Robust Face Recognition through Local Graph Matching

Ehsan Fazl-Ersi[1], John S. Zelek[2] and John K. Tsotsos[3]
[1,3]Department of Computer Science and Engineering, York University, Toronto, Canada
E-Mail: [efazl, tsotsos]@cse.yorku.ca

[2]Department of Systems Design Engineering, University of Waterloo, Waterloo, Canada
E-Mail: jzelek@uwaterloo.ca

***Abstract*— A novel face recognition method is proposed, in which face images are represented by a set of local labeled graphs, each containing information about the appearance and geometry of a 3-tuple of face feature points, extracted using Local Feature Analysis (LFA) technique. Our method automatically learns a model set and builds a graph space for each individual. A two-stage method for optimal matching between the graphs extracted from a probe image and the trained model graphs is proposed. The recognition of each probe face image is performed by assigning it to the trained individual with the maximum number of references. Our approach achieves perfect result on the ORL face set and an accuracy rate of 98.4% on the FERET face set, which shows the superiority of our method over all considered state-of-the-art methods.**

***Index Terms*—Local Feature Analysis (LFA), Gabor wavelet, Principal Component Analysis (PCA), Gaussian Mixture Models (GMM), ORL database, FERET database.**

I. Introduction

In recent years face recognition has received substantial attention from both research communities and the market, but has still remained very challenging in real applications. A large number of face recognition algorithms, along with their modifications, have been developed during the past decades which can be generally classified into two categories: holistic approaches and local feature based approaches. The major holistic approaches developed for face recognition are Principal Component Analysis (PCA), combined Principal Component Analysis and Linear Discriminant Analysis (PCA+LDA), and Bayesian Intra-personal/Extra-personal Classifier (BIC). PCA [1] computes a reduced set of orthogonal basis vectors, called eigenfaces, from the training face images. A new face image can be approximated by a weighted sum of these eigenfaces. PCA+LDA [2] provides a linear transformation on PCA-projected feature vectors, by maximizing the between-class variance and minimizing the within-class variance. The BIC algorithm [3] projects the feature vector onto extra-personal and intra-personal subspaces and computes the probability that each feature vector came from one or the other subspace.

In the local feature based approaches developed for face recognition, one widely influential work is that of Wiskott et al. [4], called Elastic Bunch Graph Matching (EBGM). By taking advantage of the fact that all human faces share a similar topological structure, EBGM represents faces as graphs, with the nodes positioned at fiducial points (e.g., eyes, nose) and the edges labeled with the distances between the nodes. Each node contains a set of 40 complex Gabor wavelet coefficients at different scales and orientations, which are called a Gabor Jet. The identification of a new face consists of determining among the constructed graphs, the one which maximizes the graph similarity function.

In contrast to EBGM, most of the available feature based approaches perform single feature matching for recognition (e.g., [5]). In training, a large set of features are extracted from the training images of each individual, and then in recognition a nearest neighbourhood classifier is used to assign a training feature to each test feature. Each of the training features belong to a certain individual and therefore, the probe image is assigned to the most referenced trained individual.

Motivated in part by the work of Wiskott et al. [4], in this paper we propose a novel technique for face recognition which takes advantage of the fact that a single feature can be confused with other features at a local scale; however, the ambiguity is less likely if we consider groups of features. Like the work of Wiskott's group, our approach compares faces using a combination of local features. However, unlike that approach, we do not use a pre-defined set of features and a complex graph matching process for locating the features. In our technique, face images are represented by a set of 3-node labeled graphs, each containing information on the appearance and geometry of a 3-tuple of face feature points, where feature points are extracted using the LFA technique, and each extracted feature point is described by a Gabor Jet.

This work is the extension of the paper titled "Local Graph Matching for Face Recognition", by E. Fazl-Ersi and J. Zelek, which appeared in the proceedings of the Eighth IEEE Workshop on Applications of Computer Vision 2007, Austin, Texas, USA.

Our method automatically learns a model set and builds a graph space for each individual. A two-stage method for fast matching is developed, where in the first stage a Bayesian classifier based on PCA factorization is used to efficiently prune the search space and select very few candidate model sets, and in the second stage a nearest neighborhood classifier is used to find the closest model graphs to the query image graphs. Each matched image graph votes for the possible identity of the probe face image and the recognition is performed based on the number of votes each individual obtains during the matching.

