SPEAKER SEGMENTATION FOR AIR TRAFFIC CONTROL

EEC Note No. 01/2008

Project: INO 2 AT

Issued: April 2008
## Title:

**Speaker Segmentation for Air Traffic Control**

<table>
<thead>
<tr>
<th>Author</th>
<th>Date</th>
<th>Pages</th>
<th>Figures</th>
<th>Tables</th>
<th>Annexes</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>Michael Neffe (1), Horst Hering (2), Gernot Kubin (1)</td>
<td>03/2008</td>
<td>vi + 35</td>
<td>10</td>
<td>2</td>
<td>-</td>
<td>50</td>
</tr>
<tr>
<td>(1) Graz Technical University, Graz, Austria; (2) EUROCONTROL Experimental Centre, Bretigny, France</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Project</th>
<th>Task no. sponsor</th>
<th>Period</th>
</tr>
</thead>
<tbody>
<tr>
<td>INO 2 AT</td>
<td>-</td>
<td>2006-2007</td>
</tr>
</tbody>
</table>

### Distribution Statement:

(a) Controlled by: Marc Bourgois, Head of INS  
(b) Special Limitations: public  
(c) Copy to NTIS:

### Descriptors (keywords):

Air Traffic Control, Safety and Security for VHF Voice Communication, Speaker Segmentation and Verification, Gaussian Mixture Models

### Abstract:

In this report a novel system of speaker segmentation has been designed for improving security of voice communication in air traffic control. While the aircraft identification tag is used to assign speaker turns on the shared communication channel to aircrafts, the main focus of this report is the investigation of a speaker verification system as an add-on attribute to enhance the security level for the air traffic control. After analysis of different speaker classification methods the one with the most promising results in the literature has been implemented. The verification task is performed by training universal background models and speaker dependent models based on the Gaussian mixture model approach. The feature extraction and normalization units are especially optimized to deal with small bandwidth restrictions and very short speaker turns. To enhance the robustness of the verification system, a cross verification unit is furthermore applied. The designed system is tested with the SPEECHDAT-AT and WSJ0 databases to demonstrate its suitability for real-world applications.
This page intentionally left blank
FOREWORD

Since the begin Air Traffic Control (ATC) was based on voice communication between the pilots and a responsible ATC operator. Upcoming data link communication concepts tend to reduce the voice communication acts between pilots and controller. But voice communication will stay the main communication facility between air and ground beyond the 2020 time frame.

The used technical communication standard is basically unchanged since its introduction. This standard doesn’t include any technical security feature for the voice communication. Especially the 9-11 event raised ups the security aspect for air ground communication. The introduction of watermark techniques (Speech Watermarking for ATC, EEC note 2005-05) could help to secure the transmission between the technical communication equipment. But Watermarking techniques could not help to identify the user that is authorised to use the communication equipment. Advanced speaker identification could secure the air ground communication at a high level. Speaker identification requires high quality speech and long speech samples. ATC voice communication doesn’t fulfil these requirements.

Each participant of the air ground voice communication broadcasts its messages. Under the assumption positive speaker identification isn’t required, as we need to indicate a potential intruder only. This problem is simpler and can be done with speaker segmentation techniques. The note describes the research and the development of a software based demonstrator.

Horst Hering
Innovative Studies
This page intentionally left blank
# Speaker Segmentation For Air Traffic Control

**EUROCONTROL**

**INO 2 AT – EEC Note No. 01/2008 vii**

## TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>FOREWORD</td>
<td>V</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>VIII</td>
</tr>
<tr>
<td>1. INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1. OVERVIEW</td>
<td>1</td>
</tr>
<tr>
<td>1.2. FROM SPEAKER SEGMENTATION TO SPEAKER VERIFICATION</td>
<td>2</td>
</tr>
<tr>
<td>2. SPEECH PROCESSING METHODS - AN OVERVIEW</td>
<td>4</td>
</tr>
<tr>
<td>3. RADIO TRANSCEIVER CHARACTERISTICS AND INTER-SPEAKER BEHAVIOURS</td>
<td>6</td>
</tr>
<tr>
<td>3.1. VHF TRANSCEIVER EQUIPMENT AND ITS LIMITATIONS</td>
<td>6</td>
</tr>
<tr>
<td>3.2. INTER-SPEAKER BEHAVIOURS</td>
<td>6</td>
</tr>
<tr>
<td>4. FRONT-END UNIT</td>
<td>7</td>
</tr>
<tr>
<td>4.1. VOICE ACTIVITY DETECTION</td>
<td>7</td>
</tr>
<tr>
<td>4.1.1. Energy-based VAD</td>
<td>7</td>
</tr>
<tr>
<td>4.1.2. Wavelet based VAD</td>
<td>8</td>
</tr>
<tr>
<td>4.1.3. Speech Parameterization</td>
<td>9</td>
</tr>
<tr>
<td>4.2. DYNAMIC INFORMATION</td>
<td>11</td>
</tr>
<tr>
<td>5. SPEAKER CLASSIFICATION METHODS</td>
<td>12</td>
</tr>
<tr>
<td>5.1. TEMPLATE MODELS</td>
<td>12</td>
</tr>
<tr>
<td>5.2. NEAREST NEIGHBOURS</td>
<td>12</td>
</tr>
<tr>
<td>5.3. VECTOR QUANTIZATION (VQ) SOURCE MODELLING</td>
<td>13</td>
</tr>
<tr>
<td>5.4. HIDDEN MARKOV MODELS</td>
<td>13</td>
</tr>
<tr>
<td>5.5. MULTILAYER PERCEPTRONS</td>
<td>14</td>
</tr>
<tr>
<td>5.6. SUPPORT VECTOR MACHINES</td>
<td>14</td>
</tr>
<tr>
<td>5.6.1. Scoring with SVM</td>
<td>15</td>
</tr>
<tr>
<td>5.7. GAUSSIAN MIXTURE MODELS</td>
<td>16</td>
</tr>
<tr>
<td>5.7.1. GMM and Single Speaker Detection</td>
<td>17</td>
</tr>
<tr>
<td>5.7.2. Adapted Gaussian Mixture Modelling Using Universal Background Model</td>
<td>18</td>
</tr>
<tr>
<td>5.7.3. Background Modelling Using the Same Data</td>
<td>18</td>
</tr>
<tr>
<td>5.7.4. Speaker Adaptation</td>
<td>19</td>
</tr>
<tr>
<td>5.7.5. Thresholding</td>
<td>19</td>
</tr>
<tr>
<td>5.7.6. Score Normalization</td>
<td>19</td>
</tr>
<tr>
<td>6. DEVELOPED SYSTEM AND PARAMETER SETTINGS</td>
<td>20</td>
</tr>
<tr>
<td>6.1. FRONT-END PROCESSING</td>
<td>20</td>
</tr>
<tr>
<td>6.1.1. Voice Activity Detection</td>
<td>20</td>
</tr>
<tr>
<td>6.1.2. Feature Extraction and Normalization Unit</td>
<td>20</td>
</tr>
<tr>
<td>6.2. CLASSIFICATION</td>
<td>21</td>
</tr>
<tr>
<td>6.3. CROSS VERIFICATION</td>
<td>21</td>
</tr>
<tr>
<td>7. PERFORMANCE EVALUATION IN SPEAKER VERIFICATION</td>
<td>23</td>
</tr>
<tr>
<td>8. DATABASE AND EXPERIMENTAL SETUP</td>
<td>24</td>
</tr>
</tbody>
</table>
9. RESULTS....................................................................................................................25

10. DEMONSTRATOR......................................................................................................28

11. CONCLUSION ............................................................................................................30

12. REFERENCES............................................................................................................31

LIST OF FIGURES
Figure 1-1: Illustration of the voice communication between pilots and controller in one control sector. ................................................................................................................ 2
Figure 1-2: Cooperation of AIT and SV on the party-line in air-ground voice communication. .............................................................................................................. 3
Figure 2-1: Task definition in the context of speech processing. ................................. 4
Figure 4-1: Example of the effect of the Long-term high-level noise detection method on real air traffic voice communication recordings. .................................................. 8
Figure 4-2: Modular representation of a Filterbank feature extraction unit. Taken from [Bimbot et al., 2003] ..................................................................................................... 9
Figure 4-3: Example of a mel-warped frequency filterbank with 12 channels. Taken from [Slaney, 1998] ........................................................................................................ 10
Figure 5-1: Example of a three state HMM ........................................................................ 13
Figure 6-1: Illustration of designing a SV system in 4 phases ........................................ 20
Figure 6-2: Histogram and fitted Gauss curves for the score distributions of imposters (left) and true speakers (right) .................................................................................. 22
Figure 7-1: Sample DCF run as a function of the threshold. .............................................. 23
Figure 9-1: DET curves with EER point (plus sign and circle) ........................................ 25
Figure 9-2: DET-curve with EER point for the SV system without VAD (NoVad), energy-based (EVad) and WT-based VADs (WaVad) ................................................... 27
Figure 10-1: Screenshots of Demonstrator ....................................................................... 28
Figure 10-2: Screenshots of Demonstrator ........................................................................ 29

LIST OF TABLES
Table 9-1: Performance results as a function of the frame length/frame rate and the number of Gaussian components (#GC) tested with SPEECHDAT-AT database. .... 26
Table 9-2: EER results derived from both databases for different VADs without (wo) and with (w) applying the hangover scheme. .................................................... 27
1. INTRODUCTION

1.1. OVERVIEW

The main task of this report is the investigation of the suitability of applying a speaker verification method (SV) in Air Traffic Control (ATC). The reliability analysis of such a system considering all restrictions and demands arising from ATC are of main interest. To show the applicability of a state-of-the-art SV method a SV system which meets the restrictions of ATC has been developed. In the first part of the work the overall speaker segmentation task in the ATC voice communication is characterized and it is shown how to reduce this problem into a speaker verification task. After classifying the problem in the field of speaker recognition, the front-end preprocessing unit containing voice activity detection, feature extraction and the feature normalization used in the system are described. Using the results of the classification described above, different speaker verification methods are introduced. An exact description of the developed system using two databases is given. This report is concluded with performance studies and a description of a demonstrator.

