IDIAP RESEARCH REPORT

IDIAP

Martigny - Valais - Suisse

Speaker Verification Experiments on the XM2VTS Database

Juergen Luettin

IDIAP–RR 99-02

January 1999

revised in August 1999

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

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

IDIAP Research Report 99-02

Speaker Verification Experiments on the XM2VTS Database

Juergen Luettin

January 1999

revised in August 1999

Abstract. This paper describes two speaker verification algorithms, a text-independent method based on a second order statistical measure and a text-dependent method based on hidden Markov modelling. We investigate the effect of different features, sampling rates, and threshold setting methods and introduce a N-best words pruning method that aims to compensate for the effect of poorly trained client models. Experimental evaluation is performed on the publicly available XM2VTS database according to a published protocol for three different operating points and a priori threshold setting.

1 Introduction

The M2VTS project (Multimodal Verification for Teleservices and Security Applications) is supported by the European commission within the ACTS programme. The main objective of the project is to address the issue of secured access to local and centralised services using multimodal biometric authentication technologies. Potential applications include access control for buildings, ATMs (Automatic Teller Machines), tele-banking, tele-shopping, central databases, and information and reservation services. Several authentication technologies have been pursued within the project, including speaker recognition, frontal face recognition, face profile recognition, and face recognition based on structured light. These methods are all non-intrusive and therefore more user-friendly than intrusive methods like finger-print recognition or retina scans. However, the performance of face- and speech-based recognition techniques is usually lower than for intrusive methods. Non-intrusive methods therefore often don't meet the high performance requirements imposed by typical applications. These issues are addressed by the M2VTS project which has the objective of developing multi-model verification systems that are both high performant and that have a high user acceptance.

Person recognition can be classified into two tasks: identification and verification. Identification is concerned with determining that person from a closed set who's model best matches the incoming observation (e.g. visual or acoustic data). Verification, on the other hand, is concerned with validating a claimed identity and either accepting or rejecting an identity claim. Verification, should be able to reject subjects, referred to as IMPOSTORS, that were not enrolled in the system. The verification task is much more important in practice since most applications require the verification of an identity claim.

This paper describes the speaker verification technology developed within the project. Speaker recognition systems can be divided into text-dependent and text-independent tasks. For the text-dependent task, the speech used for training and test is usually constrained to be the same, while for the text-independent task it is unconstrained. Two speaker verification systems are described: a text-dependent method based on hidden Markov models (HMM) and a text independent method based on second-order statistical measures.

2 Speaker Verification based on Second-Order Statistical Measure

The text-independent speaker verification system is based on a second-order statistical measure [4] computed on the parameterised speech features of the training and test utterance. This method has several important properties such as simplicity of the measure, relatively high performance, low computational complexity, and text-independence. A further advantage is that it does not require to set any number of model parameters.

2.1 Silence Removal

The first processing step aims to remove silent parts from the raw audio signal as these parts do not convey speaker dependent information. We use the speech activity detector proposed by Reynolds *et al.* [9]. This procedure basically estimates the instantaneous signal-to-noise ratio (SNR) as the ratio of the short-time and a long-time signal energy, and removes signal parts for which the SNR is below a certain threshold.

2.2 Feature Extraction

The cleaned audio signal is converted to linear prediction cepstral coefficients (LPCC) [1, 8] which were obtained by the auto-correlation method. The use of LPCC parameters has been shown to often lead to increased performance compared to alternative parameterisation methods. A principal advantage of cepstral coefficients is that they are generally decorrelated. Their distribution can therefore be modelled more accurately when hidden Markov models (HMMs) with diagonal covaricance matrices are used. We us a pre-emphasis factor of 0.94 to give more weight to high frequency components. The analysis window of length 25 ms and frame interval of 10 ms is multiplied by a Hamming window so that discontinuities at the edges are attenuated. Both the LPC analysis order and the cepstral order were 12. The energy is normalised by mapping it with the tangent hyperbolic function to the interval or $[0, 1]$. The normalised energy is included in the feature vector, leading to 13-dimensional vectors.

In addition to this feature set, we have performed experiments with a parameterisation where cepstral mean subtraction (CMS) was applied on the resulting LPCC features, i.e. for each coefficient, the mean cepstral parameter is estimated across each speech file and subtracted from each frame. CMS has shown improved performances especially for experiments with varying microphone and channel characteristics. Another set of experiment was performed for different sampling rates, which was either 8 kHz or 16 kHz.

2.3 Speaker Model

Let $\mathbf{O}_c$ be a sequence of feature vectors of client c and $\mathbf{O}$ a sequence of feature vectors of an accessing person a. Client c is modelled by the covariance matrix Σ_c, computed over the sequence $\mathbf{O}_c$. Similarly, the accessing person is modelled by the covariance matrix Σ_a, computed over $\mathbf{O}$.

We use a weighted form of the arithmetic-geometric sphericity measure $D_{SPH}(\Sigma_c, \Sigma_a)$ [3] as similarity measure between the client and the accessing person. The two asymmetric terms $D_{Sc}(\Sigma_c, \Sigma_a)$ and $D_{Sc}(\Sigma_a, \Sigma_c)$ are weighted by a function of the number of training and test vectors, M and N, respectively, to account for the different lengths of training and test data:

$$\begin{aligned} D_{SPH}(\Sigma_c, \Sigma_a) = {} & \frac{M}{M+N} \log(tr(\Sigma_a \Sigma_c^{-1})) + \frac{N}{M+N} \log(tr(\Sigma_c \Sigma_a^{-1})) \\ & - \frac{1}{p} \frac{M-N}{M+N} \log(\frac{det\Sigma_a}{det\Sigma_c}) - \log(p) \end{aligned} \tag{1}$$

where tr denotes the trace of a matrix, det the determinant of a matrix, and p the dimension of the feature vector. The similarity values are mapped to the interval $[0, 1]$ with a sigmoid function $f(D_{SPH}) = (1 + exp(-(D_{SPH} - t)))^{-1}$ where $f(t) = 0.5$.

