FP6-506811

AMI

Augmented Multiparty Interaction

Integrated Project
Information Society Technologies

D4.1 Report on Implementation of Audio, Video, and Multimodal Algorithms

Due date: 31/12/2004 **Submission date:** 20/12/2004
Project start date: 1/1/2004 **Duration:** 36 months
Revision: 1

Lead contractor: TUM

Project co-funded by the European Commission in the 6th Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	✓
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

D4.1 Report on Implementation of Audio, Video, and Multimodal Algorithms

Editor: Marc Al-Hames, TUM

Abstract: WP4 is concerned with the automatic recognition from audio, video, and combined audio-video streams. Research topics include robust speech recognition for multiparty meetings, gesture and action recognition, emotion recognition, source localization and object tracking, keyword spotting, and person identification. Deliverable D4.1 is a report on the implementation and first evaluations of these audio, video, and multimodal algorithms.

Contents

1 Introduction

WP4 is concerned with the automatic recognition from audio, video, and combined audio-video streams, with an emphasis on developing models and algorithms to combine modalities. Algorithms will be implemented in the AMI domain and evaluated on common datasets. The models that will be applied include HMMs, Bayesian networks, neural networks, multistream approaches, and multisource decoding.

1.1 Involved partners

Table 1 shows the involved partners and person-months in WP4.

Part.	UEDIN	DFKI	ICSI	TNO	BUT	TUM
PMnth.	18	3	48	26	54	65
Part.	IDIAP	USFD	UT			FC
PMnth.	54	18	28			4

Table 1: Involved partners and person-months in WP4

1.2 Splitting of work

Instead of dividing the tasks into speech, visual, and audio-visual groups it was decided to split the tasks into problem-based groups. Solutions are not distinguished by their approach (for example visual or audio identification of persons). Therefore different approaches can be evaluated and compared on a common data set with a given standard (for example how many persons have been identified correctly during the meeting?). We identified seven main questions and therefore split WP4 into seven sub-groups:

- Baseline speech recognition system
- Event spotting
- Person segmentation / clustering / identification
- Emotion recognition
- Localization and Tracking
- Gestures and actions
- Focus of attention

1.3 Aim in the first year and outline of this deliverable

The expected result of WP4 is a set of multimodal recognizers for robust speech recognition for multiparty meetings, gesture and action recognition, emotion recognition, source localization, object tracking, and person identification.

In the first year we developed and ported a wide range of algorithms to the AMI domain. We decided about common interfaces and have draft evaluation schemes available. This allows us to compare different approaches and algorithms on common AMI data. This deliverable reports about the progress that has been made in porting and implementing these algorithms for audio, video, and multimodal algorithms for AMI. The methods are described in detail in Sec. 2 - 8, where each section describes the progress that has been made in one of the seven sub-groups (cf. Sec. 1.2).

2 Automatic Speech Recognition

2.1 Objectives

The ASR subgroup is concerned with the development of a speech recognition system for the use on AMI data. As AMI data for training and is not yet available preliminary system building work has started. Past experience in meeting transcription has shown that bootstrapping of meeting systems from other ASR systems is beneficial. Hence it was decided to initially develop models based on conversational telephone speech (CTS) and on the ICSI meeting corpus. The aim in this development is to prepare the necessary setup for ASR training and testing and the initial models for bootstrapping. Hence the following sections describe a system for CTS, followed by description of initial system results for ICSI meeting data.

For acoustic model training the Hidden Markov Model Toolkit (HTK) was used whereas language model training and testing was based on the SRI language model toolkit. The recognition process itself is based on HDecode, a speech recogniser in development at Cambridge University. In addition to these fundamental tools a large number of scripts and programs has been developed to enable simple and efficient execution of fundamental steps in the construction of speech recognition systems. This framework has and will enable straight-forward migration to different corpora and/or data sources.

2.2 Transcription of Conversational Telephone Speech

Work on conversational telephone speech is based on the 3 corpora: Switchboard-I, CallHome English, and Switchboard cellular. Word level transcripts and audio segmentations for training covering most of these corpora were obtained from Cambridge University (h5etrain03). Recognition experiments are conducted using the official 2001 NIST Hub5E evaluation set. For testing of word lists and language models in addition the both the 1998 and 2002 NIST Hub5E evaluation were used.

The following describes wordlist generation, dictionary construction, work on acoustic and language modelling as well as acoustic adaptation.

2.2.1 Wordlist generation

The wordlist of a speech recognition system is the set of words that it should be able to recognise. For a given corpus the total number of complete words (i.e. excluding false starts and partial words) can be relatively small (approximately 10,000 for ICSI meetings data for example). So it is likely that a test set will contain words that are common but did not appear in the acoustic training data. It is therefore necessary to augment the wordlist with words from a larger source.

To generate the wordlist for the CTS task the set of complete words from the Switchboard corpus is augmented with the most frequently occurring set of complete words from the HUB4 broadcast news corpus until a wordlist of the desired size is obtained. The resulting list can be compared to the words in the test set. Words occurring in the test set but not in the wordlist are called out-of-vocabulary words (OOVs). It is desirable to minimise the number of OOVs while keeping a reasonably sized wordlist. Note that no knowledge of the test set is used in the selection of words.

Table 2 shows the OOV rate for various wordlist sizes on the CTS test sets. A wordlist containing 50,000 words was finally chosen.

2.2.2 Dictionary

Dictionary development has comprised two main sub-tasks, the preparation of a baseline pronunciation dictionary and the addition of new entries to this dictionary from word lists extracted as described above.

The baseline dictionary is derived from the UNISYN pronunciation dictionary, a multi-accent dictionary developed at the Centre for Speech Technology Research, University of Edinburgh. The dictionary

Length of wordlist	CTS OOV rate (%)	ICSI OOV rate (%)
40000	0.351	0.315
45000	0.312	0.302
50000	0.285	0.291
55000	0.271	0.274
60000	0.263	0.264

Table 2: Out-of-vocabulary rate on CTS and ICSI meetings tasks for various wordlist sizes.

is configured to give general American English pronunciations with British English spelling conventions using a reduced ARPABET phoneset of 44 phones plus silence. The final baseline dictionary comprises a total of approximately 115,000 pronunciations.

Pronunciations from UNISYN have been supplemented by the addition of manually checked pronunciations to produce custom dictionaries of 8100 and 2500 words respectively for use with the switchboard and ICSI corpora. A procedure for generating new pronunciations has been developed as follows:

- Generation of an Out-Of-Vocabulary (OOV) word list from the text-normalised word list
- Automatic generation of pronunciation hypotheses using a decision tree based letter-to-sound system trained on the baseline dictionary (achieving 80% word accuracy and 96% phoneme accuracy)
- Automatic generation of pronunciation hypotheses for partwords
- Cross-checking pronunciation hypotheses with other dictionary resources
- Manual correction of the pronunciation hypothesis
- Newly added pronunciations are added to existing dictionary resources to aid in later dictionary development

The resources used for dictionary development have been collected on Sheffield's host system complete with documentation.

2.2.3 Language modelling

Creation of language models requires a significant quantity of text that has a similar style to the text in the test domain. Language models optimised for a specific domain are created by interpolating many different models where each one is created from a different text corpus.

Various text corpora were collected in order to create language models:

1. Switchboard: includes Switchboard, CallHome and Switchboard Cellular data.
2. HUB4 LM96: a broadcast news corpus.
3. ICSI meetings: the meetings data collected at ICSI.
4. Web data: data collected at the University of Washington by searching the Internet for text that is similar to a given corpus.
5. M4: meetings data collected on the M4 project at IDIAP.
6. BBC: data collected from a variety of broadcasts by the BBC including news and documentaries.

Text corpus	Number of words
Switchboard	3,494,406
HUB4 LM96	151,846,263
ICSI meetings	952,173
Web data (swb)	162,913,566
Web data (fisher)	484,214,055
Web data (fisher topics)	156,322,948
Web data (meetings)	128,282,257
M4	30,949
BBC	33,049,016
Total	1,121,105,663

Table 3: The various text corpora used for creating Language models.

Table 3 lists the corpora that were used and the number of words in each after text normalisation was performed.

Text normalisation is a process in which a corpus is modified to yield optimal consistency. By consistent, we mean that the same strategy for spelling, use of hyphenation and special symbols is used across all corpora. It was decided that the recogniser should be based on British English spellings (despite having US English pronunciations in the dictionary!). Thus 'colour' is not spelled 'color' and 'normalise' has an 's' not a 'z'. Unfortunately, there are a few exceptions that are not easily translated without referring to the context in which the words are used. Such words include 'meter' versus 'metre' and 'check' versus 'cheque'. Where this was the case then the US spelling was used instead. Other normalisation considerations include the correct conversion of digits (including dates, times, currency and numerical values) to a spoken form, the correct expansion of abbreviations such as Mrs. to Missus and whether to expand acronyms (e.g. 'UNICEF' is unchanged but 'FBI' becomes 'F. B. I.'). Normalisation of the "Web data" corpora was particularly important. This included, amongst other things, the removal of HTML tags and a frequency based normalisation scheme in which the most frequent erroneous OOVs were manually corrected.

CTS language models were optimised for perplexity and tests were performed on the three CTS test sets. Tables 4, 5 and 6 show the perplexity results of the CTS optimised bigram, trigram and four-gram language models respectively. There is a significant reduction in the perplexity when incorporating the Switchboard and Fisher Web data. Little is gained from incorporating meetings data as it is a completely different domain.

Experiments were conducted to determine the effect of discarding infrequent trigrams and four-grams to reduce model complexity. Testing on the CTS test sets several optimised language models were constructed by interpolating with Switchboard, HUB4 LM96 and ICSI meetings language models in which no n-grams were discarded. The result is shown in table 7. There is a relatively small increase in the perplexity of the final optimised model while there is a significant reduction in its size.

2.2.4 Acoustic modelling and Decoding

Acoustic model training used the Cambridge University h5train03 training set which covers approximately 300 hours of speech. The acoustic training data is encoded using the HTK implementation of perceptual linear prediction coefficients together with the 0th cepstral coefficient. In total 13 coefficients plus first and second order derivatives were used. Further the data is normalised using cepstral mean and variance normalisation on a conversation side basis. Initial monophone models are trained. These are used to initialise from-scratch training of crossword triphone models. These initial triphone models are used for further bootstrapping using 2-model re-estimation. After repeating this procedure several times the final

PPLs	Switchboard	HUB4 LM96	ICSI	Web (swb)	Web (fisher)	Web (fshtop)	Web (mtngs)	M4	BBC
104.53	1.000								
144.11		1.000							
236.19			1.000						
132.10				1.000					
132.85					1.000				
144.93						1.000			
175.47							1.000		
586.02								1.000	
273.18									1.000
95.00	0.757	0.243							
94.89	0.741	0.228	0.031						
91.87	0.673	0.062		0.266					
91.27	0.663	0.043		0.134	0.161				
90.89	0.659	0.027		0.084	0.093	0.138			
90.89	0.656	0.022	0.008	0.076	0.086	0.141	0.011		

Table 4: The perplexity results from interpolating the bigram models created from the various corpora and tested on the CTS test data. The interpolation weights are shown in the body of the table and the perplexity (PPL) in the left most column.

PPLs	Switchboard	HUB4 LM96	ICSI	Web (swb)	Web (fisher)	Web (fshtop)	Web (mtngs)	M4	BBC
85.97	1.000								
112.50		1.000							
228.57			1.000						
102.45				1.000					
102.36					1.000				
117.09						1.000			
143.56							1.000		
609.41								1.000	
239.65									1.000
72.55	0.676	0.324							
72.43	0.657	0.312	0.032						
68.34	0.569	0.102	0.017	0.312					
66.93	0.562	0.056	0.009	0.123	0.251				
66.75	0.562	0.049	0.008	0.092	0.186	0.103			
66.75	0.564	0.048	0.007	0.089	0.188	0.100	0.005		

Table 5: The perplexity results from interpolating the trigram models created from the various corpora and tested on the CTS test data. The interpolation weights are shown in the body of the table and the perplexity (PPL) in the left most column.

PPLs	Switchboard	HUB4 LM96	ICSI	Web (swb)	Web (fisher)	Web (fshtop)	Web (mtngs)	M4	BBC
84.12	1.000								
109.07		1.000							
235.67			1.000						
97.04				1.000					
95.42					1.000				
111.87						1.000			
137.58							1.000		
616.79								1.000	
231.23									1.000
69.04	0.651	0.349							
68.88	0.629	0.336	0.035						
63.83	0.529	0.117	0.020	0.334					
61.82	0.514	0.061	0.010	0.127	0.287				
61.58	0.514	0.053	0.009	0.098	0.229	0.096			
61.59	0.515	0.052	0.008	0.096	0.229	0.094	0.005		

Table 6: The perplexity results from interpolating the four-gram models created from the various corpora and tested on the CTS test data. The interpolation weights are shown in the body of the table and the perplexity (PPL) in the left most column.

min 3-grams	min 4-grams	model size	int PPL
4	4	4.2M 3-grams, 3.4M 4-grams	64.52
2	2	10.8M 3-grams, 11.4M 4-grams	63.83

Table 7: The effect on model size and perplexity of setting minimum counts on trigrams and four-grams. The minimum count must be reached before the n-gram is included in the model. The minimum count setting was applied to the Switchboard Web data and the model interpolated with the full Switchboard, HUB4 LM96 and ICSI meetings language models. A huge reduction in the size of the Switchboard Web data model is accompanied by a slight increase in the perplexity of the optimised model.

model set is obtained. The performance of this system (unadapted maximum likelihood trained models) is 36.7% on the 2001 NIST evaluation data. This result is comparable or better than the equivalent stages of the best system in that year. Additional acoustic modelling experiments are targeting the use of HLDA and semi-tied covariances. Furthermore decoding strategies are investigated: the use of DUcoder, a recogniser used in M4; and the scalability of HDecode. So far the use of DUcoder appears to yield considerably poorer performance. Experiments with HDecode explore the pruning parameters used. The aims were to firstly understand the effect that each parameter has on both accuracy and decoding speed, and secondly to determine a parameter set resulting in reasonably fast (1-2xRT) single-pass, trigram LM decoding with minimal loss in accuracy compared with a slower (>10xRT) configuration.

2.2.5 Adaptation

Two main techniques have been applied in the context of speaker adaptation: Vocal Tract Length Normalisation (VTLN) and Maximum Likelihood Linear Regression (MLLR). VTLN is a well known technique which is based on the fact that the spectral spread of the speech spectrum is in first approximation a linear function of the length of the acoustic tube.

The implementation used here is based on speaker dependent warping of the frequency axis of the estimated speech spectrum. VTLN is usually performed both in training and testing. The method adopted for training consists in iteratively alternating the estimation of warp factors with the re-estimation of model parameters. In particular each training step consists of single pass retraining followed by several iterations of Baum-Welch re-estimation. This procedure has been repeated until the set of warp factors for each conversation side has stabilised. Table 8 shows the behaviour of warp factor histogram for both female and male speakers. One can observe that while the warp factor distribution for female speakers (a, b, c, d) moves towards smaller values after each step, the reverse is the case for male speakers (e, f, g, h). In a final step models were trained from scratch, i.e. using the previously normalised features and inclusive the regeneration of phonetic decision trees. This model set constitutes the final VTLN model set.

For the use of VTLN in testing the following procedure was used:

1. Initial decoding using non-normalised features and models
2. Estimation of warp factors using VTLN models
3. Recomputation of normalised feature vectors
4. Decoding using the VTLN models

When using VTLN both in training and test a relative reduction of 10% in Word Error Rate was obtained. More details can be found in table 9 where WER for every training step has been reported.

Table 9 also shows results for speaker adaptation experiments using MLLR. Here one transform for speech and one for silence was estimated. Both mean and variance adaptation was performed.

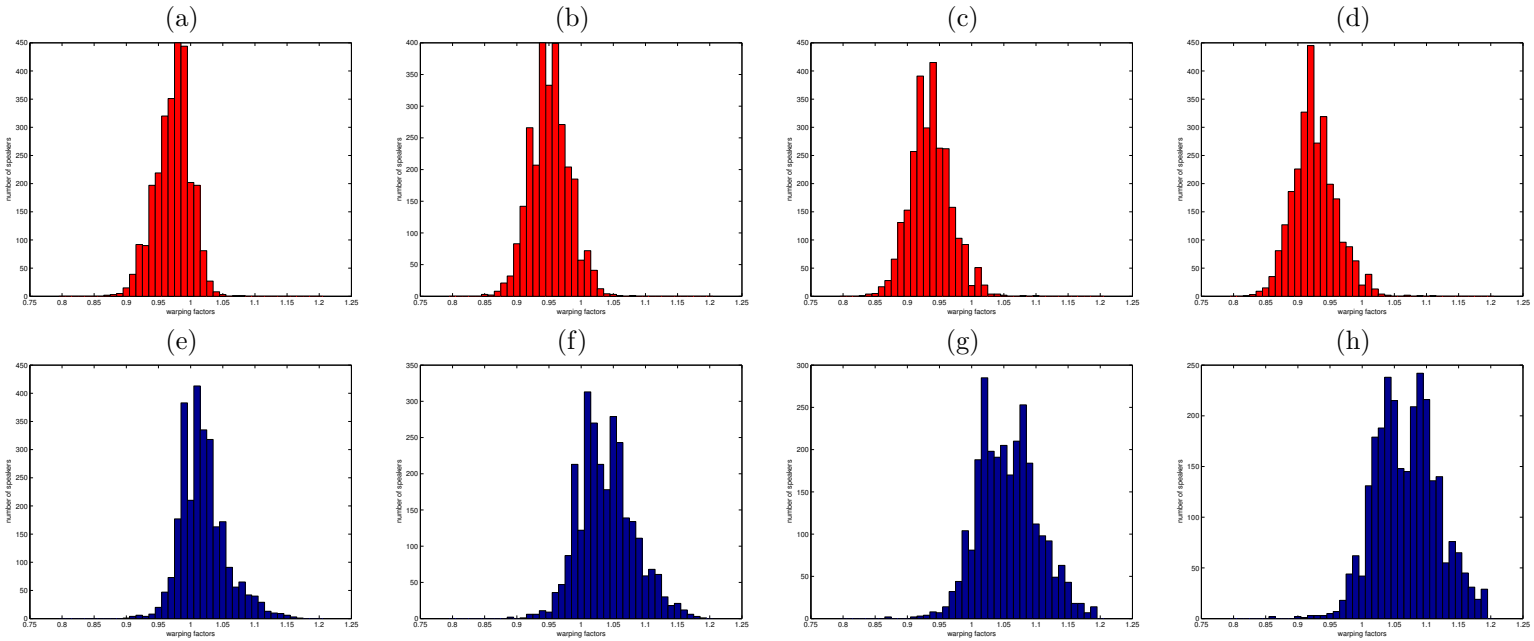

Table 8: Warping factors histograms estimated with non-normalised models (a and e), after the first (b and f), second (c and g) and fourth (d and h) step of VTLN training procedure for female (a,b,c,d) and male (e,f,g,h) speakers.

	TOT	Sub	Del	Ins	Sw1	S23	Cell	F	M
No adaptation	37.2	24.2	8.8	4.2	30.1	38.0	43.0	36.7	37.6
Test only VTLN	36.4	23.6	8.5	4.3	29.5	36.5	42.6	36.1	36.7
1^{st} pass training	35.7	22.9	8.9	3.8	29.1	35.4	42.2	35.0	36.4
2^{nd} pass training	35.0	22.5	8.8	3.7	28.5	34.6	41.4	34.2	35.8
3^{rd} pass training	34.5	22.0	8.7	3.7	27.7	34.2	40.9	33.6	35.3
4^{th} pass training	34.2	22.0	8.6	3.6	27.5	34.2	40.5	33.3	35.1
VTLN retrain	34.1	22.1	7.9	4.2	27.6	34.6	39.8	33.8	34.5
VTLN + MLLR	32.0	20.4	8.0	3.6	25.9	31.6	38.1	31.1	32.9

Table 9: Speaker adaptation results (% WER) for CTS task: the first line shows the baseline where no adaptation was performed, from the 2^{nd} row to the 6^{th} VTLN results for the iterative procedure have been reported, the 7^{th} line shows results with the same testing technique but after training from scratch, last line contains overall performances measured using both the adaptation techniques

In the context of CTS experiments this technique has been applied adapting our best VTLN models and estimating global transforms using the rough transcription given by VTLN testing. Note that the overall optimal performance is 32.0% absolute on the 2001 NIST Hub5E evaluation set.

2.3 Meeting Transcription

In this section the steps in development of a system for the transcription of the ICSI meeting data is described. Naturally the system development builds on work on CTS, hence the focus of the description is set on differences to CTS.

The following experiments focus on reporting results on the ICSI corpus part of the development and evaluation sets of the 2004 NIST RT meeting evaluations. However, as these data sets are small 2 additional test sets have been defined, one 7 hour test set (amieval-full) and one 3 hour test set taking about half an hour out of each meeting in amieval-full (amieval).

2.3.1 Audio Preprocessing

The main task has been to carry out speech/silence segmentation of the meeting data, initially just focusing on the independent headset microphone condition on the ICSI corpus. Two different approaches were explored:

1. **Simple frame energy-based technique:** In this approach each frame is classified as speech/silence based on energy in single channel or sub-band energies of multiple channels. In both cases, the speech/silence decision for a frame is made by comparing the metric to a threshold. The threshold for the energy is based on a running mean and standard deviation of noise frame energies, while the multi-channel metric threshold is based on an assumed uniform distribution for the case of no speech. Then the classification results are smoothed using a simple state machine. This is similar to the speech/silence detection implemented in HTK.

2. **TRAPS-MLP classifier approach:** In this technique, a multi-layer perceptron (MLP) is trained on using half second long temporal vector (TRAP) of each critical band logarithmic spectral energy with two target classes (speech and silence). Then posterior probabilities of these MLPs can be combined in many ways: averaging, log-averaging entropy.

