Face Recognition using Feature Extraction based on Descriptive Statistics of a Face Image

Rojana Kam-art and Thanapant Raicharoen
Faculty of Information Technology
Eartern Asia University
Thanyaburi, Pathumthani, Thailand
Email: rojana@eau.ac.th and thanapant@eau.ac.th

***Abstract*—This paper proposes a new method of feature extraction for face recognition based on descriptive statistics of a face image. Our method works by first converting the face image with all the corresponding face components such as eyes, nose, and mouth to a grayscale images. The features are then extracted from the grayscale image, based on a descriptive statistics of the image and its corresponding face components. The edges of a face image and its corresponding face components are detected by using the canny algorithm. In the recognition step, different classifiers such as Multi Layer Perceptron (MLP), Support Vector Machine (SVM), k-Nearest Neighbors (k-NN) and Pairwise Opposite Class-Nearest Neighbor (POC-NN) can be used for face recognition. We evaluated our method with more conventional Eigenface method based upon the AT&T and Yale face databases. The evaluation clearly confirm that for both databases our proposed method yields a higher recognition rate and requires less computational time than the Eigenface method.**

I. Introduction

Face recognition is an important research problem and has a wide range of application ranging from identity authentication, security monitoring, and surveillance system. It has been an active research topic and a variety of methods have been proposed for many years. A formal method of face classification was first proposed in [1]. Interest and research activities in the face recognition area have increased significantly over the past decade [2], [3], [4]. The up-to-date reviews of major human face recognition research were proposed in [5], [6]. Majority of face recognition methods apply mostly to frontal faces. Two main issues are central for face recognition: (i) what features to use to represent a face, and (ii) how to classify a new face based on the chosen representation.

One of the most thoroughly investigated approaches to face recognition is Eigenface [6], which is a conventional method and also known as Karhumen-Loeve expansion. Sirovich and Kirby [7] applied the Principle Component Analysis (PCA) for representing face images. Turk and Pentland [8] developed a well known face recognition method using Eigenfaces and Euclidean distance. Belhumeur et al [9] proposed another approach called Fisherfaces by applying first PCA for dimensionality reduction to represent a face and then use Fisher Linear Discrimination (FLD) to classify a new face. While PCA pursues a low dimensional representation of the faces, however, it's not essential with good classification capability between different faces. Another proposed method for face feature extraction is Independent Component Analysis (ICA) [10], which separates the high-order moments of the input in addition to the second-order moments [11]. The PCA and ICA methods for face recognition were compared on ”FERET” face database in [12], and it was found that both of them gave the same recognition accuracy.

The other face recognition methods are neural network, geometrical feature matching, and template matching. One of the first Artificial Neural Network (NN) techniques used for face recognition is a single layer adaptive network called WISARD, which contains a separate network for each stored individual [13]. Sung and Poggio [14] applied Multi-Layer Perceptron, and Lawrence et al [15] applied convolutional neural network for face recognition. In general, neural network approaches encounter problems when the number of classes increases. Moreover, they are not suitable for a single model image recognition test [6]. Geometrical feature matching techniques are based on the computation of a set of geometrical features from the picture of a face. One of the first researches on face recognition by using geometrical features was done by Kanade [16]. In geometrical features methods, facial features such as eyes, nose and mouth are detected. Properties and relations, such as areas, distances and angles, between the features are used to represent a face. In contrast, template matching methods [3], [4] in general operate directly on an image-based representation, such as pixel intensity array. One of the successful examples of face recognition using template matching is that based on Eigenfaces representation [18]. One drawback of template matching is its computational complexity. Another problem lies in the description of these templates [6]. However, no existing technique is free from limitations. Further works are required to enhance the performances in accuracy rate and computational time of face recognition methods.

In this paper, we focus on feature extraction, which is one of the most important steps that face recognition depends strongly on. The main idea is to apply both neural network and template matching techniques. Our method pursues a low dimensional representation of the face. The features are extracted based on the descriptive of a face image and corresponding face components. The choice of selected face features extremely impacts the accuracy of face recognition. Therefore a good feature to use to represent a face must be investigated. Nevertheless, several neural network techniques for a new face based on the chosen representation should be explored for a good classifier

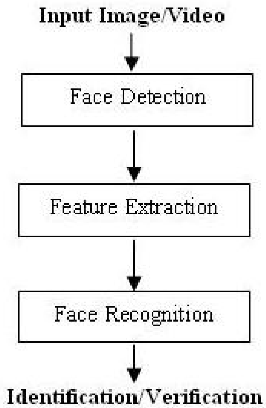

Fig. 1. Three major parts of Face Recognition Algorithm

as well.