According to Bayesian decision theory we can formulate the decision as follows:

$$D(\mathbf{O}) = \begin{cases} accept \text{ if } D_{SPH} \geq t \\ reject \text{ otherwise} \end{cases} \tag{2}$$

The processing time on an Sun Ultra-Sparc 30 for one access test for this verification method is about $\frac{1}{20}$ the time of the utterance duration.

3 Speaker Verification based on Hidden Markov Modelling

Hidden Markov models (HMMs) represent an efficient approach to model the statistical variations of speech in both the spectral and temporal domain [8].

The speaker verification technique described here makes use of speech recognition and speaker recognition technology based on HMMs [10]. The speech recognition system is used to segment the speech signal of an accessing person into words and silences, assuming that the word sequence is known. The verification process makes use of client models to represent the clients and world models to represent the statistics of speech of a large number of persons. Verification of a claimed identity is based on the likelihood ratio of the client models and the world models computed over the segmented words.

We use three distinct HMM sets: CLIENT MODELS, WORLD MODELS, and SILENCE MODELS. Let $C = \{C_1, C_2, ...C_N\}$ be the set of N clients. A client C_i is represented by the HMM set (CLIENT MODELS) $\mathcal{C}_i = \{\mathcal{C}_{i1}, \mathcal{C}_{i2}, ..., \mathcal{C}_{iM}\}$, consisting of one model for each word, where $\mathcal{C}_{ij}$ represents the word model for client C_i and word j. Let $\mathcal{W} = \{\mathcal{W}_1, \mathcal{W}_2, ..., \mathcal{W}_M\}$ denote the set of WORLD MODELS where $\mathcal{W}_j$ represents the word model for word j. The world models serve as a speaker-independent model set to represent the average speech taken from a large number of persons. These models are usually trained on a distinct set of speakers, that neither includes clients nor impostors. Finally, the set of silence models $\mathcal{S} = \{\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3\}$ consists of three models that are used to model the silent parts of the signal, where $\mathcal{S}_1$ models the silence at the beginning of a phrase, $\mathcal{S}_2$ the silence between words, and $\mathcal{S}_3$ the silence at the end of a phrase. An overview of the HMM-based speaker verification technique is shown in Fig. 1.

3.1 Similarity Measure

The similarity measure is computed on the speech segments only, silence segments are not used. Let $\mathbf{O} = \{\mathbf{O}_1, \mathbf{O}_2, ..., \mathbf{O}_W\}$ be the ensemble of observation sequences that have been segmented by the speech recognition system, where $\mathbf{O}_j$ represents the observation sequence of word j. The likelihood $p(\mathbf{O}_j|\mathcal{C}_{ij})$ can be computed for each speech segment and each corresponding client model by the Viterbi algorithm [11]. To normalise these measures with respect to different word durations, the likelihoods are divided by the number of frames F_j in the sequence $\mathbf{O}_j$. We assign equal weights to each word and sum the average frame-likelihoods $p(\mathbf{O}_j|\mathcal{C}_{ij})$ over all words M. In the log-domain, this measure becomes

$$\log p(\mathbf{O}|\mathcal{C}_i) = \frac{1}{M} \sum_{j=1}^{M} \frac{\log p(\mathbf{O}_j|\mathcal{C}_{ij})}{F_j} \tag{3}$$

The similarity measure

$$D_{HMM} = \log p(\mathbf{O}|\mathcal{C}_i) - \log p(\mathbf{O}|\mathcal{W}), \tag{4}$$

computes the likelihood ratio between the client set and the world set. We can now formulate the verification decision as follows:

$$D(\mathbf{O}) = \begin{cases} \textit{accept} \text{ if } D_{HMM} \geq t \\ \textit{reject} \text{ otherwise} \end{cases} \tag{5}$$

The processing time on a Sun Ultra-Sparc 30 for one access test by this verification method is about half the time of the utterance duration.

3.2 Feature Extraction

As the POLYCOST database contains telephone speech sampled at 8 kHz, the whole XM2VTSDB has been sub-sampled at 8 kHz to provide similar bandwidth characteristics. The same parameterisation was performed on all 3 databases. The signal was converted to 13 linear prediction cepstral coefficients (LPCC) using the autocorrelation method as described for the text-independent method. We used a pre-emphasis factor of 0.94, a Hamming window of length 25 ms, a frame interval of 10 ms, and an analysis order or 13 for both LPC and cepstrum representations. We have performed experiments with two alternative feature representations:

LPCC_DA: Linear prediction cepstral coefficients (LPCC) with first and second order temporal derivatives (speed and acceleration parameters), no energy, 39-dimensional feature vectors.

LPCC_ZEDA: Linear prediction cepstral coefficients (LPCC) with cepstral mean subtraction (CMS), energy, first and second order temporal derivatives, 42-dimensional feature vectors.

Cepstral mean subtraction was performed by subtracting the mean, calculated over the whole sentence, from each feature vector.