The air ground voice communication between pilots and air traffic controllers is hardly secured. The goal of this study is the introduction of an additional security level based on biological speech parameters. The Air Traffic Control (ATC) voice communication standards have been defined by international conventions in the forties of the last century. These standards do not address security issues. Illegal phantom communication of "jokers" playing the role of pilot or controller has repeatedly been reported. Voice communications from attackers to achieve terrorist goals are possible. In order to address the raised security demands the introduction of security measures for the air ground voice communication is required. The proposed levels of security have to align with the existing technical communication standards. Beside the technical standards for the physical transmission channel, behavioural rules for the users of the so-called party-line channel are established. Party-line communication means that the communication with all aircrafts in a controlled sector handled by a unique controller takes place in a consecutive manner on a single shared voice channel. Therefore, pilots have to identify themselves in each voice message with the attributed call sign for this flight. The ATC controller uses the same call sign in any reply to identify the destination of the voice message. Addressing messages by call signs requires permanent attentiveness of all party-line users. Call sign confusion represents an important problem in terms of ATC safety. A recent study showed that call sign confusion can be associated with about 5% of the overall ATC related incidents [Gerard, 2004].

In the legacy air ground voice communication system multiple safety and security lacks can be identified. As a consequence, the EUROCONTROL Experimental Centre (EEC) proposed the Aircraft Identification Tag (AIT) [Hering et al., 2003] in 2003, which embeds a digital watermark (e.g., call sign) in the analogue voice signal of a speaker before it is transmitted over the Very High Frequency (VHF) communication channel. AIT represents an add-on technology to the existing VHF transceiver equipment, which remains unchanged. The watermarks are not audible for humans. They represent a digital signature of the originator (pilots or controller) hidden in the received voice message. AIT allows the automatic identification of the origin of the voice communication within the first second of speaking as the schematic illustration in Figure 1-1 shows. As stated previously, spoken call signs are included in each voice message to identify originator or destination of this voice message. Many different reasons like bad technical channel quality, misunderstanding, speaking errors and so on, may make the spoken call sign unserviceable for the destination. This creates supplementary workload and call-sign confusion may affect ATC safety.
AIT will help to overcome this safety issue as it can be used to highlight the speaking aircraft in real-time. In this manner AIT also introduces some basic level of security for the communication layer.

![Figure 1-1: Illustration of the voice communication between pilots and controller in one control sector. The AIT watermarking system allows the identification of the transmitting source at a particular time.](image)

### 1.2. FROM SPEAKER SEGMENTATION TO SPEAKER VERIFICATION

Based on AIT watermarking our research introduces a new security level for the air ground communication channel by using behavioural biometric voice data of the party-line speakers. The idea is that some behavioural biometric characteristics can be extracted from the pilots’ voices and are automatically enrolled when an aircraft registers the first time to a control sector. At any later occurrence of the same AIT signature the new received speaker voice can be compared with the existing enrolled speaker dependent models to verify whether the speaker has changed as proposed in [Neffe et al., 2005]. Recapitulating, the AIT reduces the problem of distinguishing different speakers on the party-line channel known as the speaker segmentation problem to a binary decision problem of claimant vs. imposter speaker. Note that the use of a speaker segmentation system alone can't satisfy the high security demands needed for ATC.

On the one hand, because the AIT watermarking is able to determine the beginning of each speaker turn exactly and assigns it to the corresponding aircraft. On the other hand, for the verification problem the system has to make a binary decision compared to a one-out-of-N decision for the segmentation task, where N is the number of pilots on the party-line in a certain control sector. Moreover, in ATC only the information of a speaker change in one aircraft is relevant. As one can imagine the error rate for such system is higher compared to a verification system.

The enrolled speaker model should be handed over from control sector to control sector to enable a more accurate modelling of speakers with flight progress. This proposed concept secures the up and down link of the ATC voice communication. An illustration of the proposed solutions is shown in Figure 1-2. Before transmitting a voice message the push-to-talk (PTT) button has to be pressed, which introduces a click on the transmission channel. This solution may be considered as a first level of basic security. Using a click detector one may determine the start of each talk spurt. On the party-line unfortunately no information of the transmitting source can be gained with such a method. The AIT as shown on the second level in Figure 1-2 identifies besides the start of each sent voice message also the transmitting source microphone. Based on this framework, the level of security can be improved by embedding a speaker verification system which is depicted as the third level in Figure 1-2.
At the second and third level, the first numerical index determines the aircraft number and the second the speaker. As an example AC22 is not equal to AC21 which is the first speaker assigned to aircraft AC2 when AC2 enters the control sector. Hence AC22 has to be recognised as an imposter.

![Waveform with annotations](image)

**Figure 1-2:** Cooperation of AIT and SV on the party-line in air-ground voice communication. 

TS is the abbreviation for an arbitrary talk spurt, GC for a talk spurt originating from the ground (i.e., ground control) and AC for a talk spurt originating from a certain aircraft.

The first numeric index in level 2 and 3 indicates a specific aircraft and the second index in level 3 the speaker identity. In the first level the nature of the generic speaker segmentation problem is depicted. On level 2 AIT watermarking assigns speaker turns to their source and on level 3, AIT and SV are shown to solve the speaker segmentation task. The waveform at the top shows channel noise between talk spurts.

Air traffic communication can be thought of as a special case of the well-studied meeting scenario [mis, 2006, Macho et al., 2005]. Here, all participants of the meeting communicate over the VHF channel using microphones, whereas the communication protocol is strictly defined. As mentioned before, the communication is highly organized, concurrent speaking is not permitted and no direct conversation between pilots of different aircrafts is allowed, communication is intended only between pilots and controller.
2. SPEECH PROCESSING METHODS - AN OVERVIEW

The focus in this part is to define the problem introduced above as a special field of speech processing. Figure 2-1 shows a schematic representation of the main fields of speech processing. Whereas in this work the part of speaker recognition is of main interest.

Speaker Recognition can be roughly divided into the fields of Single Speaker Utterance and Multi Speaker Stream. In the first area an utterance is known to originate only from one speaker with an unknown identity. For the second area neither the identity nor the utterance boundaries are known. One has to deal with a stream of an unknown number of speakers giving utterances at unknown points in time.

The Single Speaker Utterance can be split to the two well-known areas of speaker identification and speaker verification:

- **Speaker identification**: Given a population of N known speakers, the task of speaker identification is to identify the originator of the current utterance who is known to be one of the population. In this context, one speaks of a closed-set problem. Because the current utterance has to be compared to every speaker of the population the complexity is direct proportional to N. The comparison is done by a distance measure or by computing a probability or likelihood score. The current speech is assigned to that speaker identity of the population with minimum distance or highest probability/likelihood.

- **Speaker verification**: Here a claimant is verified against the speaker model he claims to be. During verification only a binary decision has to be made, e.g., acception or rejection of the claimant. Because in this case no restriction of population size is made one speaks of an open-set problem.

Depending on the chosen method and on the application, both speaker recognition systems can be either text-dependent or text-independent. If the claimant has to utter a text known to the system one speaks of a text-dependent system, otherwise, if there are no restrictions concerning the spoken text the system works in a text-independent mode. In general text-dependent systems have a better performance than the independent ones.
Multi Speaker Stream occur, for example, in meeting scenarios. The stream is a file or recording in which an unknown number of speaker are sequentially or concurrently speaking. The task is to segment the stream into utterances, find the number of speakers and assign each utterance to the correct speaker. This is the most general case and is called Speaker Segmentation. One is speaking of Speaker Detection when the speakers' voices are enrolled (i.e. known to the system) at the beginning.

