A Comparison of Face Verification Algorithms using Appearance Models

H.Kang, T.F.Cootes, C.J. Taylor
Dept. Imaging Science and Biomedical Engineering
University of Manchester, Manchester M13 9PT U.K.

Abstract

Statistical models of shape and appearance have been successfully used in face modeling, tracking and synthesis. In this paper we describe experiments using appearance models for face verification. We compare a variety of different algorithms on a standard face database (XM2VTS). We demonstrate that simple methods of correction for head pose and face expression can significantly improve results.

1 Introduction

There is a great deal of interest in using computer vision for face verification (asking the question 'is this an image of person X'). In particular, statistical models of appearance, which can synthesise both face shape and texture, are becoming popular as they can encode a face in a relatively compact parameter vector and fast algorithms have been developed for matching such models to new images [1, 2].

In this paper we describe the results of experiments with such statistical appearance models, demonstrating their performance on a standard face database (XM2VTS [3]) with a standard protocol [4].

We try a variety of different metrics for comparing two face parameter vectors, and demonstrate that relatively simple methods for correcting the effects of pose and expression can lead to significant improvements in performance.

2 Background

Face recognition has been well studied in last decade and many algorithms have been developed in this area. The more detailed surveys see Ref [5] [6]. The procedure and some popular algorithms for face recognition/classification are summarised in Figure 1.

There are three major components to a system, the model, the normalisation/subspace correction method and the classification used. Different models can represent a face in different ways. A common approach is to use an eigenface model [7] which is based on linearly projection of the face image from image space to a low dimensional feature space using principle components analysis (PCA). An alternative is to explicitly model shape in a statistical appearance model [2]. Either approach gives a compact vector to represent a face.

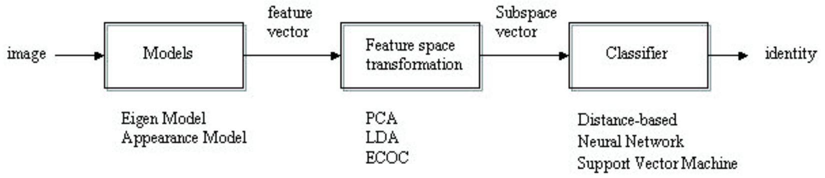

Figure 1: The procedure and some popular methods for face recognition/verification

Feature vectors are usually then projected into lower dimensional subspaces to minimise within class variations. A common approach is linear discriminant analysis (LDA) which was applied to recognising faces by Belhumeur [8]. He adopted Fisher's Linear Discriminant (FLD [9]) algorithm to project the face vector to a subspace (fisherface) to get the maximum ratio of the between class and the within class variance. As the face classification / recognition is a multi-class problem, the error correcting output coding (ECOC) was introduced for face classification by Kittler [10], in which the face vectors were mapped to a well separated code space using Multi-Layer Perceptrons (MLPs). Each code has an equal Hamming distance.

The final component is the classifier. Various classifiers have been explored to improve the accuracy of face classification. The basic approach is to use distance-base methods which measure Euclidean distance between any two vectors and then compare with a preset threshold. Based on LDA space, new metrics such as normalised correlation and optimal matching score have been considered [11]. Neural Networks are often used as classifiers due to their powerful generalization ability [12]. Support vector machines (SVMs) have been applied with encouraging results [13].

In this paper, we use appearance models to represent face image and experiment with LDA, pose and expression correction methods for space transformation. We compare the performance of different distant metrics and normalisation techniques.

3 Statistical Models of Appearance

A statistical appearance model contains models of the shape and grey-level appearance of the object of interest which can 'explain' almost any valid example in terms of a compact set of model parameters. The appearance model is built based on a set of labelled images, where key landmark points are marked on each example object. The marked examples are aligned to a common co-ordinate and each can be represented by a vector,**x**. Applying a principal component analysis (PCA) to the data, the shape model can be written as

$$\mathbf{x} = \bar{\mathbf{x}} + \mathbf{P_s}\mathbf{b_s} \qquad (1)$$

where $\bar{\mathbf{x}}$ is the mean shape, $\mathbf{P_s}$ is a set of orthogonal models of variation and $\mathbf{b_s}$ is a set of shape parameters.

After warping the texture within the region of interest to the mean shape, the texture vector $\mathbf{g}$ (a raster scan of the grey-levels) can be similarly modelled as

$$\mathbf{g} = \bar{\mathbf{g}} + \mathbf{P_g b_g} \tag{2}$$

where $\bar{\mathbf{g}}$ is the mean normalised grey-level vector, $\mathbf{P_g}$ is a set of orthogonal models of variation and $\mathbf{b_g}$ is a set of grey-level parameters.