The rest of the paper is organized as follows. The detail of our proposed method for feature extraction and face recognition is described in Section 2. The experimental results are showed in Section 3. In Section 4, conclusion is provided.

II. Methodology

Face recognition algorithm generally consists of three major parts: (i) Face Detection (ii) Feature Extraction and (iii) Face Recognition [5] as shown in Fig. 1.

Algorithms that consist of three parts are referred to as fully automatic algorithms and those that consist of only some parts (usually part (ii) and (ii)) are called partially automatic algorithms. Our paper proposes a partially automatic algorithm and is focused on feature extraction (what features to use to represent a face) and face recognition (how to classify a new face based on the chosen representation).

A. Feature Extraction

Feature extraction, a special form of dimensionality reduction, is the process of computing a compact numerical representation that can be used to characterize a pattern. The design of descriptive feature for a specific application is the main challenge in building pattern recognition systems. The representation of face features used in our proposed method is based on descriptive statistic of a face image and it's corresponding to face components. Since descriptive statistics are used to describe the basic features of the data and can be used to summarize the data. There are two essential objectives for formulating a summary statistic: (i) A measure of central tendency: to choose a statistic that shows how to different units seem similar. (ii) A measure of statistical variability: to choose another statistic that shows how they differ. Therefore, the arithmetic mean and standard deviation are chosen as a measurement of the data and also a set of features for our algorithm.

Feature Extraction Algorithm

Let Γ be a training set of *m* images: $\mathbf{I}_1$, $\mathbf{I}_2, \ldots, \mathbf{I}_m$ and $\mathbf{I}_i$ be a XxY image matrix. We assume that a face and its corresponding to face components such as left eye, right eye, nose and mouth be detected.

The algorithm to find a set of features is given as follows:

Input: a training set of m images Γ.
Process: Do for all images $\mathbf{I}_i$, $i = 1, \ldots, m$.

1. Convert an $\mathbf{I}_i$ to a grayscale image matrix.
2. Translate an image matrix to an image vector.
3. Compute mean and standard deviation of an image vector.
4. Convert corresponding $\mathbf{I}_i$ components to their grayscale components matrices.
5. Translate their components matrices to their components vectors.
6. Compute mean and standard deviation of their vectors.

Output: A feature set of m images Γ consists of all computed values of mean and standard deviation.

The proposed algorithm greatly reduces the dimension of image feature space from XxY to 10, where XxY is an image's resolution. In order to avoid some noises, lighting conditions and expression of a face, edges of a face image and its corresponding to face components are detected by using Canny method [16]. The above feature extraction algorithm is also applied on a training set of *m* edges images. The dimension of a feature set of m images is double from 10 to 20. However, it's still a good feature set to represent a face for any classifiers.

B. Face Recognition

With our extracted features, a human face can be easily recognized by using any classifier. Therefore, several classifiers, such as Neural Networks (NN) [19], Support Vector Machine (SVM) [20] or Nearest Neighbor rule [21], can be applied for our face recognition algorithm.

Face Recognition Algorithm

Let $\mathbf{I}_t$, be a test images. We also assume that a face and its corresponding to face components such as eyes, nose and mouth be detected.

The algorithm to classify a new face is given as follows:

Input: a test images $\mathbf{I}_t$ and a feature set of m images Γ.
Process: Do for a test images $\mathbf{I}_t$.

1. Compute a feature set of $\mathbf{I}_t$ using our Feature Extraction Algorithm
2. Perform a ***classifier*** on a feature set of m images Γ for training data and a feature set of $\mathbf{I}_t$ for test data

Output: Predicted Class of a test images $\mathbf{I}_t$.

III. Experiments

A. Face Datasets

We have applied our algorithm to AT&T and Yale face databases. The AT&T database [22] (formerly 'The ORL Database of Face') contains 400 images in PGM format with a resolution of 92x112. There are ten different images of each of 40 distinct persons. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open

Fig. 2. Ten different images of 2 persons in AT&T database

Fig. 3. Eleven different images of 2 persons in Yale database

/ closed eyes, smiling / not smiling) and facial details (glasses / no glasses). Examples of these images are shown in Fig. 2. The Yale Face Database [23] contains 165 images in GIF format with a resolution of 243x320 of 15 distinct persons. There are 11 images per individual, one per different facial expression or configuration: center-light, w/glasses, happy, left-light, w/no glasses, normal, right-light, sad, sleepy, surprised, and wink. Examples of these images are shown in Fig. 3.