Based on these definitions the task one has to deal with is a Speaker Segmentation problem. An unknown number of speakers are sharing the ATC party line. Using the AIT the origin of each speaker is known and a reduction of a complex problem to a Speaker Verification problem can be achieved. To be more concrete, it is Speaker Verification without enrolment because with the first utterance obtained when an airplane enters a air traffic control sector, a new speaker model has to be trained. So by definition the reference speaker is modelled from the first utterance at the start of communication between pilot and controller.
3. RADIO TRANSCEIVER CHARACTERISTICS AND INTER-SPEAKER BEHAVIOURS

3.1. VHF TRANSCEIVER EQUIPMENT AND ITS LIMITATIONS

After introducing the ATC security problem, the signal conditioning and its effects for speech quality will be analyzed. Speech quality in ATC is impaired in two ways: Firstly, the speech signal is affected by additive background noise (wind, engines) which is not completely excluded by using close talking microphones. Secondly, there is a quality degradation by the radio transmission system and channel which limits the signal in bandwidth and causes distortions and more noise. The analogue transmission of the speech signal uses the double sideband amplitude modulation (DSB-AM) technique of a sinusoidal carrier. The system is known to have a low quality. This signal is transmitted over a VHF channel with a 8.3 kHz channel spacing. This yields an effective bandwidth of only 2200 Hz in the range of 300-2500Hz [Committee, 2003] for speech transmission. The carrier frequency is within a range from 118 MHz to 137 MHz. Dominating effects which are degrading the transmitted signal are path loss, additive noise, multipath propagation caused by reflections, reflection itself, scattering, absorption and Doppler shifts. A thorough description of the signal degradation caused by the fading channel can be found in [Hofbauer et al., 2006b, Hofbauer et al., 2006a]. In the literature the SV system proposed in [Reynolds et al., 2005] deals with a speech bandwidth close to but not as narrow as the ATC system.

3.2. INTER-SPEAKER BEHAVIOURS

Considering the speaking habits during pilot and controller communication, Hering et al. in [Hering et al., 2003] have shown that one speaker turn is only five seconds in length on average. Training of speaker verification algorithms with speech material of such a short duration is a really challenging task. For comparison Kinnunen in [Kinnunen et al., 2006] used 30 seconds in mean for testing and 119 seconds for training or Chen in [Chen, 2005] uses 40 seconds for training.
4. FRONT-END UNIT

As a first step after recording, the input waveform is processed to be suitable for SV. First, a sequence of features is extracted from the speech signal to get a more compact and less redundant representation of the signal. To reduce distortions of the signal by the environment or channel, normalization techniques are applied on the features after its extraction. Two different feature extraction methods and normalization techniques are introduced and have been tested in the following. To extract only features from speech, the input signal is first fed to the voice activity detection (VAD) unit to separate speech from non-speech. Based on the VAD output, features are extracted and then normalized. In the next section two VAD methods are introduced whereas the one which yields better SV performance has been chosen finally. These processed features are finally used for speaker classification.

4.1. VOICE ACTIVITY DETECTION

A robust VAD is crucial in the SV system in order to extract suitable speaker dependent data. Non-speech data which is dominated by noise of the transmission channel may drive model training into incorrect convergence. This leads to an unreliable SV system. Two VAD algorithms are compared in [Pham et al., 2007]. One algorithm segregates speech from non-speech using energy features and the is based on the wavelet transform (WT).

4.1.1. Energy-based VAD

To segregate speech from non-speech, first the short-term log-energy $E$ is extracted from each frame with a length of 16ms and 8ms frame shift. Based on the quantile method introduced by Stahl et al. in [Stahl et al., 2000] a rough estimate noise level is obtained. There, a hard threshold is determined experimentally by taking a quantile factor in the range $[0,\ldots,0.6]$ as suggested by Pham et al. in [Pham and Kubin, 2005]. Quantile filtering is a generalization on minimum statistics and leads to high VAD performance [Pham et al., 2006]. In addition this results in low complexity because only one parameter, the log-energy, is extracted directly from the signal. By employing a quantile factor of $q=0.4$, which was selected experimentally to achieve high VAD accuracy, we expect also high performance for speaker verification. To smooth the VAD outputs due to fluctuations of non-stationary noise, a duration rule has been applied. In order to adapt our air traffic speech data, the 15ms/200ms rule as reported in [Neffe et al., 2007] has been modified to a 100ms/200ms rule to bridge short voice activity regions, preserving only candidates with a minimal duration of 100ms, and being not more apart than 200ms from each other. This excludes talk spurts shorter than 100ms and relabels pauses smaller than 200ms as speech. We propose in the following a new detection method to distinguish between speech signals and consistently high-level noise which results from the radio transmission channel itself during non-active communication periods.

**Long-Term High-Level Noise Detection:**

To account for long term high-level noise as encountered in air traffic voice communication a new rule has been introduced [Neffe et al., 2007]. The $1^{st}$ difference of the log-energy values $\Delta E$ of all frames which are stored in a buffer of 800~ms are calculated. If the difference between the maximum and minimum values of $\Delta E$ in a buffer is below a predefined threshold $k$ as shown in Equation 1, the segment/buffer is considered as high-level noise and is relabelled as non-speech frame.

$$\max_{i}(\Delta E_i) - \min_{i}(\Delta E_i) < k$$

(1)
The frame index $i$ runs from 1 to the buffer length. Informal experiments showed good performance of long-term noise detection for a threshold of $k=0.002$ on air traffic recordings provided by EEC [Hering, 2006]. The buffer update rate has been set to 80ms. In Figure 4-1 the effect of this method is shown on a true air traffic voice communication recording. The buffer window containing the values of $\Delta E$ slides over the whole signal. As one can easily recognize, the region between second 4.25 and second 7.8 contains only channel-noise. In these frames $\Delta E$ depicted as dashed line in Figure 4-1 is almost consistent and the difference between $\text{max}(\Delta E)$ and $\text{min}(\Delta E)$ is smaller than the predefined threshold in this specified region. This whole section is going to be relabelled as non-speech after applying this rule.

![Figure 4-1: Example of the effect of the Long-term high-level noise detection method on real air traffic voice communication recordings.](image)

The time-domain speech signal is depicted as reference, the solid line shows the detected speech of the VAD inside the recording and finally the dashed one the 1st discrete derivative of the log-energy $\Delta E$. The VAD outputs are shown (a) before and (b) after applying the consistent noise detector which relabels some speech frames as long-term noise duration.

### 4.1.2. Wavelet based VAD

A more robust VAD is developed from a WT-based method for phonetic classification [Pham and Kubin, 2006] with a novel adaptive quantile filtering method. The VAD has been adapted and extended for the front-end processing of the SV system [Pham et al., 2007] which is designed for ATC. First, the WT at the 2nd scale is applied on every windowed speech frame of 32ms frame length and 8ms frame rate. Then a frame-based delta feature is extracted from the obtained wavelet coefficients. To be robust against noise, the extracted feature is enhanced by applying a sigmoid function and median filtering. A noise threshold, derived by quantile filtering [Pham and Kubin, 2006], is further improved by an adaptive estimation of the quantile factor. The speech/non-speech decision is done by comparing the feature values with the adaptive noise threshold.

For smoothing fluctuations resulting from strong non-stationary noise in the VAD outputs, this VAD also uses the above bridging rule. To distinguish between speech signals and consistently high-level noise which results from the transmission channel itself during non-active communication periods, the long-term high-level noise detection has been adapted for the WT-based VAD.
4.1.3. Speech Parameterization

The aim of speech parameterization is to transform a speech signal in a set of feature vectors to yield a more compact, less redundant representation of the speech. Another advantage is the suitability of this representation for statistical modelling. Most of the speech and speaker recognition systems use the cepstral representation of speech. For speaker recognition a lot of efforts have been made to extract features from speech, which correspond directly to the biometrics of a speaker, e.g. the speaker’s anatomy. In this context the biometric hypothesis is named which states that each individual has physical characteristics that distinguishes him or herself from [Souza and Souza, 2001, Naik, 1990].

Filterbank Based Cepstral Parameters:

In Figure 4-2 a modular representation of a filterbank based feature extraction unit is shown. First the speech is pre-emphasized, a filter is applied to the signal which enhances higher frequencies of the spectrum as follows:

\[ H(z) = 1 - az^{-1}, \]  

where the value of \( a \) is chosen between \([0.95, \ldots, 0.98]\) in most applications. The decision whether to use the pre-emphasize or not relies on empirical experiments and hence is application dependent.

The speech signal is a slowly time-varying signal, i.e., a stationary signal if analyzed over a sufficiently short period of time. In speech processing short-time spectral analysis is most common to characterize the signal [Bimbot et al., 2003], where frames of 10 to 32msec in duration are analyzed separately. On each frame a window is applied to taper the signal towards the frame boundaries. Usually a Hamming or a Hann window is used. In general every 10msec a new frame is analyzed. Afterwards the windowed signals are Fourier transformed to get from the time to the frequency representation of the signal. Because one is only interested in a smooth version of the spectrum a filterbank is applied to the FFT magnitudes. Where the absolute value of the frequency bins are convolved by a low-pass to get a representative average value of the signal in each frequency band of the FFT-filterbank. The pur filter response of a filterbank can have a triangular or other shapes and can be differently located on the frequency scale. Some examples are a linear, mel or bark warped scale for the filter spacing. The mel scale is based on the human ear's critical bandwidth. The central frequencies of the mel-scaled filters using the linear scale are given by:

\[ f_{mel} = 1000 \log_{10}(1 + f_{lin}/1000) \]

\[ \log_{10} 2 \]
In Figure 4-3 an example of a mel-warped frequency filterbank is shown.