The remainder of this paper is structured as follow: in Section II, we introduce the 3-node labeled graphs and the way we extract them from face images; the learning and recognition phases of our method are described in detail in Sections III and IV, respectively; in Section V several experimental results on the ORL and FERET face datasets are reported, and finally Section VI concludes the paper.

II. IMAGE GRAPHS

Faces frequently distinguish themselves not by the properties of individual features, but by the contextual relative location and comparative appearance of these features. A tractable and efficient way for modelling this is to employ image graph models. Graphical models have been successfully used in pattern recognition and computer vision as a powerful and flexible representation mechanism (e.g., [4], [6]). In our approach, we represent face images using a set of local graphs with 3 nodes and 3 edges, where nodes are distinctive feature points of the face image, labelled with their description vectors, and edges (lines connecting the nodes) are labelled with distances between their end nodes. In our system feature points are extracted using the LFA technique, and each extracted feature point is described by a Gabor Jet. In the following sub-sections, we briefly describe the feature extraction and description techniques used in our system, and then discuss the graph properties and the way we extract the graphs from training and probe face images.

A. Local Feature Analysis (LFA)

The statistical Local Feature Analysis (LFA) technique is used in our method to extract a set of feature points from each face image, at locations with highest deviations from the statistical expected face. LFA defines a set of topographic, local kernels that are optimally matched to the second-order statistics of the input ensemble [14]. Given the zero-mean matrix X of n vectorized face images[1] with normalized energy, the eigenvalues of the covariance matrix XX^T are calculated and the first k largest eigenvalues, $\lambda_1 \ldots \lambda_k$, and their associated eigenvectors, $e_1 \ldots e_k$ are selected. Penev and Atick [14] defined a set of kernel, K as:

$$K(x,y) = \sum_{r=1}^{k} e_r(x) \frac{1}{\lambda_r} e_r(y) \tag{1}$$

The rows of K contain the LFA kernels, which have spatially local properties and are topographic in the sense that they are indexed by spatial location (see Fig.1). The kernel matrix K transforms X to the LFA output $O = KX^T$, which inherits the same topography as the input space.

LFA produces an n dimensional representation, where n is the number of pixels in the image. Since the n outputs are described by $k<<n$ linearly independent variables, there are residual correlations in the output. Penev and Atick [14] showed that the residual correlation of the outputs can be obtained using:

$$P(x,y) = \sum_{r=1}^{k} e_r(x) e_r(y) \tag{2}$$

Each point x_m in the output array $O(x)$ is correlated with other outputs via $P(x,x_m)$, so it can predict the other outputs to some extent, using the following equation [14]:

$$O_{pred}(x) = \frac{P(x,x_m)}{P(x_m,x_m)} O(x) \tag{3}$$

Penev and Atick [14] used this property and proposed an iterative sparsification algorithm for reducing the dimensionality of the representation by choosing a subset M of outputs that were as decorrelated as possible. At each time step, the output point that is predicted most poorly by multiple linear regressions on the points in M is added to M. The reconstruction (prediction) of $O(x)$ at m^{th} step is achieved by:

$$O_{rec}(x) = \sum_{m=1}^{|M|} a_m(x) O(x_m) \tag{4}$$

with:

$$a_m(x) = \sum_{l=1}^{|M|} P(x,x_l) P^{-1}(x_l,x_m) \tag{5}$$

In our system, for an arbitrary face image, the points selected through the sparsification of its LFA outputs are considered as the most distinctive features of that face image, and are used for learning and recognizing the face. The result of applying the sparsification algorithm to a sample image is shown in Fig.2. The locations and the order of the first 20 points show that the outputs with the largest deviations from the expectation (e.g., the most unusual features of the face) are selected first.

B. Gabor Jet

Each feature point in our system is described by a Gabor Jet, i.e., a set of convolution coefficients for Gabor wavelet kernels of different orientations and frequencies at the location of the feature in the image. Gabor wavelets

[1] All face images used to derive the LFA kernels, were first rectified using eye coordinates, and cropped with a semi-elliptical mask to exclude non-face area. Furthermore, the grey histograms over the face area in each face image were equalized. This preprocessing procedure is applied to all face images (gallery/probe) used in our experiments (presented in Section IV).

are biologically motivated convolution kernels in the shape of plane waves, restricted by a Gaussian envelope function [4]. The general form for a 2D Gabor wavelet is:

$$\psi_{s,o}(\vec{r}) = \frac{k_{s,o}^2}{\sigma^2}\exp\left(-\frac{k_{s,o}^2 r^2}{2\sigma^2}\right)\times\left[\exp(i\vec{k}_{s,o}\vec{r}) - \exp(-\frac{\sigma^2}{2})\right] \qquad (6)$$

where σ is a parameter to control the scale of the Gaussian (in our experiments $\sigma = \pi$) and $k_{s,o}$ is a 2D wave vector whose magnitude and angle determine respectively the frequency, *s*, and the orientation, *o*, of the Gabor kernel.