A further PCA can be applied to the shape and texture parameters to obtain a combined appearance model

$$\begin{aligned} \mathbf{x} &= \bar{\mathbf{x}} + \mathbf{Q}_s\mathbf{c} \\ \mathbf{g} &= \bar{\mathbf{g}} + \mathbf{Q}_g\mathbf{c} \end{aligned} \tag{3}$$

where $\mathbf{c}$ is a vector of parameters controlling both shape and texture together and $\mathbf{Q}_s$,$\mathbf{Q}_g$ are matrices describing the modes of variation derived from the training set. For more details see [2].

A facial appearance model has been built up based on a database containing 1267 images of 103 different people. The trained model has covered 99.5% of the variation of the faces in training data set which includes both head pose and expression change. The model has 349 modes controlling facial appearance. The first three modes are displayed in Figure 2.

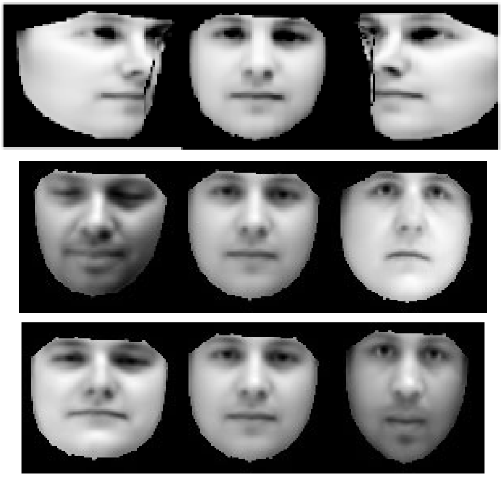

Figure 2: First three modes of the appearance model in ± 3 s.d.s

4 Protocol and Face Database

To evaluate the verification performance of the appearance model we need a large face database and a proper protocol. We adopt the XM2VTS face database [3] and the Lausanne Protocol [4]. In particular we choose the Lausanne Protocol-1 to perform our face verification test. Table 1 shows the structure of images used in the test. Three images of each of 200 clients form a registration set. In a test set there are two imags of each of the clients, and an additional 8 images of each of 70 imposters. This gives a total of 960 images in the test set.

	client/impostor	images per client/impostor	total images
Training Set	200	3	600
Testing Set	200	2	400
	70	8	560

Table 1: Lausanne Protocol configuration-1

All the images were marked up manually with 68 key points on each face. We match the models to these known points and extract the model parameters. This allows us to test the classification performance alone, without the confounding effect of search errors. The full protocol requires the use of an evaluation set to learn thresholds, which should subsequently be used with the test set. At the time of writing we have not annotated the (hundreds of) images in the evaluation set, so cannot demonstrate the full protocol. Instead, we generate ROC curve on the test set and report equal error rates (EER). This leads to results which can be compared with others published in the literature using different algorithms.

5 Algorithms for Face Verification

An appearance model can fit any new valid face image with a compact parameter vector $\mathbf{c}$ and synthesize a new image similar to the original with the parameters. Each face can therefore be represented by the parameters. We test if a new image contains the face of a claimed person by measuring a distance between the parameters representing the image and those representing the mean for the claimed person. If this is less than a threshold, the image is accepted as being of that person. There are, however, many ways of measuring the distance. In the following, we compare various different measures.

5.1 Measures of Distance

We consider Euclidean distance and (negative) dot product as basic distance measures between two vectors. Before measuring the distance we may normalise (by scaling so that $|\mathbf{c}| = 1$) or by weighting the elements by the reciprocal of the standard deviation of the model's modes.

Euclidean distance: $d = |\mathbf{x} - \mathbf{x}|$

Dot product: $d = -\mathbf{x}_1.\mathbf{x}_2$ (negative since we wish to optimise by minimising).

Normalisation: $\mathbf{x} \rightarrow \frac{\mathbf{x}}{|\mathbf{x}|}$

Weighting $\mathbf{x} \rightarrow \mathbf{W}^{-1}\mathbf{x}$

where $\mathbf{W} = diag(\sigma_1,\ \sigma_2,\ ...,\ \sigma_n)$ is a diagonal matrix containing the standard deviations of each element of $\mathbf{x}$ across the training set.

5.2 Linear Discrimination Analysis (LDA)

LDA is a popular algorithm for pattern classification which aims to find a linear subspace which maximises the between class separation and minimises the within class separation. More details can be found in [8],[14].

