ROBUST PERSON VERIFICATION BASED ON SPEECH AND FACIAL IMAGES

J. Luettin and S. Ben-Yacoub

IDIAP
CP 592, 1920 Martigny, Switzerland
luettin@idiap.ch, sby@idiap.ch, http://www.idiap.ch

ABSTRACT

This paper describes a multi-modal person verification system using speech and frontal face images. We consider two different speaker verification algorithms, a text-independent method using a second-order statistical measure and a text-dependent method based on hidden Markov modelling, as well as a face verification technique using a robust form of corellation. Fusion of the different recognition modules is performed by a Support Vector Machine classifier. Experimental results obtained on the audio-visual database XM2VTS for individual modalities and their combinations show that multimodal systems yield better performances than individual modules for all cases.

1. INTRODUCTION

Biometric authentication techniques like face recognition and speaker recognition are non-intrusive and therefore more acceptable by the user than intrusive methods such as 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.

We describe a system that combines different authentication modules which is motivated by the fact that the combination of classifiers can circumvent the shortcomings of individual methods and hence improve the overall performance [3, 4, 5].

2. SPEAKER VERIFICATION

We have investigated two different speaker verification algorithms: a text-independent method based on a second-order statistical measure [2] and a text-dependent technique using hidden Markov models (HMM) [9].

2.1. Text-independent Speaker Verification

The first processing step aims to remove silent parts from the raw audio signal as these parts do not convey speaker dependent information. The signal is sub-sampled from the original 32 kHz down to 16 kHz followed by the removal of silent parts by a silence detector. The cleaned audio signal is converted to linear prediction cepstral coefficients (LPCC) using the autocorrelation method. We use a pre-emphasis factor of 0.94, a Hamming window of length 25 ms, a frame interval of 10 ms, and an analysis order of 12. We have applied cepstral mean subtraction (CMS), where the mean cepstral parameter is estimated across each speech file and subtracted from each frame. The energy is normalized by mapping it to the interval $[0, 1]$ using the tangent hyperbolic function. The normalized energy is included in the feature vector, leading to 13-dimensional vectors.

A client model is represented by the covariance matrix $\mathbf{X}$, computed over the M feature vectors of the client's training data. Similarly, an accessing person is represented by the covariance matrix $\mathbf{Y}$, computed over the N feature vectors of that person's speech data. We use a weighted form of the arithmetic-geometric sphericity measure $D_{Sc}(\mathbf{X}, \mathbf{Y})$ [2] as similarity measure between the client and the accessing person. The two asymmetric terms $D_{Sc}(\mathbf{X}, \mathbf{Y})$ and $D_{Sc}(\mathbf{Y}, \mathbf{X})$ 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:

$$\begin{aligned} D_{SPH}(\mathbf{X}, \mathbf{Y}) \quad &= \tfrac{M}{M+N} \log(tr(\mathbf{Y}\mathbf{X}^{-1})) \\ &+ \tfrac{N}{M+N} \log(tr(\mathbf{X}\mathbf{Y}^{-1})) \\ &- \tfrac{1}{p} \tfrac{M-N}{M+N} \log(\tfrac{det\mathbf{Y}}{det\mathbf{X}}) - \log(p) \end{aligned} \quad (1)$$

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 were mapped to the interval $[0, 1]$ with a sigmoid function $f(D_{SPH}) = (1 + exp(-(D_{SPH} - t)))^{-1}$ where $f(t) = 0.5$. A claimed speaker is rejected if $f(D_{SPH}) < 0.5$, otherwise she/he is accepted. The processing time, on an Sun Ultra-Sparc 30, required by this verification module is about $\frac{1}{20}$ the time of the utterance duration.

2.2. Text-dependent Speaker Verification

The text-dependent speaker verification technique makes use of 3 sets of hidden Markov models (HMM): client models, world models, and silence models. Utterances of a client are represented by client HMMs. The world models serve as speaker-independent models to represent speech of an average person. They are trained on the

POLYCOST[1] database, that represents a distinct set of speakers. Finally, three silence HMMs are used to model the silent parts of the signal. As the POLYCOST database contains telephone speech sampled at 8 kHz, the whole XM2VTS database has been sub-sampled at 8 kHz to provide similar bandwidth characteristics.

The same features as for the text-independent system are extracted. In addition, the first and second order temporal derivatives were included, leading to 42-dimensional feature vectors. All models were trained based on the maximum likelihood criterion using the Baum-Welch (EM) algorithm. The world models were trained on the segmented words of the POLYCOST database, where one HMM per word was trained. The number of states was between 3 and 9, depending on the number of phonemes in the word. The feature distribution at each state is modelled by one Gaussian mixture component with diagonal covariance matrix. To avoid very small variance values, a variance floor has been applied. Silence models were build from speech data of clients that were not included in the protocol.

For both training and verification the sentences of the XM2VTS database are first segmented into words and silences using the world and silence models. This consists in computing the best path between the sentence and the sequence of known HMMs using the Viterbi algorithm. The client models were then trained on the segmented training words using the world models as prototypes to initialise training. The structure of client and world models is therefore identical.