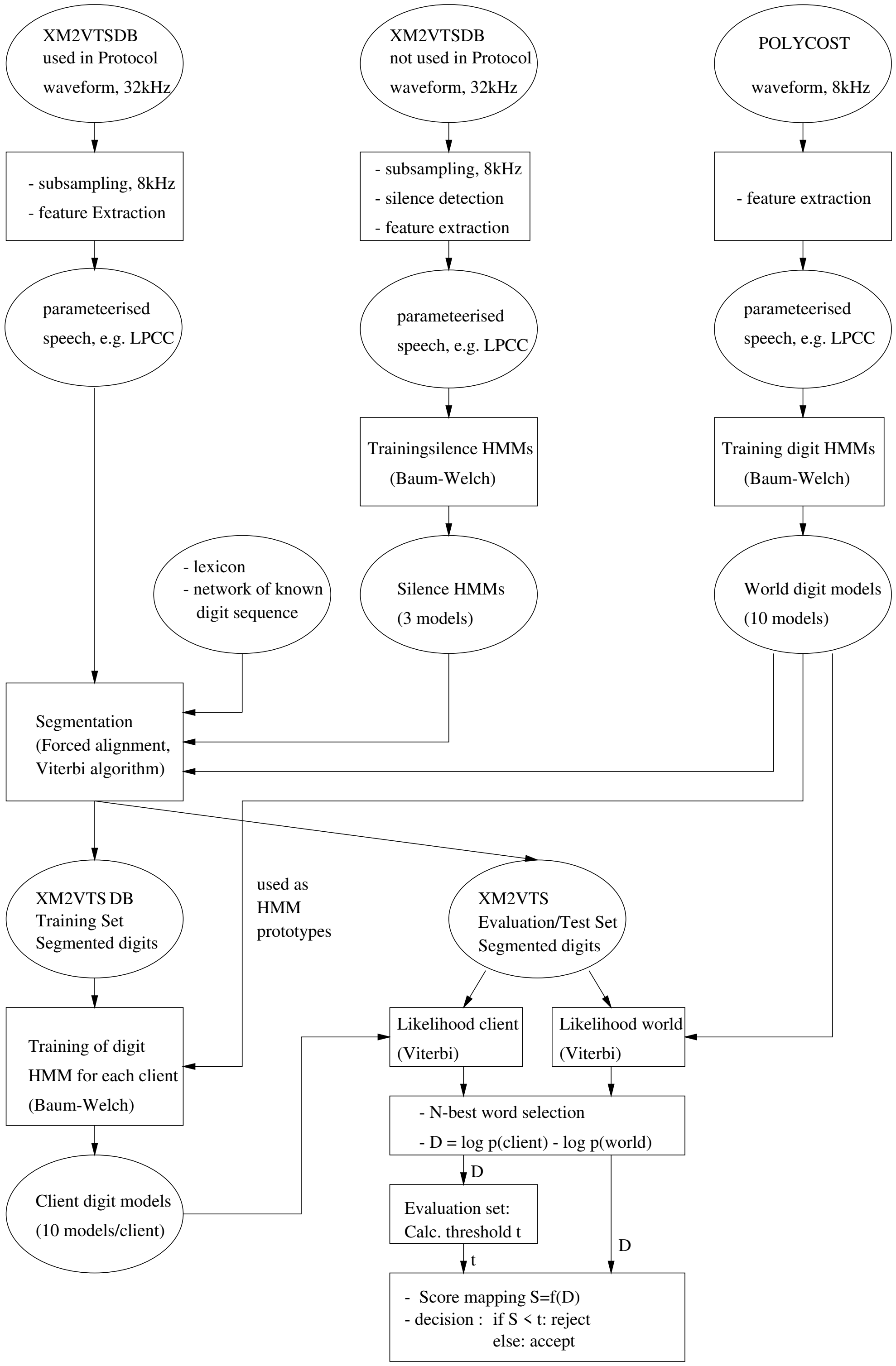

Figure 1: Overview of the HMM-based speaker verification method

3.3 Training

Three databases were used to train the three HMM sets:

- The XM2VTS database[1] [6] (XM2VTSDB) was used according to the XM2VTS evaluation protocol [5] for training, evaluation, and testing of the speaker verification systems. The CLIENT MODELS were trained on the training set.
- The data of those subjects in XM2VTSDB, that are not used in the evaluation protocol was used to train SILENCE MODELS. This part consists of 240 digit sequences, spoken by 40 subjects.
- The POLYCOST [7] database was used to train WORLD MODELS. It has the following characteristics: 134 subjects (74 male, 60 female); mainly non-native English speakers from 13 European countries; digits collected through international telephone lines; about 500 examples per digit; database has been segmented and labelled.

The POLYCOST database presents a non-optimal choice for world model training as the speech data is very different to that of XM2VTSDB. POLYCOST contains telephone speech, spoken by mostly non-native English speakers whereas XM2VTSDB contains microphone speech spoken by staff and student of a British university. Higher performance might be expected using a database that is more similar to XM2VTSDB.

Training World Models The world models $\mathcal{W}$ were trained on the segmented digits of the POLYCOST database using one HMM per digit. The number of states was between 3 and 9, depending on the number of phonemes in the digit. The parameter distribution at each state is modelled by one Gaussian mixture component with diagonal covariance matrix. Training was performed with the Baum-Welch algorithm [2]. To avoid very small variance values, a variance floor of 0.0001 has been applied, i.e. variances smaller than the variance floor are set to the variance floor.

Training Silence Models The silence models $\mathcal{S}$ were trained on the speech part that is not used in the XM2VTSDB evaluation protocol. The silence detector described in Sec. 2 was applied to detect silence parts. These parts are parameterised and used for further processing. We trained 3 silence models $\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3$ corresponding to silence at the beginning of an utterance, between utterances, and at the end of an utterance, respectively. The HMMs are based on 2 states and one Gaussian mixture component with diagonal covariance matrix. Training was performed in two stages. The first stage consists in the linear segmentation of the segments onto 2 HMM states, followed by iterative Viterbi alignment and estimation of the means and variances of the HMM states. The second stage consists in the re-estimation of the models with the Baum-Welch algorithm.

Segmentation of XM2VTSDB The XM2VTSDB was segmented into words and silences based on the trained silence models $\mathcal{S}$ and world models $\mathcal{W}$ by computing the best path between the sentence and the known sequence of HMMs using the Viterbi algorithm. The algorithm makes use of a lexicon, a network of the known digit sequence, and the trained word and silence models. The lexicon consisted of the 10 digits and the 3 silence models. The network uses the known digit sequence and allows optional silences at the beginning, between digits, and at the end, using the respective silence models.

Training Client Models The client models $\mathcal{C}$ were trained on the segmented digits of the XM2VTSDB. One HMM $\mathcal{C}_{ij}$ was build for each client i and digit j. Training was performed by the Baum-Welch algorithm. The world digit models were used as HMM prototypes to initialise training, the HMM structure was therefore identical to that of the world models. In analogy to world model training, a variance floor of 0.0001 has been applied.