We use LDA on the 600 vectors from the training data to generate such a subspace. During testing we project the parameter vectors into this subspace before measuring distance.

5.3 Face Pose Correction

Normally we use frontal faces as registration images. If the test image is not a frontal face, it can degrade the face verification performance. The face pose change includes head rotation (left - right) and nodding (up - down). To eliminate the effect from face pose change, we need to correct non-frontal faces back to frontal faces. A simple model has been shown to be sufficient [15]. A linear correlation between image vector $\mathbf{c}$ and sin/cos of view angle θ is learnt from a suitable training set:

$$\mathbf{c} = \mathbf{c}_0 + \mathbf{c}_c \cos(\theta) + \mathbf{c}_s \sin(\theta) \tag{4}$$

where $\mathbf{c}_0$, $\mathbf{c}_c$ and $\mathbf{c}_s$ are coefficient vectors derived from the training data. We refer to this as a rotation estimator. The rotation angle of any new face can be easily estimated and an unseen view of the face within -30^{o}to 30^{o}can be corrected. Similarly, a nod estimator has been built to correlate the image vector $\mathbf{c}$ and the view angle change about a horizontal axis.

Figure 3 shows images synthesized using the rotation and nod estimators. Within the range of angle θ (rotation: -30^{o}to 30^o, nod: -20^{o}to 20^o), any new face can can be easily corrected to frontal by setting the angle to zero in both directions (horizontal and vertical). This can be shown to significantly improve the performance of a recognition system.

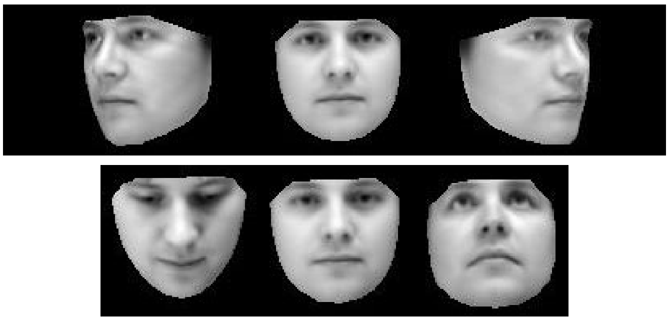

Figure 3: Synthesizing new views using head rotation and Nod Estimators

5.4 Expression Component Removal

Another major source of face variation that can affect face verification is the face expression change. This can be modelled explicitly and such a model used to correct for expression variation.

With expression images of each person in the training set, the within class covariance matrix $\mathbf{S}_w$ can be computed. Eigen analysis leads to a model of individual face variation, which can be written as:

$$\mathbf{c} = \mathbf{c}_{mean} + \mathbf{P}_{exp}\mathbf{b}_{exp} \tag{5}$$

where the $\mathbf{c}_{mean}$ is the mean face for an individual, $\mathbf{P}_{exp}$ is a matrix of eigen vectors of the covariance matrix $\mathbf{S}_w$ and $\mathbf{b}_{exp}$ is a set of parameters which modify the expression.

From the training set described above we obtain a model with 149 modes. The first three modes are displayed in Figure 4. Though such a model is predominantly expression, some lighting variations are also included (details depend on the nature of the variation in the training set).

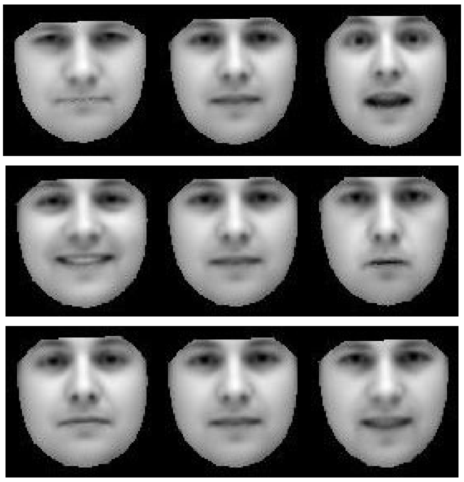

Figure 4: First three modes of expression model

To a good approximation, expression changes are orthogonal to the changes due to identity in the framework [16]. For i^{th} new image vector $\mathbf{c}^i$, the expression model parameter $\mathbf{b}^i_{exp}$ can be calculated with the Equation (6)

$$\mathbf{b}^i_{exp} = \mathbf{P}^{\mathrm{T}}_{exp}\mathbf{c}^i \tag{6}$$

To correct the expression change, we subtract the expression component ($\mathbf{P}_{exp}$ $\mathbf{b}^i_{exp}$) from the original image vector $\mathbf{c}^i$. The corrected neutral face $\mathbf{c}^i_{neutral}$ is given:

$$\mathbf{c}^i_{neutral} = \mathbf{c}^i - \mathbf{P}_{exp}\mathbf{b}^i_{exp} \tag{7}$$

