FACE RECOGNITION FOR GROUP CLASSIFICATION BASED ON KERNEL PRINCIPAL COMPONENT ANALYSIS AND SUPPORT VECTOR MACHINES

Ivanna K. Timotius[1], Iwan Setyawan[1], Andreas A. Febrianto[1]
Department of Electronic Engineering, Faculty of Electronic and Computer Engineering
Satya Wacana Christian University
Diponegoro 52 - 60 Salatiga 50711, Indonesia
email: ivanna_timotius@yahoo.com[1], iwan.setyawan@ieee.org[1], andreas@visio-lab.org[1]

ABSTRACT

Face Recognition system is a machine that is used to recognize people based on their face. In many practical applications, this face recognition system is used to determine whether somebody belongs to certain group or not. This paper presents a face recognition method for group classification by combining kernel principal component analysis (KPCA) and support vector machines (SVM). By using these methods, the classification accuracy of the system is 91.84%.

Keywords: Face recognition, Group Classification, Kernel Principal Component Analysis, Support Vector Machines.

1 INTRODUCTION

Researches in face recognition systems are motivated by the human natural ability to recognize face and enormous practical applications where human identification is needed [1]. The implementation of face recognition systems fit in numerous areas, such as biometric authentication, surveillance, human – machine interaction, and multimedia management. In the biometric authentication application, face recognition is one of the non-intrusive methods, which make face recognition technology become closer to people's daily lives.

Many face recognition applications and approaches have been proposed. Some literatures [2][3][4] developed a multiclass face recognition system, where every person is treated as one class. Reference [5] developed a two class face recognition system between two different people. In some applications, we are interested in knowing whether a person belongs to a certain group or not. For this kind of application, we need a two class classification system, where each class or group contains several people. This paper presents group face recognition system based the combination of kernel principal component analysis (KPCA) and support vector machines (SVM). The block diagram of the system is presented in Figure 1.

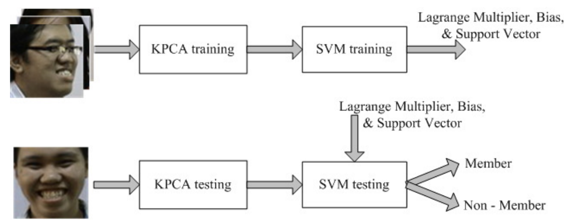

Figure 1. Block Diagram

KPCA is used as a feature extractor in this face recognition system, which aims to find the important features from the data without any supervision [6]. SVM is used as a classifier, which utilized optimalization approach to construct a hyperplane [7]. By combining these two methods, we hope that the classification accuracy will be increased. In this paper, the performance will be compared with some other commonly used method of face recognition.

This paper is organized as follows. Section 2 and 3 introduce the basic theory and the implementation of KPCA and SVM methods, respectively. Section 4 will describe the details of our experiment design. Finally, in section 5, we will present some conclusions.

2 KERNEL PRINCIPAL COMPONENT ANALYSIS

KPCA is a development of the PCA (eigenfaces) method [2]. Principal component analysis (PCA) aims to extract the principal structure of our data sets and in the same time it will lower the dimensional space of the feature. PCA is one of the orthogonal transformations, where the principal components are the basis of the new coordinate system.

ISSN 2085-1944

KPCA is the nonlinear version of PCA, which uses a nonlinear kernel function. Kernel function makes every linear algorithm that uses scalar products can be implicitly executed in a high dimension feature spaces F without explicitly knowing the mapping Φ [9]. By using KPCA we hope that the nonlinearity problem of the face recognition system will be solved and hence will help the classifier to perform better.

Given a set of centered data:

$$\mathbf{x}_k \in \mathbf{R}^N \qquad \sum_{k=1}^{M} \Phi(\mathbf{x}_k) = \mathbf{0} \tag{1}$$

In the implementation of face recognition in this paper $\mathbf{x}_k \in \mathbf{R}^N$ are taken from the face images by rearranging the pixel value order as shown in Figure 2.