Figure 4-3: Example of a mel-warped frequency filterbank with 12 channels. Taken from [Slaney, 1998]

Optionally one can take the logarithm of the filterbank output to obtain a smooth run of the average of the spectral envelope in dB. Finally the cepstral vectors are computed using the discrete cosine transform (DCT) which yields also an orthogonalization of the feature vectors:

\[
c = \log_{10}(S_k) \cos\left(k\left(n + \frac{1}{2}\right)\frac{\pi}{K}\right), \quad n = 1, 2, \ldots, K
\]

with \(S_k\) the \(k^{th}\) channel of the filterbank output.

**Feature Normalization:**

Feature normalization is performed to reduce the negative effect of additive noise and/or channel distortions in speaker and speech recognition. These distortions are real-world effects and are affecting the signal when transmitted over telephone lines, Very High Frequency (VHF) voice communication channels, or in a (reverberant) room [Chen et al., 2002, Bimbot et al., 2003]. Normalization can be performed at three different stages of a SV system. The first tries to normalize directly the features, the second normalization is carried out in the model space, hence model parameters are adjusted to better fit the data. Finally normalization can be done in the scoring stage. Here known imposter speakers, taken out from a separate pool of speakers, are used to estimate the score distribution (mean and variance) of a certain reference speaker which finally is used to normalize the score.

**Mean and Variance Normalization**

To remove additive slowly-varying noise and time-invariant influences of a channel, cepstral mean subtraction is applied to each feature vector separately. In variance normalization the variance of each cepstral coefficient is normalized to unity separately.
Histogram Equalization

Another possibility to compensate mismatched conditions in the feature space caused by speech acquired under real-world conditions is histogram equalization (HEQ) [Skosan and Mashao, 2006, Dharanipragada and Padmanabhan, 2000]. It is a nonlinear feature transformation technique which efficiently adapts an input cumulative distribution function to a target cumulative distribution function. The HEQ belongs to the so called "feature space transformations" and has the ability not only to compensate for the first and second moment but all the other moments too, of the speaker's feature distribution. Mathematically, the nonlinear transform maximizes the likelihood of the transformed input data given a target distribution function. This is the same than minimizing the Kullback-Leibler distance between the input and target densities, which is a function of the transformation $f$:

$$L(f) + D(p_z\|p_0) = \int p_z(z) \log p_z(z) \, dz - \int p_z(z) \log p_0(z) \, dz \quad (5)$$

The goal is to find $f^*$ that minimizes $L(f)$. Where $z$ is the input random variable with a probability distribution $p_z$. Furthermore the target probability distribution is given by $p_0$.

Because in a practical implementation only a finite number of observations are available from now on the cumulative probabilities is replaced by the cumulative histograms. Finally the transform must be selected such that the cumulative histogram of the transformed data match the target cumulative histogram [de la Torre et al., 2005].

Another example for a feature space transformation is the feature warping [Skosan and Mashao, 2006] and in contrast a widely used model space transformation technique is the Maximum Likelihood Linear Regression (MLLR) method. However this method is computationally more expensive and as reported in [Gales, 1997, Gopinath, 1998] results in almost the same performance comparing two Gaussian Mixture Model systems.

4.2. DYNAMIC INFORMATION

Dynamic Information of features have the ability to give information how a feature sequence evolves over time. The classic way to include this information is to calculate the $\Delta$ and $\Delta\Delta$ parameters, the slope and the curvature respectively. These are polynomial approximations of the first and second derivatives [Furui, 1981, Magrin-Chagnolleau et al., 2001, Bimbot et al., 2003].

After having introduced the front-end processing unit, in the next section classification methods used in SV are described.
5. SPEAKER CLASSIFICATION METHODS

Almost all speaker recognition systems represent the speech signal in terms of feature vectors as described in sec. 4. With these features every further measurement or classification is done.

5.1. TEMPLATE MODELS

A template is a time-ordered sequence of feature vectors of an utterance. An utterance can be a single spoken phoneme or even can contain spoken multi-words [Naik, 1990]. In a template matching scheme, an input utterance template is compared to a reference template. Because two utterances of the same text have different durations in general, the observation is assumed to be an imperfect replica of the reference template [Campbell, 1997]. Thus, for comparison of these two templates, the feature vectors have to be aligned. A well established method to compensate speaking-rate variability by using a nonlinear time compression/expansion function is known from dynamic programming as Dynamic Time Warping (DTW) [Sakoe and Chiba, 1978]. This method performs a constraint, piece-wise linear mapping of time axis to align the two signals and tries to minimize a distance between them. With \( R = \{r_1, r_2, \ldots, r_T\} \) representing the reference template and \( U = \{u_1, u_2, \ldots, u_T\} \) the input template and \( t = \{1, \ldots, T\} \) the time index, different distance measures can be defined:

\[
d(u(t), r(t)) = (u(t) - r(t))^T W (u(t) - r(t))
\]

where \( W \) is the weighting matrix. The Euclidean distance is obtained if \( W \) is the identity matrix and for example the Mahalanobis Distance [Campbell, 1997] can be calculated if \( W \) corresponds to the inverse covariance matrix. The template model is a very intuitive and a sample deterministic method.

5.2. NEAREST NEIGHBOURS

The nearest neighbor (NN) method performs a distance based classification for SV. The benefit of this method is the direct distance computation based on the feature sequence to be verified. Hence there are no models or model parameters to be estimated. This results in the drawback of computational complexity with increasing input data length. The NN method was introduced by [Higgins et al., 1993] and compares directly the input utterance with a reference utterance. Basically the nearest of each of the frames of the input utterance is computed with respect to the nearest frame of the reference utterance:

\[
d(U, R) = \frac{1}{|U|} \sum_{u_i \in U} \min_{r_j \in R} |u_i - r_j|^2 \\
+ \frac{1}{|R|} \sum_{r_j \in R} \min_{u_i \in U} |u_i - r_j|^2 \\
- \frac{1}{|U|} \sum_{u_i \in U} \min_{r_j \in R} |u_i - u_j|^2 \\
- \frac{1}{|R|} \sum_{r_j \in R} \min_{r_j \in R} |r_j - r_j|^2
\]

where \( u_i \) and \( r_j \) are the feature frames belonging to \( U \) and \( R \) respectively. To reduce computational complexity, Higgins et al. propose to set a minimum distance threshold to a value which equals the average of the cross-entropy between two utterances of the same speaker.

Whereas the first and second term of Eq. 7 are referred to as cross-entropy terms.
5.3. VECTOR QUANTIZATION (VQ) SOURCE MODELLING

After the initial data reduction obtained through feature extraction, in VQ the feature vectors are represented by an even smaller data set. This is done by a clustering procedure where each feature vector cluster is represented by only one vector in the centre of the cluster [Campbell, 1997, Kremsl, 1994]. A VQ codebook is built by using all vectors from one speaker. In the testing phase the match score is the distance of the claimant feature vectors compared to one speaker codebook in case of SV. After calculating the distance of each vector in the codebook and summing up all distances the minimum distance over all speaker codebooks is the quantity one is looking for. For T frames the match score is:

$$z = \sum_{t=1}^{T} \min d(r, u)$$  \hspace{1cm} (8)

where C is the corresponding VQ codebook. In SV the minimum distance finally is compared to a threshold to make a decision whether or not to accept an input utterance as evidence of the claimed speaker identity. In the clustering procedure the temporal information is averaged out. Hence no time alignment is necessary. However, temporal speaker-dependent information is lost. In [Xin-yi et al., 2004] other methods to compute the minimum distance are proposed is the Euclidean distance, weighted Euclidean distance, Itakuro distance or likelihood rate. The generalized Lloyd algorithm (GLA) in [Xin-yi et al., 2004] is used for training and is said to emphasize speaker dependent information. VQ can be used for both text-dependent and text-independent speaker recognition.

5.4. HIDDEN MARKOV MODELS

The Hidden Markov Model (HMM) is a statistical method and can only be used for text-prompted speaker recognition because also time dependent information is modelled. Originally this method was used for speech recognition. In this section only a rough overview to get an idea of this method is given. For the basic concept see [Rabiner, 1989]. HMMs are known to model feature sequences and each of these states correspond to observable events (e.g. phonemes). The output of one state is not random and so not flexible enough, thus in HMM the observations are probabilistic functions of the output of another state. To be more precise the model is a double embedded process where the underlying process is not directly observable but only by another stochastic process which outputs a probabilistic sequence. Now the probability density function (pdf) of a feature vector $u$ being in state $s$, is given by $p(u|s)$.

