23% by the visual features. The idea that the actors need some time at the beginning of each of the data segments to get into the role proves wrong which is shown by the results of the shortend data. The result drops about 12% for the visual features. More interessting with this experiment is, that the acoustic features are performing worst throughout the whole evaluation of model GM1. Furthermore, the

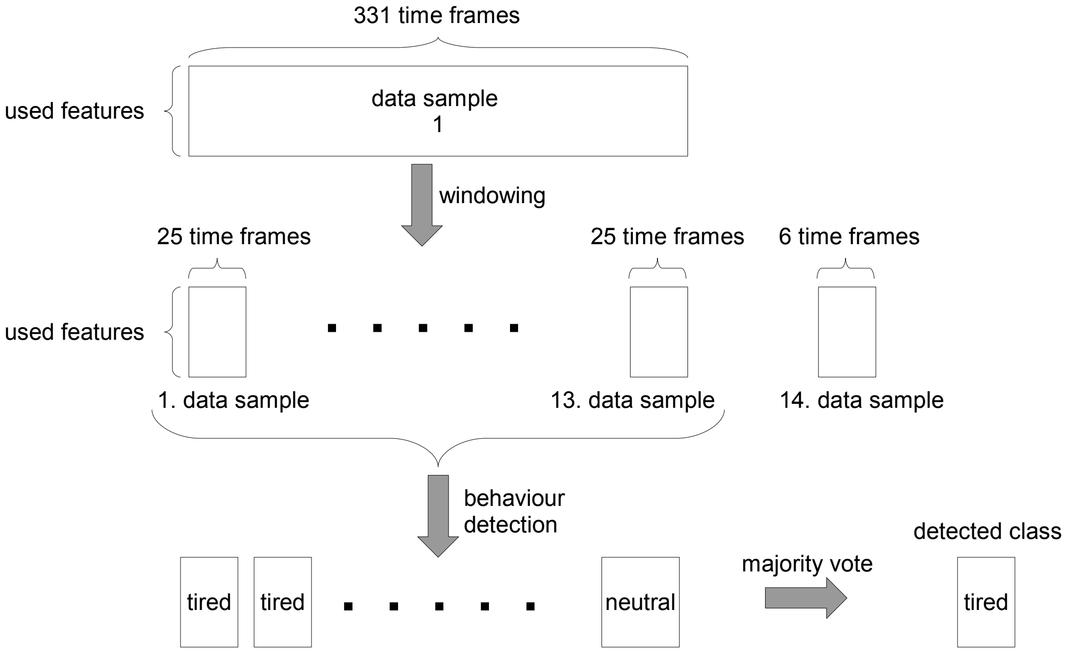


Figure 5.5: Windowing and majority vote detection process, using the simple graphical model from figure 5.3. Each data segment is split into windows of a defined size in the first step. Afterwards, for each of these windows, a classification is performed. The third step is, that a majority vote is performed over each window, which leads to the final decision for the whole data segment.

fusion of acoustic and visual data achieves the worst result. The first conclusion of experiments with shortened data is, that for the discrimination of the classes the first or the last 10% of the data segment seem very important. The second is, that the classification based on visual features can deal much better with removing 20% of the frames then the acoustic ones. The majority vote setting with a window size of 35 and the use of audio-visual features achieves 52.1% and therefore performs as good as the model using the unchanged data. When the results of the single modalities are compared with the best model, these are about 6% worse. The window size does not really influence the results of the model using the acoustic data, as the difference is only 2%. The gap between the best and the worst model applied to the visual data is more than 7% and therefore similar to the one working on the multi-modal features.

In table 5.2, the confusion matrix for the simple GM1 using the combination of acoustic and visual features is presented. The table shows that the aggressive class is nearly flawlessly detected. Only eight segments out of 94 are misclassified, but there are many segments from other classes which are assigned to the label aggressive, too. Therefore, a very good recall value is achieved, but the precision score is only about

Table 5.2: Confusion matrix of the behaviour patterns are presented for the model GM1. Furthermore, the recall, the precision and the f-measure per class are listed. All the results are achieved by using the combination of the selected acoustic and visual features and all the available data. The simple model GM1 is performing well for the classes aggressive and nervous. A big gap can already be seen for the classes neutral and cheerful. The two classes, intoxicated and tired, which are under-represented in the training data, are not really detected at all. *Aggr* stands for aggressive, *chee* represents cheerful, *into* is intoxicated, *nerv* is the short form for nervous, *neut* means neutral, and *tire* is abbreviation for tired. # is the total number of the segments per class. The last three short cuts stand for recall, precision and f-measure.

truth	aggr	chee	into	nerv	neut	tire	#	Rec	Pre	F1
aggr	**86**	7	0	1	0	0	94	91.5	51.8	66.2
chee	42	**41**	0	5	16	0	104	39.4	45.6	42.3
into	8	14	**0**	1	10	0	33	0.0	0.0	0.0
nerv	10	5	0	**59**	19	0	93	63.4	73.8	68.2
neut	17	12	0	13	**33**	0	75	44.0	39.3	41.5
tire	3	11	0	1	6	**2**	23	8.7	100.0	16.0

52%. Hence, the f-measure is only the second best for this evaluation. 63.4% of the segments of nervous are correctely detected and the precision is about 74%. This leads to the best f-measure for the model GM1 with a rate of 68.2%, compared to 66.2% for the label aggressive. A first bigger gap in the f-measure scores follows. The classes cheerful and neutral achieve a score of about 42%. Since the label cheerful is often misdetected as aggreesive the recall is only about 39%. Agressive, nervous and cheerful are the classes which are used instead of neutral during the classification, however the recall score is still 44%. Tired is mixed up mostly with cheerful and achieves a recall of only 9%. Furthermore, it is never used during the classification of any segment which belongs to another class. The f-measure of the class tired is 16%. The label intoxicated is not assigned to any of the 422 segments during the whole evaluation of the the model GM1. Due to this fact, the f-measure is zero. Most of the segments of intoxicated are labeled as neutral and cheerful.

The first conclusion which can be drawn from table 5.2 is, that the focus for further research has to be on the under-represented classes. Moreover, the precision of the class aggressive has to be improved, which is directly correlated to a better recall for most of the classes. Thus, the high number of segments which are misclassified as aggressive has to be reduced.

Table 5.3 shows some results for the evaluation of the features transformed by the principal component analysis, as described in section 3.4.2.1. The number of features, which are used in the training and classification process are increased during

Table 5.3: Experimental results for the graphical model GM1 using feature sets created by applying the principal component analysis, which is described in section 3.4.2.1. The various feature sets and parameter combinations have been tested during the evaluation. The best model uses the following parameters: 20 states, 20 Gaussian components and 20 iterations for the EM training and uses the first 20 features. All results are measured as recognition accuracy rates.

features	1–4	1–6	1–8	1–10	1–15	1–20	1–25	1–30
ACC [%]	46.7	48.1	48.6	50.0	49.1	**52.6**	49.8	51.7

the evaluation and the best result is achieved by using the first 20 features. Various parameter sets and up to 200 features, which are in order of decreasing component eigenvalues, have been tested during this evaluation. The performance of the recognition is increasing up to the first ten components, where a recognition accuracy of 50% is achieved. The performance dithers between 49% and 53%, when more than ten and less than 30 feature are concatenated to the feature set. The best average accuracy over the five folds of the cross validation is 52.6%, when the first 20 features are used. The parameters, which are applied in this case, are 20 states, 20 Gaussian components and 20 iterations for the EM training. Table 5.4 presents the confusion matrix according to this setting. The recognition accuracy is in the range between 28% to 42% for a feature set with a size between 40 and 200 features and it is about 37% for 200 features. The accuracy is decreasing due to the fact, that more training material is needed when the number of features rises and thus the complexity of the model increases.

In table 5.4, the confusion matrix of the GM1, using features derived by the principal component analysis, is presented. The results for intoxicated and tired are identical to the confusion matrix in table 5.2 derived by sequential forward selection. Furthermore, it shows that the recall of the classes aggressive and neutral is decreased by 10% and 13% compared to the previous confusion matrix. The reverse happens to the precision of both classes which increase by 14% and 18%. Therefore, the f-measure of agressive is going up by 7% and for neutral it was reduced by about 1%. The increase of the recall is about 12% for cheerful and about 8% for nervous. The precision is decreased by 8% for cheerful and by 20% for nervous. Therefore, the f-measure is going down for nervous about 6% and for neutral about 1%. On the other side it is increased by 7% for aggressive and about 1% for cheerful. Overall, the performance stays the same as for the features selected by sequential forward selection.

The much simpler approach for the feature reduction by the principal component analysis achieves the same results as the more complex sequential forward selection. Consequently, the effort for the complex method can not be justified. The biggest advantage of the principal component analysis is that the feature selection is inde-

Table 5.4: Confusion matrix of the behaviour patterns are presented for the model GM1, when the features transformed by the principal component analysis are used. Furthermore, the recall, the precision and the f-measure are calculated per class. The simple model GM1 is performing well for the classes aggressive and nervous. A big gap can already be seen for the class cheerful and another one for the class neutral. The two classes, intoxicated and tired, which are under-represented in the training data, are still not really detected at all. Compared to table 5.2, where the features selected by sequential forward selection are used, the precision of both classes aggressive and neutral increases and at the same time the recall decreases. The reverse happens with cheerful and nervous. The f-measure changes slightly for the different classes but overall the performance is unchanged. *Aggr* stands for aggressive, *chee* represents cheerful, *into* is intoxicated, *nerv* is the short form for nervous, *neut* means neutral, and *tire* is abbreviation for tired. # is the total number of the segments per class. The last three short cuts stand for recall, precision and f-measure.

truth	aggr	chee	into	nerv	neut	tire	#	Rec	Pre	F1
aggr	**77**	14	0	3	0	0	94	81.9	65.8	73.0
chee	22	**53**	0	20	9	0	104	51.0	37.3	43.1
into	11	17	**0**	2	3	0	33	0.0	0.0	0.0
nerv	1	20	0	**67**	5	0	93	72.0	54.5	62.0
neut	4	29	0	19	**23**	0	75	30.7	57.5	40.0
tire	2	7	0	12	0	**2**	23	8.7	100.0	16.0

pendent from the recognition process. Another benefit is that the computational time is much lower during the feature reduction. For the recognition, the computational time is higher than for the sequential forward selection, because all features have to be extracted every time and in a second step, these have to be transformed to a new feature space created by the principal component analysis.

The main problem of both approaches using the simple graphical model is, that the under-represented classes are not recognised. When the model is adapted during the training to these under-represented classes, in a way that the distribution of the segments per class is equal, the overall performance of the system is reduced. The recognition of the class intoxicated and tired is getting better in this case, but the performance of the other classes is reduced dramatically.

5.2.2 Single Stream Segmenting Graphical Model (GM2)

In figure 5.6, another model is presented, which is using a majority vote for the classification of the whole test segment. Compared to the majority vote model using the GM1, this model has the capability to automatically segment the test

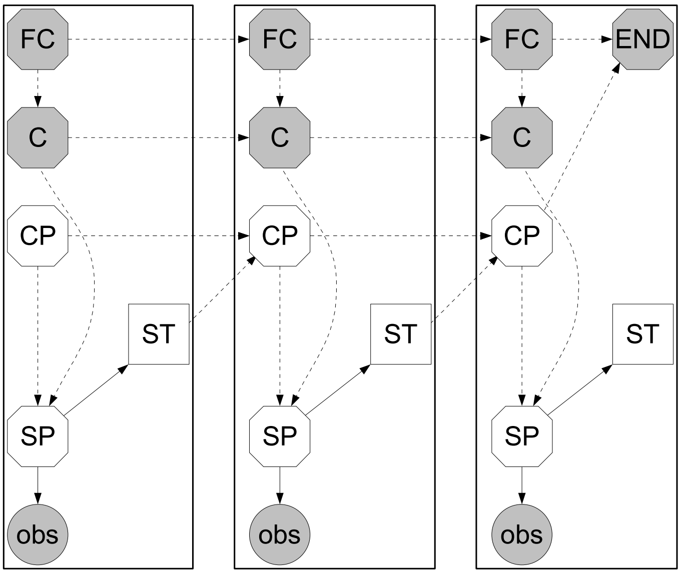


Figure 5.6: Structure for the training of the graphical model GM2. The model performs automatically a segmentation of the test segments into various sub-segments of different lengths. In a second step a majority vote is conducted on these sub-segments and the final results are achieved for the test segment.

segments into sub-segments of various lengths. GM1 uses a window of fixed size which is slided through the segment and classifies each of the windows separately. The output of each classifier for each window is used as the input for the majority vote. Unfortunately, the boundaries are predefined and therefore two classes can be located within one window. Furthermore, no information from the previous window is taken into account when fixed window sizes are used. All these disadvantages are overcome with the GM2, which finds the sub-segment boundaries within one segment automatically. Not only the segmentation is performed by the graphical model, but at the same time the classification of each sub-segment is performed. Once both segmentation and classification are finished, all sub-segments are used as an input to the majority vote. The segmentation is possible, because the model has an additional arc between two random variables C of two succeeding chunks. This would lead to a dependence between the current class and the previous one. Since a uniform distribution is chosen for the probabilities of the transitions between the classes and these probabilities are not adapted during the training, the detected class in the current chunk is independent from the previous one. This is done that way, because an adaption to the unbalanced training data is undesired. Thus, each class can be detected during the evaluation and the unbalanced number of available training segments for each class does not influence the detection process.

Table 5.5 shows the performance of the more complex graphical model. Overall it is not as good as the performance of the simple GM1 presented in table 5.2. The

Table 5.5: The performance measures of the more complex GM2 are presented. The feature selection is performed by the sequential forward selection and the best sets for the visual, acoustic and audio-visual features are used during the evaluation. The results are, compared to the simple GM1, worse. The accuracy is decreased by more than 5% for each of the feature sets compared to the best GM1 using the same feature sets. The gap is getting even bigger when the principal component analysis is used for the feature reduction. All results are measured as recognition accuracy rates.

feature	$\vec{o}_{vis}^{best}$	$\vec{o}_{ac}^{best}$	$(\vec{o}_{vis}^{best}, \vec{o}_{ac}^{best})$
ACC [%]	43.4	44.6	44.8

performance of the visual and the acoustic features is decreased by more than 5% and is around 44% now. The multi-modal audio-visual feature set does not perform significantly better than the acoustic one and compared to the best GM1 using the same feature set, the accuracy goes down by more than 8%. In the case, that the majority vote models using GM1 are compared with the GM2 the performance of the acoustic feature sets are only slightly different. The visual feature model is up to 4% worse than the majority vote model GM1 using the same features. While the acoustic modality is performing nearly equally, the accuracy of the mutli-modal majority vote GM1 achieves an accuracy of more than 52% and is about 7% better. Thus, the model GM2 is significantly worse than the best multi-modal GM1.

As mentioned before, the biggest problem of the simple GM1 is that the under-represented class intoxicated is not detected at all. This disadvantage is slightly overcome by the GM2 as table 5.6 shows, but still the two under-represented classes are detected significantly worse compared to all the four other classes. Only four segments out of 56 are detected correctly, which is compared to GM1 a significant improvement since the class intoxicated is not used at all during the evaluation of GM1. The detection of intoxicated is possible at the expense to every other class. Going more into detail of the table, shows that all recall, precision and f-measure values, except the ones for intoxicated and tired, are worse than the comparable values from the the multi-modal GM1. Especially the results of the class cheerful are decreasing. The recall is nearly bisect in value and the f-measure is reduced by 15%. With the class nervous it is similar, but all measures are decreased by about 10%. Moreover, neutral loses about 8% and aggressive about 4% for all three performance indicators.

While the performance of the two under-represented classes is increased by this graphical model, the performance for the classes aggressive, cheerful, nervous and neutral is reduced significantly. This leads to an overall performance of about 45% for the multi-modal feature set, which is compared to the simple GM1 with an accuracy of 52% a significant performance decrease. Thus, the higher complexity,

Table 5.6: Confusion matrix of the behaviour patterns are presented of the model GM2 when the audio-visual features are used. Furthermore, the recall, the precision and the f-measure are calculated per class. While the more complex model GM2 has a good performance for the class aggressive, all the other classes are performing significantly bad. Compared to table 5.2, where GM1 and the features selected by sequential forward selection are used, the class intoxicated is used during the classification. This leads to better recall, precision and f-measure for this class. All other values are below the results of the simple GM1. *Aggr* stands for aggressive, *chee* represents cheerful, *into* is intoxicated, *nerv* is the short form for nervous, *neut* means neutral, and *tire* is abbreviation for tired. # is the total number of the samples per class. The last three short cuts stand for recall, precision and f-measure.

truth	aggr	chee	into	nerv	neut	tire	#	Rec	Pre	F1
aggr	**84**	1	1	1	5	2	94	89.4	48.3	62.7
chee	43	**23**	4	9	21	4	104	22.1	37.1	27.7
into	18	7	**1**	1	6	0	33	3.0	12.5	4.9
nerv	7	10	1	**51**	21	3	93	54.8	63.0	58.6
neut	15	15	0	16	**27**	2	75	36.0	32.5	34.2
tire	7	6	1	3	3	**3**	23	13.0	21.4	16.2

which increases the computational time and the amount of training data needed, is undesireable. Since the performance gap to the GM1 is big, the evaluation using the features derived by a principal componemt analysis has not been conducted, because the improvement for GM1 is not significant.

5.2.3 Multi Stream Classifying Graphical Model (GM3)

In figure 5.7, the block diagram of the third graphical model is illustrated. The model is similar to multi-stream models as described in [YEH$^+$02, PNLM04, MHGB01]. There are two observation streams, $\vec{o}_{1,t}$ and $\vec{o}_{2,t}$, within each chunk and therefore two independent feature sets are used as an input. This is especially helpful, when different modalities are combined for improving the classification result. Acoustic and visual feature sets are applied during the evaluation and each of these sets is used as an input of a single observation. The features within one observation are depending on each other, as it is the case for all the features in the single stream graphical model GM1. The block diagram is a short form of the more complex implementation within GMTK. Therefore, the block q_t is substituted with the structure known from the simple GM1 with the difference that the structure exists twice except the vertex C. This vertex is connected to both structures and fuses the independent input streams. The production probability of graphical model

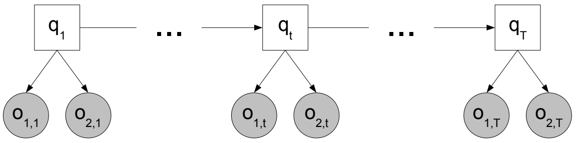


Figure 5.7: The block diagram of the two stream graphical model GM3. It contains the same structure as GM1, but it has two observation streams which are statistically independent and therefore can be drawn as two separate observation vertices.

GM3 can be easily calculated starting with figure 5.7 and yields

$$p(\mathbf{O},\mathbf{q}|\lambda) = p(q_1)\cdot p(\vec{o}_{1,1}|q_1)\cdot p(\vec{o}_{2,1}|q_1)\cdot \prod_{t=2}^{T} p(q_t|q_{t-1})\cdot p(\vec{o}_{1,t}|q_t)\cdot p(\vec{o}_{2,t}|q_t), \quad (5.3)$$

considering the state sequence $\mathbf{q} = (q_1,\ldots,q_T)$. By summing up equation 5.3 over all possible state sequences $\mathbf{q}\in\mathbf{Q}$, the following production probability of the observation $\mathbf{O} = ((\vec{o}_{1,1},\vec{o}_{2,1}),\ldots,(\vec{o}_{1,T},\vec{o}_{2,T}))$ is derived:

$$p(\mathbf{O}|\lambda) = \sum_{q\in Q} \pi_{q_1}\cdot p(\vec{o}_{1,1}|q_1)\cdot p(\vec{o}_{2,1}|q_1)\cdot \prod_{t=2}^{T} a_{q_{t-1},q_t}\cdot p(\vec{o}_{1,t}|q_t)\cdot p(\vec{o}_{2,t}|q_t). \quad (5.4)$$

Each observation $\vec{o}_{n,t}$ with $1\le n\le 2$ is thereby a vector consisting of continuous normalised features. Within equation 5.4, it is clearly visible that the two observation streams $\vec{o}_{1,t}$ and $\vec{o}_{2,t}$ are independent of each other, while features within a single stream are depending on each other.

For the first observation $\vec{o}_{1,t}$, the best visual features $\vec{o}_{vis}^{best}$ derived from the sequential forward selection are used. The best acoustic features set $\vec{o}_{ac}^{best}$ is applied to the second observation stream $\vec{o}_{2,t}$. This setting leads to an accuracy of 51.2% for the best evaluated setting of parameters during the evaluation. This accuracy is achieved by the graphical model using 20 states, ten Gaussian components for the visual stream $\vec{o}_{1,t}$, 20 Gaussian components for the acoustic stream $\vec{o}_{2,t}$ and 20 iterations for the EM training.

The confusion matrix in table 5.7 shows that the problem with the classes intoxicated and tired, which are under-represented in the data base, is not solved with the multi-stream approach. The f-measure for these classes is zero respectively 16%. Since the recall and the precision results change for each of the classes aggressive, cheerful and nervous in opposite direction, the f-measure varies insignificantly to the results from GM1 in table 5.2. Only the f-measures of the class neutral shows a significant decrease by more than 8%, since the recall is going down by 17% and the precision is improved by only 5%.

Table 5.7: Confusion matrix of the behaviour patterns are presented of the model GM3 when the best visual feature set is used in the first stream and the best acoustic feature set as second observation. Furthermore, the recall, the precision and the f-measure are calculated per class. The problem of the detection of the under-represented classes is unsolved. Furthermore, the results change only slightly compared to GM1 with the exception of the class neutral. *Aggr* stands for aggressive, *chee* represents cheerful, *into* is intoxicated, *nerv* is the short form for nervous, *neut* means neutral, and *tire* is abbreviation for tired. # is the total number of the segments per class. The last three short cuts stand for recall, precision and f-measure.

truth	aggr	chee	into	nerv	neut	tire	#	Rec	Pre	F1
aggr	**80**	11	0	2	1	0	94	85.1	51.9	64.5
chee	34	**50**	0	9	11	0	104	48.1	43.1	45.5
into	10	14	**0**	3	6	0	33	0.0	0.0	0.0
nerv	6	17	0	**64**	6	0	93	68.8	61.0	64.6
neut	14	20	0	21	**20**	0	75	26.7	44.4	33.3
tire	10	4	0	6	1	**2**	23	8.7	100.0	16.0

Overall the model GM3 achieves an accuracy of about 51%, which is a similar result compared to the best model GM1 with about 52%. The main drawback of this model is that the complexity is by far higher than for GM1, although the models are very similar. Furthermore, it does not face the challenge of detecting the classes intoxicated and tired.