The speech/silence segmentations obtained from the above two approaches were evaluated on one meeting (BMR015) using a simple frame-based False-Alarm/False-Rejection (FA/FR) evaluation proto-

col. ROC curves for different approaches are shown in Figure 1. An equal error rate (ERR) of around

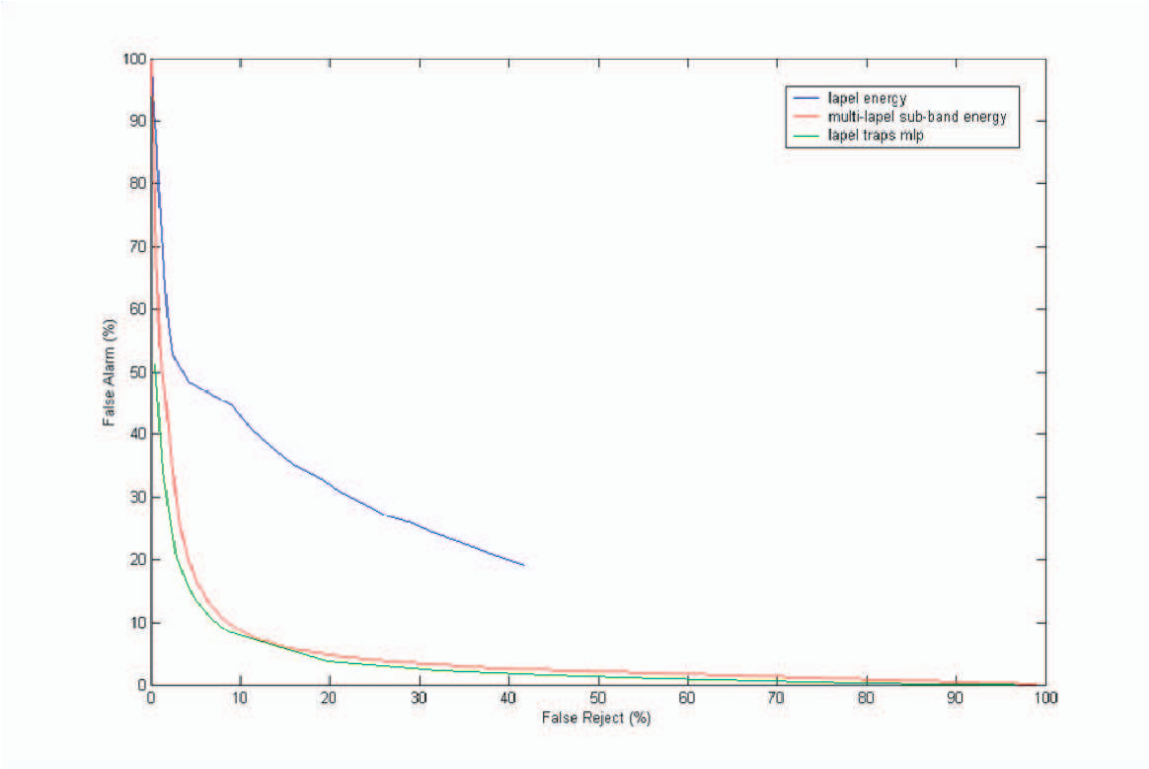

Figure 1: ROC plots for different segmentation approaches

9% (tested on a single meeting only) was obtained for both these approaches, although TRAPS-MLP classifier gave slightly better performance. The frame energy-based segmentation tools are available for AMI partners, and the MLP tools are currently being prepared for distribution.

MLP tools We developed set of tools for training and testing MLPs based upon TORCH3 package and TODE speech decoder. Currently, these tools support a single MLP training using 13 successive MFCC features (HTK format). Also two different unsepervised adaptations in training were implemented:

1. Adding a layer to input
2. Adding a layer to output

Taking the frame based classification output, chunking software has been developed to output smoothed segments suitable for input to the recogniser, using a simple procedure to enforce a minimum silence duration between segments. The priliminary results for one test meeting (BMR015) are presented in Table 10.

2.3.2 Wordlist generation

A similar procedure to that described in section 2.2.1 was used for the ICSI meetings task. Again the set of complete words from the ICSI meetings corpus was augmented by words originating from the Hub4 corpus. Table 2 shows the OOV rate for various wordlist sizes on the ICSI meetings test sets. Again a wordlist of 50,000 words was chosen.

Segmentation System	*False-Rejection*	*False-Alarm*
Frame-energy technique	10.25	11.04
MLP with no adaptation	2.63	7.56
MLP with adaptation 1	1.70	10.40
MLP with adaptation 2	2.10	9.79

Table 10: False-Rejection and False-Alarm results for various segmentation approaches

2.3.3 Dictionary

The procedure detailed above for the CTS dictionary was followed to generate a new dictionary for the ICSI Meetings Corpus from an OOV list.

2.3.4 Language modelling

The same data sources as described in section 2.2.3 were used for the construction of language models for the ICSI data. The corresponding perplexity results optimised for the ICSI test data are shown in tables 11, 12 and 13.

2.3.5 Acoustic modelling and Decoding

Data preparation for the training of ICSI acoustic models involves training and test set selection, text normalisation of the original corpus transcriptions, conversion of the transcriptions to the formats required by the training tools, analysis of the out-of-vocabulary word occurrences and subsequent word list selection. Again, identical to CTS, the data is represented in the form of 12 MF-PLP coefficients together with the 0th cepstral coefficient and first and second order derivatives. As the CTS system is based on telephone data (4kHz bandwidth) and the ICSI meeting recordings have a bandwidth of 8kHz, two sets of PLP parameters are calculated - the first set of parameters are limited in bandwidth to be the same as telephone speech, while the second set uses the full 8kHz available bandwidth. The discussion below refers to these two parameter sets as "Narrow-band" (NB) and "Wide-band" (WB) respectively. In order to facilitate experiments using MAP adaptation of CTS models, experiments are conducted on both WB and NB data sets. Cepstral mean and variance normalisation is also performed on a per headset microphone channel basis.

Initial forced alignment of the training set is performed using CTS acoustic models. The objective here is to remove utterances with poor audio/transcription quality. The ICSI NB acoustic models trained on ICSI data alone are then obtained by bootstrapped training from scratch using 2-model re-estimation with the best CTS models. This is set in contrast to the use of MAP adaptation of CTS models, as described later in this section.

ICSI WB acoustic models are generated using single-pass retraining with the best ICSI NB acoustic models, followed by standard Baum-Welch re-estimation training with mixture splitting up to 16 mixtures. Similar to CTS and ICSI NB training, bootstrapped training from scratch with 2-model re-estimation is used to further refine these models.

Three test sets are chosen to evaluate the ICSI NB and WB acoustic models. The ICSI portion of both development test and evaluation sets from the recent NIST RT04s meeting transcription evaluations (20mins and 23mins respectively) are used to gauge performance compared to results obtained in those evaluations. In addition a more significant test set was constructed consisting of 3.5 hours of speech. These sets are called, RT04s dev, RT04s eval, and AMI ICSI eval respectively. Tables 14, 15, and 16 present results for each of these sets using the narrow-band models. The results for the wide-band representation are shown in the tables 17, 18, and 19. Note that the best results here are obtained using wide-band data.

PLPs	Switchboard	HUB4 LM96	ICSI	Web (swb)	Web (fisher)	Web (fshtop)	Web (mtngs)	M4	BBC
209.55	1.000								
267.25		1.000							
141.36			1.000						
240.19				1.000					
222.21					1.000				
273.67						1.000			
237.46							1.000		
1130.97								1.000	
551.78									1.000
127.61	0.526	0.027		0.016	0.020	0.003	0.408		
126.81	0.479	0.025		0.014	0.019	0.003	0.395	0.065	
106.37	0.227	0.015	0.538	0.013	0.014	0.005	0.189		
106.37	0.223	0.015	0.536	0.013	0.013	0.005	0.189	0.007	

Table 11: The perplexity results from interpolating the bigram models created from the various corpora and tested on the ICSI test data. The interpolation weights are shown in the body of the table and the perplexity (PPL) in the left most column.

PPLs	Switchboard	HUB4 LM96	ICSI	Web (swb)	Web (fisher)	Web (fshtop)	Web (mtngs)	M4	BBC
192.90	1.000								
229.32		1.000							
129.37			1.000						
198.53				1.000					
206.92					1.000				
259.50						1.000			
210.17							1.000		
1173.28								1.000	
529.28									1.000
100.43	0.454	0.061		0.043	0.072	0.005	0.364		
99.93	0.423	0.059		0.040	0.069	0.005	0.357	0.046	
84.41	0.211	0.041	0.453	0.038	0.057	0.007	0.192		
84.42	0.208	0.041	0.451	0.038	0.057	0.008	0.192	0.005	

Table 12: The perplexity results from interpolating the trigram models created from the various corpora and tested on the ICSI test data. The interpolation weights are shown in the body of the table and the perplexity (PPL) in the left most column.

PPLs	Switchboard	HUB4 LM96	ICSI	Web (swb)	Web (fisher)	Web (fshtop)	Web (mtngs)	M4	BBC
193.43	1.000								
229.76		1.000							
135.49			1.000						
196.25				1.000					
179.81					1.000				
257.45						1.000			
203.38							1.000		
1186.97								1.000	
485.51									1.000
96.15	0.428	0.067		0.055	0.082	0.009	0.358		
95.63	0.397	0.065		0.052	0.079	0.009	0.351	0.047	
81.78	0.217	0.049	0.400	0.048	0.067	0.010	0.209		
81.79	0.214	0.049	0.398	0.047	0.066	0.011	0.208	0.007	

Table 13: The perplexity results from interpolating the four-gram models created from the various corpora and tested on the ICSI test data. The interpolation weights are shown in the body of the table and the perplexity (PPL) in the left most column.

LM	*WER(%)*	*S*	*D*	*I*	*F*	*M*
BG	24.5	14.2	8.4	1.9	26.1	23.4
TG	21.3	12.2	7.3	1.7	22.9	20.1

Table 14: ICSI NB Models: RT04s dev results

LM	*WER(%)*	*S*	*D*	*I*	*F*	*M*
BG	31.4	19.2	9.7	2.5	31.3	31.5
TG	28.8	17.5	8.9	2.5	27.5	29.4

Table 15: ICSI NB Models: RT04s eval results

LM	*WER(%)*	*S*	*D*	*I*	*F*	*M*
BG	36.4	21.6	11.1	3.7	-	-
TG	33.8	19.6	10.3	3.8	-	-

Table 16: ICSI NB Models: AMI ICSI eval results

LM	*WER(%)*	*S*	*D*	*I*	*F*	*M*
BG	22.5	13.0	8.0	1.5	24.1	21.4
TG	19.9	11.6	6.7	1.6	21.6	18.6

Table 17: ICSI WB Models: RT04s dev results

LM	*WER(%)*	*S*	*D*	*I*	*F*	*M*
BG	29.2	18.5	8.2	2.4	26.8	30.3
TG	25.7	15.9	7.4	2.4	23.2	26.7

Table 18: ICSI WB Models: RT04s eval results

LM	*WER(%)*	*S*	*D*	*I*	*F*	*M*
BG	34.6	20.5	10.3	3.8	-	-
TG	32.2	18.6	9.8	3.9	-	-

Table 19: ICSI WB Models: AMI ICSI eval results

In contrast to stand-alone training on the ICSI corpus the use of iterative MAP adaptation of CTS models to the ICSI meeting domain was tested. So far the results are obtained using a considerably weaker language model. However, the tables give an indication of the performance to be expected with a setup comparable to the previous setup.

MAP adaptation of well trained CTS models to ICSI data was applied in three different ways.

- One iteration with small τ (controlling value) - common way.
- More iterations with higher τ value - more precise models from previous iteration are used as input of actual iteration.
- More iteration with small τ value - using a two model re-estimation approach. Here the models from the previous iteration are used for state level alignment and CTS models are used as input for adaptation.

The following tables show results on the rt04dev set. Similar performance gains have been observed on rt04eval. Table 20 shows the baseline performance using CTS models alone. Table 21 show results for MAP adaptation using different values for τ. In Table 22 it is shown that further iterations can yield further substantial improvements.

	TOT	Sub	Del	Ins	F	M
rt04dev	28.7	17.5	9.8	1.4	30.1	27.8

Table 20: Baseline results given by non adapted CTS models.

	TOT	Sub	Del	Ins	F	M
$\tau = .1$	28.1	16.9	9.7	1.5	28.5	27.7
$\tau = .7$	28.1	16.9	9.7	1.6	28.5	27.8
$\tau = .9$	28.0	16.9	9.7	1.5	28.5	27.7
$\tau = 10$	28.2	17.1	9.8	1.4	28.7	27.9
$\tau = 20$	28.5	17.2	9.9	1.4	29.3	28.0
$\tau = 30$	28.9	17.4	10.2	1.4	29.6	28.5

Table 21: Results using MAP on the rt04dev test set

	TOT	Sub	Del	Ins	F	M
$\tau = .7$	26.9	16.4	9.2	1.3	28.1	26.0
$\tau = .9$	26.9	16.4	9.2	1.3	28.1	26.0
$\tau = 20$	26.8	16.2	9.3	1.3	27.9	26.1
$\tau = 30$	26.9	16.1	9.3	1.4	27.7	26.3

Table 22: Results for iterative MAP on the rt04dev test set

2.4 Summary of Systems

In the previous sections we have outlined ASR systems we have developed and their performance, both for the automatic transcription of conversational telephone speech and for transcription of meeting data.

The main features of the CTS system are

- Maximum likelihood training on 300 hours
- Up to 4-gram language models trained on about 1GW (1000 million words).
- Speaker adaptation in the form of VTLN and MLLR
- Multi-pass system generating lattices
- A dictionary with pronunciations for 50000 words.

The main features of the ICSI system are

- WB coding
- Up to 4-gram language models trained on about 1GW (1000 million words).
- A single pass system generating lattices
- A dictionary with pronunciations for 50000 words.

Note that for each system we have developed the necessary software that allows straight-forward replication of the results.

We further have developed a system that allows the unbiased automatic transcription of the complete ICSI corpus using cross-validation. The output of this system are lattices that will be used in experiments in work-package 5.

The development of the AMI CTS system is almost complete. The best performance on the 6 hour NIST 2001 Hub5E evaluation set is 32.0% WER absolute. We can compare this number with the best performing system in the 2001 NIST Hub5E speech recognition evaluations: The system by Cambridge University gave a WER of 39.1% in the first pass (unadapted). The second pass (CU-P2) yielded 31.1%. In comparison to the AMI CTS system CU-P2 uses more sophisticated training and higher complexity language modelling. Hence we expect similar or better performance with these additional system features.

The AMI-ICSI system is still simple and many components are still to be ported from CTS to this system. However, initial experiments are encouraging: On the NIST RT04 development test set we obtain with an unadapted system a word error rate of 19.9%. This compares to 17.4% on the same test set, achieved by the best system in the NIST RT 2004 Meetings evaluations.

2.5 Additional ASR Activities

Although the ASR engine development described in Sec. 2.2 - 2.4 was the most significant contribution of the ASR subgroup, there were a number of other ASR-related activities in which consortium members were engaged.

2.5.1 ASR tool development

Additional HMM training tools are developed by Lukas Burget (Brno). SERest is a tool for embedded HMM training. New key features of SERest [14] include re-estimation of linear transformations (MLLT, LDA, HLDA) within the training process, and use of recognition networks for the training. Work is in progress on the discriminative MMI (Maximum mutual information) training.

SVite allows decoding using an arbitrary recognition network. Additional tools for compilation of HMM's, pronunciation dictionaries and language models into a single network in progress (based on AT&T tools) is in progress.

For the merging of recognition results of a set of recognizers, SRover tool was developed [15]. Unlike standard ROVER based uniquely on strings of words, SRover allows for time-mediated merging of decoder outputs.

2.5.2 ASR NIST evaluation system development

ICSI continued the development of a Meetings ASR system and participated in NIST's Spring 2004 Meetings Evaluation. ICSI was one of 4 sites to participate in the ASR portion of the evaluation. We had participated in the previous NIST Meetings evaluation (held in spring 2002) using a simple port of a Switchboard-trained ASR system based on SRI's DECIPHER engine. For our 2004 participation, taking advantage of the larger collection of Meetings data becoming available, we focused on strategies for adapting acoustic models, language models, and signal processing of the telephone-speech-trained system to the Meetings domain. To give some sense of the overall progress, table 23 shows word error rate (WER) on the 2002 eval data (which became the 2004 development data) using our 2002 vs. 2004 eval systems. The table gives results for personal mics and for tabletop mics. (The 2002 eval used hand-specified segments for the personal mics whereas the 2004 evaluation required automatic segmentation, so both numbers are provided for the 2004 system.)

personal mics	ALL	ICSI	CMU	LDC	NIST
2002 eval system (hand segs)	36.0	25.9	47.9	36.8	35.2
2004 eval system (hand segs)	30.3	17.4	43.0	34.0	27.5
2004 eval system (auto segs)	36.1	20.5	50.2	43.8	30.1
tabletop mics	ALL	ICSI	CMU	LDC	NIST
2002 eval system	61.6	53.6	64.5	69.7	61.6
2004 eval system	43.8	28.4	59.1	52.3	44.0

Table 23: Performance (% WER) on 2002 eval set

The actual performance of the 2004 system on the 2004 eval data is given in table 24. The system was a streamlined system that ran in under 5x real-time. The table also includes performance of a somewhat more elaborate contrast submission using a 20x recognition protocol. More details can be found in [100, 108, 60].

	personal mics	tabletop mics
2004 5x system	34.8	46.7
2004 20x system	32.7	44.5

Table 24: Performance (% WER) on 2004 eval set

3 Event Spotting

3.1 Objectives

Acoustic event (mainly keyword) spotting (KWS) in meetings has the following goals:

1. To find all occurrences of entered word in a meeting and sort them according to confidences (in real time). This will allow for Google-like browsing of meetings using acoustics.
2. To verify if a word really occurred in a particular meeting (return its confidence). This is linked to WP5 summarization work.

At the time of writing this report, the acoustic events are limited to keywords, so that we will speak about the keyword spotting (KWS) in the following text.

3.2 Approaches to KWS

The most common approach to KWS is based on LVCSR. The word-strings or lattices are searched for keywords, for the detection, their confidences are compared to a threshold. LVCSR-based however suffers when the searched keyword is not contained in the dictionary or in case it has very low weight in the language model. Both can happen quite often for "interesting" keywords such as proper names. The recognition is run only once.

The second approach is based on composing the model of a keyword when the keyword is entered. The recognizer can be run only after this model is built. In case the keyword is not contained in a dictionary, the user must provide its approximate phonetic form, or this can be automatically created (this is related to pronunciation modeling in LVCSR).

Running acoustic KWS on large databases is however very time-consuming, and even for a fast system (say 0.01xRT), the response to a query on for example 30 hours of speech data could take ∼18 minutes, which is unacceptable. Therefore we investigate approaches based on phoneme recognition and phoneme lattices, which stand between LVCSR and purely acoustic approach.

The KWS is evaluated using Figure-of-Merit (FOM), which is the average of correct detections per 1,2,...10 false alarms per hour. We can approximately interpret it as the accuracy of KWS provided that there are 5 false alarms per hour.

3.3 Data for tests

Attention was given to the definition of experimental data-sets for KWS on ICSI meeting corpus. Although in reality, rare words (such as Bayes, minimization, etc) will be searched, in order to perform statistical evaluations, we had to define a data-set with frequently occurring keywords (otherwise the FOM metric becomes unreliable). Attention was paid to the definition of as-fair-as-possible division of data into training/evaluation/test parts with non-overlapping speakers - it was actually necessary to work on speaker turns rather than whole meetings, as they contain many overlapping speakers. We have balanced the ratio of native/nonnative speakers, balanced the ratio of European/Asiatic speakers and moved speakers with small portion of speech or keywords to the training set. The division is the following:

- training (41.3h)
- development (18.72h)
- test (17.2h)

In the definition of keyword set, we have selected the most frequently occurring words but checked, that the phonetic form of a keyword is not a subset of another word nor of word transition. The percentage of such cases was evaluated for all candidates and words with high number of such cases were removed. The final list consists of 17 keywords: *actually, different, doing, first, interesting, little, meeting, people, probably, problem, question, something, stuff, system, talking, those, using.*

3.4 Acoustic-based KWS

In these tests, the acoustic models of keywords were created. The confidence of the keyword is computed as a difference of 2 likelihoods: positive one from the last state of keyword-model and negative one from background model (a phoneme-loop). The positive path is actually prepended with a phoneme-loop too, to allow for natural "starts" of keywords.

Two approaches were tested for the modeling:

- modeling by standard HMM-GMM models (marked by HMM-GMM in the table).
- estimation of phoneme posteriors by 3 neural nets, with split-context TRAP-based features at the input (see the beginning of the next section) – marked by PHN-NN.

Table 25 summarizes the results in terms of FOM on the test set. 'log-post' and 'new-post' stand for two different functions for pre-processing the posteriors at the output of the net before the Viterbi decoder, 'new-post' is a function composed of 2 exponentials expanding both the regions around probabilities 0 and 1.

system	FOM [%]
HMM-GMM, CI models, 10 hrs ICSI training	46.8
HMM-GMM, CD models, 10 hrs ICSI training	56.73
HMM-GMM, CD models, AMI-CTS, not adapted	56.01
PHN-NN, hidden layer 500, 10 hrs ICSI training, log-post	61.01
PHN-NN, hidden layer 500, 40 hrs ICSI training, log-post	61.44
PHN-NN, hidden layer 500, 10 hrs ICSI training, new-post	64.58

Table 25: Results in terms of FOM on the test set

3.5 Phoneme-lattice based KWS

This approach starts with reliable detection of phonemes. We are using TRAP-NN system with separate modeling of left- and right-context TRAPs by 2 neural nets, with a subsequent merging by a third net. This system [92] outperformed both CI-HMM and CD-HMM on TIMIT and ICSI, it's performance while trained on a subset of ICSI meeting corpus is 46.5% phoneme error rate.

Currently, these systems are being trained on bigger data-sets (full ICSI and AMI-CTS), the results will be evaluated and used for KWS.

Phoneme lattices are generated by converting phoneme-posteriors into quasi-features and running HTK decoder HVite. The parameters of lattice generation (branching factor and word-insertion penalty) are tuned using evaluation of lower-bound phoneme error rate on lattices. We have found that for PER 46.5%, the lower-bound can reach ∼15%, however, at the price of huge lattices. The choice of optimal parameters is currently in progress.

For the detection of keywords in lattices, we have developed a toolkit for keyword-spotting in phoneme lattices using string-set-to-lattice matching. Poor FOM's were obtained while running the KWS on

lattices with small branching factors and word-insertion penalties set to the same values as in phoneme-recognition (too many misses). We are investigating the following approaches to overcome this problem:

- taking into account multiple pronunciation variants of the keyword.
- "fuzzification" of lattices using phoneme confusion matrices (taking into account possible errors of the phoneme recognizer). Either the phonetic form of the keyword or the recognized phoneme lattice can be "fuzzified".
- insertion and deletions of phonemes are handled.
- working with bigger lattices (size and parameters being optimized by lower-bound PER).
- Viterbi-style search of the keyword in phoneme lattice.