Furthermore all states are connected with each other by a transition network with corresponding transition probabilities $a_{ij} = p(s_i|s_j)$ like depicted for a three state left to right HMM in Figure 7. A speaker dependent model is trained by text-prompted utterances of the speaker. The match score for a given input feature vector $u$ of length $T$ is given by the likelihood of the input vector given the model [Campbell, 1997]:

$$p(u|\text{model}) = \sum_{\text{all state sequence}} \prod_{t=1}^{T} p(u_t|s_t) p(s_t|s_{t-1})$$  \hspace{1cm} (9)

In practice HMM text-dependent speaker recognition systems have shown to outperform most other systems.
5.5. MULTILAYER PERCEPTRONS

Multilayer Perceptrons (MLP) belong to the family of neural networks with the restriction to be a feed forward network only, in contrast to more general recurrent networks. Neural networks can be used as universal approximators. They try to model oneself on neurons of the human brain. One so called perceptron in MLP can have multiple inputs but only one single output. Several perceptrons in parallel having the same inputs are called layer. A MLP can exist of multiple (sequentially arranged) layers where some of them can be hidden. The activation of one perceptron in general depends not only on a single input but on all inputs. However, activation is dependent on a certain activation function:

\[ y_n = \phi \left( \sum_{k=1}^{K} \omega_k x_k \right) \]  

with \( k = [1, ..., K] \) different input features. It is an open issue on how to determine the numbers of perceptrons, layers and the activation function for a certain problem to solve, as for example, for the SV problem. Given the setup i.e. the number of layers and perceptrons, the node functions and the corresponding weights of their output have to be trained. The error back propagation (EBP) algorithm is normally used for learning models (i.e., estimating weights and node functions), with the limitation of poor learning speed.

In [Lee and Choi, 2004] a SV system is proposed using cohort background speaker models on the one hand for speaker model adaptation and on the other hand in the verification process (scoring) to have a strict criterion. Lee et al. report in [Lee and Choi, 2004] the usage of two algorithms to overcome the drawback mentioned above and hence make MLP suitable for real time SV. The first exploits redundancy of the training data and achieves a good speed up in learning speaker models. This algorithm is called omitting patterns in instant learning (OIL). The second is called discriminative cohort speakers (DCS) and aims to select the best fitting background speakers related to the enrolling speaker for discriminant learning of MLPs. This SV system isolates words from input utterances and classifies the words into nine streams of Korean continuants (vowels and semi-vowels) and finally learns speaker models for each continuant using the correspondingly number of MLPs. Furthermore one hidden layer is used. Since the system uses continuants the underlying density exhibits a mono-modal distribution. Compared to Gaussian mixtures this system is faster but slower when comparing directly with only one Gaussian which also is a mono-modal distribution.

This system achieves promising results under clean conditions in first classifying the continuants of digits used for test and than using the information for speaker enrolment. Open issues are if such a simple model also can be used in text-independent mode, if the classification works well in bandlimited noisy environments and finally changing the language e.g., to English and using vowels and semi-vowels would yield in reliable performance. If it would work, a benefit would be the small amount of data necessary for training (3 - 3.5 sec.) and verification (~1.5 sec.). With the open issue of real-time applicability for longer utterances in terms of computationally complexity.

5.6. SUPPORT VECTOR MACHINES

Recently also support vector machines (SVM) have been found for speaker recognition. SVM are especially well suited for SV due to its binary nature of decision. The classifier has to make the decision whether or not to except a claimant. Next the basics of SVM are described and then one possible adaptation to SV is introduced.
SVM are classifiers based on the principle of structural risk minimization [Wan and Campbell, 2000]. As said before the SVM is a binary classifier which makes its decision by constructing a boundary or hyperplane that tries to separate two classes. Such a hyperplane is defined by \( x\omega + b = 0 \) [Burges, 1998] with \( \omega \) the normal vector to the hyperplane. For a given linear separable training set \( \{x_t,y_t\}, \; t = \{1,...,T\}, \; \{x_i,y_i\}, \; y_i \in \{-1,1\} \), \( x_i \in \mathbb{R}^d \) the goal is to find a plane separating the data sets, i.e. to find the plane with maximum Euclidean distance of data points to each side of the hyperplane. The support vector algorithm tries to find the largest separating hyperplane margins. The found optimum hyperplane is a linear combination of a small set of vectors, the support vectors. These located vectors satisfy the equality \( y_i(x\omega + b) - 1 = 0 \).

If the data are not linear separable the above equation cannot be solved. To overcome this problem a slack variable is introduced. This variable should enable that also outlying points are allowed in the inequality. The task is now to minimize an error cost function to find again a hyperplane but with less restrictions.

The extension to that are non-linear boundaries of the hyperplane which in general are achieved by using kernels with certain restrictions. Two possible functions to represent the kernels are namely the radial basis functions (RBF) or polynomial kernels.

To use SVM in speaker recognition with almost the same performance than other methods some normalization and for reason of complexity data splitting has to be done. The data normalization has to be carried out because for data in a bad range of values the optimizer may find a non-satisfactory solution. So an inherent scaling of the data between zero and one in the kernel function is carried out. With this setup in [Wan and Campbell, 2000] for clean speech good results have been achieved. In general it is said that SVM need less data for training than other methods. One disadvantage of this method might be that the enrolment of a reference speaker and an adequate number of imposters need to be processed in parallel to define the right hyperplane.

### 5.6.1. Scoring with SVM

The match score for an utterance is calculated by taking the arithmetic mean of the activation of the SVM, defined as:

\[
x\omega + b
\]  \hspace{1cm} (11)

whereas the classification is given by:

\[
\text{sign}(x\omega + b)
\]  \hspace{1cm} (12)

Supposing after training the reference speaker has a positive score and imposters have negative scores, the scoring of an utterance of length \( T \) can be defined by:

\[
S = \frac{1}{T} \sum_{t=1}^{T} (x_t\omega + b)
\]  \hspace{1cm} (13)

where the Kernel function is included in \( w \). The match score can be taken either from the activation or from the classification. But in general better results are achieved by scoring with classification because of the inherent weighting by the score. Finally a threshold for verification has to be set. Below this threshold the claimant is rejected otherwise accepted.
5.7. GAUSSIAN MIXTURE MODELS

In the last years the Gaussian Mixture Models (GMM) have been shown to be very well suited for text-independent speaker recognition. Moreover, this approach has become the most dominant in this field and furthermore has achieved state-of-the-art performance [Reynolds and Rose, 1995, Bimbot et al., 2000, Kinnunen et al., 2006]. The GMMs have the advantage to be both parametric and non-parametric. The approach is parametric in the way that the behaviour of the densities can be easily changed by adjusting its parameters and non-parametric in the ability to model arbitrary density distributions. Because of the ability to model multivariate densities the GMM approach is well suited for text-independent speaker recognition. The GMM approach has shown good results in both speaker verification and identification tasks. This method belongs to the group of statistical method and is used for pattern recognition. In pattern recognition a class is typically represented by a probability density function (pdf) of a certain feature vector. Now the most challenging task is finding a proper pdf estimate which has a crucial impact on successful recognition. That’s why for more complex pdfs the GMM method is used. In the next section the basic formulas for the GMM approach are derived. Given a D-dimensional feature $x$ the pdf for the GMM is defined as:

$$ p(x|\lambda) = \sum_{c=1}^{C} w_c p_c(x) $$  \hspace{1cm} (14)

As one can see in the above equation the GMM pdf is defined as a weighted sum of unimodal Gaussian densities where $p_c(x)$ can be defined as below and the weights satisfy the constraint $\sum_{c=1}^{C} w_c = 1$.

$$ p(x) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} e^{-1/2(x-\mu)^T \Sigma^{-1} (x-\mu))} \hspace{1cm} (15)$$

where the parameter $\mu$ is the Dx1 mean vector, and $\Sigma$ the DxD covariance matrix. For clarity $w_c$, $\mu_c$ and $\Sigma_c$ are represented by $\lambda = \{w_c, \mu_c, \Sigma_c\}$. In general the covariance matrix which defines the shape of the Gaussian can be chosen as full, block diagonal or diagonal matrix. In SV the diagonal matrix is preferred firstly because the inversion of a diagonal matrix is computational less complex compared to full ones and secondly the density modelling of $M^{th}$ order of full covariance matrix can be achieved by a larger order with diagonal covariances. Furthermore in [Bimbot et al., 2000] a better performance for diagonal than for full covariance matrices is reported. This could be due to the fact that for insufficient training data the covariance matrix can not be reliable estimated.

For a given set of training data $X = \{x_1, x_2, ..., x_T\}$, for $t = [1, ..., T]$ the time index, a model in our case, a speaker dependent model, can be estimated by using the iterative expectation maximization (EM) algorithm as defined for example in [Paalanen et al., 2005]. After initializing a model which can be done by the kmeans algorithm or randomly (i.e. initial guess) the EM algorithm iteratively refines the GMM parameters and it guarantees to monotonically increase the likelihood of the estimated model given the training data. However, it is not guaranteed that the EM algorithm converges to the global maximum, hence a first good initial guess is crucial to find the global instead of the local maximum of a distribution.

