GMM-based SVM for face recognition

Hervé BREDIN, Najim DEHAK and Gérard CHOLLET
CNRS-LTCI, GET-ENST (TSI Department)
46 rue Barrault, 75013 Paris, France
{bredin, dehak, chollet}@tsi.enst.fr

Abstract

A new face recognition algorithm is presented. It supposes that a video sequence of a person is available both at enrollment and test time. During enrollment, a client Gaussian Mixture Model (GMM) is adapted from a world GMM using eigenface features extracted from each frame of the video. Then, a Support Vector Machine (SVM) is used to find a decision border between the client GMM and pseudo-impostors GMMs. At test time, a GMM is adapted from the test video and a decision is taken using the previously learned client SVM. This algorithm brings a 3.5% *Equal Error Rate (EER) improvement over the BioSecure reference system on the Pooled protocol of the BANCA database.*

1. Introduction

The wide majority of face recognition algorithms shares a common framework. At enrollment time, one or a few training pictures of the subject are taken, discriminative features are extracted and saved as the model of the subject. At test time, another small set of pictures (sometimes only one) of the subject is taken and features are extracted following the same process. These features can be geometrical (such as distance between the eyes, length of the nose, etc.) or pixel-based features (such as eigenface coefficients [13], wavelet transform, etc.), or a combination of them (Active Appearance Model [8]). Some transformations are often performed on these features to reduce dimensionality and increase their discriminative power: Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Independent Component Analysis (IDA), etc. Then, a comparison is performed between the model and these features in order to decide if the subject is the person he/she pretends to be. This comparison is often performed by computing the distance (euclidean, L1-norm, L2-norm, correlation) between the training and the test pictures. But it can also use classical classification algorithms such as K Nearest Neighbors (KNN) or One-Class Support Vector Machines (OC-SVM). These 1-to-1 or few-to-few comparisons were mostly induced by the protocols defined on the available evaluation databases. Thus, the FERET database only includes still images of face. The BANCA and XM2VTS databases do contain video sequences of talking-faces but evaluation protocols have not used them until now (for example, only 5 pictures per video are used in the BANCA protocols [1]). In [6], Gaussian Mixture Models (GMM) are used to take into account the intra-subject variability (such as face rotation, lips motion or changing light conditions), though the authors only had a few pictures manually annotated and did not use every frame of the videos. Our work starts from the same idea, but using every frame of the video in which the face is automatically located. The face tracking algorithm is quickly described in section 2 along with the features extraction process. Sections 3 and 4 present the core of our algorithm: how it is possible to apply SVM in the GMM space. The experiments we performed and the corresponding results are described in sections 5 and 6.

2. Visual front-end

Any automatic face recognition algorithm needs a preliminary step of face detection. Thus, the face has to be located before it is even possible to recognize it. To allow a fair comparison with previously published results, we used the visual front-end of the Biosecure talking-face reference system which is quickly described in the following paragraphs (a full description is available in [5]).

2.1. Face detection and tracking

The *OpenCV* implementation of [15] is first used to get a rough idea of the location of the face. Then, a moving window scans every possible rectangle in this region of interest at every position with many sizes. For each candidate, the Distance From Face Space (DFFS) [11] is computed and the candidate with the lowest DFFS is chosen as the location of the face. A temporal median filter is then applied on the

The 18th International Conference on Pattern Recognition (ICPR'06)
0-7695-2521-0/06 $20.00 © 2006 IEEE

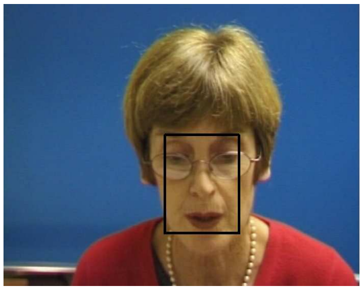

Figure 1. Face tracking

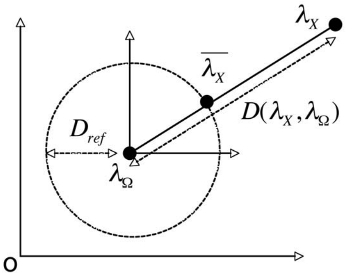

Figure 2. Normalization

location and size of the detected faces throughout the video in order to avoid local detection problems (see figure 1).