The goal in KWS based on phoneme lattices is to be able to quickly search these lattice for candidates unseen by LVCSR, and re-score them using the acoustic approach.

3.6 Plans

Our goal in AMI is to come with a complete system for keyword spotting. We would like to combine LVCSR- and acoustic/phoneme-lattice-based approaches in the following way:

- for words present in LVCSR dictionary: perform KWS by searching word-lattices at the output of LVCSR, evaluate confidences of words.
- for OOV's and in cases the keyword is de-favorized by the LM: spot similar words in the output of LVCSR and process these places using phoneme-lattice/acoustic KWS.
- for acoustic segments with unknown word or phonetic transcription (spotting by example), use the time boundaries of user-defined segment to select a "fragment" in the phonetic lattice. Search this "fragment of lattice" in the lattices of all meetings to determine similar acoustic segments.

4 Person Segmentation, Clustering, and Identification

4.1 Objectives

Algorithms for face detection, face recognition, speaker recognition, person segmentation, and clustering will be assembled and transferred to the common platform. Fusion of audio- and visual methods will be carried out.

4.2 Person identification

TNO's existing speaker recognition system was enhanced on several points, and TNO participated with their system in the NIST 2004 Speaker Recognition Evaluation. Below is a description of the submitted systems. Details about the evaluation can be found the the evaluation plan [67].

4.2.1 Introduction

The TNO speaker recognition system submission to NIST SRE 2004 consist of two basic techniques: one based on Gaussian Mixture Model/Universal background model (GMM/UBM) and one based on Support Vector Machines. We have concentrated on the single speaker conditions. Most of our development time has been spent on the GMM/UBM system, for the required core test condition we have submitted 4 slightly different GMM/UBM systems. Our primary system is a per-test-condition fusion of all other TNO systems submitted for that cell. We have performed unsupervised adaption only for the GMM/UBM systems.

4.2.2 Speech activity detection and feature extraction

We have utilized a very basic speech activity detection. The total energy in each 16 ms frame was determined. Frames were labeled 'speech' if the energy was less than 30 dB under the maximum of the speech file. Specifically, no spectral weighting or time-adaptive detection criterion was applied. Initial experiments with more elaborate speech detection schemes led to lower development test performance.

For feature extraction, we have used two forms of Perceptual Linear Prediction, one without (PLP) and one with RASTA processing. We have also used Mel Frequency Cepstral Coefficients in one system. In table 26 we've summarized the feature extraction for the non-fused systems.

PLP and RASTA extraction have the following parameters

- 32 ms frame length
- 62.5 Hz frame rate (16 ms step)
- 12 PLP coefficients and log energy
- 13 Δ coefficients calculated as linear regression over 7 consecutive frames.

MFCC extraction has slightly different set-up

- 32 ms frame length
- 100 Hz frame rate (10 ms step)
- 23 log-spaced filter bands, transformed into 18 cepstral coefficients
- 18 Δ coefficients calculated as linear regression over 7 consecutive frames.

PLP and MFCC features were normalized using short-time based feature warping [73] (256 samples, approximately 4 seconds) using and extremely inefficient but arguably elegantly simple GNU/Octave implementation. RASTA features were normalized to zero mean unit variance over the whole (speech active) file.

#	Features	Technique	UBM speech	male/females	relevance
1	PLP+Δ, warp (26)	GMM/UBM(512)	NIST SRE 2001	74/100	16
2	PLP+Δ, warp (26)	GMM/UBM(1024)	Swithboard 2 p2	324/256	4
3	MFCC+Δ, warp (36)	GMM/UBM(512)	NIST SRE 2001	74/100	16
4	RASTA+Δ (26)	GMM/UBM(512)	NIST SRE 2001	74/100	16
5	PLP+Δ, warp (26)	SVM GLDS	NIST SRE 2001	74/100	–

Table 26: Feature extraction and training parameters for the various systems

4.2.3 GMM/UBM systems

In table 26 we have also included the different UBM training conditions that for the different GMM systems 1–4. For the GMM systems, diagonal covariance, gender-specific UBMs were computed using either the training material of all NIST SRE 2001 evaluation speakers (74 male, 100 female; systems 1, 3, 4) or some of the Switchboard 2 phase 2 database (one conversation of 256 females, 324 males). The Switchboard data was first echo-canceled using the ISIP echo canceler [1]. Based on these UBMs, speaker and T-norm impostor models (using the NIST SRE 2001 evaluation training data) were computed using maximum a posteriori (MAP) estimation, [86] adapting priors, means and variances. A relevance factor of 16 was used, except for system 2 for which we used a more 'aggressive' value of 4. All systems use 512 mixtures, except for system 2 which used 1024 mixtures. We generally found little to no improvement in development test performance by increasing the number of Gaussians. Gausians were initialized using k-means clustering algorithm with 10 iteration steps, and re-estimated by 4 EM iterations of the Expectation Maximization algorithm (system 2 used 5 iterations).

4.2.4 SVM system

The SVM system is based on the work from William Campbell. [16]. We adopted his Generalized Linear Discrimination Sequence kernel (earlier coined Naive A-Posteriori Sequence kernel) method. Specifically, we used the $N_f = 26$ feature-warped PLP+Δ coefficients that were used for systems 1 and 2, and expanded them to a higher dimensional space by calculating all monomials up to order 3, leading to $\sum_{i=n}^{3} \binom{N_f+n-1}{n} = 3653$ features per frame. A diagonal sums-of-squares matrix R was constructed from these expanded features.

An SVM speaker model was trained by averaging all expanded features over time, and scaling the average by the inverse square root of the R matrix diagonal elements corresponding to the individual expanded feature dimension. These average expanded features were targeted the value '1' in the SVM training procedure. As 'background' speakers we used all NIST SRE 2001 gender-specific speakers, where for each speaker the average expanded features were targeted -1. The matrix R was estimated on all background and the one target speaker, but we have observed no performance degrade if only the background speakers were used.

T-norm models were formed by taking the background speaker expanded feature training data, and sequentially changing one target from -1 to 1 in order to obtain one impostor model. Thus, for the '1side-1side' condition the male target speakers were trained with 74 examples -1 and one 1, while the T-norm models were trained with 73 examples -1 and one 1. For training conditions with more than one side we used one positive training example per side. We used IDIAP's 'SVMTorch' implementation [18] using the simple inner product as kernel, which led to relatively fast training and testing.

4.2.5 Unsupervised Adaptation

Unsupervised adaptation was only performed for GMM systems. We used a method introduced by Claude Barras [9], where we adapt a speaker model using a test fragment where the T-normalized test score is

bigger than a fixed threshold. We used a fixed threshold of 3, which is close to the minimum Decision Cost Function threshold setting. We adapted only means, using a relevance factor 1.5 times the one used for producing the original speaker model. For the test runs with adaptation, we used the adapted model for decisions on the next test segment. T-norm or UBM models were not changed.

We have tried several methods for unsupervised adaptation of the SVM system. The first attempt used a test fragment with sufficiently high score as an additional positive target example, and then retraining a new SVM speaker model. A second attempt averaged the expanded features of a test segment with the original training data, with several mixing strategies. However, none of these attempts led to improvement during the development testing.

4.2.6 Two speaker conditions

We have submitted a 2-speaker test condition for two 1-speaker training conditions (1side and 16sides). We used a method described as 'internal segmentation' in [28] for separating the potential target speaker from the other speaker in the test segments. First, the log-likelihood-ratio for each from in the test segment was determined. This time sequence was smoothed by convolution with a 100-point (1.6 second) boxcar filter. Then, the frame scores above the 80 % percentile of the distribution were selected as detected regions. The detection time sequence was further filtered using a 101 point median filter. The smoothed log-likelihood-ratios were averaged over frames indicated by the smoothed detector. This average was used as mean log likelihood score for further detection processing.

4.2.7 Detection decision

Detection decisions were based on NIST SRE 2001 development test data. Specifically, each submitted system had separate detection thresholds for male and female speakers, and for not-adapted and unsupervised adapted systems. All threshold lie in the range 2.6–3, however, based one zero mean unit standard deviation T-normalization using impostor model scores.

4.2.8 Implementation

All processing was carried out on GNU/Linux systems, mostly 2.8 GHz Intel Xeon processors. The main training and recognition tools were scripted in GNU/Octave, combined with bash, Perl and GNU/R scripts. SVM training and classification was carried out using compiled C++ code of the SVMTorch distribution. Feature extraction was performed using the `plp` tool from SoftSound's Abbot, the `rasta` tool from ICSI's SPRACHcore distribution, and `wave2mfcc` from CMU's CMUseg package.

The NIST evaluation recognition scripts processed the NDX files in the given order. Adaptation was included in the main run, thus benefitting from T-norm calculations necessary for the non-adapted test. T-norm statistics were gathered in memory during the evaluation run, while the index of maximum UBM posterior probabilities per frame for each test segment were saved temporarily in files during the evaluation run.

4.2.9 Results

In figure 2 the DET curves for the TNO systems are plotted.

4.3 Speaker segmentation and identification

We've been working on software that reads data from a flock of birds. This software is able to collect from four devices simultaneous frame samples at a rate of 50 Hz. The obtained head orientations are combined with speaker information obtained from manual transcriptions. Speaker overlap and silence is omitted. On this data we've performed some initial research dealing with average speaker turn length's, average

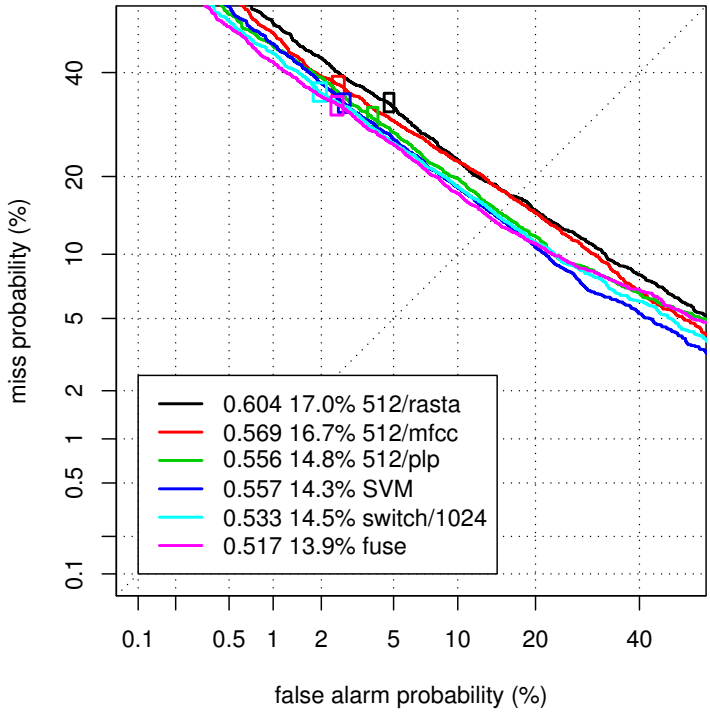

Figure 2: Detection Error Trade-off curves for the various TNO systems in the 1side-1side consition.

orientation when speaking and listening and more of this limited domain exploration. Recently we've setup a distributed experiment environment resulting in a system where we can conduct experiments at remote sites with one server sending out and collecting the samples. With this environment we intend to ask persons to evaluate the samples judging who is the current speaker. These results can be compared with machine learning algorithms on the same data.

- First results show that when given feedback people perform better over the meetings where feedback is or was given, if a new meeting is presented the performance seems to drop below the results from people who never received any feedback.
- It appears that in the middle of a speakerturn humans judge best in deriving the speaker from the head orientations alone.

4.4 Object labeling

An identification of type and person has to be evaluated for every detected object. This process is called object labeling. Resulting object type can be head or face and hand. Product of person identification is an assignment of selected object to relevant meeting participant. Object labeling has to be consistent during whole meeting independently on an activity of the participants. If each camera is evaluated separately a simple algorithm can be used to the object labeling. Known and unchangeable information about position of participant at the beginning of meeting uses this algorithm. Tracked image is divided into two same size parts with vertical line. On each side is sitting one participant and all objects belong to him. Than is supposed that the highest object is head of the participant. We can use the template matching for an improvement of this algorithm because it is possible to test if given object is really head of participant. Than two remaining lower objects are the hands. However this algorithm does not always work. For example if one participant leaves its place and walks through meeting room the identification of its objects can be lost. Other problems can occur if two identified objects are merged together or one already identified object is divided into two separate objects. In general labeling of objects has to be evaluated for all new detected objects and for all transformed objects, which are already identified.

4.4.1 Simultaneously evaluation of all cameras

More reliable solution of the object labeling is evaluation of all cameras simultaneously. A lot of additional information about a meeting room setup and participants can help during this computation. We use algorithm that is based on an elimination of impossible identification. At the beginning of meeting is evaluated labeling according to seat positions of participants using simple algorithm. A set of possible unused identifications in given time is evaluated for all unlabeled objects detected during the meeting. This set contains heads and hands, which are not assigned according to already identified objects on all cameras and possible number of participants. However other conditions given by meeting room setup have to be granted. For example it is clear that if one object on first camera is labeled as head of person B no object on second camera can be assigned to this person. Similar rules can be designed for relation between objects on first and third cameras or second and third camera. One of them for example says that if person is located on the left side of first camera it is impossible for this person to be on third camera at the same time. However if participant is located on the right side of first camera it can be assumed that unlabeled object on the left side of third camera belongs to the same participant. Some relations between objects on different cameras.

Set of possible identifications can be eliminated by other rules. There can be used known properties of human body as maximum distance between hands or head and hands of one person and also template matching can be used to discover if object contains face region. Resulting object identification is determined from eliminated set. If evaluated set contains only one member object is labeled by its identification. In other case when set contains more members with the same identification of participant

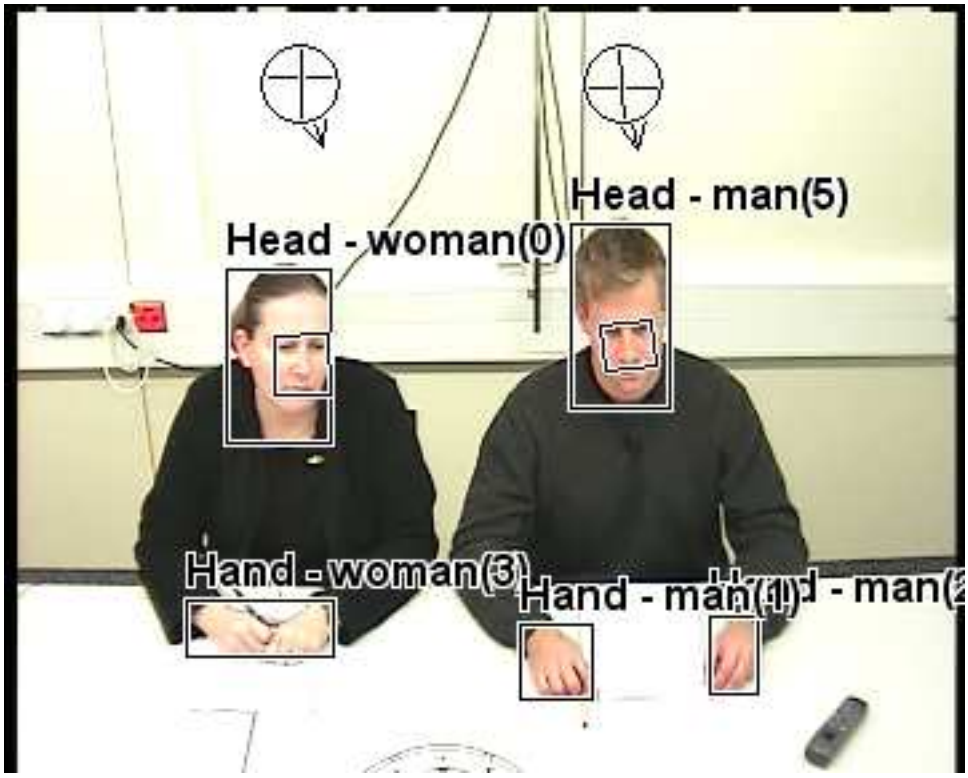

Figure 3: Looking direction

labeling can be also easy done. If the set contains several identifications with different person assignment other rules for detection of merged or divided objects and finding of lost objects can be applied. But it is possible that using of all possible rules for set elimination does not help and set of possible identification is still to big or empty and labeling cannot be completed. In fact used algorithm works in this way. The sets of possible identification are computed for unlabeled objects in given frame and eliminating rules are applied. If at least one object is successfully labeled and other object remain unlabeled in this step process of evaluation and elimination is repeated. This is done for all frames of the meeting from its beginning to the end.

4.4.2 Results

Function of described algorithms used to skin color object detection and its labeling was verified on video meeting corpus recorded in IDIAP. Some results obtained from several meeting are shown in table 27. Average number of objects with skin color occurred on all cameras during the meetings, number of detected objects and number of correctly labeled objects is shown.

Skin color objects	73155
Detected objects	66578
Detection effectivity [%]	91.0
Correctly labeled object	63587
Labeling effectivity [%]	68.9

Table 27: Experimental results

5 Emotion Recognition

5.1 Objectives

Systems for automatic emotion recognition, based on audio, video, and combined methods will be developed. Existing methods will be ported to the AMI domain and evaluated.

5.2 Introduction

The emotion subgroup of AMI-WP4 is concerned with the recognition of emotions or emotional content in meetings. For the development of emotion recognition tools, the annotation or labeling of emotional content in the AMI meeting data is of significant importance. This document will provide an overview of the specific conditions that emotion recognition in AMI is faced with as well as a discussion of various approaches to describe emotions by means of annotation of data and evaluation of recognition systems. Since the creation of AMI relevant data was not finished until the compilation of this document, recognition algorithms were developed and evaluated on various existing databases at the different partners. However, considerable progress has been made in robustness of methods based on speech and face analysis during the first period of AMI. The corresponding algorithms are introduced in section 5.5.

5.3 Emotion recognition in meeting scenarios

5.3.1 Hardware conditions

The AMI data corpus, comprises the recordings of 100 hours of meetings in specially configured meeting rooms. These so called Smart Meeting Rooms are equipped with wide-angle cameras to assure that each participant is captured by at least one of them. Furthermore close-up cameras are installed on the meeting table angled ahead of each person. Several audio streams can be recorded from lapel-micros and microphone arrays in the middle of the table.

Due to the hardware set-up mentioned, multimodal emotion recognition in AMI is limited to audio-visual modalities, i.e. mainly Speech Emotion Recognition, Facial Expression Recognition, and Gesture/Pose Recognition. The Speech Emotion Recognition can rely on high quality audio-recordings from lapel microphones of each participant. Disturbances by noises like moving chairs, clicking biros, or typing on keyboards are expected to have only minor impact. Also the cocktail-party effect of several persons speaking at the same time is assumed to be minimized by the application of lapel-micros.

On the other hand the conditions for automatic Facial Expression Recognition are more challenging. In the video streams, derived from the wide-angle cameras, faces are captured from an inadequate direction regarding the predominating orientation of the frontal plane of participant's faces. Furthermore the resolution is insufficient for applying relevant algorithms. Due to possibly frequent and extensive upper-body movements and wide head turns, the close-up cameras also provide challenging data. Though, the problems with large variation in scaling and rotation can be minimized by application of robust face localization and gaze tracking algorithms to take advantage of the high resolution captures of facial details.

5.3.2 The psychological point of view

Apart from raw hardware limitations in this real-world application of meeting scenarios, the psychological aspects turn the task of emotion recognition into an even more challenging light. Especially the limited number of participants usually unknown to each other and the formal atmosphere of common meetings have an significant impact on the behavior and therefore on the expression of emotions. Since participants finding themselves in a kind of exposed situation tend to suppress their affects or hide them behind acted emotions. Therefore most attendents' expressions of feelings can be expected to contain only colors

of emotional states or be acted even, which must be kept in mind by aiming at a set of emotions to be distinguished and the final evaluation results of recognizers compared to works based on different naturalistic data.

5.3.3 Dependencies on preceding recognition systems

Multimodal emotion recognition in AMI makes use of mainly three different information sources:

1. Prosodic properties of speech
2. Linguistic content of speech
3. Facial expressions

Tapping these sources requires results of preceding annotations or recognizers respectively:

- Speaker recognition: In order to establish user dependent models during a meeting information about the source in terms of the person speaking is necessary.
- Speech recognition: Basis of emotion recognition from the linguistic content are naturally the spoken words, transcribed by speech recognizers/annotators. Furthermore the temporal information is crucial to reliably detect tunes of speech by the length of pauses.
- Face localization and gaze recognition: Since algorithms for facial expression recognition have to be adapted on scaling, rotation and the direction of gaze these instances are to provide the essential information. Again the identity of the captured person is crucial to instantly adapt models hereby.

5.4 Adequate description of emotions in AMI

5.4.1 Discrete labels vs. continous dimensions

There is no general agreement on how to annotate or label emotional content in a natural database. A number of emotion annotation or labelling schemes have been proposed in the literature. Acted material can usually adequately be described using discrete category labels. Also for other kinds of material the categorical approach has been applied. Theorists in the discrete emotion theory tradition propose the existence of a small number of "basic" emotions, six for example [20], or seven [30]: anger, disgust, fear, joy, neutrality, sadness, surprise. However, given the gradations and subtlety of emotions occuring in natural data, the labelling of emotion using category labels is not straightforward and may result in emotional content being left unlabelled or labelled statistically unreliably [23, 38].

Instead of using discrete labels, the use of abstract dimensions is proposed. In the dimensional tradition, different emotional states are mapped in a two or sometimes three-dimensional space. The two-dimensional approach consists of a valence/evaluation dimension (positive/negative, pleasant/unpleasant, agreeable/disagreeable) and an activation/arousel dimension (active/passive) [93, 23, 21]. If used, a third dimension represents control or power.

5.4.2 Emotion annotation using *FEELTRACE*

At the Martigny workshop, the dimensional approach was demonstrated by Roddy Cowie from Belfast University. He suggested to use both the dimensional *FEELTRACE* annotation tool [21] and a set of category labels specifically chosen with respect to the (emotional content in the) AMI data set. Four members of the emotion subgroup attended the HUMAINE emotion summer-school in Belfast in September 2004. At the summer-school, among others some hands-on experience with the dimensional *FEELTRACE* tool was obtained. The AMI annotation 'problem' was discussed with HUMAINE emotion researchers.