5.2.4 Hierarchical Graphical Model (GM4)

The idea behind this graphical model is that the classification of behaviour can be split into two separate classification processes. The suspicious classes are aggressive, intoxicated and nervous and the neutral group consists of cheerful, neutral and tired. The splitting into two separate decisions, is useful to improve the confusion matrix, since many confusions are between classes which are assigned to suspicious instead to normal behaviour classes and vice versa. For example for the best GM1 in table 5.2, it is the case for the classes aggressive and cheerful where in total 49 out of the 198 segments are mixed up. The total number of confusions between suspicious and normal classes is 149. This means, that more than one third of all the 422 segments are confounded between suspicious and normal behaviour. On the other side, only 57 segments are mixed up within the same behaviour group. Thus, a major improvement is achieved when the confusion between the two behaviour groups is resolved. Moreover, a decision between only two classes is much easier to perform than a decision over six classes and the confusion between those two classes

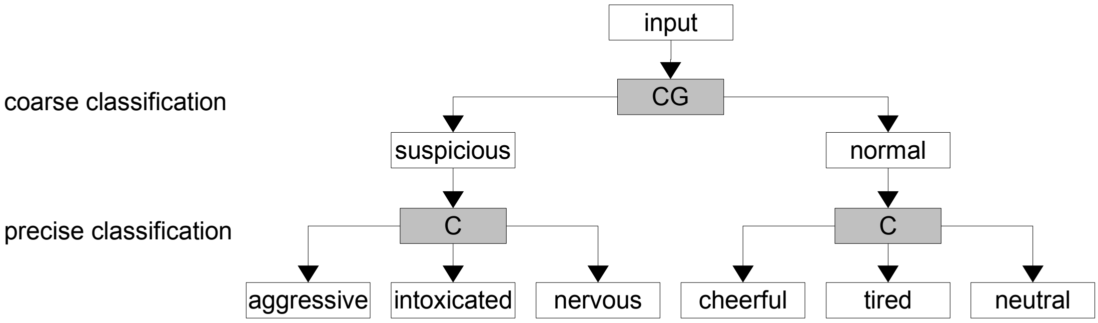


Figure 5.8: The decision structure for the hierarchical approach to behaviour detection. The behaviour of an observed person is assigned to the suspicious or the normal group in the first layer. This is enough information for some scenarios and the security officer is informed. The second layer, which is depending on the first one, splits up the classes again. Therefore, more detail about the behaviour is available for the security personnel.

should be low.

Figure 5.8 shows the decision structure which is implemented into a graphical model. The first important decision for most of the applications is, whether the current behaviour of the observed person is suspicious or normal. In the case of suspicious, an alarm can be activated and the security officer is informed about the position of the observed person. In some applications, this information is already enough and further information is not necessary for the security personnel. Since more information about the ongoing situation is always a benefit for the operation personnel, a more complex second decision is performed. This decision is depending on the first one and the classified class is further split into three more behaviour classes which describe the status of the observed person in more detail. Consequently, this approach is called a hierarchical one, because two decisions are performed and the second is depending on the first one.

In figure 5.9, the training structure for the implementation within GMTK is presented. The model has two observations like the model GM3 in figure 5.7. The main difference compared to GM3 is, that it has two observations which are connected to only one vertex SP and all the other vertices are existing only once. In the structure of GM4 the two decisions are made possible by adding additional arcs and vertices. The first decision about the behaviour group is possible within the vertex CG. The vertex C discriminates between the three classes within each of the behaviour groups. During the EM-training, the vertices CG and C are only depending on the vertex FC, because the annotation contains the needed information about the behaviour group and the behaviour class. The vertices CP, SP and ST are similar to the simple graphical model used for GM1. For the second decision within vertex C, this structure is copied and the vertices are renamed to $CP2$, $SP2$ and $ST2$. Therefore,

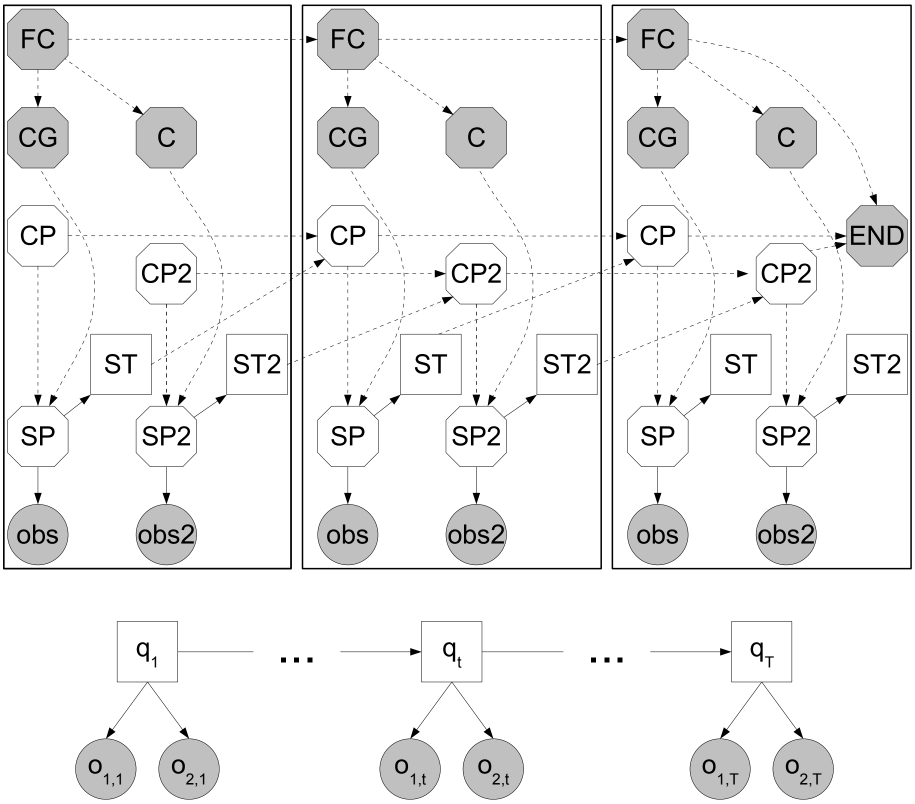


Figure 5.9: Structure for the training of the hierarchical model GM4. It consists of two simple graphical models GM1 as presented in figure 5.3. These two models are connected only by the vertex FC. The first model consists of the vertices CG, CP, SP, ST and the observation *obs*. It is performing the decision about the two behaviour groups, suspicious and normal. The second model, starting with the vertex C, discriminates between the three behaviour classes within the suspicious or normal behaviour group. Due to this training structure, it is possible to train the hierarchical model in a single training process without a further requirement of annotation or training of an intermediate layer. The complex structure, which is needed for the implementation of the model in GMTK, is substituted for the calculation of the production probability of the model in the lower part. The simplification is similar to the two stream model in section 5.2.3, since the implementation for GMTK is integrated into the vertex q_t.

the structure below the vertex CG or C is similar to GM1 with the difference that it exists twice. The more complex implementation in GMTK does not affect the substitution of the model to the same short form as for the two stream model presented in figure 5.7.

The production probability of the substituted GM, as illustrated in the lower part of figure 5.9, yields

$$p(\mathbf{O}, \mathbf{q}|\lambda) = p(q_1) \cdot p(\vec{o}_{1,1}|q_1) \cdot p(\vec{o}_{2,1}|q_1) \cdot \prod_{t=2}^{T} p(q_t|q_{t-1}) \cdot p(\vec{o}_{1,t}|q_t) \cdot p(\vec{o}_{2,t}|q_t), \quad (5.5)$$

considering the state sequence $\mathbf{q} = (q_1, \ldots, q_T)$. By summing up equation 5.5 over all possible state sequences $\mathbf{q} \in \mathbf{Q}$, the following production probability of the observation $\mathbf{O} = ((\vec{o}_{1,1}, \vec{o}_{2,1}), \ldots, (\vec{o}_{1,T}, \vec{o}_{2,T}))$ is derived:

$$p(\mathbf{O}|\lambda) = \sum_{q \in Q} \pi_{q_1} \cdot p(\vec{o}_{1,1}|q_1) \cdot p(\vec{o}_{2,1}|q_1) \cdot \prod_{t=2}^{T} a_{q_{t-1},q_t} \cdot p(\vec{o}_{1,t}|q_t) \cdot p(\vec{o}_{2,t}|q_t), \quad (5.6)$$

while each observation $\vec{o}_{n,t}$ with $1 \leq n \leq 2$ is a vector consisting of continuous normalized features.

In figure 5.10, the decoding structure of the GM4 is presented. Compared to the training structure, the vertex FC is removed and two arcs are added. The first connects the vertex CG in two sequential chunks. The second is needed for the dependence of the vertex C, which discriminates between the three behaviour classes, depending on the vertex CG, that performs the decision between suspicious or normal behaviour. Due to this arc, it is possible to implement the hierarchical structure within a single graphical model. The rest of the model is similar to the decoding structure of the simple graphical model GM1 presented in figure 5.4, with the only difference that each vertex exists twice. One vertex is always assigned to the decision about suspicious and normal behvaiour groups and the second vertex of the same type to the decision between the three classes within one of the selected groups.

For the evaluation of the model, the best features selected by sequential forward selection for the acoustic, the visual and the audio-visual features, are used. Two observation vertices are available in the model and thus the same feature set is used for both observation streams $\vec{o}_{1,t}$ and $\vec{o}_{2,t}$. Table 5.8 shows, that the two class decision is a hard task. The base line for it is about 52%, as the number of segments is not equally distributed between suspicious and normal behaviour. The best result is 67.1%, which is achieved for the audio-visual features. The parameters are ten states for the behaviour group variable CG, 20 states for the class variable C, ten Gaussian components for the first stream $\vec{o}_{1,t}$, 20 Gaussian components for the second stream $\vec{o}_{2,t}$ and ten iterations are applied to the model for the best audio-visual feature set. The best accuracy of 63.5% for the single modality is achieved by the acoustic feature set. The best set of parametes for the acoustic feature set is ten states for the behaviour group variable CG, 20 states for the class variable C, 20 Gaussian components for the first stream $\vec{o}_{1,t}$, ten Gaussian components for

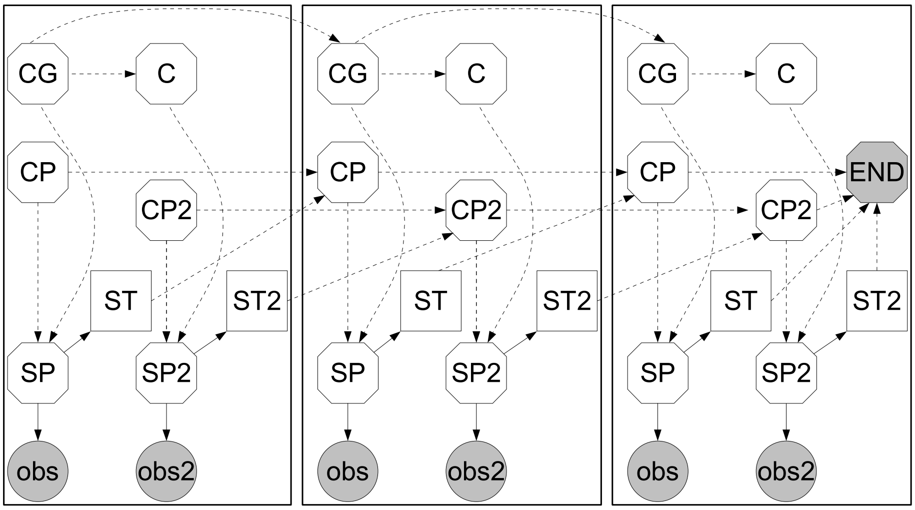


Figure 5.10: The decoding structure of the GM4. The additional arc between *CG* and *C* makes it possible to perform the hierarchical decision about the behaviour. The structure is similar to GM1, with the difference that all vertices exist twice. One is for the decision about the behaviour group and the second for the final decision about the behaviour class. The arcs are totally similar within one of the copies of GM1.

the second stream $\vec{o}_{2,t}$ and 20 iterations for the EM training. The visual features are performing insignificantly worse. Consequently, the improvement from the base line is about 15% for the feature sets and about 11% for both single modalities. For the six classes decision it looks slightly different, because the acoustic set performs worst with an accuracy of 48.8%. The visual set achieves a result of 49.3%, which is marginally better than the acoustic ones. The performance of the audio-visual feature set is best and achieves an accuracy of 53.1%, which is the best one when the feature set derived by the sequential forward selection is applied. Therefore, the accuracy is higher than the one achieved by the GM1 using the same features. It is also better than the results of the GM1 using the feature set derived by principal component analysis. In both cases the improvment is only marginally, since it is below 1% absolute. When the single modality results are compared to GM1, it is visible that the visual set performs insignificantly better and the acoustic one slightly worse. Thus, no tendency for the use of the different feature sets is found.

The confusion matrix for the best audio-visual feature set is presented in table 5.9. For the two class decision the the number of confusions compared to model GM3 is reduced. Instead of 149 confusions between the behaviour groups suspicious and normal, the number is going down to 139 which is an improvement of 6.7%. This means, that the system performs better for the two class decision as the GM3 does, but the improvement is not as big as expected. The number of misclassifications

Table 5.8: The performance measures of the hierarchical GM4 are presented. Since two classifications are performed, results are listed for the two class decision suspicious versus normal and the six classes decision. The feature selection is performed by the sequential forward selection and the best sets for the visual, acoustic and audio-visual features are used during the evaluation. Since two observation vertices are used, always the same features are applied to each of the vertices. The best performance for the two class decision is achieved with an accuracy of 67% for the audio-visual feature set. Compared to the distribution between suspicious and normal behavior it is about 15% above the base line. For the six classes decision the model achieves a better result than the GM1, but the improvement is only about half a percent, which is not significant. The accuracy of 53.1% is the best one, which is achieved during this thesis by using the feature sets derived by the sequential forward selection. All results are measured as recognition accuracy rates.

feature		$\vec{o}_{vis}^{best}$	$\vec{o}_{ac}^{best}$	$(\vec{o}_{vis}^{best}, \vec{o}_{ac}^{best})$
two class	ACC [%]	62.8	63.5	67.1
six class	ACC [%]	49.3	48.8	53.1

of GM4 within the correct behaviour group is 59 and therefore it is increased by two compared to GM3. When the results are compared with the confusion matrix of GM1 it looks different. GM1 misclassifies only 136 segments between the two behaviour groups and thus it performs better than the complex model GM4. The small performance improvement of GM4 compared with GM1 for the six classes is explained by 59, respectively 65 segments which are mixed up within each of the behaviour groups. Going into detail for the six classes decision, shows that the performance of the f-measure is improved for the classes aggressive, cheerful and neutral compared with the GM1. Only for nervous, the score is going down by about 7%. Furthermore, it is unchanged for intoxicated and tired, which means that the problem of the under-represented classes is still unsolved. The improvement of the f-measures for the classes aggressive and neutral are explained by the better precision compared to GM1. For cheerful the recall is going up about 11% and at the same time the precision is reduced by only 5%. Both measures, recall and precision, are going down for the class nervous by 4%, respectively 11%.

The hierarchical model is not only evaluated with the feature sets which are derived from the sequential forward selection, but also with the features which are transformed by the principal component analysis. Table 5.10 shows the various accuracies of the evaluation of the different parameter sets and different number of features. Since the evaluation of the simple GM1 shows that the best performance is achieved with only 20 features, only the first 30 features are tested. The accuracy is located in a range between 49.8% and 53.3% which is compared to the simple GM1

Table 5.9: Confusion matrix of the behaviour patterns are presented of the model GM4 when the best audio-visual feature set is used in both observation streams. Furthermore, the recall, the precision and the f-measure are calculated per class. In total 139 confusions occur between the two behaviour groups suspicious and normal. Within one of these groups, 59 segments are mixed up. Overall the performance is better than the result of GM1, but the hierarchical approach does not perform better for the separation of the two behaviour groups. The f-measure of the classes aggressive, cheerful and neutral are improved compared to GM1. For nervous it falls and for the under-represented classes intoxicated and tired the same results are achieved. *Aggr* stands for aggressive, *chee* represents cheerful, *into* is intoxicated, *nerv* is the short form for nervous, *neut* means neutral, and *tire* is abbreviation for tired. # is the total number of the segments per class. The last three short cuts stand for recall, precision and f-measure.

truth	aggr	chee	into	nerv	neut	tire	#	Rec	Pre	F1
aggr	**82**	8	0	1	3	0	94	87.2	62.6	72.9
chee	27	**53**	0	11	13	0	104	51.0	40.5	45.1
into	5	20	**0**	2	6	0	33	0.0	0.0	0.0
nerv	5	15	0	**55**	18	0	93	59.1	62.5	60.8
neut	10	20	0	13	**32**	0	75	42.7	43.8	43.2
tire	2	12	0	6	1	**2**	23	8.7	100.0	16.0

better. GM1 has achieved a range which is between 46.7% and 52.6%. Similar to GM1, the best performance is realised when the first 20 features derived from the principal component analysis are applied. In this case, the features are assigned to both observation $\vec{o}_{1,t}$ and $\vec{o}_{2,t}$. The parameters for the best model using the first 20 features are: one Gaussian component for the first stream $\vec{o}_{1,t}$, ten Gaussian components for the second stream $\vec{o}_{2,t}$, 20 states for the class group variable CG, 20 states for the class variable C and ten iterations for the EM-training. The rate of 53.3% is the best accuracy, which is achieved during the evaluations on this data base conducted throughout this thesis. The performance is insignificantly better than the best accuracy achieved by the same model using the features derived by the sequential forward selection. Compared to the much simpler GM1 the performance is improved by only 0.7% and thus the enhancement is marginally.

The confusion matrix for the GM4 using the first 20 features derived by principal component analysis is shown in table 5.11. 129 confusions between the two behaviour groups is the lowest number of all evaluated models within this thesis. This is an improvement of 13.4% compared to GM3, which mixes 149 segments up. The accuracy of the two class decision is equivalent to 69.4%. As 68 segments are confound within one of the behaviour groups, the overall performance is only marginally better than

Table 5.10: Experimental results for the graphical model GM4 using feature sets created by applying the principal component analysis, which is described in section 3.4.2.1. The various feature sets and parameter combinations have been tested during the evaluation. The best model uses the following parameters: one Gaussian components for the first stream $\vec{o}_{1,t}$, ten Gaussian components for the second stream $\vec{o}_{2,t}$, 20 states for the class group variable CG, 20 states for the class variable C, ten iterations for the EM-training and the first 20 features. The accuracy of 53.3% is the best, which is achivied during the evaluations performed for this thesis. All results are measured as recognition accuracy rates.

features	1–4	1–6	1–8	1–10	1–15	1–20	1–25	1–30
ACC [%]	50.2	50.2	49.8	51.0	52.4	**53.3**	52.1	51.4

for other models. The GM3 has only 57 confusions within one of the two behaviour groups, which means that the number is increased by 16.2%. The comparison with GM1, which also uses features derived by principal component analysis, shows that the confusion between the groups is reduced by eight but the confusion within the groups rises by five. Consequently, the improvements of this model compared to the GM1, which consists of only half of the amount of vertices, is insignificant. The duplication in the structure of the GM4 demands more training data and consumes more time during training and evaluation. Compared to the GM1 the amount of correct classified segments is increased by only three. The f-measure rises for the class nervous by 8% due to the fact that the recall and the precision are rising about 3%, respectively 11%. For the classes cheerful, intoxicated, neutral and tired, the change is below 1% and thus not significant. Therefore, the under-represented classes are still not detected and the problem is still unsolved. Only the f-measure for the class aggressive is reduced by 2%, as the recall and the precision is also decreased by about 2%.

5.3 Comparison of the Results

Table 5.12 presents a summary of the number of confusions between the behaviour groups and within one group. It also shows the number of confusions within the correct classified groups. Moreover, it lists the number of correct classified segments and the accuracy of those. All the numbers are taken from the best model of each structure. For the first four columns, the audio-visual feature set is applied, which is derived by the sequential forward selection. In the case of the last two, the principal component analysis is used for the feature reduction. The simple GM1 achieves for both feature reduction approaches a performance, which is only 0.7% below the more

Table 5.11: Confusion matrix of the behaviour patterns are presented for the model GM4 when the first 20 features, derived from the principal component analysis, are used. Furthermore, the recall, the precision and the f-measure are calculated per class. 129 is the lowest number, which has been achieved for confusions between the behaviour groups during this thesis. The number of confusions within one of the behaviour classes is 68 and therefore about 16.2% above the best results achieved by GM3. Only the f-measure of the classes nervous and aggressive are changed about 8%, respectively −2%. *Aggr* stands for aggressive, *chee* represents cheerful, *into* is intoxicated, *nerv* is the short form for nervous, *neut* means neutral, and *tire* is abbreviation for tired. # is the total number of the segments per class. The last three short cuts stand for recall, precision and f-measure.

truth	aggr	chee	into	nerv	neut	tire	#	Rec	Pre	F1
aggr	**75**	12	0	4	3	0	94	79.8	64.1	71.1
chee	26	**48**	0	12	18	0	104	46.2	41.0	43.4
into	6	16	**0**	1	10	0	33	0.0	0.0	0.0
nerv	2	7	0	**70**	14	0	93	75.3	65.4	70.0
neut	6	26	1	12	**30**	0	75	40.0	38.5	39.2
tire	2	8	0	8	3	**2**	23	8.7	100.0	16.0

complex GM4. Since the difference of these approaches is too small, the performance is not significanly decreased compared to the models based on structure GM4. The improvement of the principal component analysis compared to the sequential forward selection is for both models GM1 and GM4 about 0.2% or one instance more, which is classified correctly. Again, this is not significant but a tendency is visible since it happens for most of the different model paramters which have been evaluated during this evaluation. The performance of the model GM2 is nearly decreased by 8% compared to the simpler GM1. The model GM3 performs about 2% worse than the GM4. Thus the results for the principal component analysis are not presented.

In all confusion matrices the major problem of the evaluation is shown. It is, that the classes intoxicated and tired are not detected at all. This is due to the fact, that only 33, respectively 23 segments are available in the whole data base of 422 segments. Therefore, the graphical models do not adapt to these two classes during the training. In other words, the other four classes are still too general for the behaviour detection problem and still fit too similar observations for the two under-represented classes. Moreover, the table 5.12 points out, that the next major problem is about the confusion between suspicious and normal behaviour groups. In the best case 129 segments are mixed up between these two groups. These are about 30.6% of the segements which are assigned already to the wrong behaviour group, even if the decision is only between the two available groups. The range

Table 5.12: The first line shows the number of confusions between the two behaviour groups suspicious and normal. In the next line the number of segments are listed which are mixed up within one of the selected behaviour groups. The second group points out where confusions occur within the correct classified behaviour groups. The last two lines show the number of correct classified segments, respectively the accuracy. All the numbers are taken from the best model of each structure and for the two different feature reduction approaches. The best performance is achieved by the structure GM4, using the features derived by the principal component analysis. The same model also produces the fewest number of confusions between the two behaviour classes, but has the second highest number of confusions within the two groups. The model with the fewest confusions within a group is GM3. It also achieves the lowest number of confusions in the normal group. In the normal behaviour group, both models based on the structure GM4 perform best for the number of confusions.