[1]From ACTS-M2VTS project, available at http://www.ee.surrey.ac.uk/Research/VSSP/xm2vts

3.4 N-Best Words Pruning

The analysis of verification errors of a HMM-based system have shown that (1) some digits are more person discriminant than others, i.e. for some digits the likelihood ratio between the client and the "world" is much larger than for others, (2) some digits are not well recognised, i.e. very small likelihood values are obtained for some digits, especially for the client models. The first observation is likely to be due to the difference in discriminant information for different words. The second observation might be due either to the small training set used to train the client models or due to the differences in pronunciation between training and test set. We argue that higher verification performance might be obtained if these observations are taken into account in the verification process, e.g. using only those words in the similarity measure that are most likely to contain discriminant information and that correspond to well trained models. To address this issue, we propose different methods where only the N-best words are retained for the similarity measure D_{HMM}, which were chosen according to the following three criteria:

1. N-best client words, based on mean frame likelihood $\log p(\mathbf{O}_j|\mathcal{C}_{ij})/F_j$.
2. N-best world words, based on mean frame likelihood $\log p(\mathbf{O}_j|\mathcal{W}_j)/F_j$.
3. N-best client words, based on total word likelihood $\log p(\mathbf{O}_j|\mathcal{C}_{ij})$.

4 Database and Protocol

The XM2VTS database contains synchronised image and speech data as well as sequences with views of rotating heads. The database contains four recording sessions of 295 subjects taken at one month intervals. On each session, two recordings were made, each consisting of a speech shot and head rotation shot. The speech shot consists of frontal face and speech recordings of each subject during the pronunciation of a sentence.

The database was acquired using a Sony VX1000E digital cam-corder and DHR1000UX digital VCR. Video is captured at a color sampling resolution of 4:2:0 and 16 bit audio at a frequency of 32 kHz. The video data is compressed at a fixed ratio of 5:1 in the proprietary DV format. In total the database contains approximately 4 TBytes (4000 Gbytes) of data.

When capturing the database the camera settings were kept constant across all four sessions. The head was illuminated from both left and right sides with diffusion gel sheets being used to keep this illumination as uniform as possible. A blue background was used to allow the head to be easily segmented out using a technique such as chromakey. A high-quality clip-on microphone was used to record the speech. One speech shot consists of three sentences:

1. "0 1 2 3 4 5 6 7 8 9"
2. "5 0 6 9 2 8 1 3 7 4"
3. "Joe took fathers green shoe bench out"

The use of digits was chosen as this corresponds to a typical application scenario of speaker verification. The three sentences were the same for all speakers the allow the simulation of impostor accesses by all subjects. The second digit utterance was chosen to compensate for prosodic and co-articulation effects. The third item aims to represent a phonetically balanced sentence.

4.1 Evaluation Protocol

A protocol has been defined [5] to evaluate the performance of vision- and speech-based person authentication systems on the XM2VTSDB. The use of a common protocol will allow the comparison of different methods. The protocol is defined for the task of person *verification*, where the system decides whether the identity claim is true or not. This authentication task corresponds to an *open test set* scenario where persons, unknown to the system, might claim access. The subjects that are registered in the system's database are called *clients* whereas persons claiming false identity are referred to as *impostors*.

The database was divided into three sets: training set, evaluation set, and test set (see Fig. 2). The training set is used to build client models. The evaluation set is selected to produce client and impostor access scores which are used to calculate *verification thresholds* (a priori thresholds) that determine if a person is accepted or rejected. The thresholds are set to satisfy certain performance levels on the evaluation set and are then used on the test set. In the case of multi-modal classifiers, the evaluation set can be used to optimally combine the outputs of several classifiers. The test set is selected to simulate real authentication tests. The three sets can also be classified with respect to subject identities into client set, impostor evaluation set, and impostor test set. For this description, each subject appears only in one set which is an important requirement to ensures the realistic evaluation of imposter claims whose identity is unknown to the system.

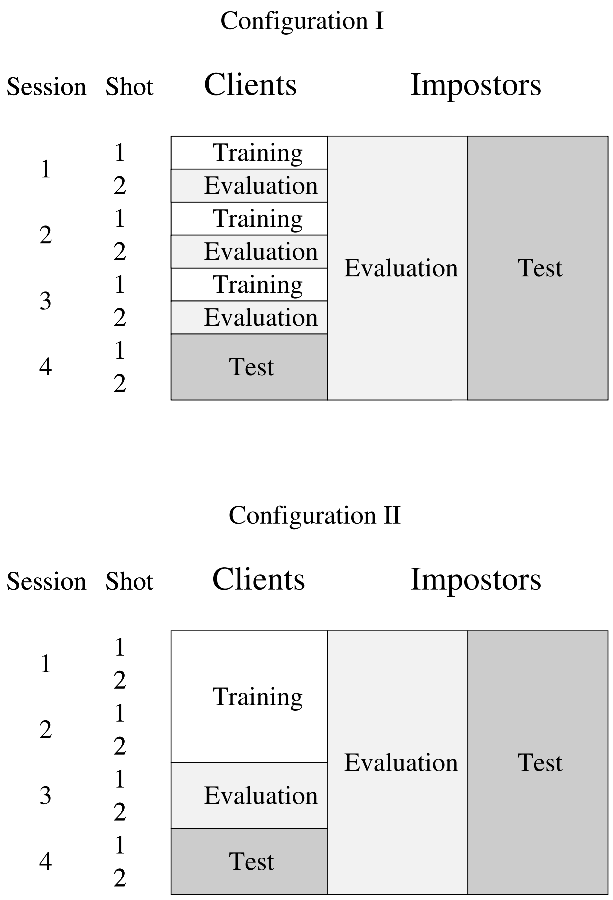

Figure 2: Diagram showing the partitioning of the XM2VTSDB according to protocol Configuration I (top) and II (bottom).

The protocol is based on 295 subjects, 4 recording sessions, and two shots (repetitions) per recording sessions. Only the first two digit sequences were used for each shot. The database was randomly divided into 200 clients, 25 evaluation impostors, and 70 test impostors (See [5] for the subjects' IDs of the three groups). Two different evaluation configurations were defined. They differ in the distribution of client training and client evaluation data as can be seen in Fig. 2. Both the client training and client evaluation data are drawn form the same recording sessions for Configuration I which might lead to good performances on the evaluation set. For Configuration II on the other hand, the client evaluation and client test sets are drawn from different recording sessions which might lead to more similar results on these two sets.