B. Feature Extraction

For our experiments, the face database is divided into two non-overlapping sets, one for training and the other for testing. The training data consist of 200 and 90 images chosen from AT&T and Yale databases respectively. The remaining 200 and 70 images are used for testing. In the experiments, the algorithm to find a set of features was performed on the training set. Table V contains the results of feature extraction from the first five face images of the 1.st and 2.nd person from AT&T database. Each row in the table represents one of the face images for a person. Table VI also contains the results of feature extraction from edges images corresponding to the face images of the 1.st and 2.nd person from AT&T database in Table V. The first two face images of the first person and corresponding to the face components (left eye, right eye, nose and mouth) from AT&T and Yale database presents in Fig. 4 and Fig. 5 respectively.

C. Face Recognition

For face recognition, four different classifiers, namely Multi Layer Perceptron (MLP), Support Vector Machine (SVM), 1-Nearest Neighbor (1-NN) and Pairwise Opposite Class Nearest Neighbor (POC-NN) [24] were used. All classifiers except POC-NN were implemented by using STPRtool [25]. Parameters for each classifier were used default from STPRtool. One-vs-One approach was used for SVM classifier. Table I summarizes the accuracy rates for testing data in percent from each classifier by using 10 features and 20 features. In most cases, the accuracy rates increase when the dimension of features is increased. However, the 1-NN and POC-NN outperform the other methods for both databases with the same

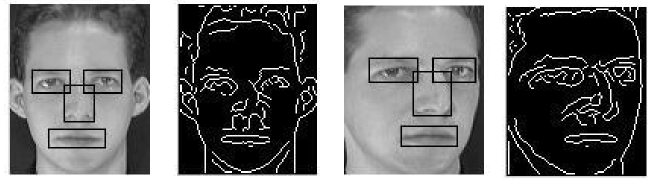

Fig. 4. AT&T database

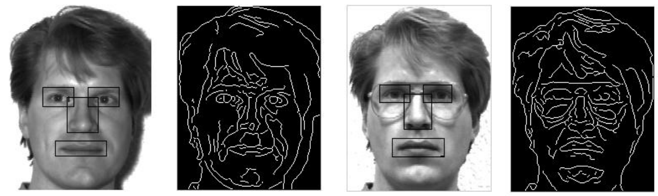

Fig. 5. Yale database

features. Nevertheless, the POC-NN reduces the computational time and the memory space of the training data. For this experiment, the number of training data was reduced from 200 to 186 and from 90 to 84 for AT&T and Yale database respectively.

TABLE I
THE COMPARISON RESULTS OF 4 CLASSIFIERS IN ACCURACY RATE IN PERCENT

Database	Dim. of Feature	MLP	SVM	1-NN	POC-NN
AT&T	10	77.00	91.50	**92.50**	**92.50**
	20	81.50	**92.50**	**92.50**	**92.50**
Yale	10	66.67	77.33	**80.00**	**80.00**
	20	69.33	78.67	**80.00**	**80.00**

Best accuracy rates among these algorithms are bold-faced.

We also compare the computational training and testing time required for the results shown in Table I. The less require computational time the more efficiency for an algorithm. All algorithms were implemented in MatLab version 7.2. All 10 experiments were done on the same Intel Pentium M-1.70 GHz notebook computer with 512 MB RAM, and the average of time from 10 experiments were reported. The time comparisons are summarized in Table II.

TABLE II
THE COMPARISON RESULTS OF 4 CLASSIFIERS IN COMPUTATIONAL TIME IN SECOND

Database	Dim. of Feature	MLP	SVM	1-NN	POC-NN
AT&T	10	3030.14	1.12	**0.03**	**0.03**
	20	4662.97	1.20	0.06	**0.05**
Yale	10	1103.53	0.17	**0.01**	**0.01**
	20	1145.36	0.23	**0.01**	**0.01**

Least computational time among these algorithms are bold-faced.

From these results, using 1-NN and POC-NN as a classifier is more effective than using the others. Therefore, we compared our algorithm using 1-NN with the conventional principal component analysis (PCA) based method, called Eigenface. The Eigenface program used for our experiment was originally developed by Omidvarnia [26]. The results obtained on the same datasets are reported in Table III.