	SFS				**PCA**	
	GM1	GM2	GM3	GM4	GM1	GM4
confusion between groups	136	153	149	139	138	**129**
confusion within groups	65	80	**57**	59	62	68
confusion in suspicious group	20	29	21	**13**	17	**13**
confusion in normal group	45	51	**36**	46	45	55
# correct segments	221	189	216	224	222	**225**
accuracy	52.4	44.8	51.2	53.1	52.6	**53.3**

of confusions between both behaviour groups is located between 30.6% and 36.3%. In the worst case more than one third of the segments are mixed up. The GM2 has with 153 confusions the worst performance. It also has the highest amount of confusions within the two behaviour groups, with 80 segments. The lowest error achieves the GM3 with 57 segments. Thus, the error of mixing up the classes even when the correct behaviour group is selected, is still located in a range from 13.5% to 19.0%. The final problem, which is visible in the table 5.12 is, that inside the normal behaviour group many segments are misclassified. For GM4 for example, 55 out of 68 segments, which are mixed up within both behaviour groups, are from the normal behaviour group. Consequently, 80.9% of these errors are generated inside this group, which is also the highest ratio during this evaluation. The lowest percentage achieves the model GM3 with 63.2%.

The whole evaluation does not take into account that occlusions or errors with finding the faces can occur [ASHR09]. For a robust and stable system, it is very important to handle this, because it can lead to detection errors.

5.4 Comparison with other Approaches

In the past, other promising approaches have been evaluated on the ABC corpus and those will be compared to the results achieved during this work. The best result in this work is achieved by the GM4 using 20 features derived by principal component analysis. It reaches an accuracy of 53.3%.

In [AHSR09], an accuracy of 52.6% is presented when HMMs are used. These HMMs have been implemented with HTK [YEH$^+$02]. For the features the same 81 visual features have been applied as used in this work. The description for these features can be found in section 3.2.2. Since the gaps between the HMMs and the best model of this work is only about 0.7% the improvement of the performance is not significant. On the other hand, the GM4 uses only 20 features, which leads to lower computational effort and the computational power is also reduced. Thus, GM4 should be prefered to the HMM presented in [AHSR09]. When the HMM is compared to GM1, it performs about 3.5% better. It looks different when the number of used feautres are compared, since the HMM uses 81 features and the GM1 applies only eight visual features. The simple GM1 can be seen as a HMM implemented in GMTK and therefore the performance should be similar with the HMMs from [AHSR09]. The performance gap has due to with the integration of HMMs in various toolkits. It is not the first time that a mismatch between similar models evaluated with HTK and GMTK occurs. The performance gap between these two implemenations has been analysed for online recognition of whiteboard notes in [SHBR09]. To sum up, graphical models are performing on a similar level as HMMs, but use by far less features.

There are also other approaches with static classifiers as Support Vector Machines and different modalities of features which are using the ABC corpus. In [Ars10], the first comparable result is performed by a simple Support Vector Machine (SVM) approach which is trained on the same visual features as used in this work (see section 3.2.2). The accuracy of this approach is 60.1% and therefore outperforms all the dynamic classifiers evaluated in this thesis.

Another visual feature set, which is evaluated in [SWA$^+$07] is created by deformable models. These deformable models are fitted to the passengers' faces in order to extract the features. Since the number of extracted features is huge, a sequential forward selection has been performed to reduce the feature space to 157 dimensions. After the reduction, a time series analysis is performed over each feature for all the available segments from the data base. The used SVMs perform with 61.1% only marginally better as the approach with the simple global motion features. Since the computational effort for the sequential forward selection and the time series analysis is massive compared to the extraction of global motion features, this approach is not the first choice.

Another visual approach which is comparable to the hierarchical graphical model developed in this thesis, is presented in [AHSR09]. The approach classifies the 81 global motions in two layers. The first layer decides between the two behaviour groups suspicious and normal. Within each of these groups a second decision is performed about three classes. The first decision results in an performance of 87.9% which is more than 18% better than the performance of the best hierarchical GM4 presented in this work. Since only 25% of the features are used by GM4, a small performance drop could be considered, but this big drop is too much. Thus, the performance of the GM4 is far below the hierarchical SVM approach. The recognition rate of the hierarchical SVM approach is 74.9% and therefore outperforms GM4 by more than 21%. Form other results of this SVM approach, which are presented in [Ars10], it is clear that the performance of the decision within the normal behaviour group is the most difficult one.

In [SWA$^+$08], an acoustic behaviour detection system is presented. This approach extracts a large set of low level audio descriptors and functionals. These features are comparable to the features used in this work. More details about the acoustic set can be found in section 3.1. The experiments result in an accuracy of 73.3%.

The best results for this data base is presented in [WSA$^+$08]. The audio-visual approach, which combines the work from [SWA$^+$08] and [SWA$^+$07], achieves an accuracy of 81.1%. This approach uses more complex facial features, as active appearance models [BH05], and a time series analysis over all the audio-visual features.

Graphical models and HMMs are dynamic classifiers and these are by far outperformed by static classifiers as SVMs. The approaches with the hierarchical structure improve the results of both classifier types. Probably, the best results are achieved when an audio-visual feature set is used as an input for a multi-layer SVM classifier.

5.5 Outlook

The accuracy is so low, because one third of the segments are assigned to the wrong behaviour group and another 15% are detected as the wrong class, even if the correct behaviour group is detected before. These are also two points, where further research can be conducted to achieve better results. One idea would be to use different types of features, for example features derived from active appearance models. A second is to add more semantic information and create a model which can deal with semantic information. For the case, that more training material is available, more complex models could be tested. A model structure, which uses audio and visual features, both separately derived by principal component analysis as two independent observation streams for each of two hierarchical observations could be tested. Furthermore, it would be possbile to create models, where not only two consecutive chunks are connected, but also more different streams could be connected. A data base could be improved to recordings from real flights with the drawback that very few suspicious situations will occur. Moreover, an approach, which takes into account that not in every segment the face is found by the visual feature extractor and on the other hand not each segment audio data is available could help to improve the performance.

6 Automatic Video Editing in Meetings

Various projects, such as the meeting project of the International Computer Science Institute at Berkley (ICSI) [MBB$^+$03], Computers in the Human Interaction Loop (CHIL) [WSS04] or Augmented Multi-party Interaction (AMI) [CAB$^+$06] investigate the use of modern pattern recognition tools to increase the efficiency of meetings. The idea arises from the problem that many people think that most meetings are just a waste of time [HNR06] and therefore should be automatically processed, for example for meeting browsing or remote participation. On the other hand, meetings are often mandatory for many of us and consume large parts of our working days. In this section, the multi-modal problem of selecting one relevant field of view from various available cameras in a meeting room is addressed. The problem derives from the fact that video conferences are getting more popular at the same time as it is getting more and more common that meeting rooms are equipped with various recording devices. This points out the two main usage scenarios of the system: On-line video conferences [SP85] and browsing of past meetings [WFG04].

Most of the current video conference systems transmit all available visual participants' recordings to each attendee and display all of them. The main difference between the systems is the way these are presenting the video data. Normally, cheap systems show different small windows on a computer monitor, while high end systems are using multiple large screens for that purpose. These approaches are limited to a few participants, because the video of the individual participant is getting smaller if more persons are connected. Few other systems are using the audio channel for selecting the stream and therefore they transmit only one video to all participants, which saves bandwidth. The main drawback of this approach is, that interesting non-verbal gestures are left aside, because these cannot be detected from the acoustic channel and thus cannot be recognized in small videos. This shows the multi-modal nature of meetings, thus it can be very important to show people who currently do not speak. In talk-shows for example, professional editors follow this rule and show facial reactions and gestures from other participants [Bel05].

The other scenario for automatic video editing is in the field of meeting recordings [JBE+03]. The capturing of meetings is very common, but an unsolved problem is how to access the recorded data in a useful way. For this purpose, meeting browsers [WFG04, WFTW05] have been developed. The browser displays the content of the meeting, but it is still very hard to watch several video streams at the same time. Normally, a remote attendee will miss some important visual hints, which a participant is able to recognize if he attends to the meeting personally. This is one of the main reasons why research in multi-modal approaches to solve problems in meeting scenarios is performed.

For both scenarios, the problem of selecting a camera is the same: for each time frame we need to choose one video mode which represents best what is happening in the meeting. We refer to the different cameras, combinations of cameras, and pictures of slides as video modes. For details about video modes and the annotation of them see section 2.1.3.1. This mode is transmitted to the other participants in the case of a video conference or stored for later browsing. Therefore, the problem can be described as video editing. Video editing will help you to catch up quickly after a missed meeting, will make it easier to attend the meeting by using small screens, for example on a cell phone, which will reduce the required bandwidth for mobile communications and will save storage capacities.

6.1 Related Work

The problem of automatic video editing is interesting for businesses, but only little research has been carried out in this field so far. In [LKF+02], a single camera with very high resolution captures the whole meeting and virtual cameras, which show parts of the meeting, are created from this single video stream. A virtual camera is finally selected, which helps to present the meeting to various remote participants. The system can work in the range of being fully automatic to being handled manually. In [LK03], the same group introduces a system that learns from human operators which virtual camera should be selected. The interesting regions in the frame, which people want to watch, are labelled by 14 persons in 22 images. By using that few images, important gestures are missed, because these are not visible in an image and therefore the regions have to be selected in the video. This leads to several virtual cameras and will make the selection more difficult. The selection is performed by using probabilities for each available virtual camera. The system is extended to more cameras with high resolution in [LSK+05]. Most of the time, the region around the presentation screen is selected as the output of the system. It is not really feasible to create close-ups from each of the participants in the meeting, thus interesting gestures, like shaking the head, are not visible to the remote participant. The introduced system fulfills many well-known composition rules. In a totally different field, automatic camera selection is performed in [SNVF03]. A multi camera system records an outdoor scenario for video surveillance. The criteria for selecting a camera is depending on the quality of the detection of the person in the video stream. The main focus of this work is to become more reliable under changing weather conditions. In [Uch01], a user study for developing a user interface for video editing in meetings was conducted with professional editors as subjects. The interface should support non professional editors by the task of video editing. For home videos a system called "Hitchcock" was described in [GBC$^+$00]. This system is searching for unsuitable frames in a single video stream, and applies standard editing rules which leads to a segmentation of the stream. All segments are ranked and the segments with high scores are selected for the final video. The system is not selecting different cameras, it is summarizing the home video. Early work in the field of video editing is based on rules. Three examples which work on the same data base are described closer upcoming paragraphs.

In [Sum04], a rule based system is introduced for video editing in lecture halls and meeting rooms. The system has two operation modes; one for a video conference and the second for stored meetings which are performed in the past. The algorithm of the system is taking into account some technical aspects of video editing, as well as aesthetic ones. For example, the duration of showing the same camera is the most important aesthetic aspect. From the technical point of view, it is important: Who is speaking? How long is someone speaking? Who is moving the head or hands? All this information is combined to a weight for each camera at each time frame and the camera with the highest weight is selected. The system is also using virtual cameras for creating more camera changes, if for a long time no camera change happens since, for example, the same participant is still speaking. The main drawback of this system is that the selected camera is highly depending on the current speaker and important non verbals, as shaking the head or nodding, are not shown in the edited video. Visual clues can be added easily to the system if they are available as annotations. All the information which is used for the camera selection are hand annotated and no automatic detectors have been created. The only available evaluation is that various subjects have watched the created videos and have ranked them. Therefore, no comparison to other approaches using the same data base is possible.

In [AHHSR06], an approach to video editing for smart meeting rooms is introduced, which is based on different thresholds for different extracted features. Therefore, different acoustic, visual, and semantic features based on time frames are extracted from the meeting data. The influence of different feature types on the task of automatic video editing is investigated. First tests show that frame based video editing by a rule based system can lead to unintentional twitches. Therefore, additional features are derived by windowing a range of subsequent frames for acoustic and visual features. For the selection of the video mode, it is tried to choose the most relevant camera for each frame in the meeting. This is done by mapping each close-up camera as well as each lapel microphone to one participant. In case a monologue takes place, the close-up camera of the according person is chosen and therefore the camera is the most relevant one. The centre camera is used if the group is in the state of discussion, whiteboard or presentation. The selection is performed by choosing the "most active" person, that means for the acoustic feature for example the audio channel which is the loudest. This process is performed on frame based data as well as on windowed data. The results of the approach are: The direct use of acoustic and visual features is not leading to success, because of a high number of camera changes per minute. The problem is solved by using windowed features. On the other hand, too few camera changes occurred when semantic features are used, since the group state only changes every minute. A second evaluation is done by performing late fusion of the results of different features. The outcome is, that the combination of audio and video information is required for a sufficient selection task. In [AHHSR06], only the number of shot changes per minute and the time of the longest shot are objective measures which give an impression about the video quality. Furthermore, a user rating is given which expresses the significance of the evaluated features from a user's point of view. Therefore, a comparison of these results with the results achieved in this work is not possible.

In [AHHM$^+$07], a video editing system based on HMMs [Rab89] is introduced. For each of the seven video modes a HHM is trained using the EM algorithm [DLR77]. The best sequence of video modes is derived by Viterbi decoding [Vit77] and of this sequence an output video is created. The system uses low level acoustic and visual features which are similar to the features used in this work. Various combinations of features and many different parameter settings have been evaluated. The best result is a frame error rate of 47.9%.

In [Len07], an approach is presented which uses an Asynchronous Hidden Markov Model [Ben03]. It works on the same data base and the same features as used for the system described in [AHHM$^+$07] and the evaluation of this thesis is based. With a frame error rate of 32.3% it performs by far better than all known approaches, but still only two third of the frames are classified correctly. The system is state of the art for the AMI database, which is the same as used in this thesis .

Graphical models are used for video editing in [AH08]. These models are capable of automatic segmentation and classification of the seven video modes. The best performance is achieved by a combination of acoustic and visual features with a frame error rate of 47.7%. The data base and the features which have been used in the work are the same as in this thesis, thus a comparison is feasible. Since the implementation of graphical models in GMTK [BZ02] is not yet as optimised as for HMMs in HTK [YEH$^+$02], the performance can be further improved in the future by optimising the used toolkit, especially the implementation of the EM training.

In [AHR08], a system using support vector machines [Vap95] is presented. It uses global motion features and the person speaking features for the automatic selection of the video modes. The work is using the same features and the same data base

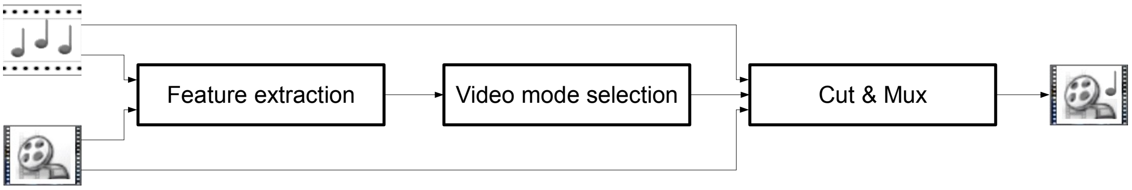


Figure 6.1: Block diagram of the video editing system is depicted. It shows the seven video streams and the four audio streams which are used as an input to the feature extraction and the cut and multiplex unit. The extracted features are the input of the selection block which selected the most relevant video mode. This video mode and the mixed audio stream of the participants are combined to the final output video.

as used in this thesis. The performance of the approach using the support vector machines is with a frame error rate of 39% better than the HMM approaches, but worse than the Asynchronous Hidden Markov Model.

6.2 The Video Editing System

In figure 6.1, an overview of the video editing system is presented. The first step is, that from the four audio channels and the seven video cameras various audio-visual features are extracted. More details about the features can be found in chapter 3. The second building block contains the classification and segmentation process which makes a decision about when and which video mode should be shown. The final block uses the audio and video sources and the selected video mode sequence to create an output video. This video contains the mixed audio sources of each participant and the most relevant video mode for each frame. The structure is similar for most approaches described in the related work section as well as for the developed models in this thesis. For similar approaches, the feature extraction is done by hand since the input data is not created reliably by automatic systems.

Figure 6.2 shows an example of an output video. It contains a sequence of shots over the time axis which are for a better illustration only presented by a single frame for each shot. Each shot is automatically selected by the video editing system using a combination of features described in section 3. The duration of one shot, which always contains only a single video mode, is also found automatically by the system during the decoding process.

6.3 Used Graphical Models

In this section different graphical models are presented which have been evaluated during this thesis. Since for the automatic video editing a segmentation of the

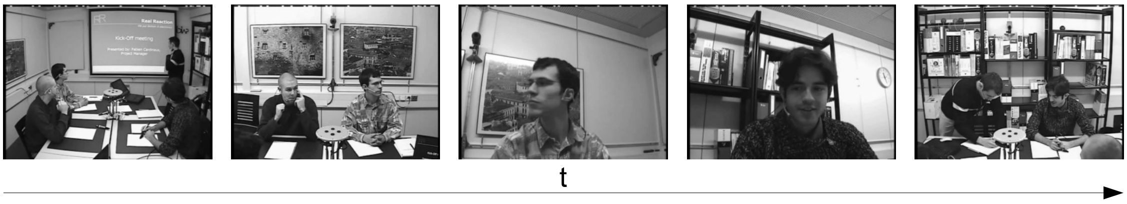


Figure 6.2: An example for a video sequence, which is represented by single frames of each selected video mode. In the video each shot has a different duration which is automatically detected.

meeting has to be performed, all models are capable of switching between video modes and finding the time frame for the change of the video mode. In this case a combined task of segmentation and classification has to be performed by the model. For this evaluation the results are measured as a frame error rate (FER) and as an action error rate (AER). The frame error rate counts all the correct selected video modes on a frame base and divides the number by the total number of frames. Therefore, the boundaries are very important for a low frame error rate since a temporal offset of a boundary leads to many wrong selected frames. For the action error rate, the correct boundaries are not as important as for the frame error rate, because only the sequence of video modes is taken into account for it. In case the correct sequence of video modes is selected the action error rate is zero, even when the boundaries are misdetected. The second and simpler test is to classify the correct video mode for all predefined segments. The boundaries of each segment are known by the system and only the correct video mode has to be found in this task. This can be done since the boundaries are available from the annotations for each of the meetings which are in the data base. This evaluation is measured with the recognition accuracy (ACC), which counts the correct classified segments and divides them by the total number of segments in one meeting.

For all evaluations, a six or a nine fold cross validation is performed and the average of the results over these folds are presented. Due to the available data set, which contains always four meetings where the same four participants are attending, six, respectively nine fold are possible without having the same persons in training and test. All the test and training sets are person disjoint, because a participant is not used at the same time in the test and training set. This is important for the evaluation since only little training material is available and the system has not the chance to adopt to participants. During the evaluation, many different combinations of audio, visual and semantic features are tested. Furthermore, the parameters of each model have been changed in a way that the best results are achieved.

For the next three subsections the six fold cross validation and seven video modes are used. Each of the seven video modes represents a single camera. This data set has a duration of only two hours. For the last subsection in this section a different

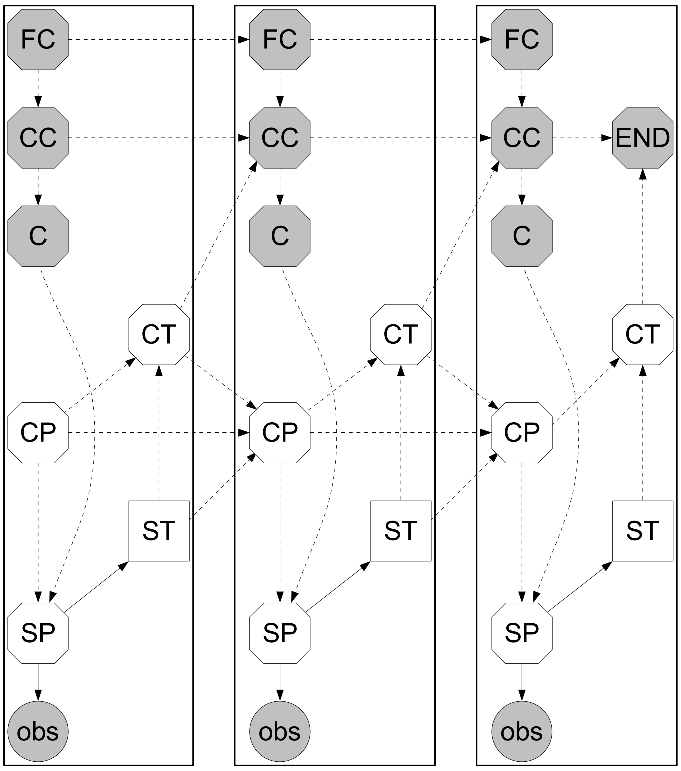


Figure 6.3: The training structure of the graphical model GM1 is shown. This structure makes it possible to train models with the necessary information about the correct boundaries between two video modes. An elaborate description of all the used vertices can be found in chapter 4.

setting is used. An extra video mode, which includes images of slides to the output video, is added and the nine fold cross validation is applied. This video mode is used for slides which are important for the ongoing meeting. The slides are visible in the centre camera but the text is too small to read it easily. Thus, eight video modes are available and the data base is extended to three hours. For the evaluation of the eight video mode setting the same graphical model structures are used as described in the next three subsections.

6.3.1 Single Stream Segmenting Graphical Model (GM1)

The training structure of the graphical model GM1, which is a continuous single stream model, is shown in figure 6.3. The structure is presented that way because it is possible to implement it in the same style in GMTK. In [BB05], a model is presented which is not capable of learning the segment boundaries, but a sequence of different segments can be learned. The structure shown in the figure is an improved version of this model. The three vertices *FC*, *CC* and *C* are necessary for the training, to make it possible to learn segment boundaries. In this case, the segment boundaries are the shot changes between cameras, which are again the points of time between two subsequent video modes. It is important for the segmentation task to learn the segment boundaries, because during the decoding the correct boundaries are necessary for low frame error rates. For more details about the structure see chapter 4.

The single vertex q_t contains all vertices of the model except the observation. This is possible, since all the information of the model can be summed up to the state of the model for a certain time frame t. The derived structure leads to the following production probability for the model with the sequence $\mathbf{q} = (q_1, \ldots, q_T)$ of all states which are run through

$$p(\mathbf{O}, \mathbf{q}|\lambda) = p(q_1) \cdot p(\vec{o}_1|q_1) \cdot \prod_{t=2}^{T} p(q_t|q_{t-1}) \cdot p(\vec{o}_t|q_t). \tag{6.1}$$

Substitution and marginalisation yields the following equation

$$p(\mathbf{O}|\lambda) = \sum_{q \in Q} \pi_{q_1} \cdot p(\vec{o}_1|q_1) \cdot \prod_{t=2}^{T} a_{q_{t-1},q_t} \cdot p(\vec{o}_t|q_t). \tag{6.2}$$

The marginalisation is done by summing up over all possible state sequences $\mathbf{q} \in \mathbf{Q}$ and the substitutions are taken from the forward-backward algorithm [Rab89]. The input of the model is the observation sequence $\mathbf{O} = (\vec{o}_1, \ldots, \vec{o}_T)$.