Table 28: The 26 Emotions percieved in meetings

at ease	frustrated	agreeable	confident
bored	amused	contemplative	decisive
joking	relaxed	encouraging	impatient
annoyed	interested	sceptical	concerned
nervous	cheerful	friendly	serious
satisfied	uninterested	attentive	curious
	disappointed	confused	

After the HUMAINE summer-school, it was decided to use *FEELTRACE* as a baseline tool for emotion annotation in AMI and to setup a number of annotation trials to find out its appropriateness in the context of AMI. A survey was conducted to find *FEELTRACE* landmarks suitable for meetings (see next section). The following questions are asked in the first trial:

1. Dimensional approach: Is the 'dimensional annotation approach' suitable given the targets within AMI? Is there consistency in labeling (inter-annotator agreement)?
2. Categorical labeling: How does the additional categorical labeling task work out? Is the provided list of labels suitable?
3. Emotional content: What does a first exploration of emotional content in real AMI data tell us? What do annotators think?
4. *FEELTRACE* : What are the experiences of annotators using the *FEELTRACE* tool in the context of AMI data? Do they have suggestions for improvements?
5. Annotation manual: Are the instructions and training that are given to the annotators without any prior knowledge on emotion annotation sufficient (validation of annotation manual)?
6. Landmarks: Is there a noticable effect of using/not using landmarks on inter-annotator agreement?
7. Technical details: Are there any technical issues?

The first trial is planned to take place in December 2004 and Januari 2005 and uses the AMI pilot recordings.

5.4.3 Landmark survey

A user study was conducted to determine meetings specific emotion labels. A survey listing 243 terms describing emotions was compiled from the lists at the Queens University in Belfast, the University of Geneva and a few other sources. It requested each participant to select twenty emotions that they most frequently perceived in their meetings. It was completed by 37 participants from various companies and with various job descriptions, including lecturers, researchers, managers, secretaries and students. The 243 emotional labels were clustered by meaning into groups. The most frequently chosen one or two labels were shortlisted from each group. Taking some labels from each group ensures that there is sufficent coverage of the emotion space. Table 28 lists the 26 shortlisted emotions.

Studies on purely continuous dimensions have been performed by [21] using a tool called *FEELTRACE* (see figure 4). They show that individuals draw unequal semantic interpretation of specific areas within the emotion space. Such shortcomings are partly overcome when the tool is "landmarked" by category labels. That is, a small set of sample emotions are marked on the *FEELTRACE* tool at their most appropriate positions. Their tests indicated that landmarks result in greater inter-annotator agreement.

A second survey was conducted to determine where each of the shortlisted labels should appear on *FEELTRACE* . The survey first presented participants with the five labels: anger, irritation, sadness, happiness and contentment. These were presented so that participants unfamiliar with *FEELTRACE* would have minimal experience in its use. Then the emotions listed in table 28, were presented twice: the first to allow the participant to gain additional training and the second to collect data.

Data was collected from 33 participants. Eleven landmarks were eventually selected by analysing the data collected from both surveys taking into consideration the number of landmarks that would be useful to an annotator. Too many landmarks leads to a cluttered tool and a risk of annotators doing categorical labelling while the full benefit of landmarks is not exploited when there are too few. The *FEELTRACE* tool with the meeting specific landmark labels is shown in figure 4.

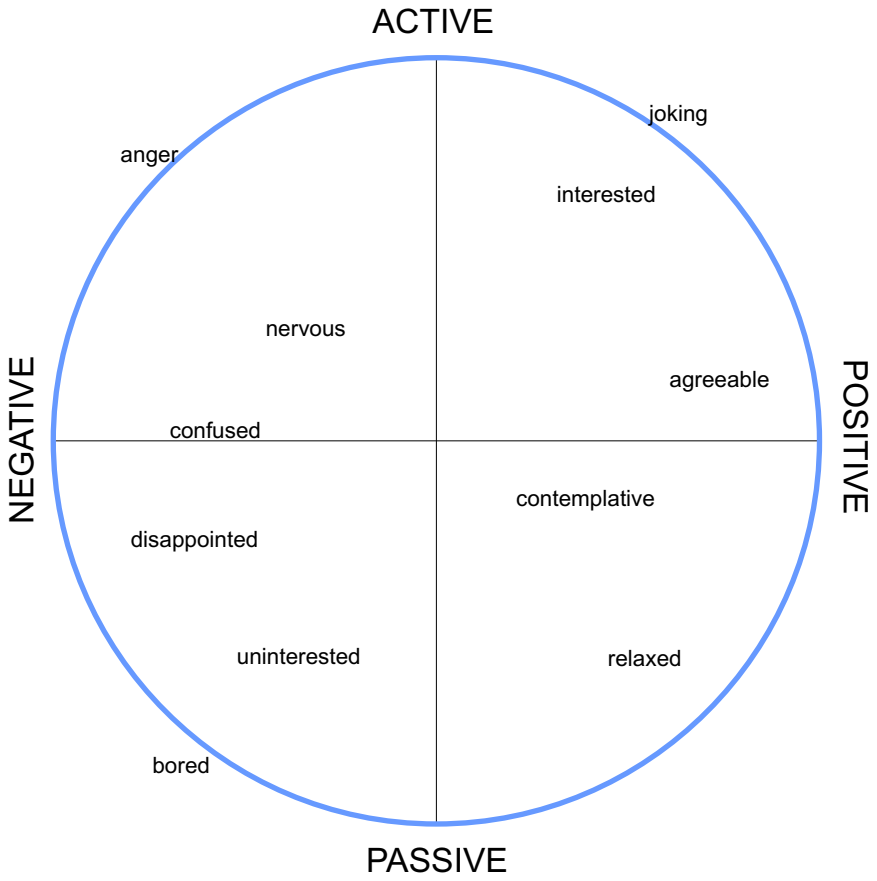

Figure 4: Proposed *FEELTRACE* landmarks

5.5 Algorithms for emotion recognition

5.5.1 Speech emotion recognition

Overview During the first year of AMI existing algorithms have been enhanced and ined to be prepared for the work on real AMI meeting data. In speech emotion recognition approaches towards a discriminative combination of acoustic features and language information for a robust automatic recognition of speakers'

affects were investigated. Throughout the work seven discrete emotional states are distinguished. Firstly we describe a model for the recognition of emotion by the acoustic properties. The derived features of the signal-, pitch-, energy-, and spectral contours are ranked via a sequential forward floating search, based on a nearest-mean classification as wrapper. Secondly an approach to emotion recognition by the spoken content is introduced applying Bayesian Network based spotting for emotional key-phrases. Finally the two information sources will be integrated in a soft decision fusion by using a Neural Net. The achieved gain will be evaluated and compared to common methods. Two emotional speech corpora used for training and evaluation are described in detail and the results achieved are discussed with respect to their considerable advance in automatic affect recognition.

Introduction Most of the advances to speech emotion recognition rely on acoustic characteristics of an emotional spoken utterance. However, in recent approaches more emphasis is also put on the spoken content itself [6], and the most reasonable advance seems to be the discriminative integration of acoustic and linguistic information. In the work presented we therefore strive to combine these two information sources in a most robust way. Firstly we aim to show an optimal acoustic feature set and classification method in a comparison, respecting high performance and speaker independence. Secondly we concentrate on the language information. While in other works the probability of an emotion is estimated by conditional probabilities of single words in an utterance we introduce an emotional phrase spotting algorithm based on Bayesian Networks. The idea behind this effort is to include the context of a whole utterance as negations of feelings and allow for a speaker's indication of the emotional extent. Consider on this the exemplary phrase: "*I do not feel too good at all*". The keyword *good* is neglected and furthermore *too* alludes the actual extent. After this discussion of acoustic and language based emotion recognition a novel approach to the fusion of these shall be presented. While the combination has yet been accomplished mostly in a late semantic fusion manner, we introduce a soft decision fusion saving available information for the final decision process. As still no unity about a general classification scheme for emotions in technical applications exists, and the use of discrete emotional user states is far spread among researchers in the field of automatic affect recognition, we consider the emotional states named in the MPEG4 standard here: anger, joy, disgust, fear, sadness, and surprise. This set is supplemented by a neutral state for a dissociation from a non-emotional state. In view of international comparability [32][39] we decided upon this set of seven emotions in our work. The estimation of an emotion shall respect a whole spoken utterance.

Emotional Speech Corpus The emotional speech corpus EMO-CAR has been collected in the framework of an internal project on integration of emotion in automotive user interfaces. A dynamic microphone was used in an acoustically isolated room to record the emotional utterances. German and English sentences of 13 speakers, twelve male, one female, were assembled. A first corpus consists of 2828 acted emotional samples used for the training and evaluation in the prosodic and linguistic analysis. The samples were recorded over a period of one year to avoid anticipation effects of the actors. While these acted emotions tend to form a reasonable basis for a first impression of the obtainable performance, the use of spontaneous emotions seems to offer more realistic results, especially in view of the spoken content. A second set consists of 700 selected utterances in automotive infotainment speech interaction dialogs recorded for the evaluation of the fusion. In the project disgust and sadness were of minor interest. Therefore these have been provoked in additional usability test-setups to ensure equal distribution among the emotions in the data set. To obtain a basis for comparison the speakers had to reclassify their own samples in a random order at the end of the test series. Table 29 shows their average performance. A rather marginal overall standard deviation among the human classifiers of 2.11% was observed. Thereby *ang* abbreviates anger, *dis* disgust, *fea* fear, *neu* neutral, *sad* sadness, and *sur* surprise.

Initially the raw contours of pitch and energy are calculated as they rely rather on broad classes of sounds. Spectral characteristics in general are known as dependent on phonemes, thus on the phonetic

Emotion	ang	dis	fea	joy	neu	sad	sur
Error, %	8.0	19.7	18.7	14.7	16.5	23.7	12.5

Table 29: Error rate at human reclassification, mean 16.3%

content of an utterance. Therefore, as the only spectral information, spectral energy below 250Hz and 650Hz is used. 20ms frames of the speech signal are analyzed every 10ms using a Hamming window function. The values of energy resemble the logarithmic mean energy within a frame. The pitch contour is computed by the Average Magnitude Difference Function (AMDF), which proves robust against noise but susceptible to dominant formants. A low-pass filtering, applying a symmetrical moving average filter of the filter-width of three, smooths the raw contours prior to the statistical analysis. In a next step higher level features are derived out of the contours, freed of their mean value and normalized to their standard deviation. As the optimal set of global static features is broadly discussed [33][39], we considered an initially large set of more than 200 features. Subsequently the emotional relevance of the features has been investigated via Sequential Forward Floating Search (SFFS) with a three-fold stratified cross-validation. Due to computational efforts, a linear nearest-mean classifier was used as wrapper. Evaluations showed that under the given amount of training data a final 33 dimensional feature-vector produced the best results.

Classification of acoustic features Various different methods have been taken into consideration for the classification on the acoustic layer. In a test-series the classifiers listed in Table 30 have been tested applying the large speech corpus. Two thirds have each been used for training, one third for testing in three cycles, i.e. three-fold cross validation. A speaker dependent (S DEP) training with only one speaker, and speaker independent (S IND) evaluation were considered. The mean error rates are shown in Table 30. Standard deviations reached from 0.01% to 0.03%.

Classifier	S IND Error, %	S DEP Error, %
kMeans	57.05	27.38
kNN	30.41	17.39
GMM	25.17	10.88
MLP	26.85	9.36
SVM	23.88	7.05
ML-SVM	18.71	9.05

Table 30: Comparison of the acoustic feature classifications

The table shows a predominance of Support Vector Machines (SVM). In the field of pattern recognition a great interest in SVMs can be observed recently. They tend to show a high generalization capability due to their structural risk minimization oriented training. Non-linear problems are solved by a transformation of the input feature vectors into a generally higher dimensional feature space by a mapping function where linear separation is possible. Maximum discrimination is obtained by an optimal placement of the separation plane between the border of two classes. The plane is spanned by the support vectors leading to a reduction of references. A number of approaches to solve multi-class problems exists. In this evaluation we show two different solutions. Once each class is trained in its own SVM against all other classes, and the decision is made for the class with the highest distance to the other classes. In a second advance Multi-Layer SVMs (ML-SVM) are introduced. Figure 5 shows the principle.

A layer-wise two class decision is repetitively made until only one class remains. The clustering of

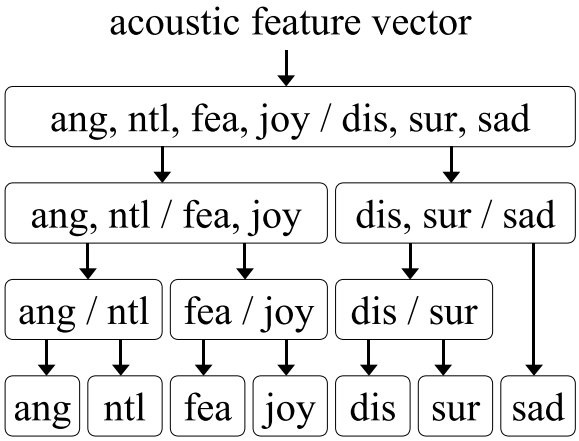

Figure 5: Optimal alignment of emotion classes using Multi-Layer-SVMs

the emotions and alignment on the layers significantly influences recognition performance. As a rule throughout the evaluation we found that hardly separable classes should be divided at last. This can either be modeled by expert knowledge or automatically extracted from the confusion matrices of the previously introduced SVM approach. A radial basis kernel as mapping function showed the best results. One disadvantage however is, that this method does not provide confidences for each class. This variant is therefore not used in the fusion.

Semantic analysis of speech In general only a small amount of utterances will consist of emotional content. Even if an utterance carries information about the current emotion of the speaker, this information will in most cases be only in fragments of the complete utterance. Therefore a spotting, based on searching for emotional keywords or phrases in natural language, seems a must. Once the emotionally relevant parts are identified, a probabilistic assignment to the corresponding affects has to take place to allow for the consideration of ambiguities in the emotional connotation of words and phrases. Therefore we chose Bayesian or Belief Networks as mathematical background for the modeling. All affects are modeled together within a single network converging from the word-level to seven root nodes, each representing an emotional state. Every relevant content observed within an utterance causes a true recognition probability in each root node, which is crucial for a discriminative fusion with the results from the acoustic analysis.

As basis for the analysis of the spoken content we apply a standard Hidden-Markov-Model based automatic speech recognition (ASR) engine without language model. Since affect recognition from semantics presumes natural language we designed the ASR-unit omitting a language model, but with a provision of n-best phrase hypotheses including single word confidences. An optimal performance of the entire semantic analysis system has been observed at n equals 5, i.e. the 5-best ASR-recognition hypotheses are processed separately. Let $\underline{h}_i$ be the 7-dim. vector of calculated affect probabilities caused by hypothesis i, for $1 < i < 5$. Our investigations showed, that the vector $\underline{h}_i$ showing the overall maximum entry should be marked as recognition result for the post-processing.

The spotting is performed with respect to the words contained in the bottom layer of the semantic model (Figure 6). In case that a word of the model is observed within an automatically transcribed utterance, the knowledge of uncertainty in the form of the ASR-recognition probability p_w can be transfered as soft-evidence P_w into the bottom-level nodes of the Bayesian Network. However, for normalization reasons a linear function performs adequate scaling and offset correction: $P_w = f_{lin}(p_w)$.

Within this deliverable we intend to provide only a brief insight in the theory of Bayesian Networks,

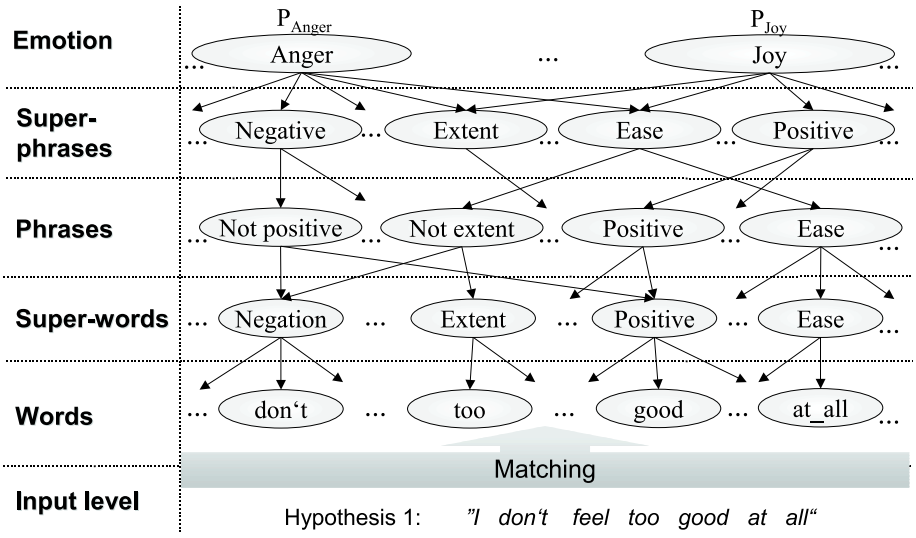

Figure 6: Overview of the interpretation model for emotions

which enjoy growing popularity in various scientific tasks. Each network consists of a set of nodes related to state variables X_i, representing a finite set of states. The nodes are connected by directed edges expressing quantitatively the conditional probabilities of nodes and their parent nodes. A complete representation of the network structure and conditional probabilities is provided by the joint probability distribution. Let N denote the total of random variables, and the distribution can be calculated as:

$$P(X_1, ..., X_N) = \prod_{i=1}^{N} P(X_i | parents(X_i))$$

Methods of interfering the states of some query variables based on observations regarding evidence variables are provided by the network. Similar to a standard approach to natural speech interpretation, the aim is to find the emotion hypothesis that maximizes the posterior probability of the word sequence given the acoustic observation. The root probabilities of the net are equally distributed in the initialization phase and resemble the priors of each emotion. On the other hand on the bottom layer the quantitative contribution $P(e_j|w)$ of any word w to the belief in an emotion e_j is calculated in a training phase by its frequency of occurrence under the observation of the affect on basis of the hand-labeled large emotional speech corpus. In four levels a clustering from words to super-words, phrases, super-phrases, and finally affects takes place as can be seen in Figure 6.

As mentioned, the inference calculus, based on the known Bayes' Rule performs the propagation of evidences at the word nodes up to the root nodes. Since this calculus is commutative, which means that the order of occurrence of evidences has no influence on the obtained results, a modification is needed to allow for the handling of word sequences during the clustering from the super-word level to the phrase level. Therefore we propose a further definition: The spatial arrangement of nodes within a layer corresponds to the order of expected evidence derived from the modeled sequence. The algorithm provokes a dependency of the strength of a evidence to the order of appearance. Thus this modification takes care that the observing of the exemplary simple phrase *not...good* will cause strong evidence within the corresponding phrase model, while *good...not* would not be considered.

The proposed method has previously been developed for the natural language controlling of an automotive infotainment system. Within this task it proved its remarkable robustness to speech recognition

errors and natural out-of-vocabulary commands. At affect recognition, based on the proposed emotion set of seven, a error rate of 40.4% could be achieved, while a hit presumed that the correct emotion was unambiguously detected with the maximum probability. Thereby 12% of the utterances of the hand-labeled corpus EMO-CAR did not contain emotional connotated phrases or words. As the section 5.5.1 will show, the proposed semantic interpretation algorithm provides valuable information to enhance the performance and robustness of the whole system significantly.

Discriminative integration of acoustic and semantic analysis In this chapter we aim to fuse the acoustic and linguistic information obtained. In other works the fusion is suggested as a late semantic logical OR combiner [57]. Since we strive to integrate information of more than two classes, a first approach might be to consider a couple-wise mean score for each emotion based on the acoustic and language information score followed by a maximum likelihood decision. As an advantage soft scores of both aspects are used in the computation prior to the final decision. However, this rather simple fusion neglects the fact that for each emotion the prior confidences in acoustical and language-based estimations differ. Furthermore a discriminative approach helps to integrate the knowledge of all accessible emotion confidences in one decision process. We therefore suggest the use of a Muli-Layer-Perceptron (MLP) for the fusion. The 14 dimensional input feature vector consists of the seven confidences of each, the acoustic, and linguistic analysis. Seven output neurons provide the final emotion probabilities by a softmax function. A use of 100 hidden-layer neurons showed the maximum performance. The MLP was trained on a second data set disjunctive of the initial training sets. For the evaluation of the combination a third data set was used. Table 31 shows results achieved using the EMO-CAR dialog corpus and optimal configurations. Thereby 12% of the utterances contained only acoustic information of the underlying emotion.

Model	Acoustic Information	Language Information	Fusion by means	Fusion by MLP
Error, %	25.8	40.4	16.9	8.0

Table 31: Performance gain means-based and MLP fusion

Summary Research activities on automatic speech emotion recognition made great progress during the first year of AMI. Especially a novel approach on the combination of acoustic and linguistic information as a solid model was introduced and a significant gain was achieved reducing error rates up to 8.0%. In the emotion estimation by acoustic information a set of features ranked via Sequential Forward Floating Search was specified. The use of SVMs predominated in robustness on this layer. Additionally a novel approach to linguistic information interpretation in view of a speaker's emotion using Bayesian Network based phrase-spotting with modifications for handling sequences of words could be shown. Finally the results of these analyzes were integrated in a reasonable and discriminative MLP soft decision fusion and lead to a significant improvement in overall performance.

6 Localization and Tracking

6.1 Objectives

Available audio-visual localization and tracking algorithms will be ported and combined to multimodal tracking procedures, combining e.g., motion features and acoustic sources obtained from microphone arrays.

6.2 Data annotation

A spoke audio-visual corpus for the localization and tracking tasks, called *AV16.3* was collected in the IDIAP meeting room [55]. 16.3 stands for 16 microphones and 3 cameras, recorded in a fully synchronized manner. The central idea was to use calibrated cameras to provide continuous 3-dimensional speaker location annotation for testing audio localization and tracking algorithms. Particular attention was given to overlapped speech, i.e. when several speakers are simultaneously speaking. Overlap is indeed an important issue in multi-party spontaneous speech, as found in meetings. We defined and recorded a series of scenarios so as to cover a variety of research areas, namely audio, video and audio-visual localization and tracking of people in a meeting room. In order to allow for such a broad range of research topics, we included a large variety of situations, from "meeting situations" where speakers are seated most of the time, to "motion situations" where speakers are moving most of the time. This departs from existing, related databases. The goal was to provide annotation both in terms of "true" 3-D speaker location in the microphone arrays' referent, and "true" 3-D head/face location in the image plane of each camera. Such annotation permits systematic evaluation of localization and tracking algorithms. To the best of our knowledge, there is no such audio-visual database publicly available.