For a given independent feature sequence like defined above the likelihood function is computed as:

$$ p(X|\lambda) = \prod_{t=1}^{T} p(x_t|\lambda) \hspace{1cm} (16)$$
This function tells about the likelihood of the data given the parameter of the distribution. Moreover this is the function to be maximized in the training. Usually this is done by taking the logarithm and one is than speaking of the log-likelihood function

\[ \log p(X|\lambda) = \log \sum_{i=1}^{T} p(x_i|\lambda) \]  

(17)

In speaker recognition good recognition results have been achieved by normalising the log-likelihood by the length of the input feature sequence. This normalisation has no theoretical basis and can only be explained such that the likelihood is not really a probability and that the assumption of independence of the features used to calculate the likelihood is also not entirely fulfilled. For increasing length of an input utterence the score would decrease no matter whether or not the utterance was given by the reference speaker or an imposter. In order to make two utterances of different length comparable they have to be normalized to unknown unit length.

5.7.1. GMM and Single Speaker Detection

In this section different methods using GMMs for Single Speaker Detection are introduced. As described in [Bimbot et al., 2003] the single-speaker detection can be thought as a hypothesis test between two hypothesis given an input feature vector \( U \):

- \( H_0 \): \( U \) is from the hypothesized speaker model \( S \).
- \( H_1 \): \( U \) is not from the hypothesized speaker model \( S \).

By taking the ratio of these two quantities, one can carry out a hypothesis test called likelihood ratio:

\[
\frac{p(U|H_0)}{p(U|H_1)} > \theta, \text{accept } H_0, \\
\frac{p(U|H_1)}{p(U|H_0)} < \theta, \text{accept } H_1,
\]

(18)

where \( p(U|H_0) \) is the likelihood of the hypothesis \( H_0 \) given the feature \( U \). In general \( H_0 \) is the speaker model and \( H_1 \) is considered as the anti-speaker model. This has to be done because the likelihood function is unlike a \textit{a posteriori} probability an unnormalised quantity. To calculate the \textit{a posteriori} probability the prior probability has to be know which in general is not the case in speaker recognition systems. The goal is to design a speaker recognition system which is optimum in testing the hypothesis in Eq. 18. Let \( \lambda_{spk} \) be the speaker dependent model trained using training data of only one speaker and represent the hypothesis \( H_0 \). In contrast \( \lambda_{aSpk} \) is assumed to be the anti speaker model and represents the alternative hypothesis needed for normalization. Using this definitions the likelihood ratio can be introduced as \( \sum_{t=1}^{T} p(u_t|\lambda_{spk})/\sum_{t=1}^{T} p(u_t|\lambda_{aSpk}) \). In practice instead of the likelihood often the log-likelihood is used:

\[
g(x) = \sum_{t=1}^{T} \log p(u_t|\lambda_{spk}) - \sum_{t=1}^{T} \log p(u_t|\lambda_{aSpk})
\]

(19)

After this definition and the knowledge of training a speaker dependent model the definition of the anti-model is now considered. First of all the anti hypothesis is not at all as straight forward defined as \( H_0 \) is, which yields in several solutions. In general always a set of \( N \) speakers is used for \( H_1 \) and is represented as:

\[
p(X|\lambda_{aSpk}) = f(p(x|\lambda_1), p(x|\lambda_2),..., p(x|\lambda_N))
\]

(20)

where this set of speaker in the most general case should represent the entire space of possible anti speakers. \( f(\cdot) \) is some weighting function to calculate the overall likelihood. Another approach is first to pool speech from various speakers and training a so called background model which
corresponds to H1. Research has shown that the best results have been achieved by using speaker specific background sets for each speaker [Bimbot et al., 2003]. After introducing different methods to design the background or normalization models one can consider this problem from the point of GMM model space. For a given GMM model space each Gaussian component can be viewed as a point in this space. Therefore a speaker space is defined as the space the GMM, trained from the speakers speech, occupies. According to this occupied space different neighbourhoods can be defined [Tran and Sharma, 2004]:

- Zero neighbourhood: the speaker space itself.
- Tight neighbourhood: the space in the immediate vicinity of the speaker space.
- Medium neighbourhood: a medium size neighbourhood surrounding the speaker space. This neighbourhood includes possibly all competing models close to the speaker space.
- Large neighbourhood: space which should cover all imposter models.
- Infinity neighbourhood: this space covers the entire model space, which also represents non-speech events.

Based on the neighbourhood definition the design of a background model can be viewed as a certain neighbourhood tried to model. Depending on the data the system is designed for, different numbers of speakers, speech lengths of speakers, age and gender distribution have to be chosen. In the next section two tested approaches are introduced and performance results are given in sec. 9.

5.7.2. Adapted Gaussian Mixture Modelling Using Universal Background Model

The adaptation of the speaker dependent model from a universal background model (UBM) trained offline has achieved the best performance over the last years [Reynolds and Rose, 1995]. For training an UBM two methods are commonly used. One is to merely pool all the speech data and train one UBM model. Where one should keep caution that the subpopulations like gender, microphone or transmission channel of the data are well-balanced.

The other is to train each subpopulation separately and then pool the models. This pooling can be a supervised weighting to yield a defined balance. If there is an a priori knowledge of some subpopulation the model can be tested only against this speech. In state of the art systems using UBMs, 512 to 2048 Gaussian components are used for the modelling.

5.7.3. Background Modelling Using the Same Data

This method was first proposed by Tran and Sharma [Tran and Sharma, 2004] in 2004. As mentioned in the title of this section both the speaker dependent and the background model are build by the same speaker data. The difference of the model can be found in the resolution (i.e. the number of Gaussian components) of the models. The background model uses a smaller number of Gaussian components compared to the speaker model. The design can be seen as a tight neighbourhood modelling. One can expect two advantages using this method: Firstly there is no acoustic mismatch of the data. The influences of the environment, the transmission channel and transducer to capture the signal (microphone) are the same. Secondly it is computationally less demanding than other background methods which have to be fed by a large data base. However, the drawback for training a model with only short durations of speech could be the phonetic influence on the models. Hence, because of the lack of statistics the speakers space might not be included in the modelling of the utterance.
5.7.4. Speaker Adaptation

As announced previously, the speaker dependent model is derived from the UBM model by MAP (Maximum a posteriori) estimation. This MAP estimation in practice is done by using the EM algorithm. Using the UBM model and the speaker dependent data the UBM can be adapted to the data. The number of iteration carried out to yield the model is a design parameter and has influence on the performance. Beside the number of iteration used for adaptation, the question whether to update $\mu$, $\sigma$ and priors or only some of them is sufficient, should be considered. Here several strategies do exist. Further information can be found in [Reynolds and Rose, 1995, Bimbot et al., 2000].

5.7.5. Thresholding

After composing a model corresponding to the hypothesis $H_0$ and one corresponding to $H_1$ the objective is to determine a threshold $\Theta$ to decide whether or not an input feature sequence is from the speaker model [Duda et al., 2001]. Using Bayesian decision theory the threshold can be determined by the conditional risk for two category classification:

$$
\frac{p(x|\lambda_{spk})}{p(x|\lambda_{aSpk})} > \frac{\beta_{12} - \beta_{22}}{\beta_{21} - \beta_{11}} \frac{P(spk)}{P(aSpk)}
$$

with $P(i)$ the prior probabilities which in general are not known and $\beta_{ki} = \beta(\alpha_k|\lambda_i)$ the loss function. The loss function is defined as the loss associated with taking action $\alpha_k$ for $k = 1, 2$ assumed to know the true state $\lambda_i$. For a given $\beta_{ki}$ the threshold is independent of the feature vector and is determined by the priors only, which are not known. This equation is only valid under the assumption that $\beta_{21} > \beta_{11}$. For the tested database this evaluation can be carried out for different values of the prior probabilities. Fixing a value for the threshold one can influence the false acceptance and false rejection rate.

5.7.6. Score Normalization

Recently some score normalization techniques were introduced to cancel out effects arising from the mismatch of handsets (HNORM), the mismatch of training and test data (TNORM) or to normalise the score distribution (ZNORM). Summaries on normalization technique can be found for example in [Auckenthaler et al., 2000, Barras and Gauvain, 2003]. Because in our problem changes in the handset type is not likely the HNORM is not considered. The Zero-Normalization (ZNORM) technique attempts to normalise the score distribution. Here the speaker model is compared to example imposter utterances. The mean and variance of these scores are used for normalization as follows:

$$
S = \frac{\log P(x|\lambda_{spk}) - \mu_i}{\sigma_i}
$$

with $\mu_i$ and $\sigma_i$ are the estimated mean and variance of the imposter score distribution.

The Test-Normalization (TNORM) relys also on the estimation of the mean and variance for normalisation. The idea in TNORM is to calculate the scores of the test utterance given a set of imposter models. Again mean and variance are estimated and than the score is normalised based on Eq. 22. One benefit among other normalisation techniques is that for the estimation the same utterance is used and an acoustic mismatch is avoided.
6. DEVELOPED SYSTEM AND PARAMETER SETTINGS

In this section the implemented system with its classification method and parameter settings is introduced. Based on the front-end processing unit speaker classification is performed. The basic design of this SV system consists of four phases as shown in Figure 6-1. In phase 1, gender dependent UBMs are trained. This step is carried out because of the lack of data to train speaker dependent models necessary to get independent to phonetic influences contained in the speech. These models are used in phase 2 for speaker dependent modelling using gender information from gender recognition. Retraining of a speaker model is performed in phase 3 and finally in phase 4 the verification task is done.