Similar to [4], in our method wavelet responses at 5 frequencies (in which $|k_{s,o}| = \{\pi/2, \pi/\sqrt{8}, \pi/4, \pi/\sqrt{32}, \pi/8\}$) and 8 orientations (varying in increments of $\pi/8$ from 0 to $7\pi/8$) are used for description, resulting in 40-element description vectors.

C. Graph Properties

The most important and distinctive property of each graph is its appearance, which is computed from the description vectors of the graph nodes. To compute this, one option is to concatenate the description vectors (Gabor Jet) of the graph nodes into a single description vector, representing the appearance of the graph.

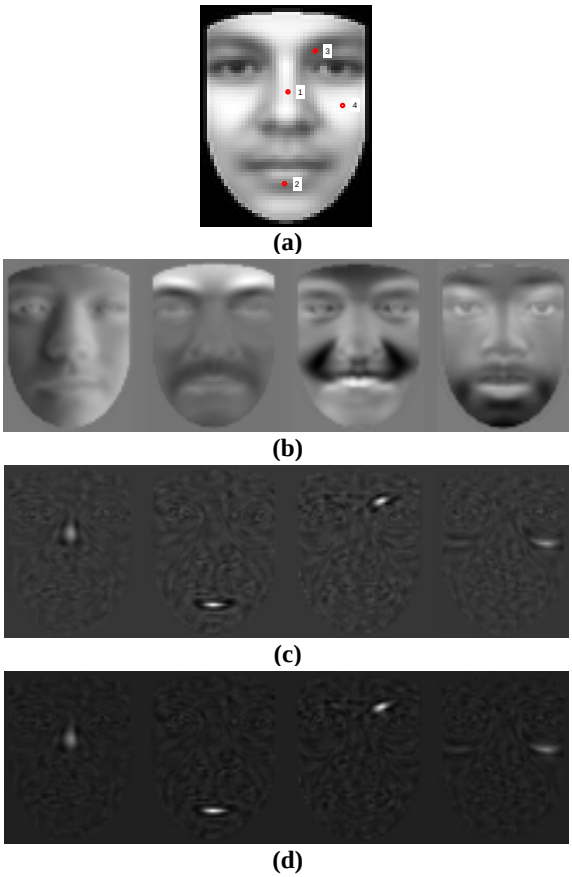

Figure 1. K(x,y) and P(x,y) derived from a set of 1196 face images. The average of the face images (the statistical expected face) is shown in (a), marked with four positions 1-4, (b) shows the first four derived PCA kernels, (c) and (d) show K(x,y) and P(x,y) respectively, at the four marked positions.

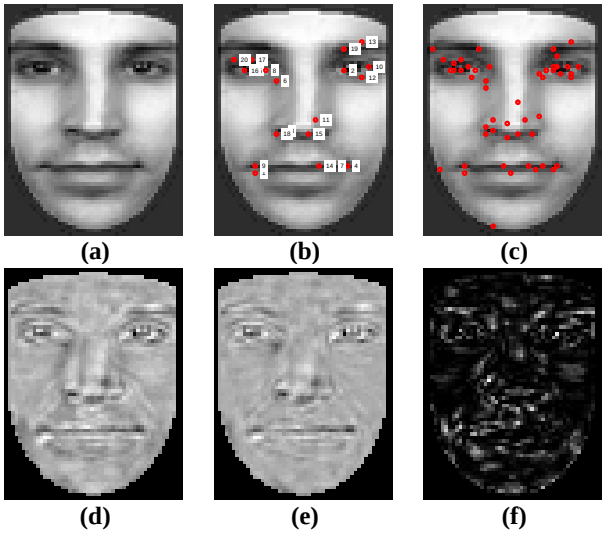

Figure 2. (a) shows the original image, (b) illustrates the first 20 selected points overlaid on the original image and numbered sequentially, (c) illustrates the first 50 selected points, (d) shows the LFA output of the original image, , (e) shows the reconstructed output, and (f) shows the reconstruction error scaled from 0 to 255, respectively.