While investigating for existing solutions for speaker location annotation, we found various solutions with devices to be worn by each person and a base device that locates each personal device. However, these solutions were either very costly and extremely performant (high precision and sampling rate, no tether between the base and the personal devices), or cheap but with poor precision and/or high constraints (e.g. personal devices tethered to the base). We therefore opted for using calibrated cameras for reconstructing 3-D speaker location.

The annotation requirements, scheme, tool, and procedure have been put in place. The definition of the specific annotation format for exchange among participants, and the annotation process are in progress.

6.3 An Architecture for Dedicated Real-Time Tracker Development and Management

6.3.1 Introduction

The work that TNO has done in this area gives a generic approach for thinking about and implementing visual object tracking systems. A concrete architecture is proposed that reflects this generic approach. Part of the work was testing this architecture for various tracking applications and to obtain hands-on experience with this architecture.

6.3.2 Applications

Within the AMI project, tracking systems can be applied in a few distinct cases and to various levels of integration. Overview cameras give the possibility to localize persons within the whole meeting room. Distinctions can be made for people entering and leaving the room, and for people moving to, from and at the presentation place. Additionally, persons, and parts of persons such as torsos, heads and hands can be localized through the close-up cameras. This application borders to gesture recognition, which

is described in section 7. These applications could also be combined to refine the localization result, by combining the input from all cameras (close-up and overview) to give an even better localization effort. The proposed architecture has a focus on performance. This enabled us to add features:

- Results are available immediately after the meeting.
- It allows for event detection during meetings.
- It becomes possible to control cameras directly and respond to events, e.g. zoom-in on details.

6.3.3 Terminology and Separation

When discussing and developing (near) real-time systems that track objects in video it is important to have clearly defined terms for various aspects of such a system. A clear distinction between various components and modules helps in coordinating the system resources. It is also essential for efficient re-use of these components and modules. Such terms are proposed here and used in the following paragraphs:

- The first term is "Scene". This refers to a portion of the real world that is observed through sensors such as cameras.
- The next term is "Object". This refers to a real-world object, such as a specific person, a chair, laptop or perhaps a fixed object, such as a wall, presentation screen or table.
- An "Image of an object" is the projection of the object through a sensor such as a camera. This image is probably different for each new frame coming from a camera.
- And finally a "Template" of an object is an internal representation of the object that a system uses to find the image of the object in new frames.

These terms describe various ways to look at data and the ways pieces of data relate to each other. More terms can be defined that describe the life cycle and activities of a small piece of the system that tries to follow one object or image through the various video frames. This piece is referred to as a "Tracker". Two types of trackers are defined: "Object Trackers" and "Template Trackers". An object tracker tracks objects through scenes or images. A template tracker tracks templates through images. This separation is especially helpful when using multiple cameras on one scene, or when multiple pieces of one object -such as hands, arms, legs, head and torso for humans- are tracked separately and must be combined with special logic to form a complete object.

The life cycle of a tracker starts with the decision to start a new tracker. A "Tracker Factory" makes this decision. This decision can depend on human input, motion detection on the video frames or input from other sensors. After an initialisation phase for a tracker, the tracker is said to be "Alive". During the life of a tracker it builds up a "Track" of the template or object that it is following. This track is typically stored in a series of coordinates and sizes. One could argue that shape should also be stored. Not all tracking systems have to be able to handle occlusions, so these are not part of the collection of most generic terms. Support for occlusion handling is however provided on more specific levels of the proposed architecture. It is time to end the lifetime of a tracker when this tracker has decided that it has "Lost" the image or object that it was following. The tracker becomes a "Dead" tracker. It is important to not just remove all data about this tracker when it becomes a dead tracker, since the very fact that it lost it's object or image could be very important to the application at hand. It was found that these terms could be applied to all tracking systems that we worked on. It was very helpful in the design process of such systems to keep a distinct separation between applications of the terms.

6.3.4 Architecture

The proposed architecture implements the distinctions described in the previous paragraphs. An implementation was made in C++ that relies on the object oriented inheritance model to define the various modules and to extend functionality of modules when needed for specific paradigms or applications. To test this architecture an implementation was made on MS DirectShow. As described in Figure 7 the General Tracking Architecture can be split in two parts. The Basic Tracking Library implements the terminology described above. It provides only in the most basic functionality. The Functional Tracking Library implements all sorts of helpful or more specific additions, like occlusion handling, template resizing, coordinate system transformations, motion detection, etc.

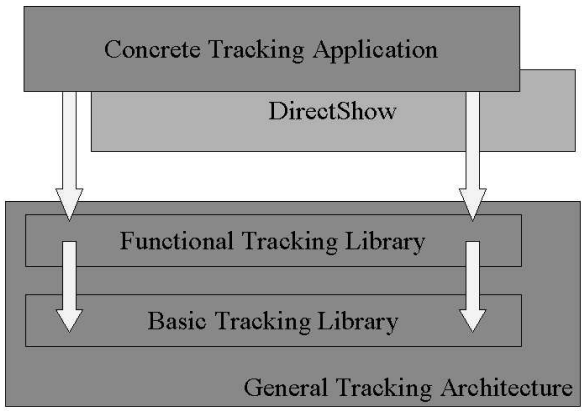

Figure 7: General Tracking Architecture

By using this architecture we were able to quickly implement and test features like:

- Merging of template tracks based on their history.
- Handling of occlusion on tracker level and above tracker level.
- Using different coordinate systems than the flat image coordinate system.
- Coordinating different types of trackers in one run.

An example Object-Oriented model for a small collection of tracker factories is described in Figure 8. In this figure the split between the collection of motion-triggered tracker creators and a tracker creator that uses custom parameters, like human input, to create a new tracker. For the class of motion triggered tracker factories, there also exists a factory that creates salient point trackers.

Future work on this architecture includes:

- Handling of multiple camera inputs of one scene
- Implementation on other video decoding platforms (i.e. Gstreamer)
- Testing the architecture and extending it's supporting features by trying to implement different types of trackers.

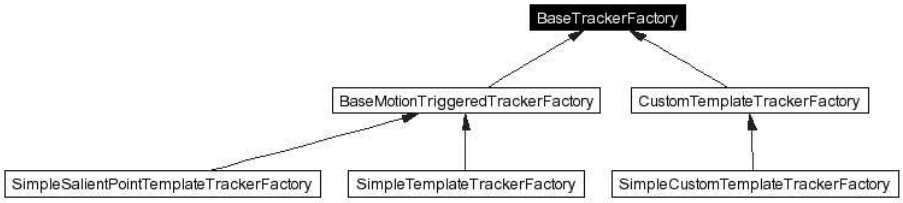

Figure 8: Example object-oriented model

6.4 Visual Localization and Tracking

6.4.1 ICondensation based visual Tracking

Localizing and tracking people is a very relevant task for smart meeting room applications, but is on the other hand quite challenging. Our module basically uses one of the characteristic properties of the head - the elliptical structure - to derive the position of any human in the meeting room. This technique enables tracking of not only frontal faces - as much of the other state-of-the-art approaches do -, but also of profile or even back views of a person. The architecture of our idea is mainly based on a stochastic particle filtering framework called ICondensation [40, 41], which provides a number of hypotheses for the position of a person.

Single Person Tracking At the beginning of the tracking procedure, particles will be initialized on skin colored regions. For finding areas with skin colored information, the RGB values are transformed into the rg-chroma space. In this plane skin color can be described by a Gaussian Mixture Model and thus a probability for each pixel to be skin colored can be computed by

$$p(skin) \propto \exp[-\frac{1}{2}(\begin{pmatrix} r \\ g \end{pmatrix} - \mu)^T C^{-1} (\begin{pmatrix} r \\ g \end{pmatrix} - \mu)]$$

Thus a binary mask like the one depicted in Fig. 9 will be created, where each pixel with $p(skin)$ below

(a) **(b)**

Figure 9: Binary mask representing only skin-colored areas in the original image

a certain threshold is set to zero and otherwise to one, i.e. blobs will be indicating skin colored areas by white regions. After that particles - also called hypotheses - are generated containing the parameters (position, length of the major axis, ratio major to minor axis, angle) of an ellipse which best fits one of the skin colored areas. The basic principle of the following procedure is depicted in Figure 10. From the initial particle set a certain number of samples is randomly drawn respective to its weight (probability for occurrence) in the actual image. In this way some of the particles will be chosen several times, while

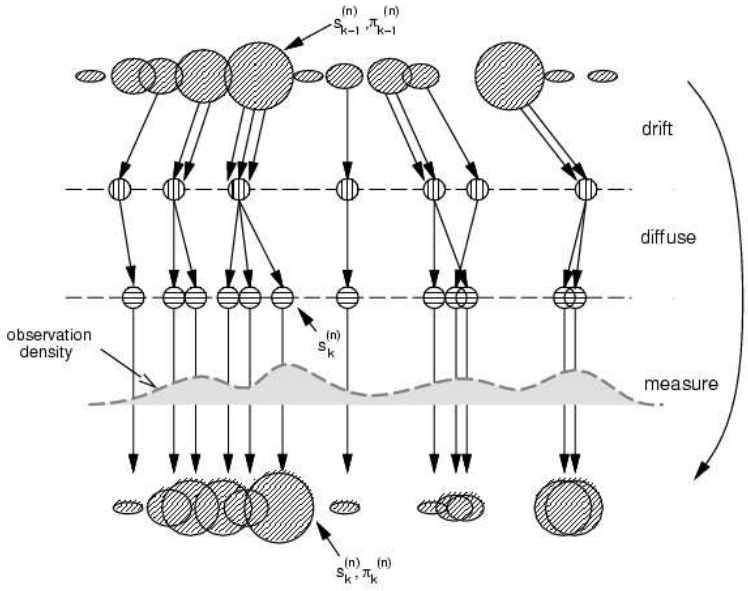

Figure 10: Basic procedure of the Condensation algorithm

others with relatively low weights will not be chosen at all. The sampled particles will be predicted now with noisy linear dynamics, named as drift and diffusion process in the Figure. In this way a new sample set has been generated, which has to be finally updated by measuring the observation density (weight) at each particle position. For this weighting step the gradient image and normal vectors at the sample positions of the ellipses have to be calculated (cf. Figure 11). Now the dot products between the gradient and the unit normal vector along the normal vectors (green lines) depicted in Figure 11 is computed. Thus the weight can be obtained by summing up over all maximum values of the dot products per sample point, normalized by the number of sample points of the ellipse. This procedure is repeated for all hypotheses and the observation probability is approximated in this way using the discrete weights.

Multiple person tracking For multiple person tracking this approach was further extended by some "super-particles" indicating a guess for the number of persons visible in the image. Similar to the technique explained in the paragraph above, these super-particles are also cycling through the drift-diffusion-measurement iteration, but now the skin colored blobs, we have already extracted, are used for the measurement. A combination of the configuration coverage (skin colored area covered by particles relative to the skin colored area) and the configuration compactness (skin colored area covered by particles relative to the area covered by particles) serves as basic indicator for the number of persons in the image. With these guesses, particles containing the elliptical properties are initialized on the different locations we obtained by the super-particles and thus are enabled to perform head tracking as proposed in the last section.

Results In the sequence in Figure 12 a typical scene has been taken from a meeting video with images taken at intervals of approximately 2 seconds. Two persons are sitting behind a table and our tracker now initializes fully automated and keeps tracking the heads. As depicted there appear sometimes also particles on the hands of the persons (light blue or violet) but the estimation keeps on the two heads (indicated by the green ellipses).

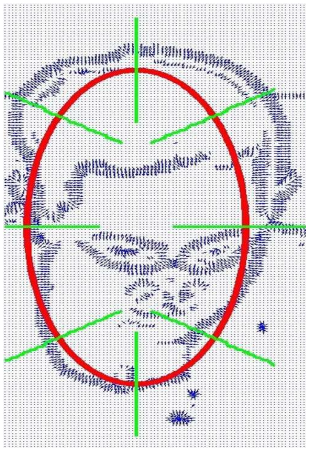

Figure 11: Gradient image, in which edges are represented by small blue arrows. Furthermore one hypothesis (red ellipse) has been plotted marked additionally with the normal vectors (green lines)

Figure 12: Office scene with partial occlusion caused by another skin colored object

6.4.2 Distributed Partitioned Sampling

We proposed a multi-object visual tracker using particle filters (PFs) [95]. We first define a joint multi-object state space, which constitutes a rigorous implementation of the problem. The state contains the configuration for every person in the scene, where a single-person configuration contains translation and scaling parameters for each participant.

Tracking a significant number of objects in a joint-object framework becomes increasingly difficult as adding new objects to the scene increases the search space exponentially. A sampling strategy known as Partitioned Sampling (PS) helps reduce the dimensionality problem by handling one object at a time,

but introduces problems with bias and impoverishment of the particle representation, dependent on the object ordering. We propose sampling using Distributed Partitioned Sampling (DPS), which redefines the distribution as a mixture model composed of subsets of particles, each of which performs PS in a different ordering [95]. In our approach, PS is performed using a different ordering for each subset to fairly distribute the bias and impoverishment effects between each object. The subsets are then reassembled and evaluated normally.

The observation model used in this work consisted of 8-bin color-space (HS) histograms with spatial components. The resulting multi-dimensional histogram consists of a concatenation of 2-D HS histograms, each built from pixels taken from different areas of the head (eyes, mouth, hair, etc) according to a template. The observation likelihood is defined as a product of single-object likelihoods, where the observation is the image region enclosed by the proposed single-object configuration, and each object likelihood is defined as an exponential distribution over the distance based on the Bhattacharyya coefficient between the observation and the specific object template histogram.

Head tracking experiments were conducted in the meeting room to test the ability of DPS to overcome impoverishment problems associated with PS. Specifically, DPS and PS were tested on meeting data for their ability to recover from occlusion (impoverishment hinders this ability) over 50 runs per method, to account for the stochastic nature of the tracker. Performance was measured by the *success rate* (SR), the percentage of successful runs (a successful run occurs when the tracking estimate overlaps the ground truth throughout the entire sequence). As reported in [95], DPS significantly outperformed both a simple multi-object PF and a PS tracker.

6.4.3 Template based face localization

We are using the Gabor Wavelet method for detecting significant parts in the face as area with eyes, nose and mouth. For each person is used the set of templates for different look directions. This method is also capable to recognize single person faces by comparison the template matches.

Figure 13: Face region detection

We have enough information for determine direction, angle and slope of the head. The slope in the side is computed as the slope angle of the inner rectangle covering eyes and mouth, which is parallel with vertical head axis. The look direction in vertical and horizontal direction is computed as ratio of left and right distances inner rectangle covering eyes and mouth and outer rectangle covering whole head. The additive information is from the type of the template. If it is used the template for side look, then is horizontal look direction defined constantly +90 or -90 degrees.

6.5 Multiple audio source detection and localization

Over the last year, methods were investigated to address the issue of both detecting and locating multiple speakers, that is persons speaking at the same time. This is indeed a necessity: in many real, multi-party speech situations people interrupt each other and talk at the same time (overlapped speech). It falls in the general "cocktail party" category.

Since spontaneous speech is very sporadic (short utterances separated by silences), the problem was attacked as an instantaneous decision to make. That means to divide a signal recorded with a microphone array into short time frames (16 ms), and to answer two questions for each frame separately: how many active speakers (it could be zero)? Where are they located? In existing literature, the detection and localization problem are usually considered separately, while here we need to consider them jointly. Moreover, many results are presented on simulated data.

Two types of approaches were developed: time-domain [54], and frequency-domain [50]. Both attempt to answer the two questions by first dividing the space around a microphone array into sectors (volumes of space), and decide for each sector whether or not it contains at least one active source. The long-term goal is to reiterate the approach in a coarse-to-fine way. Best results were obtained with the frequency-domain approach, which was extensively tested on more than 1 hour of real data, including both loudspeakers and humans, recorded by a circular, 8-microphone array.

To sum up, the proposed sector-based detection/localization approach proved its ability to detect and locate up to 3 simultaneous speakers on real meeting room recordings. Recent work successfully applied the same technique to the speech enhancement task in cars, in collaboration with Daimler-Chrysler [48].

6.6 Audio-visual localization and tracking

6.6.1 Introduction

The goal of this research is to produce a system capable of localising and tracking one or more speakers using both audio and video cues in a neurobiologically plausible manner. Specifically, we are interested in using binaural cues from a manikin in conjunction with visual cues to determine the spatial location of an individual speaker.

6.6.2 Audio cues

The acoustic inputs to each ear of the binaural manikin are sampled at 48 kHz and are processed by a model of the auditory periphery. The frequency selectivity of the basilar membrane is modelled by a bank of 64 gammatone filters [72] whose centre frequencies are spaced on the equivalent rectangular bandwidth (ERB) scale [37] between 50 Hz and 8 kHz. The auditory nerve response is approximated by half-wave rectifying and square root compressing the output of each filter [75].

Interaural time difference (ITD) is the main localisation cue used by the human auditory system [12] (see also [61] for a review). The conventional technique for estimating the lateralisation of a signal is by calculating a cross-correlation function using the left and right channels. This technique can be considered to be equivalent to the neural coincidence model of Jeffress [43].

Computing the cross-correlation for each channel gives a cross-correlogram, which is computed at 40 ms intervals resulting in a frame rate of 25 fps to match the video input. Since there may be small time differences between sounds reaching the two ears, channels dominated by a particular source will exhibit a peak at a correlation lag related to the physical azimuth of the source.

When the sound source dominates a number of frequency channels, a characteristic 'spine' can be observed at the source azimuth. For example, Fig. 14 shows the cross-correlogram for a 155 Hz complex tone which has been lateralised to the right by 45 degrees.

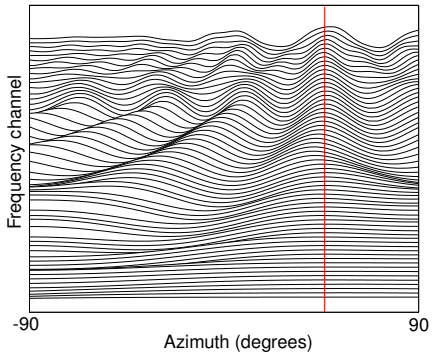

Figure 14: Cross-correlogram of a 155 Hz complex tone lateralised to the right by 45 degrees. The vertical line indicates the position of the 'spine'.

6.6.3 Video cues

Since the meetings are conducted in a relatively unchanging environment (i.e., the cameras are stationary and the lighting is consistent), a number of simple (and computationally efficient) techniques have been used.

Objects are detected by calculating the difference between the current frame and a reference frame (usually found at the beginning of a recording when the room is still empty) and motion is detected by calculating the difference between adjacent frames. These difference images are thresholded to produce binary masks. In order to produce a binary mask for face regions, we identify those pixels whose RGB values satisfy a given function [96]:

$$\begin{aligned} R > 95 \wedge G > 40 \wedge B > 20 \wedge \\ maxR, G, B - minR, G, B > 15 \wedge \\ |R - G| > 15 \wedge R > G \wedge R > B \end{aligned} \tag{1}$$

In each of the three masks, spurious pixels are discarded by using a region growth algorithm in which a pixel is only kept if its eight immediate neighbours are also 'on'. These candidate regions can still, however, be of any size and shape. To eliminate small regions, all groups whose area is less than a given figure are discarded (300 pixels for faces and 3000 pixels for other objects). An additional stage is included to produce the final face mask. To ensure only face-shaped (oval) regions remain, the length to breadth ratio is determined and used to discard non-oval regions.

6.6.4 Audio-visual integration

Two neural oscillator networks represent visual (2D network) and audio azimuth activity (1D network). The audio network has 181 nodes each representing an integer azimuth from -90 degrees to 90 degrees; the video network consists of a grid of 720x576 nodes in which each node represents a particular pixel of the binary input mask. The three video features are combined to produce a single binary input mask. If a face region is found to coincide with an object region then the face region is included in the final mask and the object region is discarded. All remaining regions are included in the input mask. Fig. 15 shows a schematic of the system.

Each network consists of an array of oscillators based upon LEGION [107]. Within LEGION, oscillators are synchronised by placing local excitatory links between them. Additionally, a global inhibitor

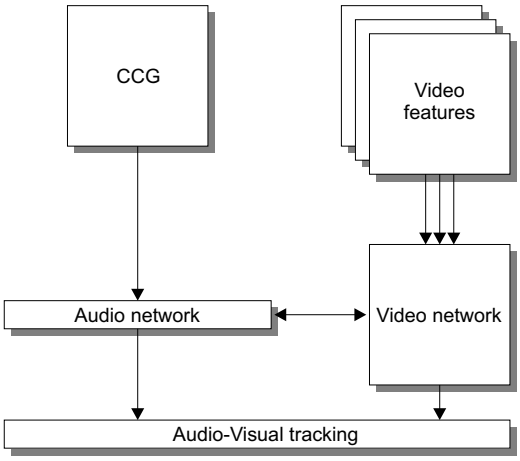

Figure 15: System schematic. The cross-correlogram (CCG) provides azimuth input to the audio network and a combination of the three video features provide input to the video network. The audio-visual object locations are then used as input to the inertia-based tracker.

receives excitation from each oscillator, and inhibits every oscillator in the network. This ensures that only one block of synchronised oscillators can be active at any one time. Hence, separate blocks of synchronised oscillators (segments) arise through the action of local excitation and global inhibition. Thus, within-network segmentation emerges as a property of network dynamics.

In order to fuse related audio and video activity, the two networks are linked by a number of weights (placed between azimuth nodes and video columns). These A-V mapping weights are generated using a two-stage process. The first stage uses a Hebbian learning rule during a training phase in which repeated, simultaneous video activity at column V and audio azimuth A strengthens the link between audio network node A and video network column V. However, since it is unlikely that the training phase will contain enough activity to generate weights for every possible audio-video pair, the second phase fits a sigmoidal function to the sparse A-V mapping data using the simplex search method [47].

Following a short period of time required for the networks to converge on a stable segmentation result, the individual A-V groupings can be determined. Any audio and video network activities which occur at the same time (their oscillators are synchronised) are said to be grouped (forming 'A-V objects'). Remaining audio or video activity which occurs independently is said to be ungrouped. Any A-V objects are candidates for object tracking.

6.6.5 Object tracking

Object tracking is implemented using an inertia-based system in which a leaky integrator models the velocity of an object. An inertia-based model ensures the tracking focus continues to move when position information of a moving tracked A-V object has been lost (possibly by visual occlusion or incomplete data). Provided the occlusion is brief and that the tracked A-V object continues at a steady velocity, the tracking focus will be close to the object when position information becomes available again.