Figure 6-1: Illustration of designing a SV system in 4 phases.

6.1. FRONT-END PROCESSING

The input speech signal is first fed to the VAD unit to divide speech from non-speech one. Based on the VAD output, features are extracted and then normalized to reduce influences not arising from the speaker shown in the dashed surrounded box in Fig. 8. These processed features are finally used for speaker classification.

6.1.1. Voice Activity Detection

As voice activity detection both the energy based and the WT-based methods have been tested. As shown in sec. 9 the WT-based methods outperforms the energy-based one in terms of SV performance sec. 7.

6.1.2. Feature Extraction and Normalization Unit

Before the sequence of features will be extracted, mean subtraction and amplitude normalization of the input speech signal is performed. For each speech segment detected by the VAD features are extracted separately. This is necessary to avoid artificial discontinuities when concatenating speech frames. 14 cepstral coefficients are extracted using a linear frequency, triangular shaped filterbank with 23 channels between 300 Hz and 2500 Hz for each frame.
Finally the whole feature set comprises these cepstral coefficients calculated in dB and the polynomial approximation of its first and second derivatives [Bimbot et al., 2003]. Altogether 42 features per frame are used. Performance results using this feature setup but for different frame lengths and frame rates are listed in section 9.

Experimental results showed that HEQ outperforms the commonly used mean and variance normalization technique [Bimbot et al., 2003]. Our implemented HEQ method maps an input cumulative histogram distribution onto a Gaussian target distribution. The cumulative histogram distribution is calculated by sorting the input feature distribution into 50 bins. This number has been selected that small to get sufficient statistical reliability in each bin.

6.2. CLASSIFICATION

Here we use the GMM-UBM approach first introduced by [Bimbot et al., 2000]. In contrast to other GMM-UBM SV systems [Bimbot et al., 2003] we decided to train gender dependent UBMs which are finally not merged to one global UBM. Because of the lack of training data and the higher computational complexity, diagonal covariance matrices instead of full ones have been taken. For training the UBM, the basic model has been initialized randomly and then trained in a consecutive manner by the speech data using maximum a posteriori (MAP) adaptation. For retraining of the model to yield the final gender dependent UBM, we used three EM - steps and a weighting factor directly proportional to the ratio of the total speech length used so far for training and the new utterance length, the model is going to be retrained too. This is done in phase 1 as shown in Fig. 8. To form a SDM, first the log-likelihood of each gender dependent model given the input data is calculated. The gender is determined by selecting the gender-model with the higher score:

$$G_e = \arg \max (L(X, \lambda_{UBM_f}), L(X, \lambda_{UBM_m}))$$

where $L(X, \lambda)$ is the log-likelihood of the model $\lambda$ given the data $X$ with $f$ and $m$ the female and male UBMs respectively. The corresponding gender dependent UBM is used to adapt a SDM as shown for phase 2. For speaker adaptation three EM - steps and a weighting factor of 0.6 for the adapted model and correspondingly 0.4 for the UBM is used to merge these models to the final SDM. In phase 3 further adaptation of the SDM with new data is done by retraining the model as described for the UBM retraining. The score $S(X)$ which is used for verification in phase 4 is calculated by comparing the hypothesized speaker namely the speaker model $\lambda_{spk}$ with its anti-hypothesis the gender UBM $\lambda_{UBM_G}$:

$$S(X) = L(X|\lambda_{spk}) - L(X|\lambda_{UBM_G})$$

Finally the cross-verification unit is applied as defined in 6.3 to further improve SV performance.

6.3. CROSS VERIFICATION

Alternatively to meet the high security expectations in ATC voice communication a cross verification unit is applied as add-on. If an utterance is shorter than a predefined minimum length (i.e., 8 seconds) and the score is not confident enough (positive or negative) the system waits for another utterance and conducts a cross verification. To explain the meaning of cross verification let $X_1$ and $X_2$ be the feature vectors of the first and second utterance to be investigated, respectively. Furthermore $\lambda_1$ and $\lambda_2$ are the adapted speaker models. If the following equation is satisfied.

$$S_{\lambda_1}(X_1) \cap S_{\lambda_2}(X_2) > \rho$$

i.e., both scores are above a threshold &x03C1; and are verified to be from the same gender as defined in Eq. 23, then it is assumed that both utterances are from the same person and thus are concatenated.
This method shows to increase the robustness of the verification system. Two wrong behaviours may occur in the cross verification unit. Firstly, if two utterances from one speaker are not determined to be uttered from the same speaker, the two utterances are not concatenated and hence are not used together for verification. Furthermore the overall performance stays the same. Secondly, if two utterances from different speakers occupy the same model space e.g., have almost the same statistical properties they are verified to be uttered by one speaker and are hence concatenated. But this leaves the score almost the same. Using this method the performance increases because by concatenating two utterances more data for verification are available. This procedure works well in the overlapping region of reference speaker score distribution and imposter score distribution where the probabilities of intruders and claimants are almost the same i.e., the confidence is low. Figure 6-2 shows the region of insufficient confidence in the score distribution histogram. Imposters and true speakers are illustrated separately. The region of low confidence for our experiment as shown in this figure in the white box with dashed borders has been set to $-1.8 \pm 0.2$.

Figure 6-2: Histogram and fitted Gauss curves for the score distributions of imposters (left) and true speakers (right). The rectangle with dashed borders illustrates the score region of low confidence.
7. PERFORMANCE EVALUATION IN SPEAKER VERIFICATION

For performance evaluations of SV systems two different methods are commonly used. The first is the Receiver Operating Characteristic (ROC) curve and the second the Detection Error Tradeoff (DET) curve. The ROC is defined as the plot of false alarm rate vs. correct detection rate. Martin et al [Martin et al., 1997] suggested that it is more appropriate to plot two error rates against each other and therefore defined the DET curve as the plot of false acceptance (FA) rate vs. false rejection (FR) rate. Recently the DET curve is favoured for performance evaluations in SV to ROC. The point in the DET curve where the false acceptance rate equals the false rejection rate is called the Equal Error Rate (EER) which is established in SV. The NIST consortium [Przybocki and A, 1997] defined another cost for the DET curve which is called detection cost function (DCF) which is a measure for the correctness of detection decision and is defined as:

$$\text{DCF} = 0.1 \times P_{FA} + 0.99 \times P_{FR}$$

with $P_{FA}$ and $P_{FR}$ the probabilities of false acceptance and false rejection respectively. In Figure 7-1 the DCF is plotted as a function of the score threshold. The minimum DCF value corresponds to the optimum operating point for the system and this cost. Normally one is only interested in the optimum operation point for the DCF i.e., the smallest value in the DCF curve.

Summing, the difference between the ROC and the EER is the different weighting of the two types of errors in both methods. For the design and test of our developed system we assumed that both errors have the same importance and hence we took the EER measurement as basis.
8. DATABASE AND EXPERIMENTAL SETUP

For development purposes the telephone database from SPEECHDAT-AT [Baum et al., 2000] was used. It is emphasized that for development, training and testing separate parts of this database are used. Further testings have been carried out on the WSJ0 database [Garofalo et al., 1993] where different speakers utter the same text. In order to simulate the conditions of ATC all files were band-pass filtered with a Butterworth filter of order 8 to a bandwidth from 300 Hz to 2500 Hz and down-sampled to a sampling frequency of 6 kHz. For the experiment a total of 200 speakers comprising 100 females and 100 males were randomly chosen from the entire database. A representative distribution of dialect regions and age was maintained.

Background models were trained gender dependent using two minutes of speech material for each of 50 female and 50 male speakers. For training the UBM the speech material of five speakers had been concatenated. This model then was used for training the next five speakers to yield the UBM in the end. Since also the influence of the number of Gaussian components on the performance is of interest it will be analyzed in more detail in sec. 9. Out of the remaining 100 speakers 20 were marked as reference speakers. Afterwards their speech material was used to train speaker dependent models.

Both, for the remaining 99 speakers, known as imposters as well as for the reference speaker, six utterances were used for verification. So each reference speaker was compared to 600 utterances, yielding a total of 12000 test utterances for 20 reference speaker models all together. To match ATC conditions the database was cut artificially in segments of 5 seconds. The experiment was performed twice.

Firstly only one segment of 5 seconds was used for training and secondly, which is assumed to be the general case, three segments in a row. For the tests conducted on the WSJ0 database again the files were pre-processed as for the SPEECHDAT-AT database. The CD 11_2_1 of WSJ0 database comprising 23 female and 28 male speakers was used to train the gender dependent UBMs.

Since in this database each speaker produces the same utterance 100 seconds of speech were randomly selected from each speaker and used for training. For testing CD 11_1_1 with 45 speakers divided into 26 female and 19 male ones was taken. Here again the speech files for the reference speaker as well as for the claimants were selected randomly. Speech material used for training the reference speaker was labelled and hence excluded from verification.