In figure 6.4, the decoding structure of the previous training structure is shown. These structures have been presented in [HAR09]. The main differences are that the vertex C, which represents the video mode, is not observed anymore and the vertices FC and CC are removed. Moreover, an additional arc between two consecutive vertices C is added. This is necessary, because the class has to be classified. Depending on the vertex CT, which controls the change of the video mode, the arc is either a deterministic or statistic type. This arc makes it possible, that the video mode is taken from the last vertex, in case no class transition in vertex CT occurs, or a new video mode is selected depending on the probability distribution of the video modes learned during the training.

Table 6.1 points out the results of the simple graphical model GM1. The first part of the table shows the performance of the low level features. The base line for the frame error rate is 71.4%, if only the video mode is used which occurs most in the annotations. Therefore, the skin blob features do not perform as well as the base line. The major problem of the skin blob features is, that the face and the hands are often misdetected. For example, even if a participant is not seated, the skin colour detector finds a face in the bookshelf, because some books have a similar colour as the skin. The shape of the detected skin blob is similar to the face and therefore it is not possible to remove the detected blob. Furthermore, the hands

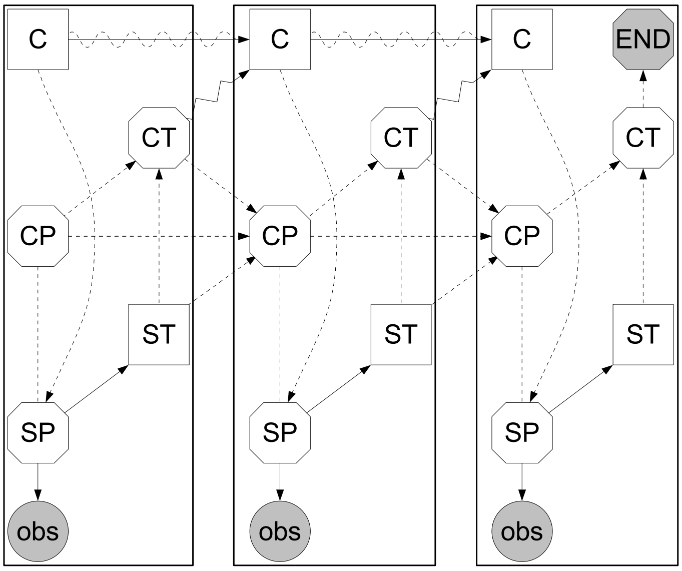


Figure 6.4: The decoding structure of GM1 as it is used by GMTK. The arc between two consecutive class vertices C switches between a statistic and deterministic dependence. This makes it possible to automatically detect segment boundaries in the test data. A description of all the used vertices and arcs can be found in chapter 4.

often overlap each other and the separation of them is not possible. This leads to a not detected hand or that some other blobs are misdeteced as the other hand. In the case global motion features are used, the system performs about 18% better than the one based on skin blobs. With a performance of 61.3% these features are about 10% below the base line, but still only one out of three frames is classified correctly. A further improvement is achieved when acoustic features are used. The frame error rate is going down to 50.1% and therefore every second frame is worng. The acoustic modality is more than 21% better than the base line. The action error rate shows that the acoustic features, used as an observation, create more changes of the video mode. Therefore, the rate is higher than the one for the global motion features. For the skin blob features, the action error rate is so high that the video is not watchable because of too many changes of the video mode in a short period of time. Sometimes in the video the changes happen so fast that it is impossible for a person to recognise what is happening when the video is watched.

The lower part of table 6.1 shows four results for an early fusion of different features. The best result, with an frame error rate of 45.3%, is achieved by the combination of acoustic and global motion features. This result is about 5% better than the best single modality model using the acoustic features. The comporatively low action error rate in comparison to the single modality models is also important, because too many changes of the video modes are very disturbing for the person

Table 6.1: The evaluation of different low level and semantic features applied to GM1 is shown here. Furthermore, combinations of low level features are presented. During the evaluation, various combinations of features and parameter settings for the models have been tested. Only the best results are listed in this table. The first three rows are using a single modality. The last four rows are achieved by a feature fusion, which uses the audio and the visual modality. AER stands for action error rate, FER means frame error rate and ACC is the recognition accuracy rate. AER and FER are results of the combined task of segmentation and classification. For the ACC performance, the boundaries are known and only a classification is performed.

Model	AER	FER	ACC
Audio (A)	158.7	50.1	47.6
Global Motion (GM)	82.4	60.1	34.5
Skin blob (SK)	600.3	78.6	16.8
Single stream (A&GM)	60.0	45.3	49.3
Single stream (A&SK)	66.1	61.3	36.2
Single stream (GM&SK)	65.2	60.3	31.0
Single stream (A&GM&SK)	174.4	48.7	43.7

watching the output video. A video with an action error rate of about 60% is already watchable for a viewer, even when still many frames are misclassified. Moreover, the combination of acoustic, global motion and skin blob features achieves a better result as all the single modalities with a frame error rate of 48.7%. It is an improvement of 1.4% to the acoustic features. The action error rate is also reduced to a level, where the output video is acceptable. The fact that the combination with skin blob features achieves good results is interessting, since the use of these features as a single modality leads to much worse results compared to all the other features. Still the combination of the skin blob features with all features lowers the performance, for example in combination with the acoustic features by more than 11% compared to the audio only model.

6.3.2 Multi Stream Segmenting Graphical Model (GM2)

In figure 6.5, the two stream graphical model GM2 is shown. The vertex q_t represents the whole structure of the model, which is similar to two models GM1. Most of the structure exists twice, only the vertex C exists once and connects both structures. The observation streams are called $\vec{o}_{1,t}$ and $\vec{o}_{2,t}$. These two observations are statistically independent and therefore different modalities can be connected. The modalities can be differentially weighted. This is an additional parameter, which has

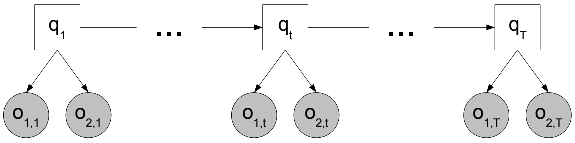


Figure 6.5: The block diagram of the two stream graphical model GM2. The two observation streams are statistically independent for each chunk. This is important to model various modalities separately and make it possible to weight them differently. The structure above with two independent observations is similar to the GM1 and therefore only the block diagram is drawn.

to be adjusted and optimised during the evaluation. In equation 7.3, the production probability for the shown model GM2 is presented.

$$p(\mathbf{O},\mathbf{q}|\lambda) = p(q_1) \cdot p(\vec{o}_{1,1}|q_1) \cdot p(\vec{o}_{2,1}|q_1) \cdot \prod_{t=2}^{T} p(q_t|q_{t-1}) \cdot p(\vec{o}_{1,t}|q_t) \cdot p(\vec{o}_{2,t}|q_t) \qquad (6.3)$$

with $\mathbf{q} = (q_1, \ldots, q_T)$ as the state sequence and $\mathbf{O} = ((\vec{o}_{1,1}, \vec{o}_{2,1}), \ldots, (\vec{o}_{1,T}, \vec{o}_{2,T}))$ as the observation. It is possible to have more than two observation streams as it is drawn in the figure. This influences the model in a way that additional streams $\vec{o}_{n,t}$ are connected. This means for the calculation of the production probability that $p(\vec{o}_{n,1}|q_1)$ and $p(\vec{o}_{n,t}|q_t)$ has to be added to equation 6.3. By marginalisation over all possible state sequences $\mathbf{q} \in \mathbf{Q}$ equation 6.3 yields to the following production probability

$$p(\mathbf{O}|\lambda) = \sum_{q \in Q} \pi_{q_1} \cdot p(\vec{o}_{1,1}|q_1) \cdot p(\vec{o}_{2,1}|q_1) \cdot \prod_{t=2}^{T} a_{q_{t-1},q_t} \cdot p(\vec{o}_{1,t}|q_t) \cdot p(\vec{o}_{2,t}|q_t). \qquad (6.4)$$

In equation 6.4, it is visible that the two observation streams $\vec{o}_{1,t}$ and $\vec{o}_{2,t}$ are independent, because the production probability is factorised for $p(\vec{o}_{n,1}|q_1)$ and $p(\vec{o}_{n,t}|q_t)$ with $n = \{1, 2\}$. The same factorisation is available for multi stream models with $n > 2$ which have been used during the evaluation, too. The performance of three stream models using low level acoustic and visual features is bad, that is why no results are presented here.

The results of the multi stream model GM2 are shown in table 6.2. Since only two stream models outperform the best single stream models, only those are listed. The combination of acoustic features for the first stream and global motion features for the second performs with a frame error rate of 44.6% better than the best single stream model GM1 with a rate of 45.3%. This is an improvement of 0.7%, which is significant but the viewer is not able to recognise the difference of the output video, since the changes mostly concern the boundaries of video mode changes. The

Table 6.2: The table shows two results of the best audio-visual feature combinations using GM2. Various other combinations and parameter settings have been evaluated, but the results are not better than the single stream models. The first model uses 20 states per class and the second five. The feature combinations contain acoustic (A), global motion (GM) and skin blob features (SK). AER stands for action error rate, FER means frame error rate and ACC is the recognition accuracy rate. AER and FER are results of the combined task of segmentation and classification. For the ACC performance, the boundaries are known and only a classification is performed.

Model	AER	FER	ACC
Two streams (A/GM)	60.8	44.6	52.9
Two streams (A/SK)	64.8	57.1	38.7

slightly higher action error rate does not really influence the quality of the output video. The second result is achieved for the combination of acoustic and skin blob features. The frame error rate and the action error rate are improved by 4.2%, respectively by 1.3% compared to the same features used in the single stream model GM1.

6.3.3 Two Layer Graphical Model (GM3)

In figure 6.6, the block diagram of the third graphical model is presented. This model has the capability to combine both semantic information and low level features during the detection process. The low level acoustic and visual features are described in section 3.1, respectively in section 3.2. Semantic information comes from group and person actions, person movements and slide changes. All this information can be derived by automatic systems, which have been developed in the last couple of years. In [Rei08], the group actions are recognised by using Markov random fields. Moreover, an approach applying graphical models is found in [AHDGP$^+$06] and both approaches are working on the same meeting data from the M4 corpus [Ren02a, Ren02b]. It would be feasible to transfer both approaches to the meeting corpus used in this thesis. An HMM based approach for the recognition of person actions is described in [AHHSR06]. During this thesis some tests have been conducted in the field of person movement recognition. HMMs have been used for the classification of the movements of the four participants in the smart meeting room. Slide changes are detected by an easy approach which was also developed during this thesis. More details about this can be found in [Zak07].

Most of these approaches can be implemented in GMTK, therefore the whole model can be represented in a more complex GMTK implementation. Due to following reasons, the complex structure is not presented in this thesis and for the proof

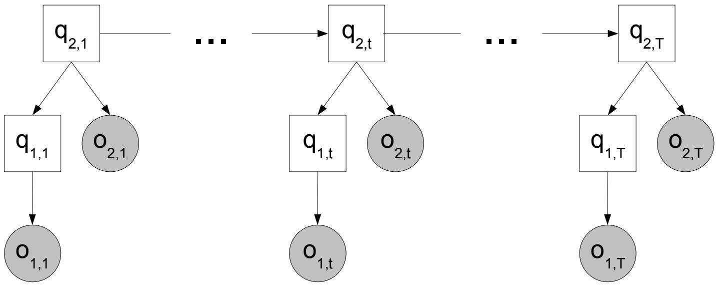


Figure 6.6: The block diagram of the two layer graphical model GM3 is shown. It combines semantic informations and low level descriptors within a single graphical model. The first layer detects various semantic informations and these results are used as additional inputs for the second layer, which uses the low level descriptors, too. The semantic informations are: group and person actions, person movements and slide changes. For example, the group action is derived by graphical models in [AHDGP^{+}06] or by Markov Random Fields in [Rei08]. Thus, the structure of the first layer can be represented in GMTK. Moreover, an implementation of the complete structure is feasible in GMTK, but the combined training of the whole model would not work, due to the huge number of dependencies and random variables. The second layer contains the same structure as GM1 or GM2.

of concept of the two layer model the hand made annotations are used:

- The training of the whole model is probably not feasible due to the huge number of dependencies and random variables.
- The amount of training data of the semantic information and of the video editing is not sufficient for the training of a discriminative model of these complex structures.
- The training would last several days for each of the evaluations.
- The structure of the models has to be changed for each test as various combinations of semantic information have to be evaluated.
- The results of the recognitions systems for the semantic information is good, but there is much space for improvement.
- An evaluation and comparison of the models detecting the semantic information is impossible, because of the multi-layer structure.

The second layer can be of the similar structure as model GM1 or GM2. During this thesis, various combinations of semantic and low level features and model

Table 6.3: The results of the evaluation with the additional semantic information are shown. The first two parts use no additional low level features for the classification of the video mode. The third part uses acoustic features as low level descriptors. As additional semantic features in combination with the acoustic feature have been used: person speaking (PS), person actions (PA) and group actions (GA). AER stands for action error rate, FER means frame error rate and ACC is the recognition accuracy rate. AER and FER are results of the combined task of segmentation and classification. For the ACC performance, the boundaries are known and only a classification is performed.

Model	AER	FER	ACC
Group Actions (GA)	84.8	61.0	26.2
Person Actions (PA)	72.2	62.8	28.2
Person Speaking (PS)	62.2	51.5	48.3
PA&PS	58.3	40.6	53.3
GA&PA&PS	58.3	39.6	54.8
Two layer (GA)	63.1	49.2	48.3
Two layer (PA)	60.2	42.5	51.5
Two layer (PS&PA)	56.6	39.0	53.6
Two layer (GA&PA&PS)	56.2	38.1	53.9

structures for the second layer have been evaluated. The best results of these combinations are presented in table 6.3. In the upper two parts, results are presented which are using only semantic information. When only the group actions, which are only a one dimensional feature vector, are used as an input for the model, the frame error rate is 61% and thus it is already about 10% better than the base line. The base line of 71.4% is achieved when only the video mode is selected, which is selected mostly by the human annotators during the annoation of the meeting corpus. The four dimensional person actions are performing with a frame error rate of 62.8% and about 9% better than the base line. The best performing semantic information is to know who is speaking, which tells a four dimensional person speaking feature vector. The frame error rate is 51.5% and consequently still every second frame is misclassified. As all of these semantic features stay very stable over time and only changes happen when something important occurs in the meeting, the action error rate is between 62% and 84%. This is compared, for example, to the acoustic features already a good result. When these results are compared with the best single modality model of GM1, there is a gap of 1% to the acoustic, which achieves a performance of 50.1%. The main difference is that GM1 uses 156 features compared to 4 features used by GM3 and that the action error rate is reduced by nearly 100% absolute or by 60% relative compared to GM1.

The second part in table 6.3 shows selected results of the combination of the semantic features. The first combination of person actions and person speaking achieves a frame error rate of 40.6% and outperforms the base line and the best multimodal model GM2 by more than 30%, respectively by 4%. A further improvement is achieved by adding group action features to the combination of person actions and person speaking. This combination performs at a frame error rate of 39.6% and an action error rate of 58.3%. It is also the best model of all evaulations for the classification task with a recognition accuracy of 54.8%.

The results of the two layer model are presented in the last part of table 6.3. For all these models, the acoustic features are used as additional low level descriptors. By adding the group action to the acoustic features, the improvement of the frame error rate is below 1% but the action error rate is about 63% and therefore reduced by nearly 100% absolute or by 60% relative compared to GM1 using the acoustic features. The person actions performs about 9% better for the frame error rate and the action error rate is about 60%. The performance is further improved by using more than a single semantic information. The combination of person actions and person speaking has a frame error rate of 39% and the combination of group actions, person actions and person speaking performs with 38.1% best. The action error rate is for both combinations about 56% and respectivley the recognition accuracy is about 54%. This is the lowest frame error rate which is achieved by any approach using graphical models or HMMs and those are presented in [HASR09]. Only the approach using an Asynchronous Hidden Markov Model presented in [Len07] works better with a rate of 32.3%.

6.3.4 Video Editing with Eight Video Modes

The difference to the previous subsections is that, instead of seven eight video modes have to be detected. The additional video mode is used for the integration of presentation slides to the output video. This video mode is especially used when a foil is changed during a presentation. The image of the slide is important, because in the centre view details located on the slide are not readable in the created output video. For the evaluation the same graphical models GM1 to GM3 are used. The only difference is, that the vertices have a higher dimension since an additional video mode has to be detected.

In table 6.4, the results of the evaluation for the eight video modes are presented. The audio features applied to GM1 perform with a frame error rate of 51.8% even better than the combinations of audio and global-motion features, with a rate of 53.1%. The global-motion features only perform with a rate of 57.0%, about 5% worse than the acoustic features. The same model using all available semantic information reaches a frame error rate about 7% better than the audio features. The best structure using only low level features is GM2 with a performance of 42.7%. The same performance is achieved by a setting using audio features with

Table 6.4: Instead of seven video modes as for GM1 to GM3, here are results presented for the more complex setting with eight video modes. The eighth video mode is used when a slide change happens and it is important to show the information of the slide. All the model structures GM1 to GM3 and various combinations of features and settings have been tested. The best results are listed here. As additional semantic features in combination with the acoustic feature and/or the global motion features have been used: person speaking (PS), person actions (PA), group actions (GA) and slide changes (SC). AER stands for action error rate, FER means frame error rate and ACC is the recognition accuracy rate. AER and FER are results of the combined task of segmentation and classification. For the ACC performance, the boundaries are known and only a classification is performed.

Model	Features	AER	FER	ACC
GM1	Audio (A)	64.4	51.8	45.2
GM1	Global-Motions (GM)	62.6	57.0	37.6
GM1	A & GM	61.3	53.1	40.6
GM1	GA&PA&PS&SC	62.1	44.4	48.0
GM2	A/GM	59.6	42.7	49.6
GM3	A&GA&PA&PS	60.4	42.7	48.8
GM3	A/GM/GA&PA&PS	59.4	43.8	49.2
GM3	A/PA&PS&SC	60.1	42.5	48.7

semantic information as input for GM3. When additionally to these features, the global-motion features are added, the frame error rate rises about 1%. In both cases, the slide change features are not used. The best feature combination which achieves a rate 42.5% is: acoustic features as low level descriptors and person action, person speaking and slide changes as semantic featuers. The result is only 0.2% better than two other models, especially as one of these is not using any semantic information at all. The action error rate of all the presented models is between 59% and 64% and thus the output video quality is acceptable for the viewer.

6.4 Comparison of the Results

For the seven video modes setting, the results show that semantic information leads to an improvement of about 6% when the frame error rates are compared. The best model using semantic information achieves a rate of 38.1%. It is the two layer model using audio features and all semanctic features. In the case semantic information, group actions, person actions and person speaking, are used alone the rate is 39.6%.

Consequently, an improvement of 5% to the low level descriptors is achieved. The action error rate of all these models is about 58%. The combination of acoustic and global-motion features by the two stream model GM2 is the best setting using only low level descriptors. The frame error rate of this model is 44.6%. When low level descriptor models are compared, it is observable that the more complex multi-stream model GM2 outperforms the simple single stream model GM1. The model GM2 improves the performance by 0.7% when the feature combination acoustic and global-motions is used. For the acoustic and skin blob features the improvement is about 4%.

When the results achieved in this thesis are compared with other approaches, it becomes clear that some improvements have been achieved. The performance of the best multi-modal low level feature model using a HMM is reported with 47.7% and thus the frame error rate is reduced by more than 3% in this work. In the case, semantic information is added to a similar model the performance can be improved by 9.6% absolute. The achieved frame error rate is 38.1%. This result is comparable to the support vector machine approach also using semantic information about who is speaking. The performance of this approach is 39%. The best results achieved by an Asynchronous Hidden Markov Model is 32.3%. The gap to this model has been reduced from 15.4% to 5.8% by using similar graphical models.

The performance of the eight video modes setting is nearly as good as for the seven video mode settings. The best performance is achieved by model GM3 using acoustic features and the semantic information of person action, person speaking and slide changes. The frame error rate is 42.5% and is only 4.4% worse than the best model GM3 for the seven video mode settings.

6.5 Outlook

The best approach using an Asynchronous Hidden Markov Model has still a frame error rate of 32.3%, which means that one third of the frames are misclassified. The relatively high number points out, that in one third of the meeting the wrong camera is shown, but still the output video is watchable. Watchable, means that human viewers are not disturbed by watching the video and get a good, but not the best, impression what is going on in the recorded meeting. This is shown by a user study conducted during a lab course. The user study with 60 subjects shows that the viewers like the video and they do not recognise a difference between the annotated video and the automatically created one. Moreover, the frame error rate and the action error rate are very high, even the video is watchable and contains all information from the subjects' point of view. Due to this reasons, a new measurement for the quality of the output video is necessary.

One third of misclassified frames leaves a lot of space for improvement. The main problem of all the approaches is finding the correct boundaries in between two video modes. Different features, for example slide changes, have the capability to segment the meeting into meaningful boundaries. This should help to segment the output video and thus the quality of it can be improved. Further research can be conducted in the field of different types of fusion. An integration of different feature modalities and types into a model structure which deals with different time scales of these inputs should also lead to an improvement.

7 Activity Detection in Meetings

In every face-to-face meeting – even if the participants do not know each other at the beginning – an order of dominance is established after a short period of time [RM79, LO81]. An important aspect of dominance is the ability to influence other participants and thus it stands in an close relation to power and prestige. Dominance is very important in group meetings since decisions have to be made or solutions have to be found. According to [Wig79], dominance is one of the most important dimension in social interaction.

However, not only a dominance level will be found in the meeting, also the activity of the different participants is observable [Sha71]. This dimension includes more physical action than social interaction and it indicates the level of attention of the participants to the ongoing meeting. These social signals are connected to each other, for example if one participant is talking more than others, this one will be recognized as more active and furthermore as more dominant than the other participants [BC79]. Still, it is possible that a person who is not talking at all has a high level of dominance, because he is shaking the head for a short moment in the ongoing discussion and so his level of activity will be low.

Not only an order of dominance and activity level, but also a hierarchical ranking is observable in a face-to-face meeting [FO70]. One of the most important factors which influences the rank is that high quality contributions are normally assigned to participants who rank high. Therefore, a person, even with a low hierarchical position inside the company, can get a high ranking during the meeting if his contributions are of high quality. Furthermore, it is also highly correlated with the role each participant has in the group or in the company, for example a manager will have one of the highest ranks at least at the beginning.

The system, described in this chapter, is developed to automatically detect a combination of these three factors during a meeting for each of the participants separately, since a close relation between the dominance, the activity and the hierarchical rank is given. It helps the moderator of the meeting to react for example if a participant talks too much or one is getting too dominant compared to the others.

On the other hand, it can be used as a coaching system for group moderators or team leaders who often will be in charge of a meeting.

7.1 Related Work

Previous work on dominance detection used more high level features such as speech transcriptions. The main problem with these approaches, is the high latency and the high real-time factor[1], due to the fact that an automatic speech recognition system is needed to create the transcriptions first. Moreover, it is necessary to have the speakers separated correctly, since interruptions are very important for the order of dominance. In [BCCP01], the approach uses a DBN, which models the interaction between two players for the classification and is performed on features such as person motion energy, speech energy, voicing state, and the number of speaker turns. Thus, the approach does not explicitly model the interaction of the group.