TABLE III
THE COMPARISON RESULTS OF OUR METHOD AND EIGENFACE IN ACCURACY RATE AND COMPUTATIONAL TIME

Database	Accuracy Rate		Computational Time	
	Our method	Eigenface	Our method	Eigenface
AT&T	92.50	90.00	0.06	569.91
Yale	80.00	80.00	0.04	396.58

The experimental results shown in Table III signify the effectiveness of our method and its performance in both accuracy rate and in computational time to those obtained by a classical Eigenface method. Especially in computational time, our method performs very fast and can be used for real time face recognition. However, the more number of training data used the more computational time required. In order to reduce the computational time, the POC-NN instead of 1-NN can be used as a classifier for our algorithm.

To increase statistical significance of the results on the data set whose training and test sets are not separated, we conduct the *K*-fold cross-validation technique which probably is the simplest and most widely used method for estimating prediction error [27], and report the average cross-validation estimate of accuracy rate and computational time as shown in Table IV. Comparison with the results in Table III, for the five-fold cross-validation (*K*=5), our algorithm still shows better results in accuracy rate and computational time than results from the *Eigenface* in all cases.

TABLE IV
THE COMPARISON RESULTS FOR THE FIVE-FOLD CROSS-VALIDATION OF OUR METHOD AND EIGENFACE IN ACCURACY RATE AND COMPUTATIONAL TIME

Database	Accuracy Rate		Computational Time	
	Our method	Eigenface	Our method	Eigenface
AT&T	95.75	94.77	0.05	585.41
Yale	75.27	75.09	0.03	330.87

IV. CONCLUSION

A new method of feature extraction for face recognition based on descriptive statistics of a face image has been proposed. The features are extracted based on the descriptive statistics of a grayscale face image and its corresponding face components. The descriptive statistics of edges images and its corresponding face components can also be used for a feature set. These feature values are able to apply to any classifier for the face recognition step. Our algorithm is straightforward and easy to implement. However, our method shows better performances on AT&T and Yale database in both accuracy rates and computational time than the results from the Eigenface method. For further investigation, the other types of descriptive statistics, such as Geometric or Harmonic mean, should be investigated.

REFERENCES

[1] F. Galton. Personal identification and description. *in Nature.* June 21, 1888, pp.269-274.
[2] A. Samal and P.A.Iyengar. Automatic recognition and analysis of human faces and facial expressions: A survey. *Pattern Recognition.* vol. 25, 1992, pp.pp.65-77.
[3] D. Valentin, H. Abdi, A. Toole, and G.Cottrell. Connectionist models of face processing: A survey. *Pattern Recognition.* vol. 27, 1992, pp.1209-1230.
[4] R. Chellappa, C.L. Wilson, and S. Sirohey. Human and machine recognition of faces: A survey. *Proc.IEEE.* vol. 83, 1995, pp.705-740.
[5] W. Zhao, R. Chellappa, P.J. Phillips, and A.Rosenfeld. Face Recognition: A Literature Survey. *ACM Computing Surveys.* vol. 35, No.4, 2003, pp.399-458.
[6] A.S. Tolba, A.H.El-Baz, and A.A.El-Harby. Face Recognition: A Literature Review. *International Journal of Signal Processing.* vol. 2, No.4, 2006, pp.88-103.
[7] L. Sirovich and M.Kirby. Low-dimensional procedure for the characterization of human faces. *Journal Optical Soc. of Amer.* vol. 4, 1987, pp.519-524.
[8] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal Cognitive Neuroscience.* vol. 3, 1991, pp.71-86.
[9] P.Belhumeur, J.Hespanha, and D.Kriegman. Eigenfaces vs. Fisherfaces: recognition using class specific linear projection. *IEEE Trans.PAMI.* vol.19(7), 1997, pp.711-720.
[10] M. S. Bartlett, H.M. Lades, and T.J. Sejnowski. Independent component representation for face recognition. *Proc. Of SPIE.* 2399, 1998, pp.528-539.
[11] P. Comon. Independent component analysis - a new concept?. *Signal Processing.* vol.36, 1994, pp.287-314.
[12] B. Moghaddam. Principal manifolds and bayersian subspaces for visual recognition. *Proc. ICCV.* 1999, pp.1131-1136.
[13] T.J. Stonham. Practical face recognition and verification with WISARD. *Aspects of Face Processing.* 1984, pp.426-441.
[14] K.K Sung and T.Poggio. Learning human face detection in cluttered scences. *Computer Analysis of Image and Patterns.* 1995, pp.432-439.
[15] S. Lawrence, C.L. Giles, A.C. Tsoi, and A.D.Back. Face recognition: A convolutional neural-network approach. *IEEE Trans. Neural Networks.* vol.8, 1997, pp.98-113.
[16] T. Kanade. Face recognition: Picture processing by computer complex and recognition of human faces. *Technical report, Dept. Information Science, Kyoto Univ.* 1973.
[17] Canny, J., A Computational Approach To Edge Detection, *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1986, vol.8, pp.679-714.
[18] M.A. Turk and A.P.Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, vol. 3, no. 1, 1991, pp. 71-86.
[19] S.S. Hakin. Neural Networks, A comprehensive foundations. New York: Macmillan, 1999.
[20] C. Cortes and V.N. Vapnik. Support Vector Networks. *Machine Learning*, 20(3), 1995, pp. 273-297.
[21] T.M. Cover,and P.E. Hart. Nearest neighbor pattern classification. *IEEE Trans. Inform. Theory* 13, 1967, pp. 21-27.
[22] The AT&T Database of Face. Avaiable: *http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html*.
[23] The Yale Database.Avaiable: *http://cvc.yale.edu/projects/yalefaces/yalefaces.html*.
[24] T. Raicharoen and C.Lursinsap, A Divide-and-Conquer Approach to the Pairwise Opposite Class-Nearest Neighbor (POC-NN) Algorithm *Pattern Recognition Letters*, Vol.16 (10), 2005, pp.1554-1567.
[25] V. Franc and V.Hlavac, *Statistical Pattern Recognition Toolbox for Matlab*, Czech Technical University in Prague, 2004.
[26] A. H. Omidvarnia, 'Eigenface' Face Recognition System. Avaiable: *http://www.mathworks.com/matlabcentral/fileexchange/17032*.
[27] T. Hastie, R. Tibshirani and J. Friedman, *T*he Elements of Statistical Learning, Springer Series in Statistics, Springer-Verlag, 2001.