Although this technique is simple and straightforward, the high dimensionality of the resulting appearances (3x40) restricts its applicability in the matching, particularly when the dataset of model graphs, created from the training images, is very large. A very powerful alternative to this is to model the joint distribution of the nodes' appearances through a probabilistic framework. Using this technique, not only is the graph described based on the description vectors of its individual nodes, the spatial relationships between the appearances of the graph nodes and the interaction between them are also taken into account. Besides the power of this appearance modelling scheme, a very fast matching is also inherently achieved. We will discuss further the modelling of the appearance of the graphs in more detail in Section III.

Another graph property which is used in our system as a descriptor for the graphs is graph geometry. Graph geometry, i.e., the way the three nodes in a graph are arranged spatially, could play an important role in discriminating graphs, particularly when the use of appearance alone causes some ambiguities. In our approach, the lengths of graph edges are used to describe the geometrical properties of each graph. The lengths of graph edges, e_i, are simply computed by measuring the Euclidean distances between the locations of each two end nodes.

D. Image Graph Extraction

Considering that we extract around 150 feature points from each face image, approximately 3.3e+6 (150x149x148) 3-node graphs could be generated for each image. Evaluating this number of graphs for each probe image would be very computationally expensive. On the other hand, even in the learning phase, where the computational time is not usually a crucial issue, this

could be a problem for the modelling of the appearance of the graphs (Section III.A). This necessitates the selection of only a subset of all possible graphs for the subsequent processing. One potential way is to define a graph radial threshold, R_{th}, and then consider only those graphs that the distance between each node of the graph with respect to the graph centre is lower than R_{th} (in our system R_{th} is set to a fixed value of 20, where the size of the face image is 112x92). This method is efficient, since it selects only those graphs in which the nodes are sufficiently close to each other, and also it averagely selects around 0.6% of the graphs, which is a very good reduction rate. However, this technique is reasonable to be used for graph extraction in the training stage only, not in the run time (recognition); since computing the graph centre and edge distances for all graphs of a probe image, is still a time consuming process. Therefore we need to find another graph selection method to be applied to the probe images. Since our graphs have three nodes, the most straightforward way for doing this is to apply a triangulation technique[2]. This has the advantage of selecting very small number of graphs which can be used as the representatives of the probe images[3].

III. LEARNING

In this section, we develop and discuss our approach for learning model graphs, extracted from training face images. In practice, given M training face images for N individuals, our learning algorithm works as follow:

1. Initialize N empty model sets.
2. For each training image I_i:
   a. Extract and describe its feature points (see Section II.A and II.B).
   b. Extract the image graphs (see Section II.D).
   c. Place the extracted graphs in one of the N model sets, according to the identity of the training face image I_i.
3. Construct an appearance model for each created model set from its model graphs.

All steps of the above algorithm have been discussed before, except the last step which is the constructing of an appearance model for each created model set. In the following we explain this step in detail.

A. Constructing Appearance Models

Given $G_i = \{(x_{i_1}, x_{i_2}, x_{i_3})\}$, representing the i^{th} 3-node graph extracted from a probe face image, where x_{i_1}, x_{i_2} and x_{i_3} are the description vectors of the graph nodes[4]; we are interested in obtaining the probability of G_i belonging to each model set (individual) to assign it to the one that maximizes the *posterior probability*:

$$R_{MAP} = \arg\max_n P(\{(x_{i_1}, x_{i_2}, x_{i_3})\} \mid C_n) \qquad (7)$$

This is called the Maximum a Posteriori (MAP) solution[5]. In the above equation, C_n is the appearance model or graph space for the n^{th} individual. The aim of this section is to develop a method for estimating C_n for each individual, using the model graphs extracted from its training images.