For verification, the Viterbi algorithm is used to calculate the likelihood $p(X_j|\mathcal{M}_{ij})$, where X_j represents the observation of the segmented word j; $\mathcal{M}_{ij}$ represents the model of subject M_i and word j. We normalize the log-likelihood of word j by the numbers of frames N_j and sum them over all words W, which leads to the following measure:

$$\log p(X|M_i) = \frac{1}{W}\sum_{j=1}^{W}\frac{\log p(X_j|\mathcal{M}_{ij})}{N_j} \qquad (2)$$

This measure is calculated for the models $\mathcal{M}_c$ of a given client M_c and for the world models $\mathcal{M}_w$. The similarity

$$D_{HMM} = \log p(X|\mathcal{M}_c) - \log p(X|\mathcal{M}_w) \qquad (3)$$

is computed and compared to a threshold t. The claiming subject is rejected if $D_{HMM} < t$, otherwise she/he is accepted. The quantities D_{HMM} were mapped to the interval $[0, 1]$ as described in Section 2.1. The processing time during verification is about half the time of the utterance duration.

2.2.1. N-Best Word Selection

The analysis of verification errors of the HMM-based system has shown that, (1) some digits are more person discriminant than others, i.e. the likelihood ratio varies across digits, (2) some digits are not well recognised, i.e. small likelihood values are obtained for some digits. This might be due to the small training set used to train the models or due to pronunciation differences between training and test set. Higher verification performance might be obtained if only selected words that contribute the most to speaker discrimination are used in the similarity measure. We have performed several experiments where only the N-best words were retained for the similarity measure D_{HMM}, which were chosen according to different criteria. Best performance was obtained using the N-best client words, with the highest mean frame likelihood [7].

[1]http://circwww.epfl.ch/polycost

3. FACE VERIFICATION

The face verification system is based on a robust form of correlation between the reference image and the test image [8]. The function aims to find the global extreme in a search space that considers transformations such as translation, scaling, and rotation. Only a selected set of features is used during recognition, that has been determined in the training phase to minimise the intra-class variance and at the same time to maximise the inter-class variance. The system runs on real-time on a high performance PC.

4. CLASSIFIER COMBINATION

Several studies have shown that the combination of different modalities can result in improved performance, particularly when the modalities are un-correlated. One of the main difficulties is the combination of classifiers that exhibit different performance levels. We use the Support Vector Machine (SVM) [10] for the fusion of classifiers [1]. Whereas classical learning approaches are based on empirical risk minimisation (error on a training set), SVM is based on structural risk minimisation (SRM). Consider a hyperplane that separates two classes into two sets. The SVM approach aims to find the optimal separating hyper-plane that has the largest margin to the closest data points. This hyperplane guarantees to minimise the classification error and to maximise generalisation.

We assume that we have a data set $\mathcal{D}$ of M points in a *n* dimensional space belonging to two different classes **+1** and **-1**:

$$\mathcal{D} = \{(z_i, y_i)|i \in \{1..M\}, z_i \in \mathbb{R}^n, y_i \in \{+1, -1\}\}$$

A binary classifier should find a function *f* that maps the points from their data space to their label space. It has been shown [10] that the optimal separating hyperplane is expressed as:

$$f(x) = sign(\sum_i \alpha_i y_i K(z_i, x) + b) \qquad (4)$$

where K(x,y) is a positive definite symmetric function, b is a bias estimated on the training set, and α_i are the

solutions of the following Quadratic Programming (QP) problem:

$$\begin{cases} \min_{\mathcal{A}} W(\mathcal{A}) = -\mathcal{A}^t I + \frac{1}{2}\mathcal{A}^t D\mathcal{A} \\ \text{with the constraints:} \\ \sum_i \alpha_i y_i = 0 \text{ and } \alpha_i \in [0, C] \\ \\ \text{where:} \\ (i,j) \in [1..M] \times [1..M] \\ (\mathcal{A})_i = \alpha_i \\ (I)_i = 1 \\ (D)_{ij} = y_i y_j K(z_i, z_j) \\ C = 1000 \end{cases}$$

The kernel functions $K(x, y)$ define the nature of the decision surface that will separate the data. They satisfy some constraints in order to be applicable (Mercer's conditions, see [10]). We have used a Gaussian kernel $K(x, y) = exp(-4||x - y||^2)$ in our experiments. Experiments reported in [1] have shown that the choice of the kernel and kernel parameters is not critical.

The computational complexity of the SVM during training depends on the number of data points rather than on their dimensionality. The number of computation steps is $O(n^3)$ where n is the number of data points. At run time the classification step of SVM is a simple weighted sum. The classification of 112400 claims requires 5.6 sec on an Ultra-Sparc 30.