4.2 Performance Measures

Two error measures of a verification system are the *False Acceptance rate* (FA) and the *False Rejection rate* (FR). False acceptance is the case where an impostor, claiming the identity of a client, is accepted. False rejection is the case where a client, claiming his true identity, is rejected. FA and FR are given by

$$FA = EI/I * 100\% \qquad FR = EC/C * 100\% \tag{6}$$

where EI is the number of impostor acceptances, I the number of impostor claims, EC the number of client rejections, and C the number of client claims. Both FA and FR can be influenced by the threshold. There is a trade-off between the two error rates, i.e. it is possible to reduce either of them with the risk of increasing the other one. For both protocol configurations, I is $112'000$ (70 impostors $\times$ 8 shots $\times$ 200 clients) and C is 400 (200 clients $\times$ 2 shots).

Verification system performance is often quoted in *Equal Error Rate* (EER). The EER can be obtained after a full authentication experiment has been performed. The true identities of the test subjects are then used to calculate the threshold for which the FA and FR are equal. The EER is an unrealistic measure. It does not correspond to a real authentication scenario and might not well predict the expected system performance. Another commonly used performance measure is the Receiver Operator Curve (ROC) that plots either of the FA and FR against the other or as a function of the verification threshold.

In practical applications the threshold needs to be set a priori, e.g. on an evaluation set, to meet certain FA and FR requirements. For example, the threshold might be set to obtain a certain maximal FA rate, whereas the FR rate might be less critical, or vice versa. An important measure for the performance of a system is therefore the deviation of the FA and FR scores on the test set from the evaluation set.

According to the protocol, the thresholds have to be set on the evaluation set to obtain certain FA and FR rates. The same threshold will then be used on the test set to obtain the final error rates. Since application requirements might constrain the FA and FR to stay within certain limits, the system is evaluated for three different thresholds t corresponding to $FAE = 0$, $FRE = 0$, and $FAE = FRE$ with

$$\begin{aligned} t_{FAE=0} &= \arg\min_t (FRE|FAE = 0) \\ t_{FAE=FRE} &= (t|FAE = FRE) \\ t_{FRE=0} &= \arg\min_t (FAE|FRE = 0) \end{aligned} \tag{7}$$

where the errors FA and FR on the evaluation set are denoted as FAE and FRE, respectively.

5 Score Normalisation and Threshold Determination

To facilitate the combination of several classifiers the similarity values of the speaker verification methods were mapped to the interval $[0, 1]$ where 1 corresponds to high similarity and 0 to low similarity. To emphasize similarity values around the verification threshold t we use the following mapping function:

$$f(x) = \frac{1}{1 + \exp(-(x - t))}, \tag{8}$$

where x corresponds to one of the similarity measures D_{HMM} or D_{SPH} and where $f(t) = 0.5$. A person is rejected if $f(x) < 0.5$, otherwise she/he is accepted.

We have performed experiments according to two different methods of *a priori* threshold determination:

- global threshold – one threshold used for all subjects.
- individual thresholds – one threshold per subject.

For the global threshold method, the desired threshold for a certain operating point can be found by a dichotomic method. The determination of the individual thresholds is more difficult as only a very small number of client accesses is available. According to the used protocol (see section 4.1) the thresholds were determined for three different desired operating conditions. We choose the following strategy, where SI represents an impostor access, SC a client access, $FR(t)$ the false rejection rate for threshold t, and $FA(t)$ the false acceptance rate for threshold t:

FRE=0:

$$t = \arg\max_i (SI_i | SI_i < SC_{min}) + \epsilon \tag{9}$$

FRE=FAE:

$$t = \arg\max_i (SI_i | FA(SI_i) \leq FR(SI_i)) + \epsilon \tag{10}$$

FAE=0:

$$t = \begin{cases} SI_{max} + \epsilon & \text{if } FA(SI_{max}) > 0 \\ (SI_{max} + SC_{min})/2 & \text{otherwise} \end{cases} \tag{11}$$

The term ϵ is used to set the threshold just above the impostor score SI_i, we used $\epsilon = 0.0001 * SI_i$.

6 Experiments

Experiments were performed on the XM2VTS database according to the protocol described above for the text-dependent and the text-independent speaker verification system.

6.1 Text-Independent Speaker Verification

Table 1 shows the results for the basic algorithm using a global threshold and for the different improved versions for protocol Configuration I. It can be seen that each method improves the overall performance of the system. Cepstral mean substraction reduces the error rate by about 50% and a sampling rate of 16 kHz generally reduces the error rate as well. Individual threshold setting (IT) generally leads to similar results across all three operating points, however, the predictability of errors is reduced. Global threshold setting leads to higher error rates for the two operating points FRE=0 and FAE=0, while sustaining relatively good predictability. The table also shows the results on the test set for posterior threshold determination that are considerably better than for a priori threshold determination. The histogram for the impostor and client scores for the best performing system are displayed in Fig. 3.

Table 2 displays the results for protocol Configuration II using CMS and 16 kHz sampling frequency. The training, evaluation, and test sets have been recorded at different times for this configuration. The performance on the evaluation set is therefore lower than for Configuration I, but similar to the performance on the test set. Compared to Configuration I, the performance on the test set is slightly worse.

6.2 Text-Dependent Speaker Verification

Experiments were performed based on two alternative LPCC parameterisations: LPCC_DA and LPCC_ZEDA, referred to as DA and ZEDA, respectively, in Table 3 and Table 4. We have evaluated two different methods for threshold determination: global threshold (GT) and individual threshold (IT).

Due to the computationally extensive task of a full verification test and due to the very low error rate of the initial system, we have selected a very small test and evaluation set of those subjects that

Table 1: Performance of the **text-independent** method for protocol **configuration I**. We denote cepstrum mean subtraction by CMS, global threshold by GT, and individual threshold by IT. All error rates are given in %.