2.2. Features extraction

Once the face is detected, it is size-normalized to the size of the eigenfaces. The decomposition of the detected face on the eigenfaces is computed and used as features for face recognition [13].

3. Gaussian Mixture Model

3.1. Principle

The use of GMMs for speaker verification has been studied in depth in the literature [12]. Given a subject X and a corresponding training set $\mathbf{x} = \{x_t, t = 1...N\}$ of D-dimensional features extracted from a video of subject X, a gaussian mixture model $\lambda_X = \{w_i, \mu_i, \Sigma_i, i = 1...M\}$ is learned maximizing the following log-likelihood:

$$\log p(\mathbf{x}|\lambda_X) = \frac{1}{N}\sum_{t=1}^{N} \log p(x_t|\lambda_X) \qquad (1)$$

where

$$p(x_t|\lambda_X) = \sum_{i=1}^{M} w_i p_i(x_t|\lambda_X) \qquad (2)$$

and

$$p_i(\bullet|\lambda_X) \sim \mathcal{N}(\mu_i, \Sigma_i) \qquad (3)$$

Then, at test time, the likelihood $p(\mathbf{y}|\lambda_X)$ that the samples $\mathbf{y}$ come from subject X is computed and compared to a threshold θ: if it is higher than θ then the subject Y is decided to be the subject X.

Training sets for each subject usually contain a relatively small number of samples which may lead to unreliable models. Therefore, using this small training set, subject GMMs are adapted from a world model λ_Ω that was previously trained on a much larger set of samples: we used MAP adaptation in our experiments (see [3] for more details). At test time, the log-ratio of $p(\mathbf{y}|\lambda_X)$ and $p(\mathbf{y}|\lambda_\Omega)$ is compared to a threshold θ. In the following, all models λ are adapted from a common world model λ_Ω.

3.2. Distance between GMMs

In [2] Ben introduces a distance between two GMMs, based on the Kullback-Leibler (KL) divergence. At test time, given a set of samples $\mathbf{y}$, a test model λ_Y is adapted from the world model λ_Ω. In the particular case where neither the weights w_i nor the covariances Σ_i are adapted (gaussians means adaptation only) and using diagonal covariance matrices, the distance is defined as follows:

$$D(\lambda_Y, \lambda_X) = \sqrt{\sum_{i=1}^{M}\sum_{d=1}^{D} w_i^\Omega \frac{\left(\mu_{i,d}^X - \mu_{i,d}^Y\right)^2}{\Sigma_{i,d}^\Omega}} \qquad (4)$$

Following the same idea as in the classical GMM case, this distance can be normalized by the distance to the world model and then compared to a threshold:

$$D(\lambda_Y, \lambda_\Omega) - D(\lambda_Y, \lambda_X) > \theta \qquad (5)$$

3.3. Normalization

In [3] an additional normalization step is proposed which happens in the adapted GMMs space. Considering the world model λ_Ω as the origin of the space, every adapted GMM λ_X is normalized so that the distance between λ_X and λ_Ω equals a reference distance D_{ref} (see figure 2). In our particular case (MAP with means adaptation only), this normalization can be summarized by the equation 6 (see [3] [2] for more details).

$$\overline{\mu_{i,d}^X} = \frac{D_{ref}}{D(\lambda_X, \lambda_\Omega)}\mu_{i,d}^X + \left(1 - \frac{D_{ref}}{D(\lambda_X, \lambda_\Omega)}\right)\mu_{i,d}^\Omega \qquad (6)$$

4. Support Vector Machines

4.1. Principle

SVMs [14] are classifiers used to find the *best* separator between two classes. They are very efficient in solving clustering problems which are not linearly separable. The fundamental idea is to project (using a mapping function ϕ) the input vectors into a new feature space of greater dimension in which it is possible to find a hyperplane producing a linear separation between classes.
In practice, SVMs use kernel functions to perform the computation of scalar products in the feature space without using the definition of ϕ. The *best* hyperplane is chosen in order to maximize the distance between the separating hyperplane and the training vectors the closest to the border: they are called support vectors. The classification of a sample x is given by equation 7

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i k(x, x_i) + b \tag{7}$$

where k is the kernel function, x_i are the training samples and $y_i \in \{-1, +1\}$ their respective class labels, α_i and b are the parameters of the model obtained after training. At test time, $f(x)$ is compared to a threshold θ.