The second, an unsupervised approach, uses a model named Influence Diffusion Model and is described in [OMI02]. It counts the terms which are reused by the current speaker from the previous one. From these numbers, a ranking of influence is created where the highest number shows the most influential speaker. This approach uses only a single feature, which is derived from the speech transcriptions.

In [ZGPBR05], a third approach is described, which combines the behaviour detection of the persons and the group in a single model. This unsupervised model is named “The team-player influence model” and is a two layer DBN. In the first layer is the players level and the second represents the team level. The number of speaker turns, the number of topic turns, and the speaking length are the used features. The structure of the model is shown in figure 7.1. Part (a) presents the Markov Model of player i with the observations for each time. In (b), it is visible that the state of each player is depending on the team state and the player’s state, both from the previous time slot. The players do not directly influence each other, but the influence is possible via the team state. The team state is only depending on current states of each player. The influence of the players on the team state is learned from training data and this is represented by the distribution assigned to the variable Q in part (c) of figure 7.1. Therefore, the variable Q estimated the influence of the players on the team. This approach is applied to 30 five minute meetings. Three annotators were asked to assign the level of dominance to each of the four participants in a way that for each meeting one single dominance level is assigned to one participant.

The fourth approach, which is based on the results of the three listed above is described in [RH05]. The number of features which are used is increased to twelve. The features are the speaking time, the number of turns in a meeting, the number

[1] This systems can not be used for a online scenario during the meeting, as the processing of the results consumes multiple of the duration of the meeting.

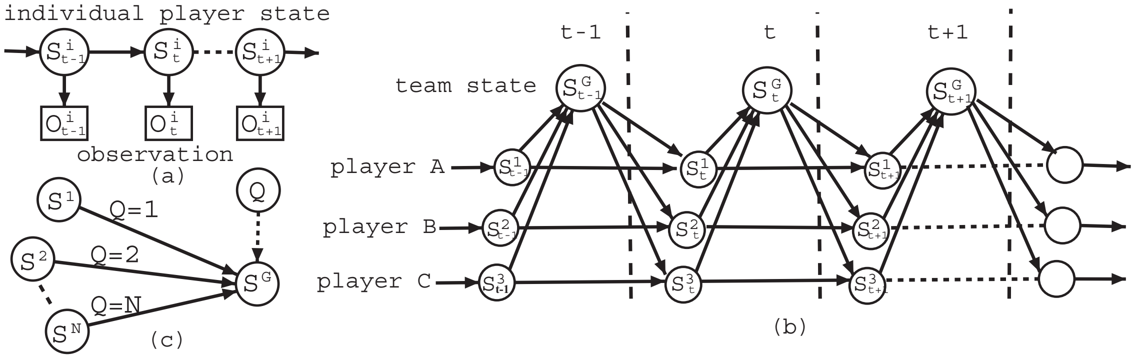


Figure 7.1: The team-player influence model taken from [ZGPBR05]. (a) Markov Model for individual player. (b) Two-level influence model (for simplicity, the observation variables of individual Markov chains and the switching parent variable Q are omitted). (c) Switching parents. Q is called a switching parent of S^G, and $S^1 \dots S^N$ are conditional parents of S^G. When $Q = i$, S^i is the only parent of S^G.

of words spoken in the whole meeting, the number of successful interruptions, the number of times interrupted, the ratio between the number of interrupts, the number of times the person grabbed the floor, the number of questions asked, the number of times addressed, the number of times privately addressed, the person's influence diffusion and the normalised person's influence diffusion. These features are derived by hand or by script which processed the speech transcriptions of the meeting. The annotation for the meeting data used in [RH05] is created by ten annotators who have selected for each of the four participants in half of the meetings a dominance rank between one and four. Each rank has to be assigned once. For the evaluation, various types of static classifiers, as Naive Bayes or SVM, are used.

7.2 The Activity Detection System

In this work, a system is described which can detect the dominance/activity of the participants in meetings from low level features. This system uses video- and audio-data from cameras and microphones, which are available for remote participants of a meeting as well as for people located in a smart meeting room. Thus, the system can be used during meetings, for video conferencing or processing recorded meetings.

Low level acoustic and visual features, as described in section 3.1 and section 3.2, are extracted from the audio and video streams which are captured in a meeting room. These low level features and various combinations of them are used as input for the different detection systems. Furthermore, semantic features are added to the system, which are described in section 3.3. Consequently, the system uses no speech transcriptions and has latency of a single frame, in this case only the low level features are used. The semantic features can not be derived in real time and some of these have a high latency. Since the low level feature extractors perform in real time and the detection system can be run in an online mode, the whole activity detection system is real time capable.

The system detects five different classes of activity and dominance for each of the four participants separately. One class represents the state when a participant makes a decision during the meeting. The other four classes describe the participant's activity from not active to most active. It is possible, that the same class is assigned to several participants at the same time and thus a ranking of the participants is not always unambiguous.

7.3 Used Graphical Models

In this section all the developed structures of the graphical models, which have been evaluated for the activity detection, are presented. During the evaluation, various combinations of features are used as input to three different model structures. For more details about each of the vertices and the arcs see chapter 4, which gives a short introduction to graphical models and the implementation structure of GMTK.

The experiments in this work consist of two separate tasks. For the first one, the boundaries of each segment are known and therefore only a classification is performed. Those results are shown in the tables as recognition accuracy rate (ACC) and the rate is equal to the number of correctly classified segments divided by the total number of segments. The second experiment is the real task of the system, because the boundaries and the labels have to be detected automatically. For this task the action error rate (AER) and the frame error rate (FER) are used. AER gives an impression of the sequence of the labels, but does not take into account the right boundaries. The FER describes the number of correctly detected frames divided by the total length of the meeting, thus it is an adequate measure for the real task. For all evaluations, a nine fold cross validation with person disjoint test and training sets is performed. Low rates of FER and AER and in the case of ACC higher ones are better.

7.3.1 Single Stream Segmenting Graphical Model (GM1)

In Figure 7.2, the training structure of the continuous single stream graphical model GM1 is depicted. It has the capability to learn the segment boundaries and thus during the decoding process an automatic segmentation is possible. For the training of the model, the EM algorithm [DLR77] is used and during the decoding the Viterbi algorithm [Vit77] is applied to the unknown test data.

All vertices except the observation can be summed up to a single vertex q_t. This vertex represents the state of the model for a certain time frame t. The derived

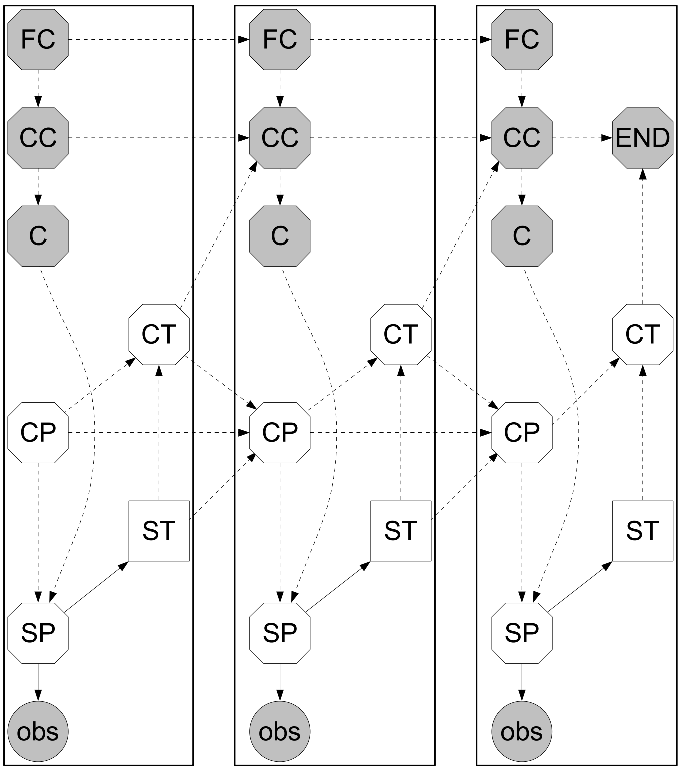


Figure 7.2: The continuous single stream graphical model GM1 is used in this thesis for the segmentation of the meeting and the classification of five labels of activity and decision making. Therefore, five classes have to be trained by the EM algorithm. The figure shows the training structure as it is implemented in GMTK for the training of the model. This structure trains in a single process both the segmentation and classification. An elaborate description of all the used vertices can be found in chapter 4.

structure leads to the following production probability for the model

$$p(\mathbf{O}, \mathbf{q}|\lambda) = p(q_1) \cdot p(\vec{o}_1|q_1) \cdot \prod_{t=2}^{T} p(q_t|q_{t-1}) \cdot p(\vec{o}_t|q_t), \tag{7.1}$$

considering the sequence $\mathbf{q} = (q_1, \ldots, q_T)$ of all states which are run through. When the substitutions, known from the forward-backward algorithm for HMMs, and the marginalization, by summing up over all possible state sequences $\mathbf{q} \in \mathbf{Q}$, are applied to the equation 7.1 the following equation is derived:

$$p(\mathbf{O}|\lambda) = \sum_{q \in Q} \pi_{q_1} \cdot p(\vec{o}_1|q_1) \cdot \prod_{t=2}^{T} a_{q_{t-1},q_t} \cdot p(\vec{o}_t|q_t). \tag{7.2}$$

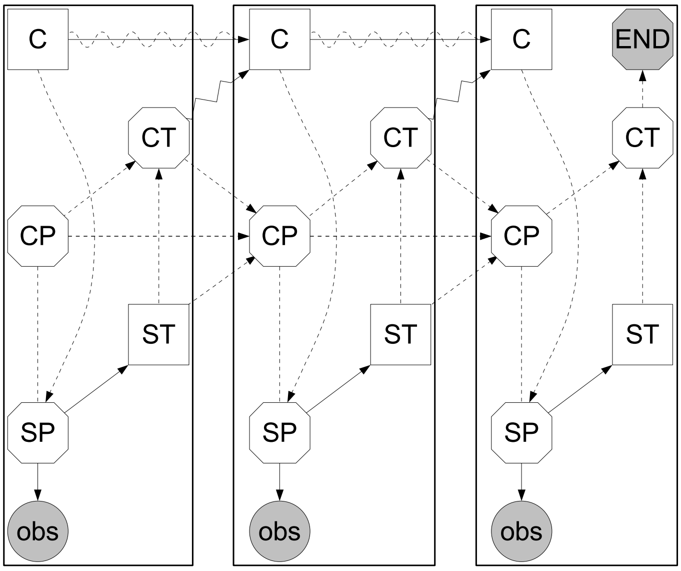


Figure 7.3: The decoding structure of GM1 as implemented in GMTK. An arc, which changes between a statistic or deterministic dependence, connects two consecutive class vertices C. This is necessary for the automatic segmentation during the Viterbi decoding. A description of all the used vertices and arcs can be found in chapter 4.

$\mathbf{O} = (\vec{o}_1, \ldots, \vec{o}_T)$ is the observation sequence, which is used as an input for the model. This shows that the model is similar to a HMM, known from [Rab89].

Figure 7.3 shows the decoding structure to the previously described training structure of this model. Since the model can automatically segment the test data via the Viterbi algorithm, an arc which changes the type between statistic and deterministic is used in between two consecutive class vertices C. This is necessary, because in the case of a class transition, the new class C is depending on the probability distribution of classes learned during the training. Otherwise, the class is copied from the last chunk, which is represented by the deterministic arc.

Table 7.1 presents the evaluation of the single stream model with various combinations of modalities. Moreover, it lists the number of features which have been used for the evaluation of the models. The first three rows contain the results for single modality models. Each of these modalities perform best in case 20 states are reserved in the models for each class. The best result is achieved by the acoustic modality with a frame error rate of 47.7%. This means, that nearly every second frame is misclassified during the segmentation process. The base line of 62.8%, when only the most usual class is selected, is outperformed by 15%. The performance of the visual features is very weak, as only one third of the frames are classified correctely. The performances for the classification task are about 29% for skin blob features and about 37% for the global motions. This means that global motion fea-

Table 7.1: Evaluation of different single stream modalities and combinations of these. Various numbers of states per class have been tested and the best results are presented. The fusion models are combinations of acoustic (A), global motion (GM) and skin blob features (SK). The third column shows the number of features which are used to achieve the listed results. AER stands for action error rate, FER means frame error rate and ACC is the recognition accuracy rate. AER and FER are results of the combined task of segmentation and classification. For the ACC performance, the boundaries are known and only a classification is performed.

Model	# states	# features	AER	FER	ACC
Audio (A)	20	39	47.2	47.7	54.9
Global Motion (GM)	20	7	64.4	63.7	36.6
Skin blob (SK)	20	15	66.6	71.3	28.6
Single stream (A&GM)	15	46	48.4	48.6	51.8
Single stream (GM&SK)	20	22	63.2	63.2	36.2

tures are performing as well as the base line. The skin blob features do not achieve the base line, due to the fact that the skin colour detector is misled by objects which have a skin-like colour and are placed behind the participants in the bookshelves. This means for example a head is detected above the empty participant's seat while the participant is giving a talk in front of the presentation screen. Therefore, a lot of misleading information is derived by the skin colour detector which leads to very bad results when the skin blob features are used. The acoustic features also perform best during the simple classification task with an accuracy of 55%. For the classification, the acoustic features are about 18% above the base line and for the combined process about 10%.

Results of the single stream feature fusion are listed in the lower part of table 7.1. The fusion of global motion and skin blob features achieves the same performance as the model using the global motions only. The improvement of the results is only about 1% for the combined task and therefore it is not worth to use more features. For the classification task, the performance is reduced slightely. This fusion model uses also 20 states per class as the single modality models. The fusion of acoustic features with global motion features reduces the performace of the combined task by less than 1% compared to the model using the audio modality only. The accuracy for the classification process is reduced by more than 3%. 15 states per class achieve the best result for the single stream feature fusion.

The combination of different modalities does not lead to an improvement of the performance when a feature fusion is performed by using this model. This model handles all features which are applied to the input in the same way. In the next

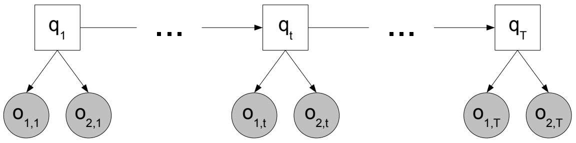


Figure 7.4: The block diagram of the two stream graphical model GM2. It contains the same structure as GM1 twice but it has two observation streams which are statistically independent and therefore can be drawn as two separate observation vertices.

section a more complex approach to feature fusion is described, which is capable of dealing with different weights for different feature types.

7.3.2 Multi Stream Segmenting Graphical Model (GM2)

In figure 7.4, the block diagram for a two stream graphical model is depicted. Compared to GM1 it has two observation vertices and consequently two statistically independent data streams can be connected to the model. The features inside one stream are depending on each other, similar to the GM1. These two observation streams are called $\vec{o}_{1,t}$ and $\vec{o}_{2,t}$ and contain a vector consisting of continuous normalised features. The block diagram leads to the production probability in equation 7.3 with the state sequence $\mathbf{q} = (q_1, \ldots, q_T)$.

$$p(\mathbf{O}, \mathbf{q}|\lambda) = p(q_1) \cdot p(\vec{o}_{1,1}|q_1) \cdot p(\vec{o}_{2,1}|q_1) \cdot \prod_{t=2}^{T} p(q_t|q_{t-1}) \cdot p(\vec{o}_{1,t}|q_t) \cdot p(\vec{o}_{2,t}|q_t) \quad (7.3)$$

Summing up equation 7.3 over all possible state sequences $\mathbf{q} \in \mathbf{Q}$, yields the following production probability of the observation $\mathbf{O} = ((\vec{o}_{1,1}, \vec{o}_{2,1}), \ldots, (\vec{o}_{1,T}, \vec{o}_{2,T}))$

$$p(\mathbf{O}|\lambda) = \sum_{q \in Q} \pi_{q_1} \cdot p(\vec{o}_{1,1}|q_1) \cdot p(\vec{o}_{2,1}|q_1) \cdot \prod_{t=2}^{T} a_{q_{t-1},q_t} \cdot p(\vec{o}_{1,t}|q_t) \cdot p(\vec{o}_{2,t}|q_t). \quad (7.4)$$

The independence between the two observation streams $\vec{o}_{1,t}$ and $\vec{o}_{2,t}$ is visible in equation 7.4. Different modalities are usually used as inputs of these observation vertices to achieve better results. In this evaluation, audio and visual features are applied to the observation inputs.

In table 7.2, the results for the multi stream model GM2 are listed. The combination of acoustic and the global motion features achieves the best results with an frame error rate of 55.8% and an accuracy of 49.2%. Therefore, the frame error rate is about 7% lower than the base line when only the most usual class is selected. This

Table 7.2: Evaluation of different multi stream models with various modality combinations. The feature combinations contain acoustic (A), global motion (GM) and skin blob features (SK). The third column shows the number of features which are used to achieve the listed results and the split of the features to the two observation streams. AER stands for action error rate, FER means frame error rate and ACC is the recognition accuracy rate. AER and FER are results of the combined task of segmentation and classification. For the ACC performance, the boundaries are known and only a classification is performed.

Model	# states	# features	AER	FER	ACC
Multi stream (A/GM)	15	10(3/7)	63.6	55.8	49.2
Multi stream (A/GM&SK)	15	25(3/22)	60.7	57.6	44.1
Multi stream (GM/SK)	20	22(7/15)	60.7	60.1	40.5

model uses 15 states per class and ten features in total. In the case skin blob features are additonally used, the performance goes down by 2%, respectively 5%. For a multi stream model only using visual features the frame error rate is reduced by 4% compared to the best multi stream model. The accuracy is degraded by nearly 9%. The skin blob features do not lead to an improvement. When the best model is compared to GM1, it is visible that the simpler GM1 outperforms the mutli stream model. The acoustic features used in GM1 achieve an accuracy which is nearly 6% higher than the GM2 using acoustic and visual features. The same feature combination achieves for the GM1 an accuracy which is about 3% better. The frame error rate is for the acoustic GM1 about 8% lower and for the multi-modal GM1 about 7% lower. Since the results for two streams are already below the results of the best single stream models, the possibility to expand the model to more independent streams, is not conducted.

7.3.3 Two Layer Multi Stream Graphical Model (GM3)

In figure 7.5, a block diagram of the third model, which has been evaluated in this thesis, is shown. It uses not only low level descriptors, as acoustic or global motion features, but also semantic information. These semantic information are described in section 3.3 and contain information about person movement, what the person is doing or the group action. All this information can be derived by automatic systems which have been developed in the last couple of years. For example, a group action recognition system using Markov random fields can be found in [Rei08]. Another approach based on graphical models is described in [AHDGP$^+$06]. The person actions are recognised by an approach using HMMs in [AHHSR06]. For the movement classification, an HMM based system has been developed. A combined structure of most approaches can be implemented in GMTK, but the training of the

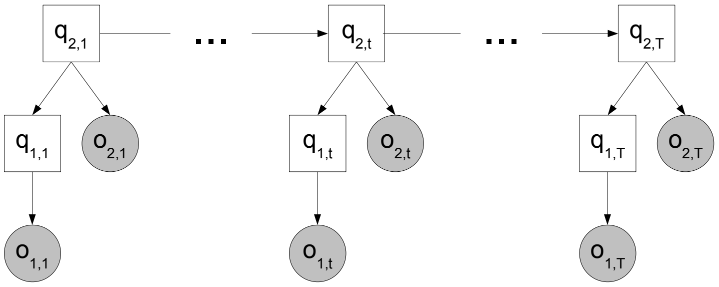


Figure 7.5: The block diagram of the two layer graphical model GM3. The first layer contains various different approaches for the detection of semantic information and therefore not the whole GMTK implementation is presented. For example, the group action is derived by graphical models in [AHDGP$^+$06] or by Markov Random Fields in [Rei08]. An implementation of the structure is feasible in GMTK, but the combined training of the whole model would not work, due to the huge number of dependencies and random variables. The second layer contains the same structure as GM1.

model would not work since the number of dependencies and random variables in the whole model is huge and not enough training data is available. All of these systems are not working yet in real time. Furthermore, the recognition performance of these systems is good but for a proof of concept not good enough. Due to these facts, ground truth data of the semantic information have been created by annotators and are used for the evaluation of the two layer model.

Table 7.3 shows the results of the evaluation. Since the best results of GM1 are achieved by using 39 acoustic features, the same acoustic features are used as low level descriptors for the second layer of the GM3. All of the listed models use 20 states per class. All the results of the two layer model are about the same level. There is no difference between the three used semantic features, movement, person action or group action. The frame error rates of the different models are in a range between 64.2% and 65.4%. This means the base line, which means only the most frequent class is selected, is reached, but there is no improvement. Therefore, they are about 17% higher than for GM1 using the acoustic features. Moreover, the accuracy is about 34% for each combination of semantic and acoustic features and thus about 20% below the accuracy of the best GM1 setting. Other low level descriptors have been tested too, but the results have been even worse.

7.4 Comparison of the Results

The visual features are performing weakest for all evaluated models. GM1 achieves the best results with a frame error rate of 47.7% if only the acoustic features are used. A comparable result is achieved in case, global motion features are added

Table 7.3: Evaluation of different modality combinations and the two layer graphical model. The two layer models always perform best in the case of 20 states per class. Additional semantic features in combination with the acoustic feature have been used: movement (M), person actions (P) and group actions (G). The third column shows the number of features which are used to achieve the listed results. AER stands for action error rate, FER means frame error rate and ACC is the recognition accuracy rate. AER and FER are results of the combined task of segmentation and classification. For the ACC performance, the boundaries are known and only a classification is performed.

Model	# states	# features	AER	FER	ACC
Two layer (M)	20	40	87.5	64.2	34.2
Two layer (P)	20	40	87.7	65.4	34.0
Two layer (G&M)	20	41	87.5	64.3	34.3
Two layer (G&P)	20	41	87.7	65.3	34.0

to the single stream containing the acoustic ones. Various combinations of multi stream approaches using GM2 perform at least 8% worse than the single stream model GM1. Only GM1 and GM2 are performing significantly better than the base line with an improvement of 15%, respectively 7%. The use of semantic information does not improve the results at all. The best GM3 achieves a nearly 20% higher frame error rate. The results are presented in [HR09]. This stands in conflict with results presented in [HASR09], where semantic information improves the performance of a video editing system for meetings using the same data base.

7.5 Outlook

Results in [Rie07, Zha06] show that a static classifier achieves a performance of 70.6% compared to a dynamic classifier with 54.4%. The evaluations of these approaches have been performed on the same data base and similar features, but the best models are not using the same features. Since the static classifier outperforms the dynamic one, an evaluation of various static classifieres would be interesting on the data base used in this thesis.

Furthermore, the features derived from speech transcripts can be used as additional input for the system. This should lead to a better performance, since context information of the ongoing meeting is important, especially for the decision level.