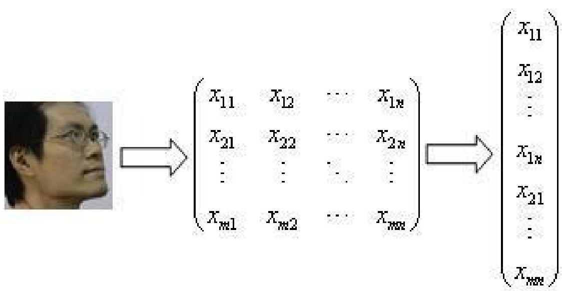

Figure 2. Pre-processing Step

PCA diagonalized the covariance matrix in F [10]:

$$C = \frac{1}{M}\sum_{j=1}^{M} \Phi(\mathbf{x}_j)\Phi(\mathbf{x}_j)^T \tag{2}$$

We have to find the eigenvalues $\lambda \geq 0$ and eigenvectors satisfying

$$\lambda \mathbf{v} = C\mathbf{v} \tag{3}$$

Since $\mathbf{v}$ lie in the span of $\Phi(x_1)$, $\Phi(x_2)$, … , $\Phi(x_M)$, there exist coefficients α_i (i = 1, 2, …, M) such that,

$$\mathbf{v} = \sum_{i=1}^{M} \alpha_i \Phi(\mathbf{x}_i) \tag{4}$$

By defining the kernel function as

$$k(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i)\cdot\Phi(\mathbf{x}_j) \tag{5}$$

and the elements of $M \times M$ matrix K by

$$K_{ij} := \Phi(\mathbf{x}_i)\cdot\Phi(\mathbf{x}_j) \tag{6}$$

we get

$$M\lambda K\boldsymbol{\alpha} = K^2\boldsymbol{\alpha} \tag{7}$$

where $\boldsymbol{\alpha}$ denotes the column vector with entries α_1, α_2, … , α_M. To find the solution, we solve the eigenvalue problem

$$M\lambda\boldsymbol{\alpha} = K\boldsymbol{\alpha} \tag{8}$$

This paper uses the radial-basis function network as a kernel function:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\tfrac{1}{2s^2}\left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2\right) \tag{9}$$

where s is the width which is specified empirically.

For the principal component extraction, the projections onto the eigenvectors $\mathbf{v}^k$ in F are needed. Given $\mathbf{x}$ as a test point and Φ(x) is its image in F, then its nonlinear principal components is

$$\mathbf{v}^k\cdot\Phi(\mathbf{x}) = \sum_{i=1}^{M} \alpha_i^k \left(\Phi(\mathbf{x}_i)\cdot\Phi(\mathbf{x})\right) \tag{10}$$

The steps to compute the principal components can be summarized as:

1. Compute the matrix K,
2. Compute its eigenvectors and normalize them in F,
3. Compute projections of a test point onto the eigenvectors.

3 SUPPORT VECTOR MACHINES

SVM is a generalized linear classifier that tries to construct a hyperplane as a decision surface that have a maximum margin of separation between two training classes. This separating hyperplane is defined as a linear function drawn in the feature space obtained from a nonlinear mapping [7].

Given a training sample $(\mathbf{x}_i, y_i)_{i=1}^{N}$, where $\mathbf{x}_i$ is a training vector and y_i is its class label, SVM aims to find the weight vector $\mathbf{w}$ and the bias b of the separating hyperplane such that [7][11]:

$$\begin{aligned} y_i(\mathbf{w}^T\varphi(\mathbf{x}_i) + b) &\geq 1 - \xi_i, \quad \forall i \\ \xi_i &\geq 0, \quad \forall i \end{aligned} \tag{11}$$

with $\mathbf{w}$ and the slack variables ξ_i minimizing the cost function:

$$\Phi(\mathbf{w}, \xi_i) = \tfrac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{N}\xi_i \tag{12}$$

where the slack variables ξ_i represent the error measures of data, C is a user-specified positive parameter which is the penalty assigned to the errors, and $\varphi(\cdot)$ is a nonlinear mapping which maps the data into a higher dimensional feature space from original input space.