During the lifetime of an object track, it is unlikely that audio localisation information will always be available: for example, binaural audio localisation is not robust (especially in reverberant environments) and speakers tend to make frequent pauses during speech. In this situation, the tracking algorithm 'backs off' to tracking the nearest video feature until audio information (and hence an A-V object) becomes available.

6.6.6 Conclusions

The system can extract video and audio features and successfully group video and audio activity when at the same position and segregate incongruous audio and video data (frame-based). A-V objects are tracked using an inertia-based mechanism so that sources can be tracked through brief visual occlusions. We are currently working on the tracking algorithm to allow objects to be tracked in complex multi-object environments. We will subsequently investigate the ability to track multiple objects simultaneously and incorporate psychophysically-motivated tracking competition behaviour. We are also investigating the enhancement of the audio azimuth estimation algorithm by using the visual motion estimates in particular frame regions to alter the degree of temporal integration at different azimuths.

6.6.7 Particle Filtering based audio-visual tracker

We proposed a multi-object audio-visual tracker using particle filters (PFs) [34]. We use an approach in which a person's head is represented by its silhouette in the image plane. The state-space is defined as a joint multi-object representation, where both the location and the speaking activity of each participant are tracked. We employ a mixed-state formulation, where in addition to continuous variables for head motion, a discrete variable is included to model the speaking status of each participant.

Our methodology exploits the complementary features of the AV modalities. Audio localization information in 3-D space is first estimated by an algorithm that reliably detects speaker changes with low latency, while maintaining good estimation accuracy. Audio, color, and shape information are jointly used in the observation likelihood. We also use an AV calibration procedure to relate audio estimates in 3-D and visual information in 2-D. The procedure uses easily generated training data, and does not require precise geometric calibration of cameras and microphones. We have dealt with the dimensionality of the multi-object state space by combining Markov Chain Monte Carlo (MCMC) and PF, which provides efficient sampling in a formalism that is naturally suitable for interaction modeling.

We have tested the method on a set of sequences from the IDIAP meeting room. After manual initialization, the four meeting participants can be simultaneously tracked, and their speaking status inferred at each time. An objective evaluation procedure involved the computation for each participant of tracking success rate, and the F-measures (which combines precision and recall) for location and speaking status, over 20 runs of the trackers. As reported in [34], the results show that our proposed approach outperforms a basic multi-object PF in both ability to track and estimation of the speaking status.

7 Gestures and Actions

7.1 Objectives

Several alternative systems for gesture and action recognition using both video only and combined audio-video input will be developed and ported to the AMI domain and evaluated.

7.2 Introduction

In the initial phase of the AMI project we focused on recognising gestures and actions which involve body and hand positions These include standing/sitting down, leaning forward/back and pointing. To facilitate algorithm evaluation TNO has contributed to the task of manually labelling ground-truth data for head positions.

Computer vision methods for analysing human behaviour have been divided into the following categories [35]; 2-D approaches without explicit shape models, 2-D approaches with explicit shape models and 3-D approaches Our research follows two complimentary approaches to gesture recognition which fall into the first of these categories.

1. Model-based gesture/action recognition using deformable templates.
2. Motion analysis using keypoints

The two approaches are described in more detail below. In the long term, these approaches will be combined in order to address the following problems; building shape models (semi-)automatically, adding anatomical context to interpret the motions of local image regions and making gesture/action classification more robust with classifier fusion techniques.

7.3 Model-based gesture/action recognition using deformable templates

Our approach matches shape information from image edges and/or segmented foreground objects to previously learned deformable templates. We have begun by using an Active Shape Model [19], or ASM, to describe body shapes. An ASM uses a set of labelled training images in order to build a statistical model of a shape which is described by a compact set of model parameters; an ASM can only generate shapes similar to those found in the training set. Previously, Baumberg and Hogg [10] have used a single ASM to track pedestrians in lower-resolution surveillance images and more recently Baker et al [8] have used Active Appearance Models (closely related to ASMs) to interpret facial gestures. To describe the full range of shape variations for the gestures and actions found in AMI data with a single shape model would require a complicated model with many, difficult to constrain, degrees of freedom. Instead we intend to take a modular approach in which multiple simple shape/appearance models compete to fit the data and a probabilistic framework is used to choose between the competing models. When different models correspond to different gestures and actions, making the right model choice corresponds to recognising a gesture/action. This approach has similarities to the hierachical template matching approach taken by Gavrila [36] for pedestrian detection but whereas Gavrila uses classes of similar discrete templates to facilitate matching, we will use classes of template model instances to enable classification.

Figure 16 shows an example in which an ASM tracks a subject who is leaning back in his chair. A leaning forward motion corresponds to the model 'losing lock' and signals the transition from one posture class to another. Work to efficiently build gesture/action specific models and implement a decision-making framework is in progress.

Figure 16: Example in which an ASM tracks a subject who is leaning back in his chair

7.4 Motion analysis using salient regions

Various algorithms exist for automatically detecting keypoints (salient points) in images and transforming local image regions around these keypoints to feature vectors. In recent years much interest has surrounded this approach to image processing and it has proved to be a powerful method of matching objects in a way that is robust to the effects of scale, rotation, occlusion, background clutter and changes in viewpoint. Keypoints-based methods have recently been applied to matching objects in video streams [94] and in this setting temporal information has been used to link information from moving objects and facilitate object-level matching. Until now, the potential of exploiting keypoints for gesture/behaviour analysis has not been explored. Our research focuses on exploiting the ability of keypoint matching to robustly match objects between frames in order to analyse human motion and recognise gestures and actions in AMI data.

Figure 17 shows an example where keypoints are used to track a persons hand. Note that a keypoints-based approach has advantages over standard template-based approaches because initialisation is handled implicitly as are several types of image transformation. This results in robust tracking and fewer tunable parameters.

7.5 Body pose estimation and action recognition

From a framework where video data and knowledge about the background is combined, silhouettes of persons can be extracted. Skin color information is used to find head and hands. An estimation of the joint angles of the body was obtained by fitting a body model to the silhouette. A first application was to estimate poses of a presenter. A 16 degree of freedom (DOF) human body model was used, see table 7.5. Since no ground truth about joint angles could be inferred from real video data, four synthetic movies have been made using Curious Labs Poser. Tests with a human body that matched the 3D character yielded an average joint angle estimation error below 10 degrees.

The second application was estimating body poses from the Scripted Meeting Recordings, recorded at the IDIAP smart room. A 10 DOF body model was used, as can be seen in the right column of table 7.5. A frame from one of these recordings, with the estimated body pose superimposed is shown in Figure 18. The estimated joint angles can be used to mimic the poses on an avatar, allowing for 3D evaluation of the estimation. Also, a virtual meeting room is constructed where the meetings can be replayed. Motion smoothing is applied to filter out noise.

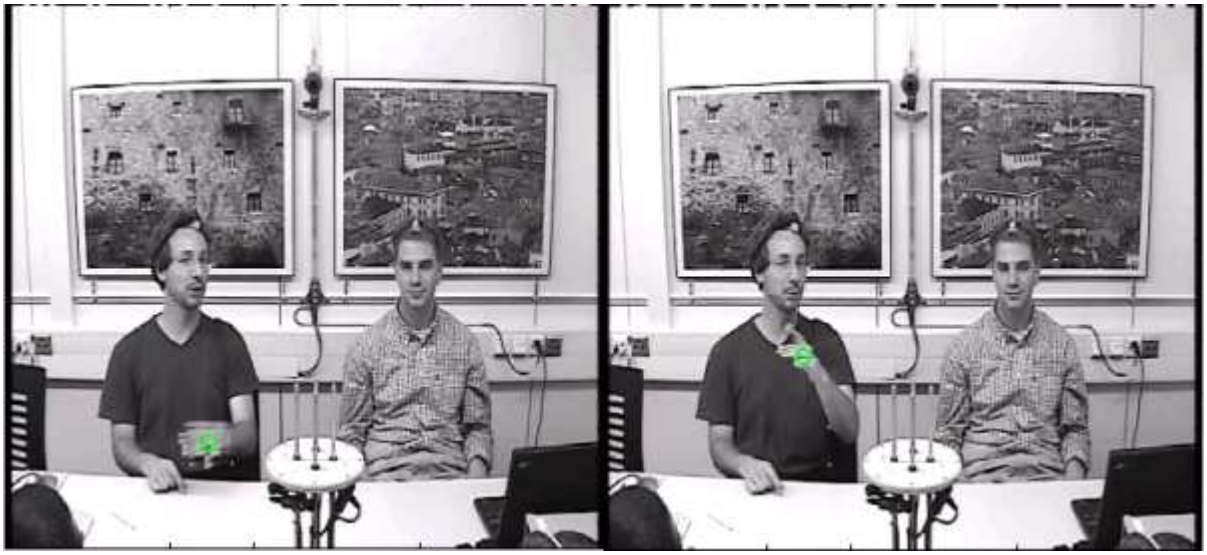

Figure 17: Example where keypoints are used to track a persons hand

For the estimated angles, individual actions can be recognized. From the dimensions described in appendix B head actions can't be recognized since only elevation of the head is measured. Similar, because no orientation of the hands is measured, not all hand actions can be recognized. Neural networks were used to classify actions on a per-frame basis. Table 7.5 shows the results for this classification, both using the 10-dimensional joint angle input vector and the 6-dimensional vector that contains (x, y) pairs of coordinates of the head and hands centers. Note that not enough training data was annotated to estimate location based actions. Current research efforts focus on recognition of individual actions using sequential machine learning techniques.

	Presenter	Participant
Left shoulder	3	3
Left elbow	1	1
Right shoulder	3	3
Right elbow	1	1
Left hip	3	-
Left knee	1	-
Right hip	3	-
Right knee	1	-
Back bend forward	-	1
Neck bend forward	-	1
Total	16	10

Table 32: Degrees of freedom within human body model for presenter and participant application

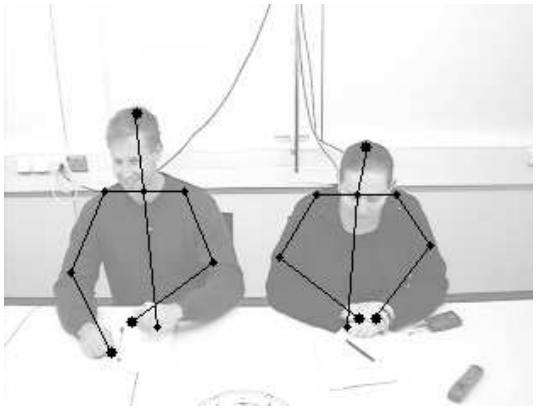

Figure 18: Example frame with superimposed body pose estimation.

	Joint angles	Face & hand coords
Hand actions {writing, no gesture}	91 %	89 %
Body actions {lean forward, lean backward, no gesture}	86 %	83 %

Table 33: Action classification results using neural networks

7.6 Meeting Event segmentation and recognition

This section encompasses the analysis and of meetings for a segmentation into sub-genres, so called meeting events (see also [84],[85]). The data for this work consists of the 53 scripted meetings, recorded in the IDIAP Smart Meeting Room. Each recorded meeting consists of a set of predefined meetings in a specific order. The events that were discriminated were

- Monologue (one participant speaks continuously without interruption)
- Discussion (all participants engage in a discussion)
- Note-taking (all participants write notes)
- White-board (one participant at front of room talks and makes notes on the white board)
- Presentation (one participant at front of room makes a presentation using the projector screen)

7.7 Feature extraction

This section illustrates in short the low level algorithms that provide the single actions of each meeting participant like speaker turns and various individual actions.

7.7.1 Speaker turn detection

The results of the speaker turn detection have been taken over from another partner in this project. A generic, short-term clustering algorithm is used that can track multiple objects for a low computational

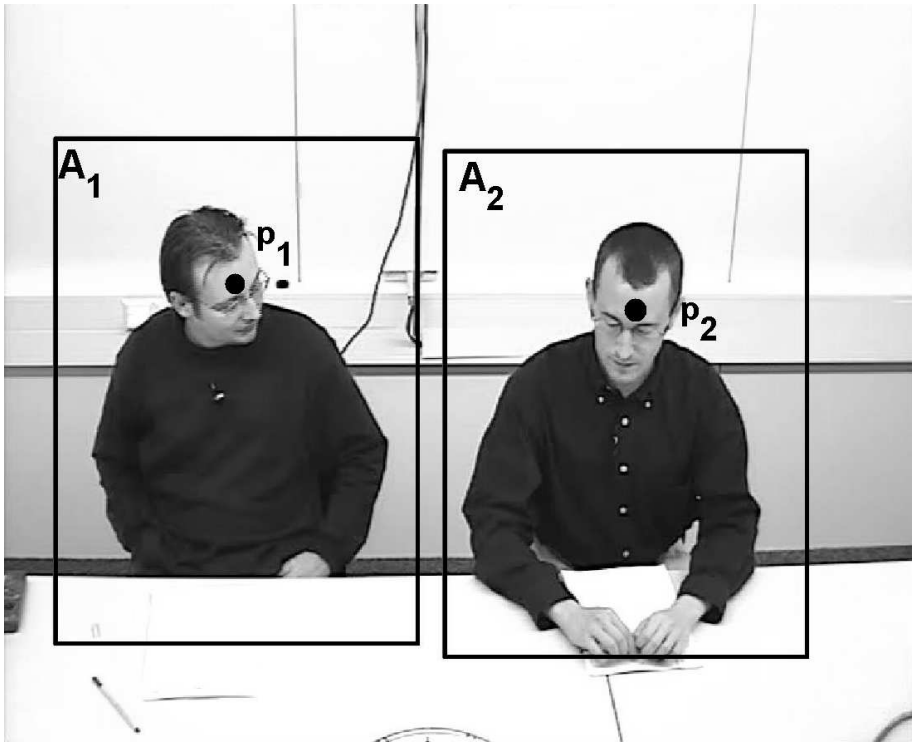

Figure 19: Video frame marked with action regions and center of the head provided by the tracking algorithm

cost. In [53] the three-step algorithm consisting in frame-level analysis, short-term analysis and long-term analysis is presented in detail.

7.7.2 Gesture recognition

Actions can be defined as movements in a certain surrounding of any person. In order to recognize actions one approach will be to extract features representing the motion in those surrounding areas. In [112], global motion features have turned out as suitable features to recognize gestures. For every single person in a so called action region A_i the individual actions are detected (cf. Fig. 19).

The video stream is divided into segments by a special algorithm. Then these segments are fed to a HMM based recognizer which has been trained on roughly 1000 gestures consisting of *writing*, *pointing*, *standing up*, *sitting down*, *nodding* and *shaking head*. In Table 34 the recognition results are shown for a continuous HMM with 6 states and 4 mixtures.

7.8 Classification of Meeting Events

The results of the recognizers described above can now be used to classify a temporal segment of a meeting into a meeting event as mentioned in Section 7.6. Thus, using this information, a static feature vector can be derived that contains the relative percentage of the various individual actions. We use for example the length of writing of a single person with respect to the whole considered time window. The same procedure applies for the remaining features like talking, nodding and so on.

	writing	pointing	standing up	sitting down	nodding	shaking head	Recognition rate [%]
writing	471	19	0	0	42	18	85.64
pointing	0	68	1	0	3	0	94.44
standing up	1	1	9	0	0	0	81.82
sitting down	0	0	2	7	0	0	77.78
nodding	8	7	7	0	225	43	77.59
shaking head	3	0	2	0	22	16	37.21

Table 34: Confusion matrix of single person action recognition

For the classification of the meeting events we chose the following classifiers:

- a simple hybrid Bayesian Network (BN) consisting of a discrete node as parent with five states (one for each meeting event) and nine continuous nodes directly connected to the parent node, representing the nine dimensions of the feature vector,
- Gaussian Mixture Models (GMM) with various numbers of Gaussians depending on the number of training material,
- a Neural Net with Multilayer Perceptrons (MLP) with 3 layers,
- a Radial Basis Network (RBN)with maximum 10 neurons,
- Support Vector Machines (SVM) with RBF-Kernel.

Each of the classifiers has been trained with the meeting events of the 30 training meetings. For evaluation purposes the remaining 23 meetings were used. For the recognition task alone, where the segment boundaries are given, the MLP performs best and achieves a recognition rate of 95.90%. Two classifiers (RBN and SVM) yield a quite good result with 95.08% whereas the GMMs seem not to be able to adapt well enough and achieve a recognition rate of 70.61%. The Bayesian Network is somewhere in between with 93.44%. One cause of this difference may be the relatively small amount of training material available.

7.9 Segmentation of Meeting Events

While segmentation of individual actions has been done manually as outlined in Section 7.7.2, an attempt has been made to automatically perform the segmentation of the meeting data into meeting events.

7.9.1 Integrated approach

The integrated approach combines the detection of the boundaries and classification of the segments in one step. The strategy is similar to that one used in the BIC-Algorithm [102] and is illustrated in Figure 20. Two connected windows with variable length are shifted over the time scale. Thereby the inner border is shifted from the left to the right in steps of one second and in each window the feature vector is classified. If there is a different result in the two windows, the inner border is considered a boundary of a meeting event. If no boundary is detected in the actual window, the whole window is enlarged and the inner border is again shifted from left to the right. This procedure can be described by the following

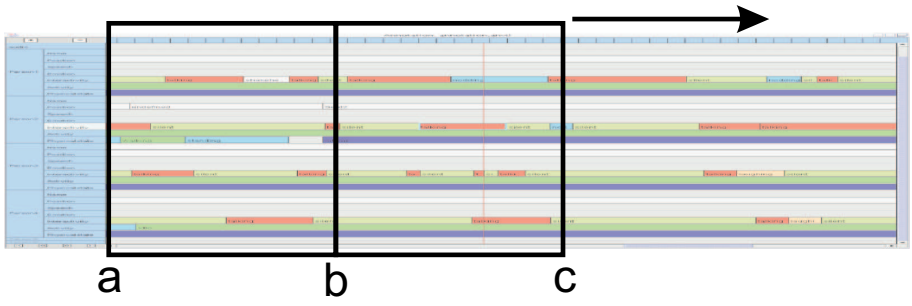

Figure 20: Two connected windows are shifted over the time scale to produce potential boundaries.

algorithm (a is the left border, b is the inner border, c is the right border of the window, L is the minimum length of a meeting event, $K(a,b)$ is the classification result of the interval $[a,b]$):

```
(1) initialize interval [a,c]:
    a = 1; b = a + L; c = a + 3L;
(2) if K(a,b) ≠ K(b,c) then
       save b as boundary
       a = c; b = a + L; c = a + 3L;
    else
       b = b + 1;
(3) if (c − b) < L
       c = c + 1; b = a + L;
       goto (2)
    else
       goto (2)
```

This algorithm is run until the right border c has reached the end of the video file.

7.9.2 Dynamic programming approach

Here the segmentation task is performed in two steps. At first, potential segment boundaries are searched; in the second step from all these possible boundaries those are chosen that give the highest overall score.

First the possible boundaries have to be found. Again two connected windows are shifted over the time scale as shown in Figure 20. This time the length of the windows remains fixed at 10 seconds each. Inside these two windows the feature vector is calculated and classified. If the results differ a potential segment boundary is assumed. In the same step a clustering of all found boundaries is performed. As long as the classification result $K(a,b)$ in the left window remains equal, the new assumed boundary is appended to the existing cluster $G\{i\}$. Otherwise a new cluster $G\{i+1\}$ is created. After that all clusters that contain less than three possible boundaries are discarded so that only important boundaries remain. Now we have a collection of arrays $G\{i\},\ \ i = 1,\ldots,N$, where N is the number of clusters, consisting in the potential boundaries.

Having found all boundaries that come into question, in each cluster $G\{i\}$ the in some sense 'best' boundary has to be chosen. This is accomplished via Dynamic Programming (DP). This approach assumes that the meeting events are mutually independent. So each boundary of a meeting event can be found if only the direct predecessor is known. The first and the last boundary are known a priori (beginning and end of the meeting), so the task is to choose the remaining inner boundaries that give the highest overall score. The score of a meeting event is calculated as the pseudo-probability that the

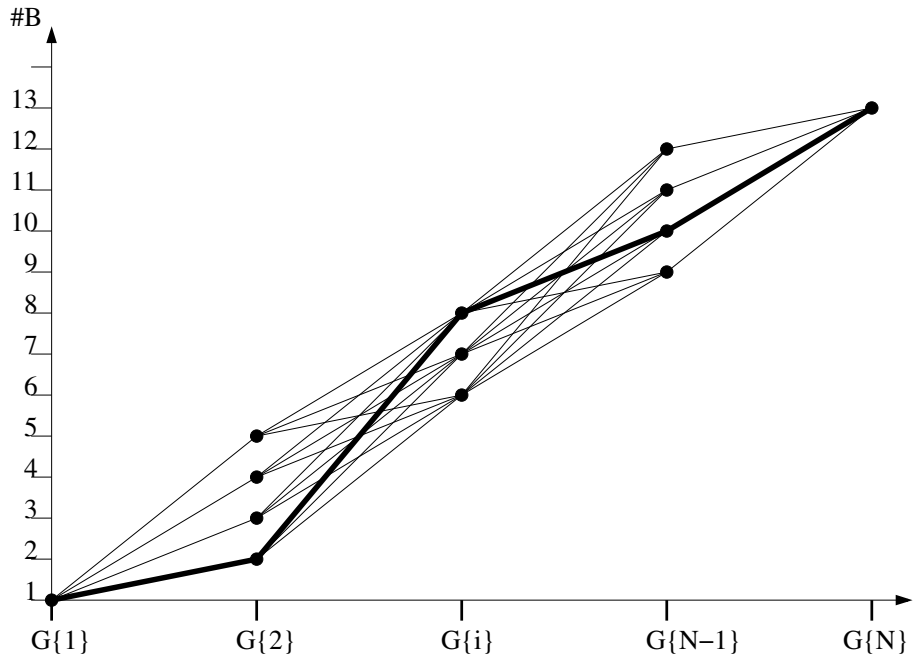

Figure 21: Finding the optimal boundaries: the path with the highest overall score is found through backtracking. The abscissa denotes the clusters of potential boundaries, the ordinate the number of the boundary.

classifier returns for the examined interval. This could be for example the normalized probability of the GMM or the normalized output of the neural net. As additional constraint only those boundaries could be chosen that ensure a minimum length of a meeting event of 15 seconds.