Modelling the total joint likelihood of all 3-node graphs of a model set, that is, to construct the graph space (appearance model), becomes a *(3xD)*-dimensional distribution, where D is the length of the description vector of each node, which is 40 in our approach (see Section II.B). To make the appearance model estimation more accurate and tractable, we need to apply a dimensionality reduction technique over the set of all description vectors, extracted from training images of all individuals. Principal Component Analysis (PCA) is a standard technique for dimensionality reduction and has been applied to a broad class of computer vision and machine learning problems. While PCA suffers from a number of shortcomings such as its implicit assumption of Gaussian distributions and its restriction to orthogonal linear combinations, it remains a popular method for dimensionality reduction due to its simplicity and low computational time. Given the set of all description vectors, X, PCA calculates the eigenvalues of the covariance matrix XX^T, and selects the first k largest eigenvalues and their associated eigenvectors to form the PCA projection matrix W. Using the PCA, the original 40-dimensional description vectors X are factorized into k-dimensional vectors S, where $S=W*X$. In our system we experimentally chose k to be 8, therefore the joint likelihood of the 3-node graphs becomes (3x8)-dimensional rather than (3x40)-dimensional, which is more manageable and tractable.

Given s_i the factorized description vector of x_i, we now estimate the $P(\{s_{i_1}, s_{i_2}, s_{i_3}\})$, rather than estimating $P(\{x_{i_1}, x_{i_2}, x_{i_3}\})$. This can be modelled for each model set by using parametric or non-parametric distribution estimation techniques. In our approach, we estimate the joint distributions of graph appearances for each individual by using a Gaussian Mixture Model (GMM) as a parametric approximation technique. In a Gaussian Mixture Model [10], different Gaussian distributions represent different domains of the data, and have different output characteristics; GMMs try to describe a complex system using combination of all the Gaussian clusters,

[2] A triangulation of a discrete set of points P is a subdivision of the convex hull of the points into simplices such that any two simplices intersect in a common face or not at all and the set of points that are vertices of the subdividing simplices coincides with P.

[3] Triangulation could not be a good choice for extracting graphs in the training stage, because then the number of extracted graphs from training images would not be sufficient for reasonably estimating the graph spaces (discussed in Section 3).

[4] Generally G_i should be represented by its all properties, however since in this section we are only working with the description vectors of its nodes, we represent G_i as: $G_i = \{(x_{i_1}, x_{i_2}, x_{i_3})\}$.

[5] Given that the classes C_n have equal prior probability.

instead of using a single Model. In mathematical terms, a GMM can be defined as:

$$P(x \mid \theta) = \sum_{i=1}^{K} w_i . N(x \mid \mu_i, \sigma_i) \quad (8)$$

where,

$$N(x \mid \mu_i, \sigma_i) = \frac{1}{\sigma_i \sqrt{2\pi}} . e^{-\frac{(x-\mu_i)^2}{2\sigma^2}}$$

are the components of the mixture, K is the number of components (Gaussians) in the mixture model, $\theta = \{w_i, \mu_i, \sigma_i^2\}_{i=1}^{K}$ consists of the means and variances of the Gaussians, and w_i is the weight of Gaussian i [10].

Since the family of mixtures of Gaussians is parametric, the density estimation problem can be defined more specifically as the problem of finding the parameter vector θ that specifies the models from which the data are most likely to be drawn (the mixing weight vector w should also be estimated). Among the available techniques for estimating the parameters of Gaussians, we choose to use the standard Expectation-Maximization (EM) technique [11]. The EM algorithm is an efficient iterative procedure to compute the Maximum Likelihood (ML) estimate in the presence of missing or hidden data. In ML estimation, we wish to estimate the model parameters (Θ) for which the observed data are the most likely:

$$\theta^* = \arg\max_{\theta} P(x \mid \theta) \quad (9)$$

Each iteration of the EM algorithm consists of two processes: The E-step (or expectation step), and the M-step (or maximization step). In the E-step, given the observed data and current estimate of the model parameters, the missing data are estimated. This is achieved using the conditional expectation. In the M-step, the likelihood function is maximized under the assumption that the missing data are known (For more details about the EM algorithm see [11]).

Once all the graph spaces for each individual's model set are estimated through Gaussian mixture models and Expectation-maximization technique, the learning is done and matching can be performed based on the constructed appearance models.