	Evaluation Set						Test Set					
	FR=0		FR=FA		FA=0		FR=0		FR=FA		FA=0	
Method	FR	FA	FR	FA	FR	FA	FR	FA	FR	FA	FR	FA
8 kHz, GT	0.00	73.5	3.33	3.28	35.7	0.00	0.75	75.4	21.0	4.7	80.3	0.00
8 kHz, CMS, GT	0.00	17.7	1.50	1.50	14.3	0.00	0.50	22.3	7.25	2.04	52.5	0.00
16 kHz, CMs, GT	0.00	33.7	1.17	1.17	6.50	0.00	0.00	32.6	5.00	1.60	35.0	0.01
posterior T							0.25	27.3	3.25	3.25	56.3	0.0
16 kHz, CMS, IT	0.00	0.41	0.17	0.40	0.83	0.00	7.00	1.44	7.00	1.41	7.00	1.04
posterior T							0.0	0.41	0.0	0.41	13.3	0.0

Table 2: Performance of the **text-independent** method for protocol **configuration II**.

	Evaluation Set						Test Set					
	FR=0		FR=FA		FA=0		FR=0		FR=FA		FA=0	
Method	FR	FA	FR	FA	FR	FA	FR	FA	FR	FA	FR	FA
16 kHz, CMS, GT	0.00	72.0	5.00	4.99	60.0	0.00	0.00	69.6	4.25	5.53	59.8	0.01
posterior T							0.0	45.1	4.75	4.75	68.0	0.0
16 kHZ, CMS, IT	0.00	1.13	0.25	0.98	15.3	0.00	8.25	2.11	9.00	1.88	12.5	0.99
posterior T							0.0	0.9	0.0	0.9	21.8	0.0

Table 3: Performance of the **text-dependent** method for protocol**Configuration I** for different features (DA/ZEDA), different threshold determination strategies (GT/IT), and optional N-Best word pruning (NB). All error rates are given in %.

	Evaluation Set						Test Set					
	FR=0		FR=FA		FA=0		FR=0		FR=FA		FA=0	
Method	FR	FA	FR	FA	FR	FA	FR	FA	FR	FA	FR	FA
DA, GT	0.0	73.6	3.33	3.33	19.5	0.0	0.0	73.6	3.25	3.51	31.5	0.001
DA, IT	0.0	2.52	1.17	1.71	5.33	0.0	1.9	3.32	2.7	2.51	8.8	0.061
DA, GT, NB	0.0	0.64	0.17	0.30	30.0	0.0	2.5	0.72	3.00	0.33	59.3	0.0
DA, IT, NB	0.0	0.018	0.0	0.018	0.83	0.0	2.5	1.18	2.50	1.17	2.5	1.17
ZEDA, GT, NB	0.0	2.30	0.33	0.33	10.83	0.0	0.0	2.84	0.75	0.41	17.5	0.004
posterior T							0.0	1.76	0.75	0.75	36.5	0.0
ZEDA, IT, NB	0.0	0.015	0.0	0.015	0.167	0.0	0.0	1.48	0.0	1.48	1.75	0.036
posterior T							0.0	0.026	0.0	0.026	1.25	0.0

Table 4: Performance of the **text-dependent** method for protocol **Configuration II**.

	Evaluation Set						Test Set					
	FR=0		FR=FA		FA=0		FR=0		FR=FA		FA=0	
Method	FR	FA	FR	FA	FR	FA	FR	FA	FR	FA	FR	FA
ZEDA, GT, NB	0.0	16.9	1.25	1.25	14.8	0.0	0.0	20.4	0.5	1.57	13.3	0.004
posterior T							0.0	2.0	0.5	0.5	25.5	0.0
ZEDA, IT, NB	0.9	0.115	0.0	0.115	2.0	0.0	0.0	1.58	0.0	1.55	3.0	0.073
posterior T							0.0	0.015	0.0	0.015	2.0	0.0