In Figure 21 the procedure for finding the optimal segment boundaries is illustrated. For each boundary $x \in G\{i\}$ the score $s_x(y)$ to each boundary $y \in G\{i-1\}, \quad i = 2, \ldots, N$ is calculated. Then the maximum score $s_{max}$for each x is chosen.

$$s_{x,max} = \max \; s_x(y); \tag{2}$$

The sum of this score and the overall score until $i-1$ is calculated and saved in a score-matrix $SG\{i\}$ together with the predecessor y.

$$SG\{i\} = \begin{bmatrix} \vdots & \vdots & \vdots \\ x & s_{x,max} + SG\{i-1\}_{y,2} & y \\ \vdots & \vdots & \vdots \end{bmatrix}; \tag{3}$$

This is done for all clusters $G\{i\}$. Afterwards the best path through all score matrices is found through backtracking. Starting with the last score matrix $SG\{N\}$, which contains only one boundary, and following the indices in the third column those boundaries are chosen that produce the best overall score. In a completing step two segments that contain the same meeting event are merged.

This approach has the advantage of being computationally much less expensive, since there are much less segments to test due to the fixed length of the sliding windows.

Classifier	Insertion	Deletion	Accuracy	Error
BN	0.1474	0.0622	7.9316	0.3903
GMM	0.2475	0.0233	10.8718	0.4140
MLP	0.0861	0.0167	6.3326	0.3244
RBF	0.0689	0.0300	5.6654	0.3164
SVM	0.1779	0.0083	9.0838	0.3576

Table 35: Segmentation results using the integrated approach (BN: Bayesian Network, GMM: Gaussian Mixture Models, MLP: Multilayer Perceptron Network, RBF: Radial Basis Network, SVM: Support Vector Machines). The columns denote the insertion rate, the deletion rate, the accuracy in seconds and the classification error rate (see text).

Classifier	Insertion	Deletion	Accuracy	Error
BN	0.1650	0.0467	6.6667	0.3664
GMM	0.2971	0.0250	33.2812	0.4911
MLP	0.1871	0.0317	16.0696	0.3896
RBF	0.1738	0.0083	16.0127	0.3969

Table 36: Segmentation results using Dynamic Programming.

7.9.3 Segmentation results

From the 53 available meetings, mentioned in Section 7.6, 30 were chosen for the training of the classifiers, the remaining 23 wereused for evaluation purposes.

The results of the segmentation are shown in Table 35 and Table 36 respectively (BN: Bayesian Network, GMM: Gaussian Mixture Models, MLP: Multilayer Perceptron Network, RBF: Radial Basis Network, SVM: Support Vector Machines). Each row denotes the classifier that was used. The columns show the insertion rate (number of insertions in respect to all meeting events), the deletion rate (number of deletions in respect to all meeting events), the accuracy of the found segment boundaries (mean absolute error in seconds) and the recognition error rate (cf. [29]). In all columns lower numbers denote better results.

As can be seen from the tables, the results are quite variable and heavily depend on the used classifier. With the integrated approach (cf. Table 35) the best outcome is achieved by the radial basis network. Here the insertion rate is the lowest. The detected segment boundaries match pretty well with a deviation of only about five seconds to the original defined boundaries.

The results of the segmentation with dynamic programming were in general slightly worse. Due to the impossibility to get a score from the SVMs, these were not used here. Remarkable is the difference of ten seconds in the accuracy of the found boundaries between the Bayesian Network and the Neural Networks. The Bayesian Networks miss the given boundaries by 6.6 seconds on average. The neural network approaches make a greater mistake and produce a deviation of approx. 16 seconds.

8 Focus of attention

8.1 Objectives

The original objective of this part was to study tasks related to either the focus-of-attention (FOA) of meeting participants, or to the meeting focus-of-attention. However, as the latter we difficult to define, we decided to first concentrate on the the individual FOA. Moreover, defining the general FOA of people also appeared to be problematic: first, a person might have multiple FOA (e.g. a person browsing some notes while listening to the speaker); secondly, as it is related to the mental state of participant, it might be difficult to ground-truth (e.g. from the audio-video recordings, identifying the current speaker as the FOA of somebody looking at the table is a matter of interpretation), which would prevent then a proper evaluation of developped algorithms. As a consequence, we decided **to define the focus-of-attention of people as the spatial locus defined by the person's gaze**.

We identified two research tracks related to the FOA:

- the first track is concerned with the **recognition** of the FOA. More precisely, given recorded meeting data streams, can we identify at each instant the FOA of people ? One obvious research direction for this task is the study and development of gaze estimation algorithms, or, as a surrogate, of head orientation estimation algorithms.
- in the second track, the objective is to identify the role played by the FOA in the dynamics of meeting. For instance, can we identify the current speaker if we know the FOA of each participant ? What do the sequence of FOA say about the ambiance in the meeting, i.e. is the meeting boring or interesting ? can we identify the main character or leader in the meeting from it ? Answering such questions would thus be useful to understand the relationship between the FOA and other cues (such as speaker turns) as well as to more precisely identify the interactions between participants (e.g. by contributing to the recognition of the higher level dialog acts), which in turn could translate into better FOA recognition algorithms.

In the AMI project, we have started to work in both directions. The next sections will summarize the work that has been done on head orientation estimation, and the study on speaker identification capability from head poses. As a proper evaluation of these works requests annotated data, we will first present in the next section the effort that has been done by the partners along this direction.

8.2 Databases specification, recording and annotation

To achieve the research tasks, we have considered three databases with different annotation schemes. Two of these databases are already recorded, and are the result of a close collaboration between the university of Twente (UT) and IDIAP, while the specification for the third one has been written. These databases along with their purpose and annotation details are described in the next subsections.

8.2.1 Head orientation database

Purpose : evaluation of head pose estimation and head tracking algorithms.
One first step towards determining a person's FOA consists of estimating its gaze direction. Then, from the geometry of the room (object, cameras) and the location of meeting participants, the FOA can be estimated. As estimating gaze is difficult (and requires very close-up views of people), we have developed as an approximation algorithms for estimating the head pose (see 8.3). These algorithms were first assessed by visual inspection. However, in view of the limitations of visual evaluation, and the inaccuracy obtained by manually labeling head pose in real videos, we decided to record a video database with head pose ground truth produced by a flock-of-birds device. It is important here to mention that we did not

find any publicly available database on this topic. The closer one in research focus is the POINTING[1] database, which contains only static images of people looking to discrete position of the space and with uniform background.

Specifications : the database is specified by the following elements:

- annotation : head pose with respect to the camera, which is defined by three euler angles (pan, tilt and roll).
- content : to account for a larger set of situations, we considered two scenario:
  - an office scenario: in this case, one person is performing different activities at their desk, such as typing, reading, calling a person, discussing with somebody, etc...
  - a meeting scenario: recording of groups of 4 people involved in a discussion.

An objective of the database was to have the largest amount of different faces, to better evaluate the generalization performance of algorithms.

Status : the data have been recorded and annotated. The definition of evaluation protocols are in progress. More specifically, the database comprises :

- 14 sequences of approx. 5 minutes each for the office scenario.
- 8 meetings of 4 people (duration of each meeting is approx. 6 minutes). The recording took place in IDIAP's smart meeting room. However, due to technological constraints (the magnetic field due to the setup caused distortions on the flock-of-birds readings), we were able to capture the head ground truth of only two participants.

Groundtruth has been elaborated using flock of birds magnetic sensors attached to the head. Precise care has been taken to calibrate the spatial transformation between the 3D magnetic readings, and the camera frames. The precision of the groundtruth is of 6 degrees approximately. A website is currently built to provide a free access to these data. We expect to release the database at the end of this year.

8.2.2 Focus-of-attention database

Purpose : evaluation of focus-of-attention recognition algorithms.
The purpose of this database is different than the head orientation database. Here, the emphasis is on the recognition of a finite set of specific FOA locus. Thus, while mapping estimated head orientations to FOA labels might be one approach to this problem, other methods might also be considered and need data to be evaluated on. Besides, it might be also important to evaluate how higher head pose estimation accuracy translates into higher FOA recognition rates.

Specifications : A document has been elaborated to provide the definition of the focus of attention, the set of visual focus of attention to annotate, and the requirements for the datasets. It is given in appendix A.

Status : the data specification and annotation protocols are written. The recording of the specific data sets will be performed in the upcoming months. Annotation will follow.

[1] http://www-prima.inrialpes.fr/Pointing04

8.2.3 Discussion database

Purpose : study the role of FOA in meeting situation understanding, behaviour understanding, and non-verbal communication analysis.
More precisely, we have studied so far differences in head orientation between speakers and non speakers, and the ability of both humans and machine learning algorithms to predict the current and the next speaker based solely on the head orientations of meeting participants (see Section 8.4). Other research directions with this database might be argumention analysis and adressee detection.

Specifications : All the recorded meetings are discussions. People were asked to debate three statements that were shown one after another on the white board. This resulted in nice discussions with a lot of argumentation, expressive behaviour and emotional speech.
In addition to the flock-of-birds readings, which provide information on head orientation for display purposes, the database will be annotated with speech transcription according to the AMI guidelines, and later, with dialogue acts adressee information.

Status : The Video, Audio and Flock data is available for AMI partners[2]. There are speech transcriptions of three of the height meetings, and the rest is to be completed soon. Figure 23 displays an image example of the setup.

8.3 Joint head tracking and pose estimation

Head pose estimation is often used as a first step for other higher level tasks such as facial expression recognition or gaze direction estimation. In meetings, head pose can be reasonably used as a proxy for gaze (which usually calls for close views), and can thus be useful for determination of visual focus-of-attention and addressees in conversations. Most of the existing work for head tracking and pose estimation defines the task as two sequential and separate problems: the head is tracked, its location is extracted, and the head pose is estimated from the head location. As a consequence, the estimated head pose totally depends on the tracking accuracy. This formulation misses the fact that knowledge about head pose could be used to improve head modeling and thus improve tracking accuracy.

In our approach, we couple head tracking and pose estimation using a mixed-state particle filter (PF) [7]. In this paragraph, we first recall the general particle framework. Specific elements then follow.
The Bayesian formulation of the tracking problem is well known. Denoting by X_t the hidden state representing the object configuration at time t, and by Y_t the observation extracted from the image, the filtering distribution $p(X_t|Y_{1:t})$ of X_t given all the observations $Y_{1:t} = (Y_1 \dots Y_t)$ up to the current time can be recursively computed by:

$$p(X_t|Y_{1:t}) = Z^{-1} p(Y_t|X_t) \times \int_{X_{t-1}} p(X_t|X_{t-1}) p(X_{t-1}|Y_{1:t-1}) dX_{t-1} \quad (4)$$

where Z is a normalizing constant. A PF is a numerical approximation to the above recursion in the case of non-linear and non-Gaussian models. The basic idea behind PF consists of representing the filtering distribution using a weighted set of samples $\{X_t^n, w_t^n\}_{n=1}^{N_s}$, and updating this representation as new data arrives. With this representation, Eq. 4 can be approximated by :

$$p(X_t|Y_{1:t}) \approx Z^{-1} p(Y_t|X_t) \sum_{n=1}^{N_s} w_{t-1}^n p(X_t|X_{t-1}^n) \quad (5)$$

using importance sampling. Given the particle set at the previous time step $\{X_{t-1}^n, w_{t-1}^n\}$, configurations

[2] http://hmi.ewi.utwente.nl/AMIMeeting/

at the current time step are drawn from a proposal distribution $q(X_t) = \sum_n w_{t-1}^n p(X_t|X_{t-1}^n)$. The weights are then computed as $w_t^n \propto p(Y_t|X_t^n)$.
Four elements are important in defining a PF:

1. the *state space*, which defines the elements we are looking for.
2. the *dynamical model* $p(X_t|X_{t-1})$ defines the temporal evolution of the state.
3. the *observation likelihood* $p(Y_t|X_t)$ measures the adequacy between the observation and the state. This is an essential term, where data fusion occurs, and whose modeling accuracy can greatly benefit from the additional discrete variables in the state space.
4. the *sampling mechanism* places new samples as close as possible to regions of high likelihood.

These elements along with our model are described in the next paragraphs.

Our approach to the head head tracking and pose estimation consists of coupling both problems using a mixed-state PF [7]. The state $X_t = (x_t, l_t)$ is a mixed variable. The continuous variable $x = (\mathbf{T}, s)$ specifies the head location and scale. The discrete variable l specifies an element of the head pose exemplars set. The pose at given time is obtained by marginalizing over the spatial configuration part of the state. In the following paragraph, we describe the head pose models, the dynamical model, and the observation model.

Head pose exemplars are learned using the PIE database. A total of N_θ head poses are defined by a pan angle ranging from -90 to 90 degrees discretized with 22.5-degree steps. For each head pose θ, Gaussian and Gabor features are extracted from training images, concatenated into a single feature vector, and clustered with K-means into L_θ clusters $\{e_l^\theta = (e_{l,j}^\theta), l \in \mathcal{L}_\theta\}, |\mathcal{L}_\theta| = L_\theta$. The cluster centers are taken to be the head pose exemplars. The number of elements of each cluster are used to define prior distributions π_l^θ, and the diagonal covariance matrix of the features $\sigma_l^\theta = diag((\sigma_{l,j}^\theta))$ is used to define pose probability models. The pose of an head image is estimated by extracting its feature vector $Y = (Y_j)$, and finding the pose MAP estimate by $p(Y|\theta) = \sum_{l \in \mathcal{L}_\theta} \pi_l^\theta p(Y|l)$, with

$$p(Y|l) = \prod_j \frac{1}{\sigma_{l,j}^\theta} \max(\exp -\frac{1}{2} \left(\frac{Y_j - e_{l,j}^\theta}{\sigma_{l,j}^\theta} \right)^2, T) \tag{6}$$

where T is a bound introduced to tolerate modeling errors.

The dynamical model is a second order autoregressive process $p(X_t|X_{t-1}, X_{t-2})$. Assuming that the two components x_t and l_t are independent, and that head pose depends only on the previous pose, the dynamics factorize as :

$$p(x_t|x_{t-1}, x_{t-2})p(l_t|l_{t-1}).$$

Finally, the observations are obtained by extracting the features $Y(x)$ from the image region specified by the spatial configuration x. The observation likelihood is given by $p(Y_t|X_t) = p_T(Y_t(x_t)|l_t)$, with p_T defined in Eq. 6.

Results. Head pose estimation was tested on PIE database. The best result was obtained with two exemplars per pose, with a recognition rate of 94.8% while the state-of-the-art obtains around 90% [13]. More details about evaluation can be found in [7]. The joint tracking algorithm was also tested on video sequences from our meeting room. An example with $N_S = 100$ particles is shown in Fig. 22. Tracking and head pose estimation are visually quite satisfactory. An objective evaluation of the algorithms on the database mentionned in the previous section is currently in process.

Open issues. The current features are obtained using gray-level information. While our head tracking and pose estimation system works well in general, some problems might occur when the background is highly textured. The use of color information for more robust tracking is under investigation.

Figure 22: Joint tracking and head pose estimation in meeting room. The green box and red arrow specify the estimated head location and head pose, respectively. The red circle gives information about the pose value; its radius corresponds to 90 degrees. The participants are looking at the room entrance.

8.4 Speaker prediction from meeting participants' head pose

On the data described in Section 8.2.3, we have conducted some initial research dealing with average speaker turn lengths, average orientation when speaking and listening and more of this limited domain exploration.

Figure 23: Overview of meeting data.

More precisely, we have setup a distributed experiment environment (see Fig. 24) resulting in a system where we can conduct experiments at remote sites with one server sending out and collecting the samples. This environment displays a virtual meeting room with the head of participants avatars driven by the head orientation of real participants obtained through the flock-of-birds devices attached to each participant's head (see Fig. 23). A motivations for using such a setup is the fact that we examine rich data containing several modalities. In comparison with e.g. the ICSI corpus, our small corpus contains visual data. Another advantage is that we have exact head orientations. So no rough estimations are made as e.g. in [98]. A drawback of our approach is that the measured head orientation does not fully correspond with the actual eye gaze. Studies showed however that the accuracy of focus of attention estimation based on head orientation data alone is more than 88% [99]

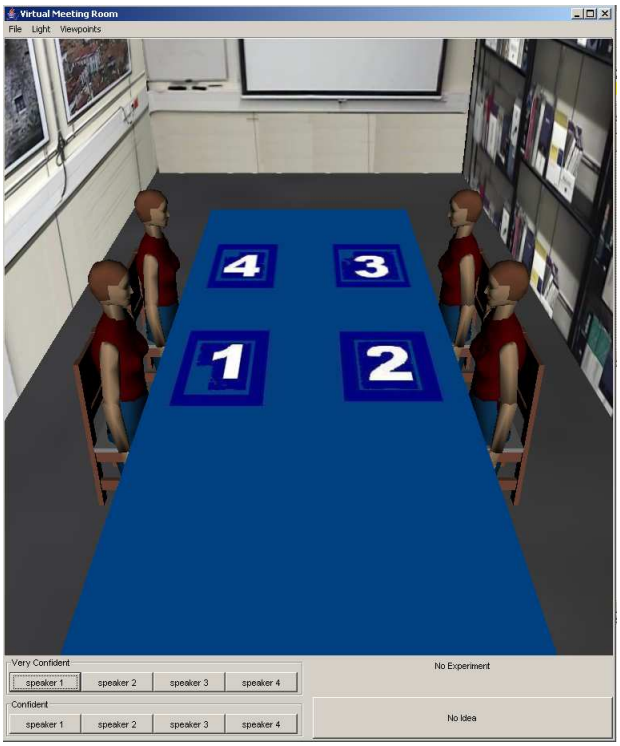

Figure 24: Virtual setup provided to the experimenters.

[97], [98]. Furthermore this approach enabled us to correct for headmovements contrary to the work of Vertegaal[104] where a 'chair with very comfortable neck support' was used to minimise headmovement.

With this environment we have conducted a first set of experiments. We asked persons to judge who is the current speaker based on 'frames' of head orientations samples taken from the database. These results have been compared with machine learning algorithms on the same data.

Algorithm	Score
Naive Bayes with discretization	70.3 %
Neural Network	63.3%
Humans	37.7%

Table 37: Classification results for the Naive Bayes classifier with discretization, Neural Networks and Humans

Table 37 and the following points summarizes some of our findings. Additional results and experimental details can be found in [87].

1. Bayesian networks with discretization can score up to 90% in correctly predicting the speaker based solely on azimuth information of all the participants when trained and tested on data from a single meeting using ten fold cross validation.

2. On all samples (more than 64000) a neural network scored on average 63%.

3. As a comparison, humans scored around 38% on 3200 presented samples.

4. Furthermore, we studied and compared the performances of people that were receiving feedback on their assignements with people getting no feedback. Results have shown that people with feedback performed significantly ($p < 0.05$ with a paired t-test) better than the others. However, when samples from a different meeting were presented to the people with feedback, their performance droped significantly below the results from people who never recieved any feedback.

9 Summary and Future Work

WP4 is concerned with the automatic recognition from audio, video, and combined audio-video streams, with an emphasis on developing models and algorithms to combine modalities.

In this report we described the implementation and first evaluations of ported and developed algorithms. Seven main tasks have been identified for WP4:

- *Baseline speech recognition system*: The automatic speech recognition subgroup is concerned with the development of a speech recognition system for the use on AMI data
- *Event spotting*: Acoustic event (mainly keyword) spotting in meetings has the goal to find all occurrences of entered word in a meeting and sort them according to confidences. This will allow for Google-like browsing of meetings using acoustics. It also has the goal to verify if a word really occurred in a particular meeting, which is linked to WP5 summarization work.
- *Person segmentation / clustering / identification*: Algorithms for face detection, face recognition, speaker recognition, person segmentation, and clustering will be assembled and transferred to the common platform. Fusion of audio- and visual methods will be carried out.
- *Emotion recognition*: Systems for automatic emotion recognition, based on audio, video, and combined methods will be developed. Existing methods will be ported to the AMI domain and evaluated.
- *Localization and Tracking*: Available audio-visual localization and tracking algorithms will be ported and combined to multimodal tracking procedures, combining e.g., motion features and acoustic sources obtained from microphone arrays.
- *Gestures and actions*: Several alternative systems for gesture and action recognition using both video only and combined audio-video input will be developed and ported to the AMI domain and evaluated.
- *Focus of attention*: The objective of this task is to study tasks related to either the focus-of-attention of meeting participants, or to the meeting focus-of-attention

In this report we described several implemented and ported algorithms and methods for each of the seven tasks. The output of the described procedures can then be used as input for WP5. Currently each of the sub-groups is defining common evaluation schemes (cf. Milestone M4.3, month 18). This allows to compare different approaches to a problem on the common AMI data set (cf. Deliverable D4.2, month 24). Furthermore the common evaluation scheme guarantees common interfaces among the involved partners, and a common, stringent output of the different recognisers. Therefore WP5 has defined inputs from WP4 - independent of the actual used algorithm. Finally the common interfaces allow the fusion of several algorithms to a larger system.

Consequently, the next steps in WP4 are: to finalize the definitions of common *evaluation schemes* and interfaces (M4.3). Then to *evaluate* the different approaches in the seven sub-groups according to the evaluation schemes on the common AMI data set (D4.2). With these results we can *improve* the algorithms and methods based on the evaluation results and *fuse different algorithms* to further improve the recognition performance.

A Focus of Attention annotation scheme

A.1 Interpretation

The most important issue is that we stick to Visual focus of attention of individuals, defined by the head orientation or eye gaze. So if someone is looking at a person but thinking about his upcoming holiday we will only label where he is looking at, since there is no end in deriving the thought of others. Ground-truth labeling is also undoable in this sense of interpreted focus of attention.

A.2 Data Set

The data on which annotations are performed is to be split in two groups. The training and the test set.

A.2.1 Training set

Specific recordings to have sufficient training data for all the labels or possible focus items. In particular, it might be important to have a variety of different people in the training set. For each label, at least 15 occurrences should be annotated, for at least 3 different people. Several parts (30 seconds/1 minute) of recorded meetings (different than those used in the test set) can be used in addition.

A.2.2 Test set

For validation, a minimum of 5 occurrences of each of the labels would be sufficient. Note however that some labels will be much more represented (e.g. looking at persons). We aim at labeling approximately 15 minutes of 2 meetings, and around 3 selected minutes of around 8 other meetings (to have evaluation with more participants).

A.3 Annotation requirements

Most of the focus are smart meeting room dependent. Thus, the geometry of the room can be used as prior knowledge. (We know for each person under which (pan, tilt combinations) we find which person) There should be no ambiguities in the annotations and inter annotator agreement.