IV. MATCHING AND RECOGNITION

In this section, a two-stage method for optimal matching between the graphs extracted from a probe image and the trained model graphs is developed, where in the first stage a MAP solution is used to efficiently select the most likely individuals' model sets based on the appearances of the graphs, and in the second stage, a nearest neighbourhood classifier is used to enable correspondence with learned model graphs of the selected individuals' model sets, by incorporating the geometry of the graphs. In practice, given N model sets and their corresponding appearance models, where N is the number of training individuals, the following algorithm is applied on the probe image for matching and recognition:

1. Extract and describe the image's feature points (see section II.A and II. B).
2. Calculate the factorized description vectors for the extracted feature points by using the PCA projection matrix, W, learned during the training (see section III.A).
3. Extract the image graphs (see section II.B).
4. For each graph G_i, i = 1…N_G, (where N_G is the total number of extracted graphs from the probe image):
   a. Obtain the probability of G_i belonging to each appearance model, C_n, $P(G_i \mid C_n)$, and select the r model sets with highest $P(G_i \mid C_n)$ - r in our system is 5% of N.
   b. Incorporate the geometrical properties of G_i and search the learned instances (model graphs) of the selected model sets, picking the model graph with highest similarity to the test graph (Graph similarity function in terms of appearance and geometry is described later in this section).
   c. Vote for the identity of the model set that one of its model graphs is matched to the considered test graph.
5. Select the individual with the maximum number of votes.

All steps of the above algorithm have been discussed before, except step 4.b, which requires a similarity function based on appearance and geometry of the graphs, to enable correspondence with learned instances of the selected model sets. Given $G_i = \{(x_{i_1}, x_{i_2}, x_{i_3})\}$, a test graph, and $G_j = \{(x_{j_1}, x_{j_2}, x_{j_3})\}$, a model graph, the similarity between the appearances of the graphs can be simply calculated by averaging the similarity between the corresponding nodes' description vectors (x_{i_k} with x_{j_k} for k = 1… 3). By employing the *cosine* similarity measure[6], the appearance similarity function can be formulated as:

$$Y(G_i, G_j) = \frac{1}{3} \sum_{k=1}^{3} \frac{x'_{i_k} . x_{j_k}}{(x'_{i_k} . x_{i_k})^{1/2} . (x'_{j_k} . x_{j_k})^{1/2}} \quad (10)$$

As discussed in Section II.C, besides the appearance, the geometry can be also used as a descriptor for a graph. In our system, the lengths of graph edges are used to describe the geometrical properties of each graph. Now, given $G_i = \{(e_{i_1}, e_{i_2}, e_{i_3})\}$, a test graph, and $G_j = \{(e_{j_1}, e_{j_2}, e_{j_3})\}$, a model graph, the dissimilarity between the geometry of the graphs can be measured by:

$$Z(G_i, G_j) = \frac{1}{3} \sum_{k} \frac{(e_{i_k} - e_{j_k})^2}{(e_{j_k})^2} \quad (11)$$

[6] $S(x_1, x_2) = \frac{x'_1 . x_2}{(x'_1 . x_1)^{1/2} . (x'_2 . x_2)^{1/2}}$

By combining Equations 6 and 7, the final graph similarity function takes the form:

$$S(G_i, G_j) = Y(G_i, G_j) - \alpha Z(G_i, G_j) \quad (12)$$

where α determines the relative importance of appearance and geometry similarities (in our system α =0.3).

V. EXPERIMENTS

In order to validate the robustness of our technique for face recognition, several experiments were performed on two of the publicly available and widely used face datasets, the ORL [12] and the FERET [16].

Our first experiment was performed on the ORL face set. ORL contains 400 images from 40 individuals (10 for each), with variations in pose, facial expression and a certain amount of scale and viewpoint (see Fig.3 for an example). The first five images for each individual have been used for training and the other five images for testing. PCA was applied to a dataset of 30,000 feature description vectors extracted from training images (150 features form each of the 40x5 training images), and the dimensionality of feature vectors was reduced from 40 to 8 (with 72.3% of non-zero eigenvalues retained). In graph space estimation, mixture models with 10 Gaussians were used for each individual, to model the joint distributions of the nodes' factorized description vectors. Our system achieved an accuracy rate of 100% in this experiment on the ORL face dataset (see [17] for a comparative study of several face recognition techniques on the ORL dataset).