4.2. SVM in the GMM space

In this paper, we propose to apply SVM directly in the GMM space. The idea is to train a SVM for each client X in order to separate the model λ_X from the rest of the world. In this purpose, pseudo-impostors models λ_{PI_i} are introduced. Typically, they are models adapted from the world model λ_Ω using subsets of the original world model dataset. GMMs are normalized following the equation 6 and the best hyperplane separating the client model λ_X (class +1) from the pseudo-impostors models λ_{PI_i} (class -1) gives the classification function f_X of client X. We used the probabilistic distance kernel given by equation 8, which is a particular case of kernels introduced in [10]:

$$k(\lambda_X, \lambda_Y) = \exp\left(-D^2(\lambda_X, \lambda_Y)\right) \tag{8}$$

where D was previously defined in equation 4. This method was first introduced and successfully applied for speaker verification by Dehak et al. in [9].

5. Experiments

5.1. The BANCA database

The BANCA audiovisual database contains 52 speakers divided into 2 groups G1 and G2 of 26 speakers each (13 females and 13 males). This division into 2 disjoint groups allows to use G2 as the world model when testing on G1 (and reciprocally). In the following, *world model* always refers to the group which is not currently tested. 12 sessions were recorded in 3 different conditions (controlled, adverse and degraded). In each session and for each speaker, 2 recordings were performed: one client access where the speaker pronounces digits and his/her (fake) own address and one impostor access where he/she pronounces digits and the address of another person.

5.2. Protocols

The experiments are performed following the Pooled BANCA protocol. The face space (needed for both the face tracking algorithm and the eigenface projection [5]) is built using all faces from the world model. Automatic face tracking is then performed on every video of the BANCA database and 80 eigenface coefficients are extracted from each frame (about 450 frames or more per video). The world GMM is learned on the features extracted from the videos of the world model. One pseudo-impostor GMM per video of the world model is adapted from the world GMM: this makes about 600 pseudo-impostor GMMs. For each subject, a GMM is adapted from the world GMM using only the features extracted from the client access of one controlled session (4-fold cross validation is achieved by using successively the 4 controlled session for client GMM training). Therefore, 1248 imposter and 936 client accesses are performed for each group: confidence at 95% is less than 3%.

A simple face recognition algorithm based on only 5 pictures randomly extracted from the recordings was also tested as a way of calibrating the difficulty of the BANCA Pooled protocol. It uses the same eigenface features and computes, at test time, the minimum euclidean distance between the five feature vectors of the client model and the five feature vectors of test. Note that, as in the SVM-GMM case, faces are located automatically (the annotation given in BANCA are not used): this might explain the poor performance of this simple algorithm (in comparison to the results obtained in the literature [6]).

5.3. Tools

GMM training, adaptation and scoring are performed using the open-source software *BECARS* [4]. SVM training and scoring are performed using the library *libSVM* [7].

6. Results

Figure 3 shows that the GMM algorithm outperforms the simple *minimum euclidean distance* algorithm. Hence,

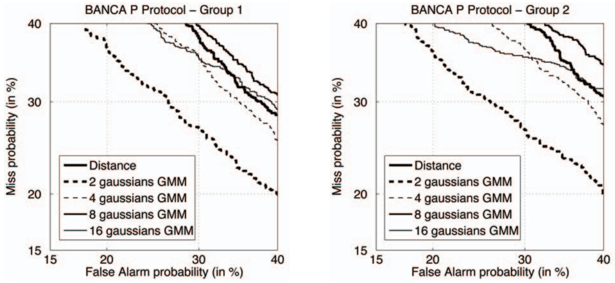

Figure 3. Euclidean distance vs. GMM

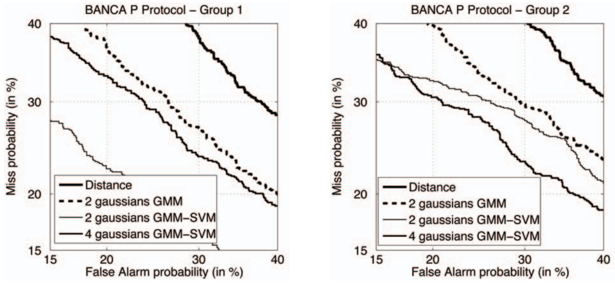

Figure 4. GMM vs. GMM-based SVM

the *distance* algorithm gets a 34% Equal Error Rate (EER) when averaged on groups G1 and G2. The best GMM algorithm leads to a 29% EER, which is obtained when only 2 gaussians per model are used. This might be explained by the fact that the training dataset available for each client is very small and therefore does not allow a good estimation of additional gaussian parameters.