Another thing, which can be considered to accomplish in the future, is that the annotation is done in a different way. The problem of the current annotation is that the segments are defined for the whole group and not for each of the participants separately. This is especially a problem, when in the middle of a defined segment a person stops speaking and another participant starts speaking. The annotation would get more precise and this is good for the training of the classifiers. It would lead to better results for the classification process, but the segmentation and classification task would get more difficult because several boundaries for each participant have to be found.

8 Multi-modal Interest Detection

For the communication between people, many different techniques are used for the interaction. Various different techniques are commonly used when people communicate and interact with each other. Not only spoken words, even gaze, facial expressions, gestures and intonation of speech are essential for the human communication and interaction. For a natural dialogue between a machine and a human subject it is important to take all these interaction skills into account. A machine has to understand the natural way of human communication, in order to provide interaction that is close to human to human dialogue [AC75, NL86].

In the last couple of years multi-model dialogue systems are becoming more common. These systems are capable to detect reactions of human users and respond to them in an appropriate way. Such reactions can be in a wide range from non-verbal over social and emotional perception to interaction and behavioural capabilities. Various publications illustrate which combinations of interaction skills are important [TKB92, Tho93]. Numerous research projects are developing such dialogue systems [SBE$^+$09, Sch10], for example the SEMAINE project[1]. In [SKMR06, SMH$^+$07], systems are introduced which detect the level of interest of a human user. The information on interest has great potential for the Human-Machine Interaction [PM05, Shr05] and consequently for many commercial applications, as advertisement systems and virtual guides. Furthermore, the level of interest has been analysed in meeting scenarios in many publications, for example [SYW02, KE03, GPMZB05]. Another application is described in [MP03], where a tutor system for children is developed. Not only in the field of interest, research has been conducted, but also systems for the detection of the curiosity have been developed [QBZ05].

In the literature, numerous publications can be found about the recognition of affective and emotional states of users. The emotional states are closely related to the interest in the topic of the ongoing dialogue. Many of these works only apply acoustic speech parameters [SRL03, BSH$^+$05]. In [KPI04, ALC$^+$07], for example,

[1] The aim of the SEMAINE project is to build a Sensitive Artificial Listener. The funding for the research project comes from the European Community's Seventh Framework Programme.

systems are described which use vision based features for the task. There are fewer approaches using visual features than systems performing on acoustic parameters. The fusion of all available input features generally helps to improve the results, yet only few publications are available which deal with multi-model approaches for the detection of emotional states [GPMZB05, MP07, SMH^{+}07, SME^{+}09]. In general, not only the performance is improved, but also the reliability and the robustness of such a system [CDCT^{+}01, PR03, SAR^{+}07, WSA^{+}08].

8.1 Related Work

There is only little published work on the same data base as used for the interest recognition in this thesis. An approach is introduced in [SKMR06], which analyses acoustic and linguistic cues of spoken segments. More than 5.000 features and functionals are extracted from each segment. The statistical analysis of individual segments is important, since most information of the level of interest is available by analysing the development of the feature values over a segment. A subsequent feature space reduction has to be performed to find the most relevant attributes. Linguistic information is added by a bag-of-words[2] representation taken from the speech recogniser. The prototype achieves a remarkable performance using support vector machines for the selection of one of the three level of interest classes.

The first audio-visual approach for the recognition of spontaneous interest has been presented in [SMH^{+}07]. Active appearance models are used for the detection of facial expressions. Moreover, the movement of the eyes are recorded by an eye tracker and the activity of the eyes is further analysed. Speech is analyzed with respect to acoustic properties. This is based on a high-dimensional prosodic, articulatory, and voice quality feature space. Furthermore, the linguistic analysis of spoken content is modeled by a large-vocabulary continuous speech recognition engine and a bag-of-words vector space including non-verbals. A person-independent system is evaluated and the results show the high potential of such a multi-modal approach applying support vector machines as classifiers.

The multi-modal approach from [SMH^{+}07] is extended by the use of temporal context information in [SME^{+}09]. Consequently, Active-Appearance-Model-based facial expression, vision-based eye-activity estimation, acoustic features, linguistic analysis, non-linguistic vocalisations, and temporal context information are combined by an early feature fusion. Support vector machines are trained for the evaluation and subject disjunct test and training sets are used. Moreover, a real-life system has been developed and is tested during a user-study. The theoretical and practical proof of effectiveness is demonstrated.

[2] Bag-of-words means, that various words are combined into one bag. A bag is found if the same words, as stored inside the bag, are recognised without taking into account the word order.

In [SR09], a comparison of bag of frames and supra-segmental features is described. Supra-segmental features are functionals, which are calculated for each segment from the low level features extracted from each frame. Based on the functionals, support vector machines are trained. For the bag of frames also support vector machines are trained, but multi-instance learning is performed on the features which are based on frames [ATH03]. A comparison shows that some feature groups, for example pitch and energy, perform far better on the segments than on frame base. The mel frequency cepstral coefficients still suffer from the frame based approach but not as much as other feature groups.

8.2 The Dialogue Control System

The dialogue control system developed in this thesis analyses both audio and visual recordings. From the audio stream various low level features and functionals are extracted, as described in section 3.1. These functionals are calculated for segments with a duration of one second, respectively 25 frames, compared to the functionals used in the related work, where these are calculated for segments of different durations. The acoustic functionals have an overlap of 24 frames, since these are calculated for each frame. Global motion features are extracted from the visual recordings, as specified in section 3.2.2. Furthermore, a principal component analysis is performed and the derived features are also evaluated. An audio-visual fusion is performed on feature level and therefore the segments are classified on a frame base. This is the main difference to most publications described in the related work section, which classifies on segment base. In [SR09], it is shown that the classification with support vector machines performs better on segments as it does on a frame base.

In this thesis, only the detection of the level of interest is described and all the evaluations are taking place on data which is recorded from a human-to-human conversation. For the user study, a shift of the paradigm is necessary. The setting of the whole dialogue control system is in a way, that a subject is seated in front of a monitor, which means a human-to-machine conversation takes place. A camera and a microphone is located close to the monitor for audio-visual recordings. A male animated embodied conversational agent is shown on the screen and it guides the subject through the topics. This dialogue system, which is capable of switching to a different topic if the subject is bored, has been developed for a user study. This user study is performed as a Wizard-Of-Oz experiment [Nie93, NSGN02], where the dialogue is controlled by a human operator. Moreover, the system is able to switch the topics automatically depending on the results of the level of interest detection. The results of a study can be found in [SME^{+}09].

Table 8.1: The table shows the results of the best low level feature groups. The audio features are the only ones which outperform the base line. For more details about the features see section 3.1. ACC is the recognition accuracy rate and higher rates are better.

Model	Features	ACC
SVM	113 RMSenergy	54.7
SVM	113 PCM_MAG_fband_250_650	55.7
SVM	113 PCM_MAG_fband_0_650	58.2

8.3 Used Models

In this section all the developed models are described. Not only graphical models are used but also support vector machines, since these have performed well in previous works on the level of interest detection. During the evaluation, different combinations of features as input and parameters for the models are tested. For more details about the graphical model structure, as it is implemented in GMTK, see chapter 4. It gives a short introduction to graphical models and to the implementation of them in GMTK.

For the evaluation, the recognition accuracy rate (ACC) is used as performance indicator. The ACC is equal to the number of correct classified segments divided by the total number of available segments. The base line for the evaluation is 53.4%. This rate is achieved in case that the class with the highest number of utterances is always selected, which is the class LOI1. Furthermore, the standard deviation (STD) is used, as the differences in the single folds of the corss-validation is very high. A leave-one-speaker-out cross validation is performed, which means 21 folds are fulfilled since 21 subjects are available.

8.3.1 Support Vector Machines

An introduction to support vector machines (SVM) can be found in [Vap95]. In this thesis they are used for the classification of each single frame. This is also the main difference to most other works [SWA$^+$07, SMH$^+$07] which are using SVMs. In those publications, SVMs are used for the classification of pre-defined segments or blocks of 25 frames. The idea behind performing the classification on segments is that the values are developing inside a segment and this development can be reproduced with functionals. For this work it is necessary to have the classification results for each single frame, as the output of the SVMs is used as an input for the graphical models. During the evaluation, all the features are grouped and for each of these groups various settings of the SVMs have been tested.

In table 8.1, the results of the three best SVMs are presented. Most of the

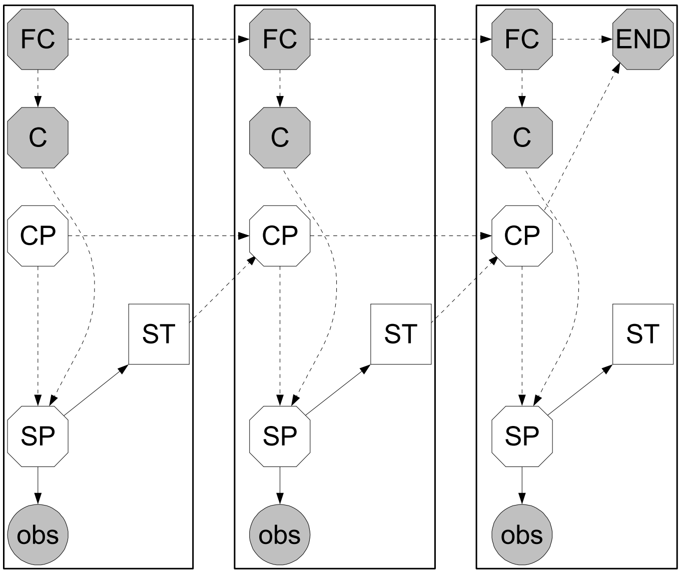


Figure 8.1: Simple continuous single stream graphical model GM1 used in this thesis for the classification of three classes of interest. The figure shows the training structure, as it is implemented in GMTK for the training of the model. A description of all the used vertices can be found in chapter 4.

29 feature groups, which have been evaluated do not perform better as the base line of 53.4%. Due to the training procedure of SVMs, usually the most dominant class in the transcriptions is selected. Therefore, the base line is always reached. A couple of feature groups perform slightly better than the base line, for example the "RMSenergy" performs a recognition accuracy rate of 54.7%. The best feature group is the spectral energy within the frequency-band of 0 to 650Hz. The accuracy for this setting is 58.2% and thus nearly 5% better than the base line. All of the results which achieved a better performance as the base line and one base line result have been transformed into an input stream for the graphical models which are evaluated in the next section.

8.3.2 Graphical Model Structure

The graphical models used for the dialogue control system are similar to the models described in chapter 5. This is possible, since the data bases are of matching structure. All the audio and video recordings are pre-segmented and thus there is no need to use a model which is capable to detect segment boundaries. There is still a main difference between these two data bases, as for the ABC corpus a hierarchical approach can be performed since two groups of behaviour, namely suspicious and normal, are available. For this case the hierarchical approach is not taken into account, as it is not feasible for the AVIC corpus. For the AVIC corpus used in

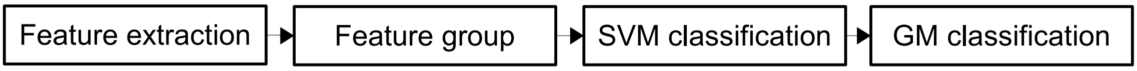


Figure 8.2: Illustration of the combined approach of support vector machines and graphical models. The features and the functionals are extracted and combined to feature groups. These feature groups are classified with support vector machines and the output is used as an input for the classification with graphical models.

this chapter only three classes are discriminated and therefore no clustering of the classes is available. The best results for the multi-modal surveillance are achieved by GM1 (see section 5.2.1) and GM3 (see section 5.2.3). These models are using one, respectively two streams as an input. For the two stream model no results are presented, because of the observation, which has been made during the evaluation, that the performance of these models are even below the quite low performance of the one stream models using features, as global-motions or mel frequency cepstral coefficients. Thus, only the results of model GM1 are presented in this chapter. During the evaluation various combinations of feature sets are used and different parameter settings for the model GM1 are tested. The single stream model is shown in figure 8.1 and a description is found in section 5.2.1.

Furthermore, an approach based on SVMs is evaluated, which is illustrated in figure 8.2. The features and functionals derived from the audio and video recordings are combined to 21 different feature groups. Each of these groups contains one single low level feature and all the 56 functionals and the 56 first order derivatives of the functionals. A feature group contains 113 features in total and these are used as an input to the SVMs. A first classification is performed by the SVMs and the output is applied to the input of the graphical model. Three features can be derived from the output of the SVM, since only three different classes are available in the data base. The output contains three numbers, one for each class, which represents the probability for each class. GM1, as described in section 5.2.1, is used for the classification of the SVM output. The idea behind this approach is, that the SVMs should help to filter the features and reduce the disruption of GM1, which occurs due to the high fluctuation of the low level features.

The results presented in table 8.2 are achieved by using various different feature types and settings. The first three results are performed by low level features which have been derived by a principal component analysis. All of the tests with low level features and graphical models lead to performances which are below the base line. A very interesting observation is, that the standard deviation of these results between the cross validation folds is very high. The best performance of 48.7% is achieved, when the first two features of the principal component analysis are used and a graphical model with seven states. In this case, the standard deviation is 17.4% with the highest accuracy of 78.1% and the lowest of 11.1% within one single fold.

Table 8.2: The table shows the results of the best feature combinations using GM1. Various other combinations and parameter settings have been evaluated, but the results are even worse than the presented ones. The upper half of the table displays results of the features derived by the principal component analysis. The lower part indicates, that only results based on features which are derived from the output of support vector machines outperform the base line. ACC is the recognition accuracy rate and used as the measure, since no segmentation is performed. STD stands for the standard deviation.

Features	Parameters	ACC	STD
4 PCA normalised	10 states	46.4	12.8
3 PCA	7 states	47.6	17.6
2 PCA normalised	7 states	48.7	17.4
3 SVM VoiceQual	7 states	47.7	16.6
3 SVM RMSenergy	7 states	52.4	14.6
3 SVM PCM_MAG_fband_0_650	7 states	54.6	18.4

A comparison of the highest accuracy for the single fold with the lowest accuracy of the same fold of a different model shows, that the accuracy rate drops below 10%. The result for the fold with the lowest accuracy is similar as the rate rises above 50%. This analysis of various results shows that the best and worst results in the cross validation are always achieved in different folds. Therefore, the accuracy rates are not depending on a special cross validation fold, which consequently means there is not a problem with the unbalacend training and test set in the data base.

The lower half of table 8.2 presents results which are achieved in case the output of SVMs are used as features for the graphical model. Only some evaluations are performed, since few SVM outputs are better as the base line, which is 54.3%. The best performance is commonly achieved by applying a graphical model with seven states per class. The only feature group used which lead to a slightly better performance (54.6%) as the base line is spectral energy within the frequency-band of 0 to 650Hz.

8.4 Comparison of the Results

Due to the fact that all the results of the related works are performed on the AVIC data base, but the total number of segments differs, a comparison can not be performed perfectly. Still, it is possible to get an impression of the quality of the results achieved by the approaches presented in this thesis. The best result presented in [SMH^{+}07], is a recognition accuracy rate of 77.1%. For this result, SVMs are used with features from acoustics, facial expression, eye activity, linguistics including non-linguistic vocalisations and context, which are extracted from each segment of the data base with 996 segments in total. The data base evaluated in this thesis has only 925 segements, due to the fact that some segments of the 996 have a duration of less than ten frames. The best performance achieved during this thesis is 54.6%, which is nearly 20% below the best performance reported in [SMH$^+$07]. Therefore, the approach using SVMs on frame base is not leading to an improvement. All graphical models do not reach the base line of 54.3%.

In [SR09], a frame based approach is compared with the approach applied to segments and the results show that SVMs perform better on segments than on frames. This can be explained, since the development of values is much better represented in functionals which are calculated from the whole segment, than in features derived from a single frame. Even functionals, which are calculated over several frames, as it has been done in this thesis, do not lead to an improvement of the performance.

8.5 Outlook

A different approach for the training of SVMs can be evaluated, as the performance in [SR09] shows that multi-instance learning helps to achieve good results even on features which are extracted on the frame level. A similar training algorithm can be developed for graphical models [ZZ03], which should help to improve the results of those models, too.

Moreover, the features can be analysed if there is much noise overlaid and filters can solve that problem. The data base can be scanned for segments where no speech signals are available and the duration of the segments can be investigated. At the moment, the shortest segments have a duration of only 400ms which is very short for a proper classification of an emotion state, such as level of interest for example.

9 Summary

In this work some real life problems have been transformed into a form which allows applying modern pattern recognition methods to them. By doing this, it is possible to come up with solutions for these problems which help people with their daily business. In this thesis, the focus is on surveillance, meetings and dialogue scenarios. For all of these the access to multi-modal recordings of the subject is possible. This is very important since performance and robustness are increased compared to single modality recordings. Various graphical models are evaluated for each of the scenarios and the performances are compared to publications on the same topic.

A graphical model is a combination of graph- and probability theory. The graph theory allows a simplification by performing algorithms based on the graph and therefore the calculations of the required probabilities can be easier. This is very important, because a slight change of the graph or of the probability leads to having to start each calculation form scratch. By using algorithms based on the graph, many calculations do not start from the beginning, but previous results can be reused. Furthermore, the graph gives an intuitive description of complex problems which depends on many different random variables. This makes a graphical model especially suitable for rapid-prototyping and expert-systems, where the probabilities are defined by the expert. Finally, graphical models can be adapted to any complex problem and not the problem has to be abstracted to fit to the pattern recognition method.

As an input to the graphical models, various types of features are used. These features are extracted from audio and visual recordings. Furthermore, semantic information is derived from the meeting data which is also used as an input. Not only low level descriptors are used, but especially from the acoustic features functionals have been calculated which help to detect a development in time of different features. In total, more than 3000 features are extracted. Since graphical models can not deal with the huge amount of features, a feature space reduction is performed depending on the different scenarios. Two types are evaluated: the sequential forward selection and the principal component analysis.

In the following paragraphs the four scenarios are summed up shortly. The first scenario is about surveillance of passengers in an aircraft. Via audio-visual recordings, a passenger which is located in his or her seat, is analysed if he or she acts suspiciously or normally. Therefore, various sets of features, graphical models and different parameters are evaluated. The best results are achieved by a hierarchical model which makes a decision about suspicious or normal behaviour in the first layer and a more specific decision within these two behaviour groups in the second layer. The recognition accuracy of the models is 53.3%. An approach using support vector machines achieves 81.1%, which is 27.8% above the graphical model and therefore the graphical models should not be used for this scenario.

The second scenario is automatic video editing in meetings. This means, that the meeting is recorded with several cameras and the most relevant view of these cameras has to be found for each time frame. This view is transmitted to the remote participants during a video conference or is shown in case an archived meeting is watched. The meeting is not only recorded with cameras but also microphones are used for that. Depending on audio, visual and semantic information the best view is found by a graphical model. The best graphical model achieves a frame error rate of 38.1% which is 5.8% lower as an Asynchronous Hidden Markov Model found in the literature. Compared to other apporaches using graphical models or hidden markov models, the performance has been improved by 9.6%.

Detection of activity and dominance in meetings is the third scenario. During the meeting for each participant separately, the level of activity is recognised from the same information as used for the video editing. Each participant is automatically assigned by a graphical model for each time frame to one of the labels in the range between absent to most active. The output can be used for assisting the moderator of the meeting, because it shows for example if a participant is not involved in the meeting. The lowest frame error rate with 47.7% is achieved by graphical models, using only the acoustic features. Due to the lack of comparable results in the litrature, it is not possible to compare them, but the best dynamic classifier found achieves a recognition rate of 54.4%, using a similar data base and different features. This is comparable with the performance of the graphical model presented in this work.

The last scenario is an interest detection system for a human-machine-interaction. A subject is talking with a machine and at the same time it is recorded by a camera and a microphone. From these recordings, an audio-visual analysis of the subject's level of interest is performed. This means that the subject is assigned to a predefined level for short segments during the dialogue. Depending on these levels the machine recognises if the topic is interesting for the subject and makes a decision if a topic change makes sense. Since graphical models perform very badly in this scenario, a new approach combining support vector machines and graphical models has been evaluated. Even this approach does not lead to a significant improvement compared to the base line. If the results are compared with other approaches using support vector machines and different features, the performance is nearly 20% below the best result presented in other works.

9.1 Conclusion

In this work graphical models have been applied to very different applications, which shows the high flexibility of graphical models. The models have been adapted to the different applications, which is very important for further research in different other fields of application. Another important point is that graphical models allow to describe complex problems in a simple form, which also helps to get into other application fields and helps to formulate problems in a way many people can understand. Furthermore, graphical models provide a universal description of various applications and therefore it is simple for an expert in graphical models to provide analysis and results for a model developed by someone else. Since a general formalism exists for these models, many instructions and algorithms are available which can be applied to all types of graphical models.

The main drawback of the possibility to develop models depending on the problem is that the number of parameters which have to be adjusted and trained can rise fast. In this case, more training samples are needed or for very complex models the training gets in intractable. The high flexibility of graphical models allows to develop multi-layer models, which take into account various levels of information, for example low level features, results from classifiers or high level semantic information. A model like this needs for the training annotated information from each layer, which means that complex annotation has to be done on each layer separately.

The implementation of toolkits for graphical models is very complex. It is sometimes necessary for the development of models to implement new types of edges or vertices, which have to be performed directly in the toolkit. This stands in conflict with the idea that graphical models can be easily transfered from paper to a running system. The graphical model toolkit used in this work is still in beta status, which means that many improvements and optimisation steps are needed to get a toolkit which is really useful. There is a gap of several percent of performance if different toolkits are used for the same model, especially between GMTK and HTK. In many different applications the HTK has outperformed GMTK by about 3%.

The performance of graphical models is highly depending on the features which are used as an input. The use of multi-modal features is promising for many applications tested in this thesis, but not for all. Furthermore, the use of functionals, which are calculated over windows of the size of one second, does not improve the results. Graphical models, which are developed for dynamic segments always analyse the entire segment and therefore the functionals are not needed. In case support vector machines are used as classifier, the functionals are the most important features, since they have the possibility to represent the development of signals over time. This is very important for support vector machines, because otherwise they only take a single frame into account for the decision about the class.

Graphical models are very powerful and achieve good results if the test data is not pre-segmented. This means, the model has to find the correct boundaries between two classes. Ideally, the model has been trained on a data set where the class boundaries are exactly known. On the other side, the performance is better if more classes have to be distinguished.

9.2 Outlook

In the future, some developments can be performed to further improve graphical models and make them even more powerful. The research should not be limited to new applications or other real life problems, but should also include some theoretical improvements for the research community.

The toolkits can be improved and optimised. It should be easier to integrate a new model into a toolkit, if the structure has been developed on paper, than with the current toolkits. This whould help to improve rapid-prototyping with graphical models and it would get more common. Theoretical improvements exist for the learning algorithms, but they are not implemented in the toolkits. Furthermore, asynchronous data streams should be possible, because for many multi-modal applications it is very difficult to get the data perfectly synchronised.

Very complex models need approximation to be tractable. Until now, only few research has been conducted in this field. The approximation during the calculations would allow the training of complex models, which for example can consist of various existing models of different applications in the same domain which influence each other. This could be done in the meeting scenario where different applications have been analysed and some are depending on each other. As mentioned above, more flexible toolkits would help to integrate such a complex model.