Figure 5 shows an example, in which the left is the original test image with surprise expression, the middle is reconstructed image after expression removal and the right is the registration image with neutral expression. If we use (negative) dot product as the measure of distance between any two image vectors, the distance between the original test image (left)and the registration image (right)is -3.36, the distance between corrected image (middle) and the registration image (right) becomes -11.72 . The measure has been significantly reduced, which means that the corrected image is much closer to the neutral image.

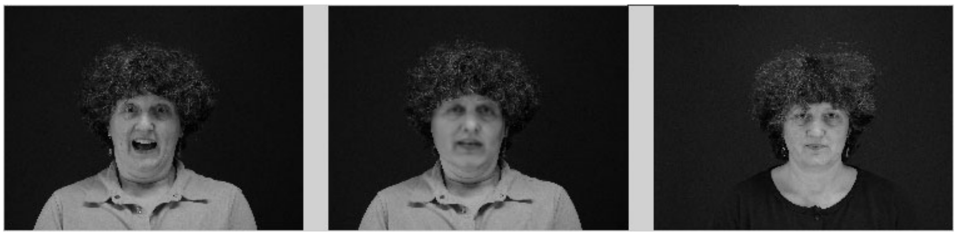

Figure 5: An example of the expression removal algorithm

Below we demonstrate that this algorithm can successfully correct expression change, leading to an improvement in verification performance.

6 Experiment Results

6.1 Measures of Distance

The test result on the combination of basic distance metrics and vector preprocessing has been shown in Table 2. We report the Equal Error Rate (EER) over the test image set to compare the performance of different distance metrics.

Metrics	Normalisation	Weighted	Un-weighted
Euclidean	$\lvert\mathbf{x}\rvert = 1$	6.0%	8.5%
	-	9.8%	10.7%
- dot product	$\lvert\mathbf{x}\rvert = 1$	6.0%	8.5%
	-	**5.5%**	16.1%

Table 2: Test Result in original feature space

Table 2 shows that the best result is obtained with the un-normalised, weighted dot product.

6.2 Linear Discrimination Analysis (LDA)

By projecting the test images to LDA subspace, the vector dimension has been reduced from 349 to 199. Repeating the same experiment, measuring distance in the reduced space leads to the results summarised in Table 3.

Metrics	Normalisation	Weighted	Non-weighted
Euclidean	$\|\mathbf{x}\| = 1$	**4.6%**	6.8%
	-	7.9%	8.8%
- dot product	$\|\mathbf{x}\| = 1$	**4.6%**	6.8%
	-	6.1%	13.5%

Table 3: Test Results on LDA

Comparing Table 3 against Table 2, except un-normalised dot product, all the EER have been reduced. In this case, with normalised and weighted vectors the EER can be reduced by 1.4%.

6.3 Face Pose Correction

Table 4 shows the results of face pose correction. In this experiment all the vectors have been weighted inversely by model's standard deviation.

Pose Correction	LDA	Normalisation	EER
Yes	No	$\|\mathbf{x}\| = 1$	6.0%
Yes	No	-	5.6%
Yes	Yes	$\|\mathbf{x}\| = 1$	**3.7%**
Yes	Yes	-	5.7%

Table 4: Test Result on Pose Correction

6.4 Expression Component Removal

Applying the expression component removal method to the test images before face classification, we can find a further improved the result in Table 5. With the expression component removal and pose correction the EER goes down to 2.6%.

Expression correction	LDA	Pose Correction	EER
Yes	No	Yes	**2.6%**
Yes	No	No	2.8%
Yes	Yes	Yes	5.1%
Yes	Yes	No	4.5%

Table 5: Performance after removing expression effects

Comparing Table 5 with Table 3, we can see that applying methods of pose correction and expression component removal to the test images before face classification can achieve a better result (EER=2.6%) than that of using LDA alone (EER=4.6%).

7 Results Comparison with Other Groups

In section 2, we briefly reviewed several other algorithms. Table 6 lists some of the other results obtained on the same database with the same protocol.