A.4 Items to be Annotated

This is the proposed list of focuses that are to be labeled for each meeting participant. There is a hierarchy in this list, so whenever there is an ambiguity (see below), the higher level is to be chosen.

- Other persons in the meeting (in general, 4 labels)
- Objects of interest : own notes, own laptop (2 labels)
- Specific locations of interest : entrance, whiteboard, slidescreen, table except personal notes/laptop (4 labels)
- Attentively looking at one object/location (different from above) (1 label)
- Unfocused, or none of the above (1 label).

In general, the size of the label set will be 12.

A.5 Annotation Details

Labeling might sometimes be difficult to assess from the visual data. If there is somebody at the white board for instance, the label could be either the person or white board. In such cases, and any other ambiguous cases, the hierarchy as specified above should be followed, which means that in the above example, the person should be labeled as the FOA since he is higher in the hierarchy.

As tool for annotation the tool will be used that is the best known to the annotators. Anvil, TasX or the interface developed for video labelling in Twente (Label'a').

B Gestures and Actions annotation scheme

Within this subgroup, the task definition and the requirements for dataset recording and annotation have been specified.

Task definition: specificperson activities and gestures relevant to the analysis and understanding of meeting have been identified. They mainly consist of:

- location based activities
- head gesture/activities
- hand gesture/activities
- body gesture/activities
- miscellaneous activities

Datasets: two types of dataset are necessary. A training datasets:

- specific data with high densities of identified activities/gestures. To be collected at one of AMI setup
- Excerpts from the AMI core corpus.

and a test dataset: Excerpts from the AMI core corpus, different than the training set.

Annotation: Annotation will be done manually. Beginning and end time points of each gesture/activity per participant will be annotated. A more elaborate version of the specifications can be found on http://www.amiproject.org/private/WP03/annotgroups/indact.

References

[1] Isip echo cancellation software package, version 2.6. `http://www.isip.msstate.edu/projects/speech/software/`

[2] *Improved speaker segmentation and segments clustering using the bayesian information criterion.* Proceedings EUROSPEECH '99, 1999.

[3] J. Ajmera, G. Lathoud, and I. McCowan. Clustering and segmenting speakers and their locations in meetings. In *Proc. IEEE ICASSP*, 2004.

[4] M. Al-Hames and G. Rigoll. An investigation of different modeling techniques for multi-modal event classification in meeting scenarios. MLMI'04 poster presentation, Martigny, Switzerland, 2004.

[5] M. Al-Hames and G. Rigoll. A multi-modal mixed-state dynamic bayesian network for robust meeting event recognition from disturbed data. Submitted for publication, 2004.

[6] M. Lang B. Schuller, G. Rigoll. Speech emotion recognition combining acoustic features and linguistic information in a hybrid support vector machine - belief network architecture. In *Proceedings of ICASSP'04*, Montreal, Canada, 2004. IEEE.

[7] S. Ba and J.-M. Odobez. A probabilistic framework for joint head tracking and pose estimation. In *Proc. ICPR*, Cambridge, UK, 2004.

[8] Baker and et al. Real-time non-rigid driver head tracking for driver mental state estimation. In *Proc. of 11th World Congress on Intelligent Transportation Systems*, 2004.

[9] C. Barras, S. Meigner, and J. L. Gauvain. Unsupervised online adaptation for a speaker verification system over the telephone. In *Proc. Speaker Odyssey*, 2004.

[10] A. Baumberg and D. Hogg. An efficient method for contour tracking using active shape models. In *Proc. of IEEE Workshop on Motion of Non-rigid and Articulated Objects*, page 194, 1994.

[11] M. Bett, R. Gross, H. Yu, X. Zhu, Y. Pan, J. Yang, and A. Waibel. Multimodal meeting tracker. In *Proceedings of RIAO2000*, Paris, France, April 2000.

[12] J. Blauert. *Spatial Hearing — The Psychophysics of Human Sound Localization.* MIT Press, 1997.

[13] L. Brown and Y. Tian. A study of coarse head pose estimation. In *IEEE Workshop on Motion and Video Computing, Orlando*, December 2002.

[14] L. Burget. Combination of speech features using smoothed heteroscedastic linear discriminant analysis. In *Proc. ICSLP*, 2004.

[15] L. Burget. *Complementarity of Speech Recognition Systems and System Combination.* PhD thesis, Faculty of Information Technology VUT Brno, 2004. Submitted.

[16] W. M. Campbell. Generalized linear discriminant sequence kernels for speaker recognition. In *Proc. ICASSP*, pages 161–164, 2002.

[17] B. Chen, Q. Zhu, and N. Morgan. Long-term temporal features for conversational speech recognition. In *Proc. MLMI'04*, Martigny, Switzerland, 2004.

[18] R. Collobert, S. Bengio, and J. Mariéthoz. Torch: a modular machine learning software library. Technical Report IDIAP-RR 02-46, IDIAP, 2002.

[19] Cootes and et al. Active shape models - their training and application. *Computer Vision and Image Understanding*, 61, 1995.

[20] R.R. Cornelius. *"The Science of Emotion. Research and Tradition in the Psychology of Emotion"*. Prentice-Hall, Upper Saddle River, NJ, 1996.

[21] R. Cowie and R. Cornelius. Describing emotional states that are expressed in speech. *Speech Communication*, 40:5–32, April 2003.

[22] R. Cowie, M. Schroeder, M. Sawey, E. Douglas-Cowie, E. McMahon, and S. Savvidou. *FEELTRACE, a tool for recording how perceived emotion develops over time*, 2004. `http://www.dfki.de/~schroed/feeltrace/`.

[23] R. Craggs and M. McGee Wood. A two dimensional annotation scheme for emotion in dialogue. In *AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications, AAAI-EAAT 2004*, Stanford University, March 2004.

[24] R. Cutler, Y. Rui, A. Gupta, JJ Cadiz, I. Tashev, L. w. He, A. Colburn, Z. Zhang, Z. Liu, and S. Silverberg. Distributed meetings: A meeting capture and broadcasting system broadcasting system. In *Proceedings of ACM Multimedia Conference*, 2002.

[25] L. Devillers and L. Lamel. Emotion detection in task-oriented dialogs. In *Proceedings of the ICME 2003*, volume III of *Multimedia Human-Machine Interface and Interaction I*, pages 549–522, Balitmore, MD, USA, 2003. IEEE.

[26] L. Devillers, L. Lamel, and I. Vasilescu. Emotion Detection in Task-oriented Spoken Dialogs. In *International Conference on Multimedia and Expo*, Baltimore, July 2003.

[27] E. Douglas-Cowie, N. Campbell, R. Cowie, and P. Roach. Emotional speech: Towards a new generation of databases. *Speech Communication*, (40):33–60, 2003.

[28] R.B. Dunn, D.A. Reynolds, and T.F. Quatieri. Approaches to speaker detection and tracking in conversationalspeech. *Digital Signal Processing*, 10, 2000.

[29] S. Eickeler and G. Rigoll. A novel error measure for the evaluation of video indexing systems. In *IEEE Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Istanbul, Turkey, June 2000.

[30] P. Ekman. An argument for basic emotions. *Cognitive Emotions*, 6:169–200, 1992.

[31] P. Ekman and W. Friesen. *"The Facial Action Coding System"*. Consulting Psychologists' Press, San Francisco, CA, 1978.

[32] A. Nogueiras et al. Speech emotion recognition using hidden markov models. In *Eurospeech 2001, Poster Proceedings*, pages 2679–2682, Scandinavia, 2001.

[33] R. Cowie et al. Emotion recognition in human-computer interaction. *IEEE Signal Processing Magazine*, 18(1):32–80, January 2001.

[34] D. Gatica-Perez, G. Lathoud, J.-M. Odobez, and I. McCowan. Audio-visual tracking of multiple speakers in meetings. In *IDIAP Research Report RR-04-66*, Dec. 2004.

[35] D. M. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73(1):82–98, 1999.

[36] D. M. Gavrilla. Pedestrian detection from a moving vehicle. In *Proc. of the European Conference Computer Vision*, page 37, Dublin, 2000.

[37] B. R. Glasberg and B. C. J. Moore. Derivation of auditory filter shapes from notched-noise data. *Hearing Research*, 47:103–138, 1990.

[38] P. Greasley, C. Sherrard, and M. Waterman. "emotion in language and speech: Methodological issues in naturalistic approaches". *Language and Speech*, 43(4):355–375, October 2000.

[39] V. Hozjan and Z. Kacic. Improved emotion recognition with large set of statistical features. In *Proceedings of the Eurospeech*, pages 133–136, Geneva, Switzerland, 2003. IEEE.

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

[41] M. Isard and A. Blake. Icondensation: Unifying low-level and high-level tracking in a stochastic framework. In *Proc. of the Fifth European Conference on Computer Vision (ECCV '98)*, volume I, pages 893–908, Freiburg, Germany, June 1998.

[42] L. Janku and J. Cernocky. Unconstrained phoneme-based keyword-spotting in meeting data. In *Proc. MLMI'04*, Martigny, Switzerland, 2004.

[43] L. A. Jeffress. A place theory of sound localization. *J. Comp. Physiol. Psychol.*, 41:35–39, 1948.

[44] M. Karafiat, F. Grezl, and J. Cernocky. Trap-based features for lvcsr of meeting data. In *Proc. ICSLP*, Jeju, Korea, 2004.

[45] M. Kipp. Anvil - a generic annotation tool for multimodal dialogue. In *Proceedings of the 7th European Conference on Speech Communication and Technology (Eurospeech)*, pages 1367–1370, Aalborg, September 2001.

[46] F. Kottelat and J-M. Odobez. Audio-video person clustering in video databases. IDIAP-RR 46, IDIAP, Martigny, Switzerland, 2003.

[47] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright. Convergence properties of the nelder-mead simplex method in low dimensions. *SIAM Journal of Optimization*, 9(1):112–147, 1998.

[48] G. Lathoud, J. Bourgeois, and J. Freudenberger. Sector-based detection for hands-free speech enhancement in cars. In *IDIAP Research Report RR-04-67*, Dec. 2004.

[49] G. Lathoud and M. Magimai.-Doss. A sector-based, frequency domain approach to detection and localization of multiple speakers. Technical Report RR04-54, IDIAP, Sep. 2004.

[50] G. Lathoud and M. Magimai.-Doss. A sector-based, frequency-domain approach to detection and localization of multiple speakers. In *Proc. ICASSP*, Philadelphia, Mar. 2005.

[51] G. Lathoud and I. McCowan. A sector-based approach for localization of multiple speakers with microphone arrays. In *SAPA Workshop*, 2004.

[52] G. Lathoud, I. A. McCowan, and J. Odobez. Unsupervised location-based segmentation of multi-party speech. In *Proc. of the 2004 ICASSP-NIST Meeting Recognition Workshop*, Montreal, Canada, May 2004.

[53] G. Lathoud, I. A. McCowan, and J.-M. Odobez. Unsupervised Location-Based Segmentation of Multi-Party Speech. In *Proceedings of the 2004 ICASSP-NIST Meeting Recognition Workshop*, Montreal, Canada, May 2004. IDIAP-RR 04-14.

[54] G. Lathoud and I.A. McCowan. A sector-based approach for localization of multiple speakers with microphone arrays. In *Proc. SAPA Workshop*, Korea, October 2004.

[55] G. Lathoud, J.-M. Odobez, and D. Gatica-Perez. Av16.3: an audio-visual corpus for speaker localization and tracking. In *Proc.MLMI Workshop*, Martigny, Jun. 2004.

[56] G. Lathoud, J.-M. Odobez, and D. Gatica-Perez. Av16.3: An audio-visual corpus for speaker localization and tracking. In *Proc. MLMI'04*, Martigny, Switzerland, 2004.

[57] C.M. Lee and R. Pieraccini. Combining acoustic and language information for emotion recognition. In *Proceedings of the ICSLP*, Denver, CO, USA, 2002.

[58] I. McCowan, S. Bengio, D. Gatica-Perez, G. Lathoud, F. Monay, D. Moore, P. Wellner, and H. Bourlard. Modeling human interaction in meetings. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, Hong Kong, April 2003. IDIAP-RR 02-59.

[59] I. McCowan, D. Gatica-Perez, S. Bengio, and G. Lathoud. Automatic analysis of multimodal group actions in meetings. IDIAP-RR 27, IDIAP, Martigny, Switzerland, 2003. Submitted to IEEE Transactions of Pattern Analysis and Machine Intelligence.

[60] N. Mirghafori, A. Stolcke, C. Wooters, T. Pirinen, I. Bulykoa, D. Gelbart, M. Graciarena, S. Otterson, B. Peskin, and M. Ostendorf. From switchboard to meetings: Development of the 2004 icsi-sri-uw meeting recognition system. In *Proc. Intl. Conf. Spoken Language Processing*, Jeju, Korea, 2004.

[61] B. C. J. Moore. *An Introduction to the Psychology of Hearing*. Academic Press, fifth edition, 2003.

[62] D. Moore. The idiap smart meeting room. IDIAP-COM 07, IDIAP, 2002.

[63] N. Morgan, D. Baron, J. Edwards, D. Ellis, D. Gelbart, A. Janin, T. Pfau, E. Shriberg, and A. Stolcke. The meeting project at icsi. In *Proceedings of the Human Language Technology Conference*, San Diego, CA, March 2001.

[64] P. Motlicek and J. Cernocky. Multimodal phoneme recognition of meeting data. In *Proc. 7th International Conference Text, Speech and Dialogue (TSD)*, Brno, Czech republic, 2004.

[65] R. Mueller, J. Moore, R. Ordelman, and V. Wan. Annotation of emotions in meeting scenarios, 2004. AMI Draft Proposal.

[66] R. Mueller and G. Rigoll. Belief networks in natural language processing for improved speechi emotion recognition. MLMI'04 poster presentation, Martigny, Switzerland, 2004.

[67] The NIST year 2004 Speaker Recognition Evaluation Plan. `http://www.nist.gov/speech/tests/spk/2004/index.htm`, 2004.

[68] EU-IST Network of Excellence HUMAINE (HUman-MAchine Interaction Network on Emotion) www.emotion research.net.

[69] R. Ordelman, D. Heylen, I. McCowan, J. Moore, M. Poel, R. Muller, and V. Wan. AMI emotion annotation discussion. Technical report, AMI working notes, 2004. AMI intranet.

[70] R. Ordelman, D. Heylen, I. McCowan, J. Moore, M. Poel, R. Muller, and V. Wan. Experiments on Emotion Annotation for the AMI Corpus. Technical report, AMI working notes, 2004. AMI intranet.

[71] EU-IST Project AMI (Augmented Multi party Interaction) www.amiproject.org.

[72] R. D. Patterson, I. Nimmo-Smith, J. Holdsworth, and P. Rice. An efficient auditory filterbank based on the gammatone function. Technical Report 2341, Applied Psychology Unit, University of Cambridge, UK, 1988.

[73] J. Pelecanos and S. Sridharan. Feature warping for robust speaker verification. In *Proc. Speaker Odyssey*. Crete, Greece, 2001.

[74] E. Petajan, I.S. Pandzic, T.K. Capin, P.H. Ho, R. Pockaj, H. Tao, H. Chen, J. Shen, P.E. Chaut, and J. Osterman. "iso/iecjtc1/sc29/wg11 n1365, face and body definition and animation parameters", October 1996.

[75] J. O. Pickles. *An Introduction to the Physiology of Hearing*. Academic Press, second edition, 1988.

[76] T. Pirinen and J. Yli-Hietanen. Time delay based failure-robust direction of arrival estimation. In *Proc. 3rd IEEE Sensor Array and Multichannel signal Processing Workshop*, Barcelona, 2004.

[77] V. Popovici, J.-P. Thiran, Y. Rodriguez, and S. Marcel. On performance evaluation of face detection and localization algorithms. In *Proc. IEEE ICPR*, 2004.

[78] R. Poppe. Real-time pose estimation from monocular image streams using silhouetted. Master's thesis, University of Twente, 2004.

[79] R. Poppe, D. Heylen, A. Nijholt, and M. Poel. Towards real-time body pose estimation for presenters in meeting environments. Submitted for publication, 2004.

[80] I. Potucek, G. Rigoll, F. Wallhoff, and M. Zobl. Dynamic tracking in meeting room scenarios using omnidirectional view. In *Proc. ICPR*, Cambridge, UK, 2004.

[81] I. Potucek and M. Spanel. Face detection in meeting room using omni-directional view. In *Proc. MLMI'04*, Martigny, Switzerland, 2004.

[82] I. Potucek, S. Sumec, and M. Spanel. Participant activity detection by hands and face movement tracking in the meeting room. In *Proc. CGI*, Crete, Greece, 2004.

[83] S. Reiter and G. Rigoll. Multimodal meeting event recognition fusing three different types of recognition techniques. MLMI'04 poster presentation, Martigny, Switzerland, 2004.

[84] S. Reiter and G. Rigoll. Segmentation and classification of meeting events using multiple classifier fusion and dynamic programming. In *IEEE Proceedings of the International Conference on Pattern Recognition (ICPR)*, pages 434–437. IEEE Computer Society, August 2004.

[85] S. Reiter and G. Rigoll. Multimodal meeting analysis by segmentation and classification of meeting events based on a higher level semantic approach. In *Proceedings of the 30th International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Philadelphia, USA, March 2005.

[86] D.A. Reynolds, T.F. Quatieri, and R.B. Dunn. Speaker verification using adapted gaussian mixture models. *Digital Signal Processing*, 10:19–41, 2000.

[87] R. Rienks, R. Poppe, and M. Poel. Speaker prediction based on head orientation. In *Proceedings of the BeneLearn 2005*, 2005. submitted.

[88] Y. Rodriguez, F. Cardinaux, S. Bengio, and J. Mariethoz. Estimating the quality of face localization for face verification. In *Proc. IEEE ICIP*, 2004.

[89] S. Schreiber and G. Rigoll. Robust face tracking and person action recognition. MLMI'04 poster presentation, Martigny, Switzerland, 2004.

[90] B. Schuller, G. Rigoll, and M. Lang. Towards intuitive speech interaction by the integration of emotional aspects. In *Proceedings of the SMC*, Yasmine Hammamet, Tunisia, 2002. IEEE.

[91] B. Schuller, G. Rigoll, and M. Lang. Hidden markov model-based speech emotion recognition. In *Proceedings of the ICASSP*, volume II, Hong Kong, China, 2003. IEEE.

[92] P. Schwarz, P. Matejka, and J. Cernocky. Towards lower error rates in phoneme recognition. In *Proc. 7th International Conference Text, Speech and Dialogue (TSD)*, Brno, Czech republic, 2004.

[93] K. R. Sherer. Vocal communication of emotion: A review of research paradigms. *Speech Communications*, 40:227–256, 2003.

[94] J. Sivic and A. Zisserman. A text retrieval approach to object matching in videos. In *Proceedings of the International Conference on Computer Vision*, 2003.

[95] K. Smith and D. Gatica-Perez. Order matters: a distributed sampling method for multi-object tracking. In *Proc. BMVC*, London, Sep. 2004.

[96] F. Solina, P. Peer, B. Batagelj, S. Juvan, and J. Kovac. Color-based face detection in the '15 seconds of fame' art installation. In *Proceedings of Mirage*, INRIA Rocquencourt, France, 2003.

[97] R. Stiefelhagen. Tracking focus of attention in meetings. In *IEEE International Conference on Multimodal Interfaces*, Pittsburgh, PA, USA, October 14–16 2002.

[98] R. Stiefelhagen, J. Yang, and A. Waibel. Estimating focus of attention based on gaze and sound. In *Workshop on Perceptive User Interfaces (PUI'01)*, November 2001.

[99] R. Stiefelhagen and J. Zhu. Head orientation and gaze direction in meetings. In *CHI '02 extended abstracts on Human factors in computing systems*, pages 858–859. ACM Press, 2002.

[100] A. Stolcke, C. Wooters, N. Mirghafori, T. Pirinen, I. Bulyko, D. Gelbart, M. Graciarena, S. Otterson, B. Peskin, and M. Ostendorf. Progress in meeting recognition: The icsi-sri-uw spring 2004 evaluation system. In *Proc. NIST 2004 Meeting Recognition Workshop*, Montreal, 2004.

[101] S. Sumec. Multi-camera automatic video editing. In *Proc. of ICCVG*, Warsaw, Poland, 2004.

[102] A. Tritschler and R. A. Gopinath. Improved speaker segmentation and segments clustering using the bayesian information criterion. In *Proceedings of EUROSPEECH*, pages 679–682, 1999.

[103] D. van Leeuwen and J. Bouten. Results of the 2003 NFI-TNO forensic speaker recognition evaluation. In *Proc. Odyssey 2004 Speaker and Language recognition workshop*, pages 75–82. ISCA, 2004.

[104] R. Vertegaal. *Who is looking at whom*. PhD thesis, University of Twente, September 1998.

[105] A. Waibel, M. Bett, M. Finke, and R. Stiefelhagen. Meeting browser: Tracking and summarizing meetings. In D. E. M. Penrose, editor, *Proceedings of the Broadcast News Transcription and Understanding Workshop*, pages 281–286, Lansdowne, Virginia, February 1998. Morgan Kaufmann.

[106] F. Wallhoff, M. Zobl, and G. Rigoll. Action segmentation and recognition in meeting room scenarios. In *Proc. IEEE ICIP*, Singapore, October 2004.

[107] D. L. Wang. Primitive auditory segregation based on oscillatory correlation. *Cognitive Science*, 20:409–456, 1996.

[108] C. Wooters, N. Mirghafori, A. Stolcke, T. Pirinen, I. Bulyko, M. Graciarena D. Gelbart, S. Otterson, B. Peskin, and M. Ostendorf. The 2004 icsi-sri-uw meeting recognition system. In *Proc. MLMI'04*, Martigny, Switzerland, 2004.

[109] S. Wrigley and G. Brown. Audio-visual source localization and tracking using a network of neural oscillators. In *British Society of Audiology Short Papers Meeting on Experimental Studies of Hearing and Deafness*, University College London, UK, 2004.

[110] B. Zhou and J. H. L. Hansen. Unsupervised audio stream segmentation and clustering via the bayesian information criterion. In *Proceedings of International Conference on Spoken Language Processing*, pages 714–717, Beijing, China, Oct. 2000.

[111] Q. Zhu, B. Chen, N. Morgan, and A. Stolcke. Tandem connectionist feature extraction for conversational speech recognition. In *Proc. MLMI'04*, Martigny, Switzerland, 2004.

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