The last very interesting field of research is the automatic learning of model structures. In this field, few research has been conducted for static problems, but no research for dynamic problems is known. Since brute force algorithms are used for static problems, they can not be transfered to dynamic problems, because of the computational power which would be needed. An algorithm which finds the best structure would come up with the best solution for the problem, as it finds the optimized dependencies between all variables.

To sum up this work, graphical models are getting more popular in the future, if the toolkits are getting more optimised und more flexible. They will be used for many real-life problems, which can be formulated as a pattern recognition task, where dynamic segmentation of the multi-modal input data is needed.

A

Used Measures

A.1 Recognition Accuracy Rate

The recognition accuracy rate (ACC) is calculated in the case that pre-defined segments are available and only a classification for these segments have to be performed. It is used for all corpuses, as the annotations are either based on segments or boundaries are defined. Therefore, segments can be extracted from the data via the annotation. The ACC is calculated by counting all the correct classified segments and dividing the number by the total count of segments. A high ACC points out that the classifiers work well.

$$\mathrm{ACC} = \frac{\text{number of correct classified segments}}{\text{total number of segments}} \times 100\%.$$

A.2 F-Measure

For the calculation of the f-measure, recall and precision have to be evaluated first. These measures are helpful to identify the performance for each of the classes separately and to identify which classes can be distinguished. Precision can be seen as a measure of fidelity, whereas recall is a measure of completeness. The f-measure is the harmonic mean of precision and recall. For the calculation, the number of relevant segments, which are taken from the annotation of the class, and the number of retrieved segments, which are taken from the classification results of the class, are needed. These scores are calculated for the ABC corpus only. All these measures are in a range between 0 and 1. The best score of them is 1.

$$\text{Recall} = \frac{|\text{number of relevant segments} \cap \text{number of retrieved segments}|}{|\text{number of relevant segments}|}$$

$$\text{Precision} = \frac{|\text{number of relevant segments} \cap \text{number of retrieved segments}|}{|\text{number of retrieved segments}|}$$

$$\text{f-measure} = 2 \times \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}}$$

A.3 Frame Error Rate

The frame error rate (FER) is used for the evaluation of videos which have to be segmented. This is necessary for the meeting corpus, as the meetings are not pre-segmented. The measure gives an impression of the quality of the automatically segmented and classified meeting. Compared to the first two measures, which do not take into account any sequence of segments, the sequence and the correct boundaries are important to achieve a good score. Each frame is compared with the annotation and if a mismatch is detected the number of wrong classified frames is increased. If all frames are compared, this number is divided by the total number of available frames. As it is an error rate, lower scores are better. The best performance whould be 0, which means that all frames are classified correctly.

$$\text{FER} = \frac{\text{number of wrong classified frames}}{\text{total number of frames}} \times 100\%.$$

A.4 Action Error Rate

The action error rate (AER) is again used for the evaluation of automatically segmented data. Compared to the frame error rate, only the sequence of classified segments are taken into account. This means, that the boundaries between two segments are not necessarily at the correct point of time. If the sequence does not match, it is checked if additional segments are added (Insertions), if segments have been removed (Deletions) or if a segment has been replaced by another one (Substitutions). The sum of these is divided by the total number of annotated segments. The best AER is 0, which does not necessarily mean that the output is good, because only the sequence is correct, but the boundaries have not to be at the correct point of time. An upper limit does not exist, as the number of insertions, deletions and substitutions can be higher than the total number of annotated segments.

$$\text{AER} = \frac{\text{Insertions} + \text{Deletions} + \text{Substitutions}}{\text{number of annotated segments}} \times 100\%.$$

Acronyms

ABC	Aircraft Behaviour Corpus
ACC	Recognition Accuracy Rate
AER	Action Error Rate
aggr	Aggressive
AMI	Augmented Multi-party Interaction
AMIDA	Augmented Multi-party Interaction with Distant Access
AVIC	Audiovisual Interest Corpus
BN	Bayesian Network
C	Vertex class
CC	Vertex class counter
CG	Vertex class group
chee	Cheerful
CHIL	Computers in the Human Interaction Loop
CP,CP2	Vertex class position
CT	Vertex class transition
DBN	Dynamic Bayesian Network
DAG	Directed Acyclic Graph
FC	Vertex frame counter
EM	Expectation-Maximisation
FER	Frame Error Rate
F1	F-measure

GM	Graphical Model
GMTK	Graphical Model Toolkit
HMM	Hidden Markov Model
HTK	Hidden Markov Model Toolkit
ICSI	International Computer Science Institute
into	Intoxicated
LOI	Level of Interest
MFCC	Mel Frequency Cepstral Coefficients
nerv	Nervous
neut	Neutral
obs,obs2	Vertex observation
Pre	Precision
Rec	Recall
RGB	Red green blue colour space
rg	Red green colour space
PCA	Principal Component Analysis
SFS	Sequential Forward Selection
SP,SP2	Vertex state pool
SPOT	Screening Passengers by Observation Techniques
ST,ST2	Vertex state transition
STD	Standard Deviation
SVM	Support Vector Machine
tire	Tired
TUM	Technischen Universität München

List of Symbols

a_t	Transition probability at time frame t
$\vec{b}(t)$	Motion vector
c_t	Class at time frame t
$\mathcal{C}_t$	Class counter at time frame t
D	Dimension of a matrix
$\Delta m_x^L(t)$	Global motion: movement of the center of motion in x-direction for a location L
$\Delta m_y^L(t)$	Global motion: movement of the center of motion in y-direction for a location L
E	Set of all edges of a graph
$\mathcal{F}_D$	D dimensional acoustic, visual or audiovisual feature set
$\mathbf{f_t}$	Feature vector at time frame t
$\mathbf{f_{t,ord}}$	Sorted feature vector at time frame t
$\mathcal{G}$	Graph containing vertices V and edges E
$\mathcal{G}^D$	Directed Acyclic Graph
$\vec{h}$	Hidden vertices of a model
$\vec{h}^*$	Configuration of hidden vertices with the highest probability of a model
I	Number of available data samples for the training or the evaluation
$I_d(x,y)$	Different image sequence
$i^L(t)$	Global motion: intensity of motion for a location L
$J(\cdot)$	Cost function for the SFS

κ	Inter-annotator agreement
L	Location within an image
$\mathcal{L}$	Data likelihood
$\mathbf{\Lambda}$	Diagonal matrix of the eigenvalues of the covariance matrix $\mathbf{\Phi}$
λ	Model parameters of the graphical model
$m_x^L(t)$	Global motion: center of motion x-value for a location L
$m_y^L(t)$	Global motion: center of motion y-value for a location L
μ	Mean value
$\mathbf{O}$	Set of all observations for one training or test
$\vec{o}_t$	Observation at time frame t
$\vec{o}_{n,t}$	Feature vector of stream n at time frame t
$\vec{o}_t$	Feature vector at time frame t
$\vec{o}_{ac}^{best}$	Best acoustic feature vector selected by SFS
$\vec{o}_{vis}^{best}$	Best visual feature vector selected by SFS
$p(x_i)$	Probability that random variable X has the value x_i
$\mathrm{pa}(V_i)$	Parents of the vertex V_i
$\mathbf{\Phi}$	Covariance matrix
q_t	State pool at time frame t
q_t^c	Class position at time frame t
R	Dimension of the by SFS selected feature space
$S_0(f_i)$	Individual significance of feature f_i
$S(f_i, \mathcal{X})$	Joint significance of feature f_i and feature set $\mathcal{X}$
σ^2	Variance
$\sigma_x^L(t)$	Global motion: mean absolute deviation of the difference pixels in x-direction for a location L
$\sigma_y^L(t)$	Global motion: mean absolute deviation of the difference pixels in y-direction for a location L
T	Length of an observation
t	time or time frame
$\mathbf{U}$	Matrix containing the eigenvectors
$\mathbf{u_d}$	Eigenvector

V	Set of all vertices of a graph
w_t	Class transition at time frame t
V_i	Vertex i of a graph
X_{V_i}	Random variable of vertex V_i
y_r	By SFS selected features
$\mathcal{Y}_k$	Current best selected feature set by SFS
$\mathcal{Y}_R$	Best selected feature set by SFS
$\#$	Total number of samples or instances

Bibliography

[AC75] M. Argyle and M. Cook. *Gaze and Mutual Gaze.* Cambridge University Press, Cambridge, England, 1975.

[Adm06] Transportation Security Administration. Train police officers to spot terrorist related activity. `http://www.tsa.gov/press/releases/2006/press_release_0655.shtm`, April 6, 2006.

[AH08] M. Al-Hames. *Graphische Modelle in der Mustererkennung.* Dissertation, Technische Universität München, Munich, Germany, 2008.

[AHDGP+06] 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, *Proceedings of the 2nd International WorkshopMachine Learning for Multimodal Interaction (MLMI)*, volume LNCS 3869, pages 52–63, Edinburgh, Scotland, 2006.

[AHHM+07] M. Al-Hames, B. Hörnler, R. Müller, J. Schenk, and G. Rigoll. Automatic multi-modal meeting camera selection for video-conferences and meeting browsing. In *Proceedings of the 8th International Conference on Multimedia and Expo (ICME)*, 2007.

[AHHSR06] M. Al-Hames, B. Hörnler, C. Scheuermann, and G. Rigoll. Using audio, visual, and lexical features in a multi-modal virtual meeting director. In S. Renals and S. Bengio, editors, *Proceedings of the 3nd International Workshop on Machine Learning for Multimodal Interaction (MLMI).* Springer Verlag, 2006.

[AHR06] M. Al-Hames and G. Rigoll. Der EM-Algorithmus und Gauß-Mixtur-Modelle. Skript zum Praktikum Praxis der Mensch-Maschine-Kommunikation, Technische Universität München, Lehrstuhl für Mensch-Maschine-Kommunikation, 2006.

[AHR08] D. Arsić, B. Hörnler, and G. Rigoll. Automated video editing for meeting scenarios applying multimodal low level feature fusion. In *Proceedings of the 5th International Workshop on Machine Learning and Multimodal Interaction (MLMI)*, September 2008.

[AHSR09] D. Arsić, B. Hörnler, B. Schuller, and G. Rigoll. A hierarchical approach for visual suspicious behavior detection in aircrafts. In *Proceedings of the 16th IEEE International Conference on Digital Signal Processing (DSP2009), Special Session Biometric recognition and verification of persons and their activities for video surveillance*, Santorini, Greece, July 2009.

[ALC^{+}07] A.B. Ashraf, S. Lucey, T. Chen, K. Prkachin, P. Solomon, Z. Ambadar, and J.F. Cohn. The painful face: Pain expression recognition using active appearance models. In *Proceedings of the 9th International Conference on Multimodal Interfaces (ICMI), Special Session on Multimodal Analysis of Human Spontaneous Behaviour*, pages 9–14, Nagoya, Japan, 2007. ACM SIGCHI.

[AOJJ90] S. Andersen, K. Olesen, F.V. Jensen, and F. Jensen. Hugin: a shell for building bayesian belief universes for expert systems. In G. Shafer and J. Pearl, editors, *Readings in Uncertain Reasoning*, pages 332–337. Morgan Kaufmann, 1990.

[Ars10] D. Arsić. *Detection and Tracking of Objects for Behavioral Analysis in Sensor Networks.* Dissertation, Technische Universität München, Munich, Germany, 2010.

[ASHR09] D. Arsić, B. Schuller, B. Hörnler, and G. Rigoll. Resolving partial occlusions in crowded environments utilizing range data an video cameras. In *Proceedings of the 16th IEEE International Conference on Digital Signal Processing (DSP2009), Special Session Fusion of Heterogeneous Data for Robust Estimation and Classification*, Santorini, Greece, July 2009.

[ASR07] D. Arsić, B. Schuller, and G. Rigoll. Suspicious behavior detection in public transport by fusion of low-level video descriptors. In *Proceedings of the 8th International Conference on Multimedia and Expo (ICME)*, pages 2018–2021, Beijing, China, July 2–5 2007.

[ATH03] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In S. Thrun S. Becker and K. Obermayer, editors, *Advances in Neural Information Processing*

Systems 15 (NIPS), pages 561–568. MIT Press, Cambridge, MA, USA, 2003.

[BB73] K. Burns and E. Beier. Significance of vocal and visual channels in the decoding of emotional meaning. *The Journal of Communication*, 23(1):118–130, 1973.

[BB05] J. Bilmes and C. Bartels. Graphical model architectures for speech recognition. *IEEE Signal Processing Magazine*, 22(5):89–100, 2005.

[BC79] R. Bales and S. Cohen. *SYMLOG: A System for the Multiple Level Observation of Groups*. Free Press, New York, 1979.

[BCCP01] S. Basu, T. Choudhury, B. Clarkson, and A. Pentland. Learning human interactions with the influence model. Technical report, MIT Media Laboratory Technical Note, 2001.

[Bel05] H. Beller. *Handbuch der Filmmontage - Praxis und Prinzipien des Filmschnitts*. TR-Verlagsunion, München, 5. edition edition, 2005.

[Ben03] S. Bengio. An asynchronous Hidden Markov Model for audio-visual speech recognition. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems (NIPS) 15*, pages 1237–1244. MIT Press, 2003.

[BH05] A.U. Batur and M.H. Hayes. Adaptive active appearance models. *IEEE Transactions on Image Processing*, 14:1707–1721, 2005.

[Bil97] J. Bilmes. A gentle tutorial on the EM algorithm and its application to parameter estimation for gaussian mixture and Hidden Markov Models. Technical Report ICSI-TR-97-021, University of Berkeley, 1997.

[Bil03] J. Bilmes. Graphical models in speech and language, and the graphical models toolkit. `http://www.clsp.jhu.edu/ws03/preworkshop/lecture_bilmes.pdf`, 2003. The Center for Language and Speech Processing Workshop 2003 at Johns Hopkins University.

[Bil04] J. Bilmes. Graphical models and automatic speech recognition. In M. Johnson, S.P. Khudanpur, M. Ostendorf, and R. Rosenfeld, editors, *Mathematical Foundations of Speech and Language Processing*, pages 191–246. Springer-Verlag, 2004.

[Bil06] J. Bilmes. Graphical models. `http://ssli.ee.washington.edu/courses/ee512`, 2006. Lecture EE512, University of Washington, Department of Electrical Engineering.

[BK02] C. Borgelt and R. Kruse. *Graphical Models: Methods for Data Analysis and Mining.* Jon Wiley & Sons, 2002.

[BM76] J.A. Bondy and U.S.R. Murty. *Graphentheoretische Methoden und ihre Anwendungen.* Elseviier Science Publishing, 1976.

[BSH$^+$05] A. Batliner, S. Steidl, C. Hacker, E. Nöth, and H. Niemann. Private emotions vs. social interaction - towards new dimensions in research on emotion. In *Proceedings of the International Workshop on Adapting the Interaction Style to Affective Factors, at the 10th International Conference on User Modelling*, Edinburgh, Scotland, 2005.

[BSS$^+$06] A. Batliner, S. Steidl, B. Schuller, D. Seppi, K. Laskowski, T. Vogt, L. Devillers, L. Vidrascu, N. Amir, L. Kessous, and V. Aharonson. Combining efforts for improving automatic classification of emotional user states. In *Proceedings of the 5th Slovenian and 1st International Language Technologies Conference (IS-LTC)*, pages 240–245, October 2006.

[Bul79] M. Bullowa. *Before speech: the beginning of interpersonal communication.* Cambridge University Press, 1979.

[Bun95] Wray L. Buntine. Chain graphs for learning. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 46–54, 1995.

[BZ02] J. Bilmes and G. Zweig. The graphical model toolkit: An open source software system for speech and time-series processing. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2002.

[CAB$^+$06] 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 meeting corpus: A pre-announcement. In *Proceedings of the 2nd Workshop on Machine Learning for Multimodal Interaction (MLMI)*, pages 28–39. Springer-Verlag, 2006.

[CBM02] R. Collobert, S. Bengio, and J. Mariéthoz. Torch: a modular machine learning software library. Technical Report 46, IDIAP, 2002.

[CDCT$^+$01] R. Cowie, E. Douglas-Cowie, N. Tsapatsoulis, G. Votsis, S. Kollias, W. Fellenz, and J. G. Taylor. Emotion recognition in human-computer interaction. *IEEE Signal Processing magazine*, 18(1):32–80, January 2001.

[CDLS99] R. Cowell, A.P. Dawid, S.L. Lauritzen, and D. Spiegelhalter. *Probabilistic Networks and Expert Systems.* Statistics for Engineering and Information Science. Springer-Verlag, corrected second edition edition, 1999.

[CF08] N. L Carter and J. M. Ferryman. The SAFEE on-board threat detection system. In *International Conference on Computer Vision Systems (CVS)*, pages 79–88, May 2008.

[Cha91] E. Charniak. Bayesian networks without tears: Making bayesian networks more accessible to the probabilistically unsophisticated. *Artificial Intelligence Magazine*, 12(4):50–63, 1991.

[Cow01a] R. Cowell. Advanced inference in bayesian networks. In M.I. Jordan, editor, *Learning in Graphical Models*, pages 27 – 49. MIT Press, 2001.

[Cow01b] R. Cowell. Introduction to inference for bayesian networks. In M.I. Jordan, editor, *Learning in Graphical Models*, pages 9 – 26. MIT Press, 2001.

[Dar09] A. Darwiche. *Modeling and Reasoning with Bayesian Networks.* Cambridge University Press, 2009.

[DK82] P.A. Devijver and J. Kittler. *Pattern recognition: A statistical approach.* Prentice Hall, 1982.

[DLR77] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society B*, 39(1):1–38, 1977.

[Don06] S. Donnelly. A new tack for airport screening: Behave yourself. `http://www.time.com/time/nation/article/0,8599,1195330,00.html`, May 17, 2006. Time Inc.

[Edw95] D. Edwards. *Introduction to Graphical Modelling.* Springer Texts in Statistics. Springer-Verlag, corrected second edition edition, 1995.

[EKR98] S. Eickeler, A. Kosmala, and G. Rigoll. Hidden markov model based continuous online gesture recognition. In *Proceedings of the 14th International Conference on Conference on Pattern Recognition (ICPR)*, pages 1206–1208, 1998.

[EWS09] F. Eyben, M. Wöllmer, and B. Schuller. openear - introducing the munich open-source emotion and affect recognition toolkit. In *Proceedings of the 4th International HUMAINE Association Conference on*

Affective Computing and Intelligent Interaction 2009 (ACII). IEEE, 2009.

[FGZ01] Z. Fang, Z. Guoliang, and S. Zhanjiang. Comparison of different implementations of MFCC. *Journal of Computer Science and Technology*, 16(6):582–589, 2001.

[FO70] M. Hamit Fisek and R. Ofshe. The process of status evolution. *Sociometry*, 33(3):327–346, 1970.

[Fuk90] K. Fukunaga. *Principal Component Analysis*. Academic Press, 1990.

[Gau04] D. Gaultier. SAFEE targets on-aircraft security. Technical report, European Commission, February 2004.

[GB68] G.A. Gorry and G.O. Barnett. Experience with a model of sequential diagnosis. *Computers and Biomedical Research*, 1(5):490–507, 1968.

[GBC$^+$00] A. Girgensohn, J. Boreczky, P. Chiu, J. Doherty, J. Foote, G. Golovchinsky, S. Uchihashi, and L. Wilcox. A semi-automatic approach to home video editing. In *Proceedings of the 13th annual ACM symposium on User interface software and technology (UIST)*, pages 81–89. ACM, 2000.

[Gha98] Z. Ghahramani. Learning dynamic bayesian networks. In C.L. Giles and M. Gori, editors, *Adaptive Processing of Sequences and Data Structures*, Lecture Notes in Artificial Intelligence, pages 168–197. Springer-Verlag, 1998.

[GPMZB05] D. Gatica-Perez, I. McCowan, D. Zhang, and S. Bengio. Detecting group interest-level in meetings. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2005.

[Gwe01] K. Gwet. *Statistical Tables for Inter-Rater Agreement*. STATAXIS Publishing Company, 2001.

[Hal77] M. Halliday. Learning how to mean: explorations in the development of language. *Language in Society*, 6(01):114–118, 1977.

[HAR09] B. Hörnler, D. Arsić, and G. Rigoll. Graphical models for multi-modal automatic video editing in meetings. In *Proceedings of the 16th IEEE International Conference on Digital Signal Processing (DSP2009), Special Session Fusion of Heterogeneous Data for Robust Estimation and Classification*, Santorini, Greece, July 2009.

[HASR09] B. Hörnler, D. Arsic, B. Schuller, and G. Rigoll. Boosting multimodal camera selection with semantic features. In *Proceedings of the International Conference on Multimedia and Expo (ICME)*, pages 1298–1301, 2009.

[Haw07] K. Hawley. Aviation security in the future: Is there a better way? In *U.S. - Europe Aviation Security Policy Conference*, July 2–4, 2007.

[Hec01] D. Heckerman. A tutorial on learning with bayesian networks. In M.I. Jordan, editor, *Learning in Graphical Models*, pages 301 – 354. MIT Press, 2001.

[HNR06] D. Heylen, A. Nijholt, and D. Reidsma. Determining what people feel and think when interacting with humans and machines: Notes on corpus collection and annotation. In J. Kreiner and C. Putcha, editors, *Proceedings 1st California Conference on Recent Advances in Engineering Mechanics*, 2006.

[HR09] B. Hörnler and G. Rigoll. Multi-modal activity and dominance detection in smart meeting rooms. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 1777–1780, 2009.

[IB98] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, 1998.

[JBE$^+$03] 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 of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Hong Kong, China, 2003.

[Jen96] F. Jensen. *An Introduction to Bayesian Networks.* UCL Press, 1996.

[Jen02] F. Jensen. *Bayesian Networks and Decision Graphs.* Statistics for Engineering and Information Science. Springer Verlag, second corrected edition edition, 2002.

[Jen10] F. Jensen. Bayesian networks. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2, 2010.

[JLO90] F. Jensen, S. Lauritzen, and K. Olesen. Bayesian updating in causal probabilistic networks by local computations. *Computational Statistics Quarterly*, 4:269–282, 1990.

[Jol02] I.T. Jolliffe. *Principal Component Analysis.* Springer-Verlag, 2 edition, 2002.

[Jor01] M.I. Jordan, editor. *Learning in Graphical Models.* MIT Press, second printing edition, 2001.

[JR64] W. Kees J. Ruesch. *Nonverbal Communication.* University of California Press, 1964.

[JS01] M.I. Jordan and T.J. Sejnowski, editors. *Graphical Models: Foundations of Neural Computation.* MIT Press, 2001.

[Kar47] K. Karhunen. Über lineare methoden in der wahrscheinlichkeitsrechnung. *Annales AcademiæScientiarum FennicæMathematica*, 1947.

[Kau71] A. Kaufmann. *Einführung in die Graphentheorie.* Orientierung und Entscheidung. Verlag R. Oldenbourg München und Wien, 1971.

[KE03] L. Kennedy and D. Ellis. Pitch-based emphasis detection for characterization of meeting recordings. In *Proceedings of the International Workshop on Automatic Speech Recognition and Understanding (ASRU)*, December 2003.