	FR	FA	TE	EER
Pose and Expression Correction [this]	-	-	-	2.6%
Normalised dot + hist.eq. + LDA [11]	2.25%	2.56%	4.81%	
Localised dot (s_o) + hist.eq. + LDA [11]	1.75%	1.70%	3.45%	
SVM (with LDA) [13]	1.37%	0.75%	1.06%	
ECOC [10]	0.75%	1.25%	-	

Table 6: Comparison with Other Techniques

Where FR is false reject rate, FA is false acceptance rate , TE the total error when using a preset threshold [4]. The method we propose outperforms the best distance measures used on the eigenface represented in [11]. Part of this improvement comes from the explicit use of shape variation in the appearance models. However, our method is significantly outperformed by the nonlinear methods (SVM and ECOC). Clearly, there is scope for using such methods on the corrected appearance model parameters to further improve results.

8 Conclusions and Future Work

We have described the performance of statistical models of shape and texture in a face verification task. Our results confirmed the related work by Kittler [11] that the normalised correlation measure (the dot product of unit vectors) is the best simple measure of distance for faces. We have shown that explicitly correcting for head pose and expression change leads to better results than using an LDA approach. The best result we obtain (2.6%) seems to be the best reported on the data set using simple single classifier techniques. However, more complex classification algorithms such as SVMs [13], ECOC [10] and classifier combination algorithms have produced even better results. The latter methods have typically been applied to eigen-face type features (not explicitly taking shape change into account). In further work we will investigate whether such techniques will improve the performance of an appearance model approach.

References

[1] G.J. Edwards, C. J. Taylor, and T. F. Cootes. Interpreting face images using active appearance models. In 3^{rd} *International Conference on Automatic Face and Gesture Recognition 1998*, pages 300–305, Japan, 1998.

[2] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. In H.Burkhardt and B. Neumann, editors, 5^{th} *European Conference on Computer Vision*, volume 2, pages 484–498. Springer, Berlin, 1998.

[3] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre. Xm2vtsdb: The extended m2vts database. In *Proc. 2nd Conf. on Audio and Video-based Biometric Personal Verification*. Springer Verlag, 1999.

[4] J. Luettin and G. Maitre. Evaluation protocol for the extended m2vts database. Technical report idiap-com 98-5, Dall Molle Institute for Perceptual Artificial Intelligence, P.O. Box592, Martigny, Valais, Switzerland, 1998.

[5] R. Chellappa, C. Wilson, and S. Sirohey. Human and manchine recognition of faces: A survey. *Proceedings of the IEEE*, 83:705–740, 1995.

[6] A. Samal and P. Iyengar. Automatic recognition and analysis of human faces and facial expression: A survey. *Pattern Recognition*, 25:65–77, 1992.

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

[8] P. N. Belhumeur, J. P. Hespanha, and D. J. Kreigman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. 4th *European Conference on Computer Vision*, 1:45–58, 1996.

[9] R.Fisher. The use of multiple measures in taxonomic problems. *Ann. Eugenics*, 7:179–188, 1936.

[10] J. Kittler, R. Ghaderi, T. Windatt, and J. Matas. Face veirfication using error correcting output codes. In Tim Cootes and Chris Taylor, editors, 12th *British Machine Vison Conference*, volume 2, pages 593–602, 2001.

[11] J. Kittler, Y.P. Li, and J. Matas. On matching scores for lda-based face verification. In M. Mirmehdi and B. Thomas, editors, 11th *British Machine Vison Conference*, volume 1, pages 42–51, Bristol, UK, September 2000. BMVA Press.

[12] S. Lawrence, C. Giles, A. Tsoi, and A. Back. Face recognition: A convolutional neural network approach. *IEEE Transactions on Neural Network and Pattern Recognition*, 8:98–113, 1997.

[13] K. Jonsson, J. Kittler, Y. Li, , and J. Matas. Support vector machines for face authentication. In 10th *British Machine Vison Conference*, pages 543–553, 1999.

[14] D.A. Devijiver and J. Kittler. Pattern recognition: A statistical approach, 1982.

[15] T. F. Cootes, K. N. Walker, and C. J. Taylor. View-based active appearance models. In 4th *International Conference on Automatic Face and Gesture Recognition 2000*, pages 227–232, Grenoble,France, 2000.

[16] N. Costen, T. F. Cootes, and C. J. Taylor. Compensating for ensemble-specificity effects when building facial models. In M. Mirmehdi and B. Thomas, editors, 11th *British Machine Vison Conference*, volume 1, pages 62–71, Bristol, UK, sep 2000. BMVA Press.