TABLE V

THE FEATURES OF 5 IMAGES FOR THE FIRST AND SECOND PERSON FROM AT&T DATABASE

Person	Image	Mean					Standard Diviation				
		face	left eye	right eye	nose	mouth	face	left eye	right eye	nose	mouth
1	1	128.34	148.11	137.60	164.15	163.41	53.32	26.41	23.94	13.66	27.15
1	2	132.59	149.17	139.83	162.54	159.38	52.65	26.75	26.08	13.79	26.61
1	3	142.77	136.75	148.63	168.62	158.50	48.26	26.89	32.88	20.89	25.03
1	4	134.48	144.06	139.29	160.42	157.24	49.34	28.86	27.45	20.77	23.96
1	5	134.24	138.86	151.56	161.70	164.91	55.40	27.99	30.57	26.28	24.47
2	1	111.99	141.90	132.76	137.21	127.06	48.43	29.38	28.10	30.28	34.79
2	2	111.54	140.99	128.96	139.17	132.22	49.27	35.30	27.70	31.44	33.37
2	3	113.69	142.20	135.26	139.87	126.20	47.38	30.16	26.51	29.32	40.06
2	4	114.42	139.50	122.97	134.14	111.93	47.65	30.59	24.27	32.18	36.30
2	5	112.12	142.69	125.91	133.35	114.19	49.97	32.20	28.94	32.42	38.07

TABLE VI

THE FEATURES OF 5 EDGES IMAGES FOR THE FIRST AND SECOND PERSON FROM AT&T DATABASE

Person	Image	Mean					Standard Diviation				
		face	left eye	right eye	nose	mouth	face	left eye	right eye	nose	mouth
1	1	0.089	0.126	0.146	0.159	0.105	0.285	0.332	0.353	0.366	0.306
1	2	0.088	0.141	0.187	0.165	0.116	0.283	0.348	0.391	0.372	0.320
1	3	0.082	0.121	0.143	0.134	0.106	0.275	0.326	0.351	0.341	0.308
1	4	0.091	0.149	0.133	0.134	0.103	0.287	0.356	0.340	0.341	0.304
1	5	0.077	0.113	0.152	0.108	0.100	0.268	0.317	0.360	0.311	0.300
2	1	0.113	0.194	0.162	0.149	0.116	0.317	0.396	0.369	0.357	0.320
2	2	0.118	0.206	0.168	0.119	0.136	0.322	0.405	0.374	0.325	0.343
2	3	0.115	0.162	0.143	0.142	0.153	0.320	0.369	0.351	0.350	0.360
2	4	0.115	0.197	0.181	0.158	0.161	0.319	0.398	0.385	0.365	0.368
2	5	0.112	0.171	0.156	0.159	0.144	0.316	0.377	0.363	0.366	0.351