Although the ORL dataset is one of the most challenging publicly available face datasets, the small number of contained individuals (40) cannot provide a good estimate of the ability of a face recognition system in working with datasets with large number of individuals (e.g., greater than 1000). To this aim, we have tested our face recognition algorithm on the FERET test set. Two FERET image sets were used in our experiments: FA**,** which contains frontal images of 1196 subjects (one image per person); and FB, which contains frontal images of 1195 of the subjects available in FA, with an alternative facial expression than in FA photographs. We detected the faces manually from all images and resized them to 112x92 pixels. The face images in FA were used for training and the FB images for testing the system. The performance of our technique in this experiment is shown in Table I[7], in comparison with the results of five state-of-the-art methods: Elastic Bunch Graph Matching (EBGM), LDA+PCA, Bayesian Intra-personal/Extra-personal Classifier (BIC), Boosted Haar Classifier [13], and the work of Timo et al. [15] based on the LBP (Local Binary Patterns) texture analysis. The comparisons indicate that our method achieved better results than the other evaluated methods.

[7] When comparing the results in Table I, note that almost similar face normalization algorithms were used in all techniques, except [4].

Figure 3. Images of an individual from ORL. The first 5 images are used for training and the other 5 for testing.

TABLE I.
COMPARISON OF OUR RESULT ON THE FERET DATASET WITH THE RESULTS OF SEVERAL STATE-OF-THE-ART FACE RECOGNITION METHODS

Methods	**Recognition Rates**
(EBGM) - Wisskott et al. [4]	95.5%
(LDA + PCA) - Etemad et al. [2]	96.2%
(BIC) – Moghaddam et al. [3]	94.8%
(Boosted Haar) – Jones et al. [13]	94.0%
(LBP) - Timo et al. [15]	97%
Our method	98.4%

The computational time of our method directly depends on the number of features, extracted from each face image, as the most time consuming process of our method is feature extraction. The computational time for extracting 150 features from a face image takes about 1.4 second on a 3.2 GHz CPU. However, in contrast to most of the available face recognition techniques, the overall computational time of our recognition method does not depend very much on the number of individuals learned, which is an advantage of our method.

VI. CONCLUSIONS

Face recognition, because of its many applications in automated surveillance and security, has garnered a great deal of attention. While there have been many papers published in this area, much of the debate has now moved outside of the academic area. Since details of many of the best commercial algorithms are not publicly available, it can be difficult to compare results or gauge progress.

In this paper we presented a novel technique for face recognition which represent face images by a set of 3-node labeled graphs, each containing information on the appearance and geometry of a 3-tuple of face feature points. Our method automatically learns a model set and builds a graph space for each individual. A two-stage method for fast matching is used in recognition, where in the first stage a MAP solution based on PCA factorization is used to efficiently prune the search space and select very few candidate model sets, and in the second stage a nearest neighbourhood classifier is used to find the closest model graphs to the query image graphs. Our proposed technique achieves perfect results on the ORL face set and an accuracy rate of 98.4% on the FERET face set, which shows the superiority of the proposed technique over all considered state-of-the-art methods.

ACKNOWLEDGMENT

The authors would like to acknowledge the Ontario Centre of Excellence (OCE) and the National Science and Engineering Research of Canada (NSERC) for their partial support.

REFERENCES

[1] M. Turk and A. Pentland. "Eigenfaces for Recognition", Journal of Cognitive Neuroscience, 1991, Vol. 3, pp. 71-86.

[2] K. Etemad and R. Chellappa, "Discriminant Analysis for Recognition of Human Face Images", Journal of the Optical Society of America, 1997, Vol. 14, pp. 1724-1733.

[3] B. Moghaddam, C. Nastar and A. Pentland, "Bayesian Face Recognition using Deformable Intensity Surfaces", In Proceedings Computer Vision and Pattern Recognition 96, 1996, pp. 638-645.

[4] L. Wiskott, J.-M. Fellous, N. Kruger and C. Von Der Malsburg, "Face Recognition by Elastic Bunch Graph Matching", IEEE Trans. on Pattern Analysis and Machine Intelligence, 1997, Vol. 19, No. 7, pp.775-779.

[5] E. Fazl Ersi, J.S. Zelek, "Local Feature Matching for Face Recognition", In Proceedings of the 3rd Canadian Conference on Computer and Robot Vision (CRV'06), 2006.

[6] P. Felzenszwalb and D. Huttenlocher, "Pictorial Structures for Object Recognition", In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2066-2073, 2000.

[7] C. Harris and M. Stephens, "A Combined Corner and Edge Detector", in Alvey Vision Conference, pp. 147-151, 1998.