Because the speech files were randomly selected, the experiment was carried out five times. Out of the 45 speakers 24 were labelled as reference speakers, 12 female and 12 male each. As for the SPEECHDAT-AT database the speech files were cut artificially into talk spurts of 5 seconds. Training of the reference speakers was performed by using three segments of speech each of 5 seconds in length.

Both, for the remaining 44 speakers, known as imposters as well as for the reference speaker, 12 utterances also five seconds in length were used for verification. So each reference speaker was compared to 540 utterances. Verification had been done for 24 reference speakers which yields in a total number of 12960 test utterances.

Here the influences of mismatches between different microphone types have not been considered, because in general it can be assumed that a pilot does not change the headset during a flight.
9. RESULTS

The impacts of front-end processing and model complexity on speaker verification performance are examined. Therefore various numbers of Gaussian mixture components and different frame lengths and frame rates configurations shown in Table 1 are studied. For all the subsequent experiments the VAD method was fixed to the energy-based one. The performance has been measured in terms of equal error rate (EER) and detection cost function (DCF) [Przybocki and A, 1997] which are depicted as special points in the detection error trade-off (DET) curve sec. 7. In Figure 9-1 DET curves for both the best system with and without the cross verification unit are shown. As previously expected, the EER for the cross-verification system is lower than for the basic system but with a slight increase of the DCF value.

![Figure 9-1: DET curves with EER point (plus sign and circle).](image)

The results are shown for both the normal system and the system with cross-verification. FA is the abbreviation for the false acceptance rate and FR for the false rejection rate. The results with subscript cr are those of the cross-verification system and those without, of the basic system. Both are using the energy-based VAD.

To see the influence of the frame length and frame rate on the one hand and the number of Gaussian components on the other hand experiments on the SPEECHDAT-AT database has been conducted several times using 3 segments of 5 seconds in length for training. Performance results of the various setups measured as EER in [%] and as DCF values are shown in Table 9-1: Performance results as a function of the frame length/frame rate and the number of Gaussian components (#GC) tested with SPEECHDAT-AT database. Table 9-1. We used 16, 38, 64 and 128 Gaussian components (#GC) for the experiments. For feature extraction five different configurations of frame lengths/frame rates (FL/FR) in [ms], (10/5, 20/5, 20/10, 25/5, 25/10), have been examined. Considering only the EER values for the FL/FR ratio of 0.5 e.g., 10/5 ms and 20/10 ms, the EER increases with increasing number of Gaussian components used for modelling. For the remaining FL/FR configurations one can easily recognize a minimum for 38 Gaussian components.
After studying these results one had taken the system with the lowest EER of only 6.51% as the best setup for this specific application. This system has been finally used for training a speaker model with only one segment of 5 seconds. Here an EER of 13.4% has still been reached. The recognition rate for the gender recognition is 96%.

Table 9-1: Performance results as a function of the frame length/frame rate and the number of Gaussian components (#GC) tested with SPEECHDAT-AT database. The first value of each table entry is the EER in % and the second one corresponds to the DCF value. For training 3 segments of 5 seconds in length are used.

<table>
<thead>
<tr>
<th>#GC</th>
<th>Frame Length/Frame Rate [ms]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10/5</td>
</tr>
<tr>
<td>16</td>
<td>8.2/0.042</td>
</tr>
<tr>
<td>38</td>
<td>9.04/0.042</td>
</tr>
<tr>
<td>64</td>
<td>9.25/0.045</td>
</tr>
<tr>
<td>128</td>
<td>10.65/0.046</td>
</tr>
</tbody>
</table>

The same experiment for the SPEECHDAT-AT database has been conducted for a background model with 1024 Gaussian mixture components.

The remaining parts have been left untouched. This has been done for comparison reasons since many systems being in place are using GMMs up to this number of components [Bimbot et al., 2000]. For this system design and its restrictions, an EER of 33% could be reached. A reason for this result could be the over modelling for speaker dependent models and verification afterwards, using short speaker turns.

Finally fixing all parameters yielding in the best SV performance, the impact on EER using two different VAD methods with and without the bridging rule are investigated. The energy-based VAD in [Neffe et al., 2007] is used to compare with the proposed WT-based VAD in term of EER. As reported in Table 9-2, for the SPEECHDAT-AT database which consists of noisy telephone recordings, the usage of both VAD methods improves SV performance significantly compared to the case without using VAD. However, for the almost noise-free WSJ0 database, the obtained results are almost similar. This shows a positive effect of VAD in removing noise-dominated non-speech segments which may lead to an unreliable trained SV system. With the more accurate WT-based VAD than the energy-based VAD, the EER is reduced from 11.7% to 9% without smoothing, and from 6.52% to 4.75% with smoothing as illustrated in Figure 9-2. Thus, by using the proposed WT-based VAD, we gain 23% and 27% relative improvements compared to the energy-based VAD in both cases. In addition, from the observed results, we discovered that not only an accurate detection of speech frames but also a smoothing to bridge short pauses between speech frames help to improve the SV performance.
To assess the impact of environmental mismatch between training and test conditions, a cross testing has been performed using SPEECHDAT-AT database for training UBMs but WSJ0 database for testing and vice versa. The WT-based VAD is employed for these experiments. In the former condition, the EER is 11.8% which is worse than above results because the models were trained by noisy speech and tested with clean speech. In the later condition, the slightly improvement of EER to 11% may result from the effect of VAD in reducing of noisy non-speech segments in testing phase. In both conditions, using VAD can’t solve the mismatch of training and testing phases.
10. DEMONSTRATOR

Finally after testing the implemented system a demonstrator has been implemented in MATLAB with a graphical user interface. It is compiled to run in stand-alone mode not only on LINUX PCs but also on MAC OSX and WINDOWS operation systems. For further information please read the instructions. This demonstrator uses exactly the system tested in the previous section. Moreover it intends to simulate somehow the behaviour of the real air traffic voice communication scenario with 3 aircrafts in a certain control sector. In Figure 10-3(a) on the left the 3 push to talk buttons (PTT ON) for each of the pilots are shown. Basically, each of the pilots have the possibility to read a speech file or to utter a command in a microphone as direct input to the demonstrator. To stop the recording the huge PTT OFF button has to be pressed. Furthermore one can see two units called Parameter Settings right top and Computational Intelligence to the left bottom. The first is to control the following:

- Score Threshholding: This is the score threshold used to make a decision as theoretically mentioned in sec. 5.7.5.
- Pause: To adjust the time lag the demonstrator is waiting for a pilot input in the Sound Animation mode.
- Speakers/AC: Press to allow a possible co-pilot in an aircraft.
- Sound Animation: Simulation of the communication between pilots and controller in a control sector; A pilot can only utter a command when no other (simulated controller/pilot) is talking.
- Microphone: To switch between file and soundcard for speech input.

In Figure 10-3(a) a pilot is entering a control sector and registering. The pilots voice is used to enrol a speaker dependent model. As illustrated in Figure 10-3(b) the pilot is again active and the voice message is used first to verify if its the same pilot. After the successful verification the speech is used to further refine the speakers model.

![Figure 10-3(a) A Pilot enters a control sector and its voice is enrolled to the system](image)

![Figure 10-3(b) Another voice message of the pilot is first verified positive and afterwards used for retraining](image)

![Figure 10-1: Screenshots of Demonstrator](image)
After the description of enrolling and retraining of a pilot communicating one is also interested on the detection of a utterance not arising from the pilot. After sufficient training an utterance of an intruder/attacker should be recognized reliable as shown in Figure 10-4(a). After the recognition also a measure of the decision confidence as shown in Figure 10-4(b) is provided. This confidence is estimated using the results of the SPEECHDAT-AT database.

Figure 10-4(a) An Intruder/Attacker tries to abuse the communications system and hence should be verified as an intruder

Figure 10-4(b): After recognition of the false speaker also a confidence measure is provided.

Figure 10-2: Screenshots of Demonstrator
11. CONCLUSION

A novel speaker segmentation system for voice communication in air traffic control is described for enhancing the safety of air traffic voice communication. As a first safety level the aircraft identification tag based on watermarking technique is used to assign a talk spurt on the shared communication channel to its source. The air traffic communication safety is then enhanced by applying a speaker verification system based on the optimized front-end processing unit for this task. Speaker dependent models are derived in cooperation with gender information for selecting the gender dependent universal background model. Results have been presented with investigation of various numbers of Gaussian components used for modelling. Furthermore its inter-relationship with the frame length and frame rate used to extract features are discussed. Finally testing two different voice activity detection methods emphasized the importance of removing non-speech portions appropriately to improve for speaker recognition performance. Based on a priori knowledge of the score distribution a cross verification unit can further reduce the equal error rate. The system has been evaluated on two databases with satisfying results. As each pilot has to identify its voice message with the call sign a possible extension of the system might be the combination of our system with a text-constrained system which would result in an even higher level of safety.
12. REFERENCES