5. THE XM2VTS DATABASE

The XM2VTSDB database[2] contains synchronized image and speech data as well as sequences with views of rotating heads. The database includes four recordings of 295 subjects taken at one month intervals. On each visit (session) two recordings were made: a speech shot and head rotation shot. The speech shot consisted of frontal face recording of each subject during the dialogue.

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 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. The speech sequence consisted in uttered English digits from 0 to 9.

[2]http://www.ee.surrey.ac.uk/Research/VSSP/xm2vts

5.1. Evaluation Protocol

The database was divided into three sets: TRAINING SET, EVALUATION SET, and TEST SET (see Fig. 1). 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 estimate verification thresholds that are then applied on the TEST SET 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. This ensures realistic evaluation of imposter claims whose identity is unknown to the system. The protocol is based on

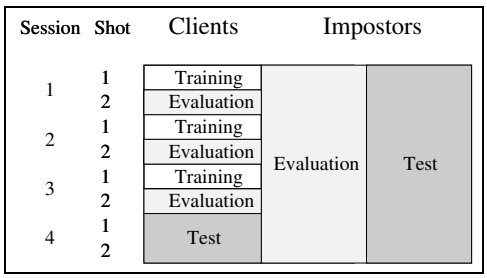

Figure 1: Diagram showing the partitioning of the XM2VTSDB according to protocol *Configuration I*.

295 subjects, 4 recording sessions, and two shots (repetitions) per recording sessions. The database was randomly divided into 200 clients, 25 evaluation impostors, and 70 test impostors (See [6] for the subjects' IDs of the three groups). Two different configurations have been defined that differ in the distribution between the client training set and the client evaluation set. In this paper, experiments were performed according to *Configuration I*, shown in Fig. 1.

5.2. Performance Measures

Two error measures of a verification system are the *False Acceptance rate* (FA) and the *False Rejection rate* (FR). A trade-off between FA and FR can be controled by a threshold. The number of impostor claims is $112,000$ (70 impostors $\times$ 8 shots $\times$ 200 clients) and the number of client claims 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. In practical applications the threshold needs to be set a priori. We would like to simulate a real applications and therefore set the threshold on the EVALUATION SET to obtain certain false acceptance (FA) and false rejection (FR) values. The same threshold is later used on the TEST SET to obtain the actual error rates.

6. EXPERIMENTS

The error rates obtained for the EVALUATION SET and TEST SET are shown in Table 1 for the three individual verification modules and for different combinations. The threshold for each verification module was determined on the EVALUATION SET to obtain an EER. These determined thresholds were used to obtain the error rates on the TEST SET. The fusion module was trained on the verification values of the EVALUATION SET and the threshold values were set to lead to an EER on that set.

Table 1: False acceptance (FA) and false rejection (FR) rates (in %) for different modules and different combinations.

	Evaluation		Test	
Modality	FA	FR	FA	FR
Face	7.64	7.67	7.76	7.25
TI	1.17	1.17	1.60	5.00
TD	0.015	0.0	0.0	1.48
Face + TI	0.86	0.83	1.18	0.0
Face + TD	0.17	0.17	1.18	0.0
TI + TD	0.0	0.17	0.38	0.5
Face + TI + TD	0.0	0.0	0.78	0.0

7. CONCLUSION

The performances on the TEST SET are generally lower than on the EVALUATION set and in most cases the FA and FR rates are no longer equal on the TEST SET. This observation is not valid for the face verification module which shows high predictability of error rates on the test set. These observations demonstrate the importance of performance evaluation using thresholds that are determined a priori on a different data set.

The experiments show that the combination of different modalities yields better results than individual modalities for all described classifier combinations. This is even the case for the combination of classifiers with very different individual performances, e.g. Face and TD. Interestingly, the combination of the two modules with the lowest performances (Face and TI) outperforms the best single module (TD).

8. ACKNOWLEDGEMENTS

The authors would like to thank J. Matas, K. Jonsson, and J. Kittler for providing the face verification results that were used in our experiments. This work has been performed under the European ACTS-M2VTS project with the financial support from the Swiss Office for Education and Science (BBW).

9. REFERENCES

[1] S. Ben-Yacoub, J. Luettin, K. Jonsson, J. Matas, and J. Kittler. Audio-visual person verification. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1999.

[2] 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.

[3] R. Brunelli and D. Falavigna. Person identification using multiple cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(10):955–966, October 1995.

[4] P. Jourlin, J. Luettin, D. Genoud, and H. Wassner. Acoustic-labial speaker verification. *Pattern Recognition Letters*, 18(9):853–858, 1997.

[5] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas. On Combining Classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):226–239, 1998.

[6] J. Luettin and G. Maître. Evaluation protocol for the extended M2VTS database (XM2VTSD B). IDIAP-COM 98-05, IDIAP, 1998.

[7] J. Luettin. Speaker verification experiments on the XM2VTS database. IDIAP-RR 99-02, IDIAP, 1999.

[8] J. Matas, K. Jonsson, and J. Kittler. Fast face localisation and verification. In *Proceedings of the British Machine Vision Conference*, pages 152–161. BMVA Press, 1997.

[9] 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.

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