[KF09] D. Koller and N. Friedman. *Probabilistic Graphical Models.* MIT Press, 2009.

[Köh06] N. Köhler. Skalierungsinvariante Augenlokalisation und Konzeptionierung eines Systems zur automatischen Mimikerkennung. Diplomarbeit, Technische Universität München, 2006.

[Kip01] Michael Kipp. ANVIL - a generic annotation tool for multimodal dialogue, 2001.

[KM06] J. Karp and L. Meckler. Which travelers have hostile intent? biometric device may have the answer. `http://online.wsj.com/public/article/SB115551793796934752-Vns0SEJ5cKlwnUhTbmkwBeMPIAE_20080510.html`, August 14, 2006. The Wall Street Journal.

[KMH$^+$07] M. Kranz, A. Maldonado, B. Hörnler, R.B. Rusu, M. Beetz, G. Rigoll, and A. Schmidt. A knife and a cutting board as implicit user interface - towards context-aware kitchen utilities. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction (TEI)*, 2007.

[KMR+07] M. Kranz, A. Maldonado, R.B. Rusu, B. Hörnler, G. Rigoll, M. Beetz, and A. Schmidt. Sensing technologies and the player-middleware for context-awareness in kitchen environments. In *Proceedings of the 4th International Conference on Networked Sensing Systems (INSS)*, 2007.

[Knö69] W. Knödel. *Graphentheoretische Methoden und ihre Anwendungen.* Ökonometrie und Unternehmensforschung. Springer-Verlag, 1969.

[KPI04] A. Kapoor, R.W. Picard, and Y. Ivanov. Probabilistic combination of multiple modalities to detect interest. In *Proceedings of the 19th International Workshop on Pattern Recognition, (ICPR)*, volume 3, pages 969–972, 2004.

[KS00] M. Kudo and J. Slansky. Comparison of algorithms that select features for pattern classifiers. *Pattern Recognition Journal*, 22:25–41, 2000.

[Lau96] S.L. Lauritzen. *Graphical Models.* Oxford Statistical Science Series. Oxford Science Publications, 1996.

[Len07] C. Lenz. Das asynchrone hidden markov modell und seine anwendung in der multimodalen meetinganalyse, 2007. Diplomarbeit, Technische Universität München.

[LK03] Q. Liu and D. Kimber. Learning automatic video capture from human's camera operations. In *Proceedings of the International Conference on Image Processing (ICIP)*, 2003.

[LKF+02] Qiong Liu, Don Kimber, Jonathan Foote, Lynn Wilcox, and John Boreczky. Flyspec: a multi-user video camera system with hybrid human and automatic control. In *Proceedings of the 10th ACM international conference on Multimedia (ACMMM)*, pages 484–492. ACM, 2002.

[LO81] M.T. Lee and R. Ofshe. The impact of behavioral style and status characteristics on social influence: A test of two competing theories. *Social Psychology Quarterly*, 44(2):73–82, 1981.

[Loè78] M. Loève. Probability theory vol. II. *Graduate Tests in Mathematics*, 46, 1978.

[LS88] S.L. Lauritzen and D.J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society, Series B*, 50:157 – 224, 1988.

[LSK+05] Q. Liu, X. Shi, D. Kimber, F. Zhao, and F. Raab. An online video composition system. In *Proceedings of the International Conference on Multimedia and Expo (ICME)*, 2005.

[MA00] R. McEliece and S. M. Aji. The generalized distributive law. *IEEE Transactions on Information Theory*, 46(2):325–343, 2000.

[Mac09] I. MacLeod. Canadian airport to test behaviour detection program. `http://www.puppetgov.com/2009/08/14/canadian-airport-to-test-Śbehaviour-detectionŠ-program/`, August 14, 2009.

[MBB+03] N. Morgan, D. Baron, S. Bhagat, H. Carvey, R. Dhillon, J. Edwards, D. Gelbart, A. Janin, A. Krupski, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters. Meetings about meetings: Research at ICSI on speech in multiparty conversations. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2003.

[MGPB+05] I. McCowan, D. Gatica-Perez, S. Bengio, G. Lathoud, M. Barnard, and D. Zhang. Automatic analysis of multimodal group actions in meetings. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27:305–317, 2005.

[MHGB01] A. Morris, A. Hagen, H. Glotin, and H. Bourlard. Multi-stream adaptive evidence combination for noise robust ASR. *Speech Communication*, 34(1–2):25–40, 2001.

[MMF+06] S. Meignier, D. Moraru, C. Fredouille, J.-F. Bonastre, and L. Besacier. Step-by-step and integrated approaches in broadcast news speaker diarization. *Computer Speech & Language*, 20(2-3):303 – 330, 2006.

[Moo96] T.K. Moon. The expectation-maximization algorithm. *IEEE Signal Processing Magazine*, 13(6):47 – 60, 1996.

[Moo02] D. Moore. The IDIAP smart meeting room. Technical Report 07, IDIAP, 2002.

[MP03] S. Mota and R. Picard. Automated posture analysis for detecting learner's interest level. In *Proceedings of the International Workshop on Computer Vision and Pattern Recognition (CVPR) for HCI*, June 2003.

[MP07] L. Maat and M. Pantic. Gaze-x: Adaptive, affective, multimodal interface for single-user office scenarios. In *Artifical Intelligence for Human Computing*, volume 4451/2007, pages 251–271. Springer LNCS, Berlin, Heidelberg, 2007.

[MR83] P.L. McLeod and R. Rosenthal. Micromomentary movement and the deconding of face and body cues. *Journal of Nonverbal Behavior*, 8(2), 1983.

[Mur01] K. Murphy. The bayes net toolbox for matlab. *Computing Science and Statistics*, 33, 2001.

[Mur02] K. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning.* PhD thesis, University of California, Berkeley, 2002.

[MYLB08] A.A. Miranda and G. Bontempi Y. Le Borgne. New routes from minimal approximation error to principal components. *Neural Processing Letters*, 27(3), 2008.

[Nea03] R.E. Neapolitan. *Learning Bayesian Networks.* Artificial Intelligence Series. Prentice Hall, 2003.

[Nie93] J. Nielsen. *Usability Engineering.* Academic Press, Inc., 1993.

[NL86] J.-L. Nespoulous and A.R. Lecours. Gestures: Nature and function. In J.-L. Nespoulous, P. Perron, and A.R. Lecours, editors, *The Biological foundations of gestures: motor and semiotic aspects*, pages 49–62. Lawrence Erlbaum Associates, Hillsdale, NJ, USA, 1986.

[NSGN02] R. Nieschulz, B. Schuller, M. Geiger, and R. Neuss. Aspects of efficient usability engineerings. *it+ti journal, "Usability Engeneering"*, 44(1):23–30, 2002.

[OMI02] Y. Ohsawa, N. Matsumura, and M. Ishizuka. Influence diffusion model in text-based communication. In *Proceedings of the 11th International Conference on world wide web*, 2002.

[Pea86] J. Pearl. Fusion, propagation, and structuring in belief networks. *Artificial Intelligence*, 29:241 – 288, 1986.

[Pea88] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann Publishers, San Fransisco, California, revised second printing edition, 1988.

[PM05] A. Pentland and A. Madan. Perception of social interest. In *Proceedings of the 10th International Conference on Computer Vision, Workshop on Modeling People and Human Interaction (ICCV-PHI)*, Beijing, China, October 2005.

[PNK94] P. Pudil, J. Novovičová, and J. Kittlera. Floating search methods in feature selection. *Pattern Recognition Letters*, 14, 1994.

[PNLM04] G. Potamianos, C. Neti, J. Luettin, and I. Matthews. Audio-visual automatic speech recognition: An overview. In G. Bailly, E. Vatikiotis-Bateson, and P. Perrier, editors, *Issues in Visual and Audio-visual Speech Processing*, chapter 10. MIT Press, 2004.

[PR03] M. Pantic and L. Rothkrantz. Toward an affect-sensitive multimodal human-computer interaction. *Proccedings of the IEEE*, 91:1370–1390, September 2003.

[Pra01] W.K. Pratt. *Digital image processing.* John Wiley & Sons, 2001.

[PSS04] I. Potucek, S. Sumec, and M. Spanel. Participant activity detection by hands and face movement tracking in the meeting room. In *Proceedings of the IEEE Computer Graphics International (CGI)*, 2004.

[PV87] J. Pearl and T. Verma. The logic of representing dependencies by directed graphs. In *Sixth National Conference on Artificial Intelligence*, volume 1, pages 374–379. American Association of Artificial Intelligence (AAAI), 1987.

[QBZ05] P. Qvarfordt, D. Beymer, and S. X. Zhai. *RealTourist - A Study of Augmenting Human-Human and Human-Computer Dialogue with Eye-Gaze Overlay*, volume LNCS 3585, pages 767–780. Springer Berlin / Heidelberg, 2005.

[Rab89] L.R. Rabiner. A tutorial on Hidden Markov Models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–285, 1989.

[RBK98] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 20(1):23–38, January 1998.

[Rei08] S. Reiter. *Multimodale Modellierung von Gruppenaktionen zur Segmentierung von Besprechungen.* Dissertation, Technische Universität München, Munich, Germany, 2008.

[Ren02a] S. Renals. The M4-Project – multimodal meeting manager. `http://www.dcs.shef.ac.uk/spandh/projects/m4/index.html`, 2002.

[Ren02b] S. Renals. The M4-Project – multimodal meeting manager, project leaflet. `http://www.dcs.shef.ac.uk/spandh/projects/m4/pdf/leaflet_lowres.pdf`, 2002.

[RH05] R. Rienks and D. Heylen. Automatic dominance detection in meetings using easily obtainable features. In S. Renals and S. Bengio, editors, *Proceedings of the 2nd International Workshop on Machine Learning for Multimodal Interaction (MLMI)*, volume LNCS 3869, pages 76–86, Edinburgh, Scotland, 2005.

[Rie07] R. Rienks. *Meetings in smart environments : implications of progressing technology.* PhD thesis, University of Twente, Enschede, Netherlands, 2007.

[Rig94] G. Rigoll. *Neuronale Netze. Eine Einführung für Ingenieure, Informatiker und Naturwissenschaftler.* Expert-Verlag, 1994.

[RJ93] L.R. Rabiner and B.-H. Juang. Theory and implementation of Hidden Markov Models. In *Fundamentals of Speech Recognition*, pages 321 – 389. Prentice Hall PTR, 1993.

[RK97] G. Rigoll and A. Kosmala. New improved feature extraction methods for real-time high performance image sequence recognition. In *Proceedings of the 22nd International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 4, pages 2901–2904, April 1997.

[RM79] E. Rosa and A. Mazur. Incipient status in small groups. *Social Forces*, 58(1):18–37, 1979.

[RSR05] S. Reiter, S. Schreiber, and G. Rigoll. Multimodal meeting analysis by segmentation and classification of meeting events based on a higher level semantic approach. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 161–164, 2005.

[RSR07] S. Reiter, B. Schuller, and G. Rigoll. Hidden conditional random fields for meeting segmentation. In *Proceedings of the 8th International Conference on Multimedia and Expo (ICME)*, pages 639–642, Beijing, China, 2007.

[SAR+07] B. Schuller, D. Arsic, G. Rigoll, M. Wimmer, and B. Radig. Audiovisual behaviour modeling by combined feature spaces. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume II, pages 733–736, Honolulu, HY, USA, 2007.

[SBE+09] M. Schröder, E. Bevacqua, F. Eyben, H. Gunes, D. Heylen, M. ter Maat, S. Pammi, M. Pantic, C. Pelachaud, B. Schuller, E. de Sevin, M. Valstar, and M. Wöllmer. A demonstration of audiovisual sensitive artificial listeners. In *Proceedings of the 4th International HUMAINE Association Conference on Affective Computing and Intelligent Interaction (ACII)*, pages 263–264, 2009.

[SCH+08] M. Schröder, R. Cowie, D. Heylen, M. Pantic, C. Pelachaud, and B. Schuller. Towards responsive sensitive artificial listeners. In *Proceedings of the 4th International Workshop on Human-Computer Conversation*, 2008.

[Sch09] J. Schenk. *Online-Erkennung handgeschriebener Whiteboard-Notizen.* Dissertation, Technische Universit¨at München, München, 2009.

[Sch10] M. Schröder. The SEMAINE API: Towards a standards-based framework for building emotion-oriented systems. *Advances in Human-Machine Interaction*, 2010.

[SER08a] B. Schuller, F. Eyben, and G. Rigoll. Beat-synchronous data-driven automatic chord labeling. In U. Jekosch and R. Hoffmann, editors, *Proceedings of the Deutschen Arbeitsgemeinschaft für Akustik, DAGA*, pages 555–556. DEGA, 2008. 10.-13.03.2008.

[SER08b] B. Schuller, F. Eyben, and G. Rigoll. Tango or waltz?: Putting ballroom dance style into tempo detection. *EURASIP Journal on Audio, Speech, and Music Processing*, 2008(Article ID 846135):12 p, 2008.

[Sha71] M.E. Shaw. *Group Dynamics: The Psychology of Small Group Behavior.* McGraw Hill, New York, 1971.

[SHAR09] B. Schuller, B. Hörnler, D. Arsić, and G. Rigoll. Audio chord labeling by musiological modeling and beat-synchronization. In *Proceedings of the 9th International Conference on Multimedia and Expo (ICME)*, 2009.

[SHBR09] J. Schenk, B. Hörnler, A. Braun, and G. Rigoll. Graphical models: Statistical inference vs. determination. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 1717–1720, 2009.

[SHML00] M. Soriano, S. Huovinen, B. Martinkauppi, and M. Laaksonen. Skin detection in video under changing illumination conditions. In *Proceedings of the 15th International Conference on Pattern Recognition (ICPR)*, pages 839–842, 2000.

[Shr05] E. Shriberg. Spontaneous speech: How people really talk and why engineers should care. In *Proceedings of the 9th European Conference on Speech Communication and Technology (Eurospeech)*, Lisbon, Portugal, September 2005.

[SKMR06] B. Schuller, N. Köhler, R. Müller, and G. Rigoll. Recognition of interest in human conversational speech. In *Proceedings of the 9th International Conference on Spoken Language Processing (ICSLP - Interspeech)*, pages 793–796, 2006.

[SKR09] J. Schenk, M. Kaiser, and G. Rigoll. Selecting features in on-line handwritten whiteboard note recognition: SFS or SFFS? In *10th International Conference on Document Analysis and Recognition*, 2009.

[SME$^+$09] B. Schuller, R. Müller, F. Eyben, J. Gast, B. Hörnler, M. Wöllmer, G. Rigoll, A. Höthker, and H. Konosu. Being bored? recognising natural interest by extensive audiovisual integration for real-life application. *Image and Vision Computing Journal (IMAVIS)*, 27:1760–1774, 2009. Special Issue on Visual and Multimodal Analysis of Human Spontaneous Behavior. Elsevier, Issue 12.

[SMH$^+$07] B. Schuller, R. Müller, B. Hörnler, A. Höthker, H. Konosu, and G. Rigoll. Audiovisual recognition of spontaneous interest within conversations. In *Proceedings of International Conference on Multimodal Interfaces (ICMI)*, pages 30–37, 2007. Special Session on Multimodal Analysis of Human Spontaneous Behaviour.

[Smy97] P. Smyth. Belief networks, hidden markov models, and markov random fields: a unifying view. *Pattern Recognition Letters*, 18(11-13):1261–1268, 1997.

[SNVF03] L. Snidaro, R. Niu, P. K. Varshney, and G. L. Foresti. Automatic camera selection and fusion for outdoor surveillance under changing weather conditions. In *Proceedings of the Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2003.

[SP85] S. Sabri and B. Prasada. Video conferencing systems. *Proceedings of the IEEE*, 73(4):671–688, 1985.

[SR09] B. Schuller and G. Rigoll. Recognising interest in conversational speech - comparing bag of frames and supra-segmental features. In *Proceedings of the 10th Annual Conference of the International Speech Communication Association (Interspeech)*, pages 1999–2002. ISCA, 2009.

[SRL03] B. Schuller, G. Rigoll, and M. Lang. Hidden markov model-based speech emotion recognition. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume II, pages 1–4, 2003.

[SS90] G. Shafer and P. Shenoy. Probability propagation. *Annals of Mathematics and Artificial Intelligence*, 2:327–351, 1990.

[SSB$^+$07] B. Schuller, D. Seppi, A. Batliner, A. Maier, and S. Steidl. Towards more reality in the recognition of emotional speech. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, April 2007.

[SSB09] B. Schuller, S. Steidl, and A. Batliner. The interspeech 2009 emotion challenge. In *Proceedings of the 10th International Conference of Speech Communication Association (Interspeech)*, 2009. to appear.

[Sum04] S. Sumec. Multi camera automatic video editing. In *Proceedings of the International Conference on Computer Vision and Graphics (ICCVG)*, pages 935–945, Warsaw, Poland, 2004. Kluwer Verlag.

[SWA$^+$07] B. Schuller, M. Wimmer, D. Arsić, G. Rigoll, and B. Radig. Audiovisual behavior modeling by combined feature spaces. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, April 2007.

[SWA$^+$08] B. Schuller, M. Wimmer, D. Arsić, T. Moosmayr, and G. Rigoll. Detection of security related affect and behaviour in passenger transport. In *Proceedings of the 9th International Conference of Speech Communication Association (Interspeech) incorp. 12th Australasian International Conference on Speech Science and Technology SST*, pages 265–268, Brisbane, Australia, 2008. ISCA.

[SWM$^+$08] B. Schuller, M. Wimmer, L. Mösenlechner, C. Kern, D. Arsić, and G. Rigoll. Brute-forcing hierarchical functionals for paralinguistics: A waste of feature space? In *Proceedings of the International Conference* *on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 4501–4504, 2008.

[SYW02] R. Stiefelhagen, Jie Yang, and A. Waibel. Modeling focus of attention for meeting indexing based on multiple cues. *IEEE Transactions on Neural Networks*, 13(4):928 – 938, July 2002.

[Tho93] K.R. Thorisson. Dialogue control in social interface agents. In *Proceedings of the Conference Companion on Human factors in Computing Systems (CHI)*, pages 139–140, New York, NY, USA, 1993. ACM.

[TKB92] K.R. Thorisson, D.B. Koons, and R.A. Bolt. Multi-modal natural dialogue. In *Proceedings of the Special Interest Group on Computer-Human Interaction (SIGCHI) Conference on Human factors in Ccomputing Systems (CHI)*, pages 653–654, New York, NY, USA, 1992. ACM.

[Uch01] S. Uchihashi. Direct camera control for capturing meetings into multimedia documents. In *Proceedings of the International Conference on Multimedia and Expo (ICME)*, 2001.

[Vap95] V. Vapnik. *The Nature of Statistical Learning Theory.* Springer Verlag, 1995.

[Vit77] A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260 – 269, 1977.

[VO06] A. Vinciarelli and J. Odobez. Application of information retrieval technologies to presentation slides. *IEEE Transactions on Multimedia*, 8(5):981–995, 2006.

[WFG04] P. Wellner, M. Flynn, and M. Guillemot. Browsing recorded meetings with ferret. In S. Renals and S. Bengio, editors, *Proceedings of the 1st International Workshop on Machine Learning for Multimodal Interaction (MLMI).* Springer Verlag, 2004.

[WFTW05] P. Wellner, M. Flynn, S. Tucker, and S. Whittaker. A meeting browser evaluation test. In *CHI '05 extended abstracts on Human factors in computing systems*, pages 2021–2024, New York, NY, USA, 2005. ACM Press.

[WH09] C. Wooters and M. Huijbregts. The ICSI RT07s speaker diarization system. In *Multimodal Technologies for Perception of Humans*, Lecture Notes in Computer Science, pages 509 – 519. Springer Berlin / Heidelberg, 2009.

[Whi91] J. Whittaker. *Graphical Models in Applied Multivariate Statistics.* Wiley Series in Probability & Mathematical Statistics. Jon Wiley & Sons, 1991.

[Wig79] J.S. Wiggins. A psychological taxonomy of trait-descriptive terms: The interpersonal domain. *Journal of Personality and Social Psychology*, 37(3):395–412, 1979.

[Win03] G. Winkler. *Image Analysis, Random Field and Markov Chain Monte Carlo Methods*. Applications of Mathematics, Stochastic Modelling and Applied Probability. Springer, second edition edition, 2003.

[WR05] M. Wimmer and B. Radig. Adaptive skin color classificator. In *Proceedings of the first International Conference on Graphics, Vision and Image Processing (ICGST)*, volume 1, pages 324–327, 2005.

[WSA$^+$08] M. Wimmer, B. Schuller, D. Arsić, B. Radig, and G. Rigoll. Low-level fusion of audio and video feature for multi-modal emotion recognition. In A. Ranchordas and H. Araujo, editors, *Proceedings of the 3rd International Conference Computer Vision Theory and Applications VISAPP*, volume 2, pages 145–151, Funchal, Madeira, Portugal, 2008.

[WSS04] A. Waibel, H. Steusloff, and R. Stiefelhagen. Chil - computers in the human interaction loop. In *Proceedings of the the NIST ICASSP Meeting Recognition Workshop*, 2004.

[WTVS61] H.R. Warner, A.F. Toronto, L.G. Veasey, and R. Stephenson. A mathematical approach to medical diagnosis: Application to congenital heart disease. *Journal of the American Medical Association (JAMA)*, 177(3):177–183, 1961.

[WZR04] F. Wallhoff, M. Zobl, and G. Rigoll. Action segmentation and recognition in meeting room scenarios. In *Proceedings of the International Conference on Image Processing (ICIP)*, 2004.

[YEH$^+$02] S. Young, G. Evermann, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland. *The HTK Book (for HTK Version 3.2.1).* Cambridge University Engineering Department, 2002.

[YKA02] 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.

[Zak07] P. Zak. Slide extraction in meetings. Technical Report, Technische Universität München, 2007.

[ZGPB$^+$04] 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 of the Second IEEE Workshop on Event Mining: Detection and Recognition of Events in Video, in Association with CVPR*, 2004.

[ZGPBR05] D. Zhang, D. Gatica-Perez, S. Bengio, and D. Roy. Learning influence among interacting markov chains. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems 18 (NIPS)*, pages 1577–1584. MIT Press, Cambridge, MA, USA, 2005.

[Zha06] D. Zhang. *Probabilistic Graphical Models for Human Interaction Analysis.* PhD thesis, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, 2006.

[ZPRH09] Z. Zeng, M. Pantic, G. I. Roisman, and T. S. Huang. A survey of affect recognition methods: Audio, visual and spontaneous expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 39–58, Januar 2009.

[ZWR03] M. Zobl, F. Wallhoff, and G. Rigoll. Action recognition in meeting scenarios using global motion features. In J. Ferryman, editor, *Proceedings of the Fourth International Workshop on Performance Evaluation of Tracking and Surveillance (PETS-ICVS)*, pages 32–36, 2003.

[ZZ03] Z. Zhou and M. Zhang. Ensembles of multi-instance learners. In *Proceedings of the 14th European Conference on Machine Learning*, pages 492–502. Springer, 2003.