[8] B. Triggs, "Detecting Keypoints with Stable Position, Orientation and Scale under Illumination Changes", in Eigth European Conference on Computer Vision, 2004.

[9] L. Guibas , J. Stolfi, "Primitives for the Manipulation of General Subdivisions and the Computation of Voronoi", ACM Transactions on Graphics, vol. 4, no.2, pp.74-123, 1985.

[10] J. McLachlan and D. Peel, "Finite Mixture Models" New York, John Wiley & Sons Ltd., 2000.

[11] R. A. Redner and H. F. Walker, "Mixture Densities, Maximum Likelihood and the EM Algorithm", SIAM Review, vol. 26, no. 2, pp. 195-239, 1984.

[12] F. Samaria and A. Harter. "Parameterisation of a Stochastic Model for Human Face Identification", Proceedings of 2nd IEEE Workshop on Applications of Computer Vision, 1994.

[13] M. J. Jones and P. Viola, "Face Recognition using Boosted Local Features", In Proceedings of International Conference on Computer Vision, 2003.

[14] P.S. Penev et al., "Local Feature Analysis: a General Statistical Theory for Object Representation". Network: Computation in Neural Systems, 1996, 7(3): 477-500.

[15] A. Timo, H. Abdenour, and P. Matti, "Face Recognition with Local Binary Patterns", Proceedings of the ECCV, pp. 469-481, 2004.

[16] P. J. Phillips et al., "The FERET Database and Valuation Procedure for Face Recognition algorithms," Image and Vision Computing, 1998, 16(5): 295-306.

[17] Li et al., "A Novel Face Recognition Method with Feature Combination", Journal of Zhejiang University: Science, 2005, 6(5).

Ehsan Fazl-Ersi received an undergraduate degree in Computer Engineering from the Azad University of Mashad, Mashad, Iran, in 2004, and a Master's degree in Systems Design Engineering from the University of Waterloo, Waterloo, Canada in 2006. He is now pursuing a Ph.D. degree in Computer Science at the York University, Toronto, Canada. His research interests include computer and robot vision, particularly, object detection and recognition, and mobile robot localization and navigation.

John S. Zelek is an Associate Professor in the Systems Design Engineering department at the University of Waterloo. He is also co-director of the Intelligent Human-Machine Interface lab. His research interests can be best summarized as being in the area of intelligent Mechatronic control systems that interface with humans; specifically, the areas are (1) wearable sensory substitution and assistive devices; (2) probabilistic visual and tactile perception; (3) wearable haptic devices including their design, synthesis and analysis; and (4) human-robot interaction. He is a co-founder of Tactile Sight Inc. He was awarded with the 2004 Young Investigator Award by the Canadian Image Processing & Pattern Recognition society for his work in robotic vision. He has also been awarded on numerous occasions with the best paper in various conferences. His work has also been featured in over 30 television, newspaper and radio stories.

John K. Tsotsos received an honours undergraduate degree in Engineering Science in 1974 from the University of Toronto and continued at the University of Toronto to complete a Master's degree in 1976 and a Ph.D. in 1980 both in Computer Science. In 1980 he joined the faculty in Computer Science at the University of Toronto where he founded the computer vision research group in the department. He served two 5-year terms as Fellow of the Canadian Institute for Advanced Research. His research focuses on the computational complexity of biological visual perception and the development of an accompanying theory of visual attention that makes strong predictions (and now with extensive supporting experimental evidence) about the psychophysics and neurobiology of human and primate perception. Another research thrust is the practical application of this work in the development of PLAYBOT, a visually-guided robot to assist physically disabled children in play.

He moved to York University in Toronto in January 2000 where he was appointed Professor in the Department of Computer Science and Engineering. He was also named Director of York's Centre for Vision Research, a position he held until November 2006. He currently holds an NSERC Tier I Canada Research Chair in Computational Vision and is also an Adjunct Professor in Ophthalmology and in Computer Science the University of Toronto.

He has served on numerous conference committees and on the editorial boards of Image and Vision Computing Journal, Computer Vision and Image Understanding, Computational Intelligence and Artificial Intelligence and Medicine. He served as the General Chair for the IEEE International Conference on Computer Vision 1999. Recent editorial activities include a Special Issue on Attention and Performance for Computer Vision and Image Understanding (with L. Paletta, R. Fisher and G. Humphreys), and Neurobiology of Attention for Elsevier Press (with L. Itti and G. Rees).