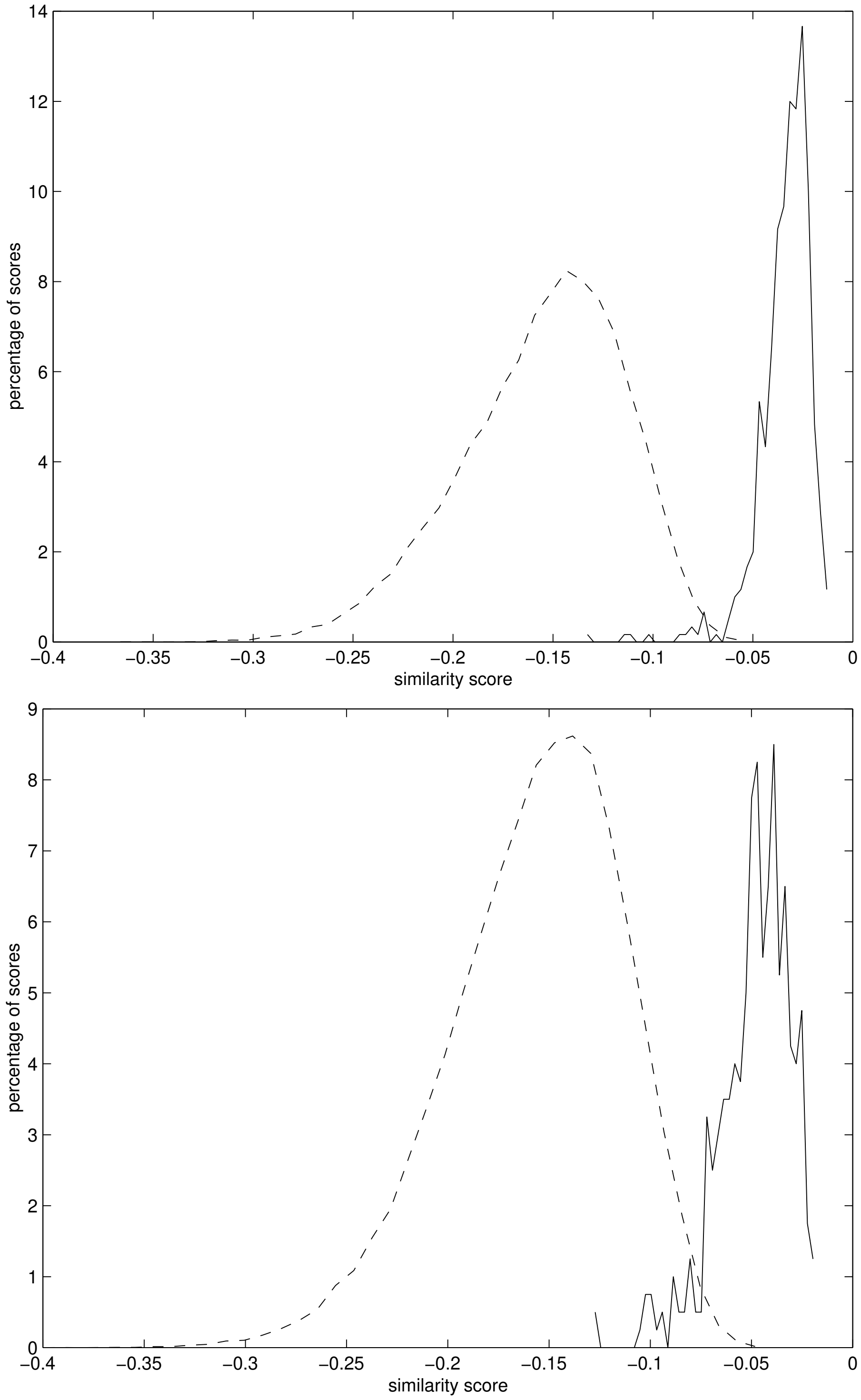

Figure 3: Histogram of the client (solid) and impostor (dashed) scores for the evaluation set (top) and the test set (bottom) for the text-independent method using 16 kHz sampling rate and CMS.

caused most of the FR and FA errors to evaluate different N-best words pruning methods. It consisted of 6 clients and 5 impostors. Table 5 shows the verification results using either all words (N=20) or using only the N-best words, which were sorted according to the 3 different criteria. It can be seen that all N-best schemes reduce the error rate for this set considerably and that the first method achieves the overall best performance. The world models were trained on considerably more data than the client models. This might explain the improved results when client scores are used in the N-best words calculation. The performance for different numbers N is fairly stable for values between around $8 - 16$.

Table 5: Performance of the **text-dependent** method for a **small difficult set** using different N-best words criteria. Errors are shown in % for the EER (a posteriori threshold) operating point.

Criteria	N	Global Threshold		Individual Threshold	
		FR	FA	FR	FA
-	20	33.33	36.87	25.0	32.18
1	18	4.17	7.5	0.0	0.625
	16	0.0	2.18	0.0	0.0
	14	0.0	0.0	0.0	0.0
	12	0.0	0.0	0.0	0.0
	10	0.0	0.0	0.0	0.0
	8	0.0	0.0	0.0	0.0
	6	0.0	0.125	0.0	0.0
2	18	16.7	16.5	12.5	13.75
	16	4.17	4.06	0.0	1.25
	14	0.0	2.5	0.0	0.0
	12	0.0	3.4	0.0	0.0
	10	4.17	4.38	0.0	0.938
	8	4.17	4.06	0.0	0.312
	6	4.17	4.06	0.0	0.312
3	18	4.17	6.25	0.0	1.88
	16	0.0	0.938	0.0	0.0
	14	0.0	0.625	0.0	0.0
	12	0.0	0.938	0.0	0.0
	10	0.0	1.25	0.0	0.0
	8	0.0	2.5	0.0	0.31
	6	0.0	3.438	0.0	0.0

The N-best words pruning methods was applied to the whole database by retaining the 14-best words in the similarity measure, sorted according to the mean frame likelihood of the client (N-best method 1). Results are shown in Table 3. It can be observed that for most experiments, the error rate is considerably reduced by the word pruning method.

Experiments show that the setting of individual thresholds often leads to better results, especially for the FRE=0 and FAE=0 operating points. Furthermore, the LPCC_ZEDA features generally lead to better results than LPCC_DA features. It can be observed that the results for Configuration II are worse than for Configuration I, but the results for the evaluation and test sets are more similar for Configuration II than for Configuration I. As it was the case for the text-independent method, this is likely to be due to the different recording times between the sets. Posterior threshold setting leads to much lower error rates as it was the case for the text-independent method. The histograms of the client and impostor scores for the evaluation and test set are shown in Fig. 4.