Figure 4 shows the improvements given by the use of GMM-based SVM for face recognition. In average (on groups G1 and G2), it brings a significant 3.5% EER improvement. Hence, the GMM-SVM algorithms leads to a 25.5% EER with 2 gaussians, and 26.8% with 4 gaussians.

7. Conclusions and future work

A fully automatic face recognition system has been presented in this paper. We showed that using every frame of the video in a GMM framework outperforms a simple system based on a distance between features extracted from only a few frames. The main contribution of this paper stays in the use of SVM in the GMM space. It brings another 3.5% equal error rate improvement.

Many points have yet to be improved. For instance, no normalization pre-processing was applied on the automatically tracked face: head rotation normalization and histogram equalization might drastically improve performances. Moreover, we expect to improve the GMM-SVM system by using more than one client model: training a SVM with only one example in one class can lead to inaccurate training.

Finally, the use of face and voice combined in a talking-face modality for identity verification is still a great challenge and we plan to improve the system by adding a speaker verification step, by fusing score and/or audiovisual features.

References

[1] E. Bailly-Baillière, S. Bengio, F. Bimbot, M. Hamouz, J. Kittler, J. Mariéthoz, J. Matas, K. Messer, V. Popovici, F. Porée, B. Ruiz, and J.-P. Thiran. The BANCA Database and Evaluation Protocol. In *Lecture Notes in Computer Science*, volume 2688, pages 625 – 638, January 2003.

[2] M. Ben. *Approches Robustes pour la Vérification Automatique du Locuteur par Normalisation et Adaptation Hiérarchique*. PhD thesis, University of Rennes I, 2004.

[3] M. Ben and F. Bimbot. D-MAP: a Distance-Normalized MAP Estimation of Speaker Models for Automatic Speaker Verification. In *IEEE-ICASSP*, volume 2, 2003.

[4] R. Blouet, C. Mokbel, H. Mokbel, E. Sanchez, and G. Chollet. BECARS: a Free Software for Speaker Verification. In *ODYSSEY 2004*, pages 145 – 148, 2004.

[5] H. Bredin, G. Aversano, C. Mokbel, and G. Chollet. The Biosecure Talking-Face Reference System. In *2nd Workshop on Multimodal User Authentication*, May 2006.

[6] F. Cardinaux, C. Sanderson, and S. Bengio. Face Verification Using Adaptative Generative Models. In *IEEE Int. Conf. on Automatic Face and Gesture Recognition*, 2004.

[7] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[8] T. Cootes, G. Edwards, and C. Taylor. Active Appearance Models. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 681 – 685. IEEE, June 2001.

[9] N. Dehak and G. Chollet. Support Vector GMMs for Speaker Verification. In *IEEE Odyssey 2006*, 2006.

[10] P. J. Moreno, P. P. Ho, and N. Vasconcelos. A Kullback-Leibler Divergence Based Kernel for SVM Classification in Multimedia Applications. In *NIPS*, 2003.

[11] G. Potamianos, C. Neti, G. Gravier, A. Garg, and A. W. Senior. Recent Advances in the Automatic Recognition of Audiovisual Speech. In *IEEE*, volume 91, pages 1306 – 1326, September 2003.

[12] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn. Speaker Verification using Adapted Gaussian Mixture Models. *Digital Signal Processing*, 10:19 – 41, 2000.

[13] M. Turk and A. Pentland. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, 3(1):71 – 86, 1991.

[14] V. Vapnik. *The Nature of Statistical Learning Theory*. Statistics for Engineering and Information Science. Springer Verlag, Berlin, 2000.

[15] P. Viola and M. Jones. Robust Real-Time Object Detection. *Int. Journal of Computer Vision*, 2002.