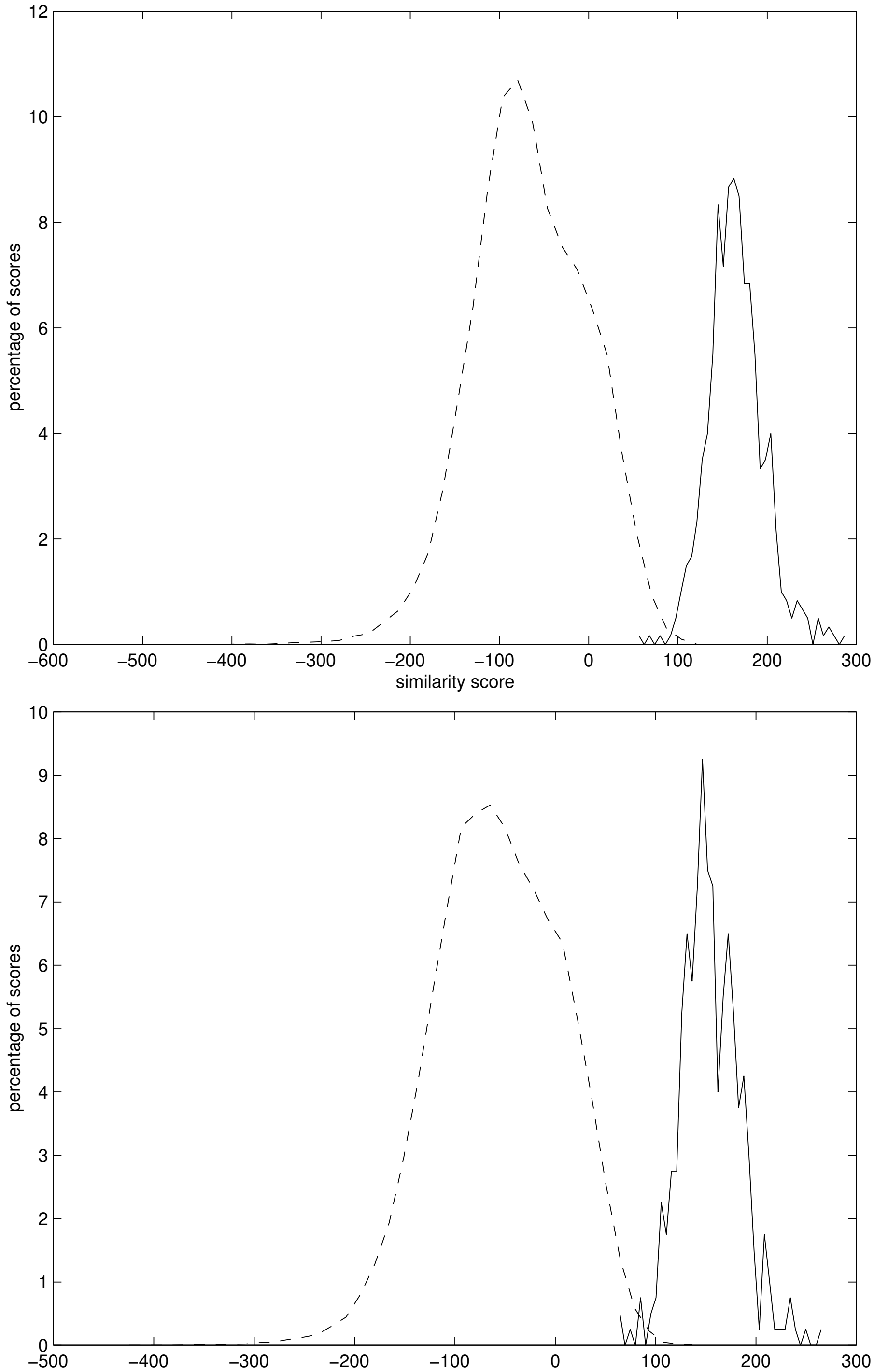

Figure 4: Histogram of the client (solid) and impostor (dashed) scores for the evaluation set (top) and test set (bottom) for the text-dependent method using ZEDA features and N-Best word pruning.

7 Conclusions

This paper has described two speaker verification algorithms, a text-independent method based on a second-order statistical measure and a text-dependent method based on hidden Markov modeling. Experimental evaluation was performed on a publicly available database according to a published protocol to allow the comparison of results. Experiments were performed for three different operating points and a priori threshold setting to simulate a realistic application.

Experiments have shown that the performance of the best text-dependent system is much higher than for the text-independent system. The performance of the best text-independent method for Configuration II obtains about 0.6% mean total error (MTE=FA+FR, averaged over all 3 operating points) on the evaluation set and about 8.3% on the test set. The same rates for the text-dependent method are about 0.065% on the evaluation set and about 1.58% on the test set. These error rates are very similar for the three operating points. These differences in error rates between the evaluation and test of a factor of about 10-20 demonstrate the importance of performing experiments with a priori threshold setting. The HMM-based system has not been optimised, thus further improvements might be obtained for example by using a higher sampling rate, training world models on a more similar database, or by optimising the HMM configuration.

Different standard techniques including different sampling rate, different features, cepstral mean subtraction, and different threshold setting strategies have been evaluated and compared.

A N-best words pruning method has been introduced to compensate for the effect of badly trained client models and has been shown to lead to considerably reduced error rates.

Acknowledgements

The author would like to thank Gilbert Maitre for contributions to the verification experiments, Eric Meurville, Gilbert Maitre, and Laurent Besacier for the implementation of the text-independent method, and Dominique Genoud for useful discussions. This work has been performed under the European ACTS-M2VTS project with financial support from the Swiss Office for Education and Science (BBW).

References

[1] B.S. Atal. Effectivness of linear prediction characteristics of the speech wave for automatic speaker identification and verification. *JASA*, 55(6):1304–1312, 1974.

[2] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occuring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics*, 41:164–171, 1970.

[3] F. Bimbot, I. Magrin-Chagnolleau, and L. Mathan. Second-order statistical measure for text-independent speaker identification. *Speech Communication*, 17(1-2):177–192, 1995.

[4] Frédéric Bimbot and Luc Mathan. Second-order statistical measures for text-independant speaker identification. In *Esca Workshop on Speaker Recognition, Identification, and Verification*, pages 51–54, Martigny, 1994.

[5] J. Luettin and G. Maître. Evaluation protocol for the extended M2VTS database (XM2VTSD B). IDIAP-COM 98-05, IDIAP, 1998. available at ftp.idiap.ch/pub/reports/1998/com98-05.ps.gz.

[6] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre. XM2VTSDB: The extended M2VTS database. In *Audio- and Video-based Biometric Person Authentication, AVBPA'99*, pages 72–77, Washington, D.C., March 1999.

[7] D. Petrovska, J. Hennebert, H. Melin, and D. Genoud. Polycost: a telephone-speech database for speaker recognition. In *Proc. Speaker Recognition and its Commercial and Forensic Applications*, Avignon, 1998.

[8] L. R. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition*. Prentice-Hall, Englewood Cliffs, NJ, 1993.

[9] D.A. Reynolds, R.C. Rose, and M.J.T. Smith. Pc-based tms320c30 implementation of the gaussian mixture model text-independent speaker recognition system. In *ICSPAT, DSP Associates*, pages 967–973, 1992.

[10] A. E. Rosenberg, C. H. Lee, and S. Gokoen. Connected word talker verification using whole word hidden Markov model. In *ICASSP-91*, pages 381–384, 1991.

[11] A. J. Viterbi. Error bounds for convolutional codes and an asymtotically optimum decoding algorithm. *IEEE Trans. on Information Theory*, 13(2):260–269, 1967.