The dual problem of SVM is given as follows. Find the Lagrange multipliers $(\alpha_i)_{i=1}^{N}$ that maximize the objective function:

$$Q(\alpha) = \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j)$$

subject to

$$\sum_{i=1}^{N} \alpha_i y_i = 0$$
$$0 \le \alpha_i \le C \qquad \forall i \tag{13}$$

where C is a user-specified positive parameter. If $0 < \alpha_i \le C$, the corresponding data points are called support vectors. The kernel used in SVM in this paper is radial-basis function network similar to Eq. 9:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{1}{2s_1^2}\left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2\right) \tag{14}$$

where s_1 is the width which is specified by the user. However, the width used in KPCA and SVM does not have to be the same.

Having the Lagrange multipliers, the optimum weight vector $\mathbf{w}_o$ could be computed by

$$\mathbf{w}_o = \sum_{i=1}^{N} \alpha_i y_i \varphi(\mathbf{x}_i) \tag{15}$$

According to Kuhn-Tucker [12], the bias could be calculated by taking the samples with $0 < \alpha_i < C$

$$b = \frac{1}{\#SV} \sum_{x_i \in SV} \left(\frac{1}{y_i} - \sum_{\mathbf{x}_j \in SV} \alpha_j y_j k(\mathbf{x}_j, \mathbf{x}_i) \right) \tag{16}$$

where #SV is the number of support vectors with $0 < \alpha_i < C$.

For the testing data $\mathbf{z}$, its predicted class can be obtained by the decision function:

$$D(\mathbf{z}) = \operatorname{sign}\left(\sum_{i=1}^{N} \alpha_i y_i k(\mathbf{x}_i, \mathbf{z}) + b \right) \tag{17}$$

4 EXPERIMENT AND RESULTS

The experiment was done by using the Video Image and Signal Processing (VISIO) laboratory Satya Wacana Christian University (SWCU) multiview face database. This database contains face images of several subjects, which are evenly distributed in gender. The age of the subjects varies between 19 and 69 years. The images are taken under controlled condition in the VISIO laboratory. Each subject is photographed against a uniform white background using a single camera and identical settings. We used a single fluorescent light source that is placed in front of the subject. The camera's automatic white balance is used for all subjects. For each subject, we take 105 photographs. Each photograph has a different combination of viewpoint, facial expression, and accessories. Each image in the VISIO multiview face database is manually cropped around the facial area and is resampled into a 64 × 64 pixel 8-bit grayscale image.

The experiment used 30 subjects from VISIO face database. These subjects are divided into two classes: 6 subjects belong to the class of VISIO member and 24 subjects belong to the class of non VISIO member. The experiment is done by using 2-fold cross validation. The parameters for KPCA and SVM are determined empirically. The average of classification accuracy by using the combination of KPCA and SVM is compared with other commonly used methods in face recognition, namely the nearest neighbor (NN), the SVM (without using KPCA as feature extractor), and the combination of KPCA (as feature extractor) and NN as classifier.

The parameters (s_1^2, C, s^2, and the number of eigen vector), average of classification accuracy, training time (in minute) and testing time (in second) are summarized in Table 1. The testing time is calculated for each image. Since in the training phase NN only store all training images, the training time is approximately zero. The best result is given by using the combination of KPCA (with the parameters: s^2 = 100000 and the number of eigen vector = 100) and SVM (with the parameters: s_1^2 = 0.025 and C = 150). The average of classification/ testing accuracy by using this combined method is 91.84%.

Table 1. Experimental Results

Method	Average of Accuracy	Training Time (min)	Testing Time (sec)
NN	88.92%	0	2.58
SVM s_1^2 = 2000 C = 20	91.08%	16.05	3.77
KPCA – NN s^2 = 100000 Numb. of eig.vec. = 100	89.62%	9.18	3.55
KPCA – SVM s^2 = 100000 Numb. of eig.vec. = 100 s_1^2 = 0.025 C = 150	91.84%	24.26	3.62

5 DISCUSSION AND CONCLUSION

Our experimental results show that the use of KPCA as a feature extractor in any classifier increases the average of classification accuracy. SVM perform better than NN. The combination of

KPCA and SVM methods shows a satisfactory result which is 91.84% of classification accuracy.

The training time is increased by the use of KPCA and/or SVM. NN holds the fastest testing time. The testing time by using SVM, KPCA – NN, and KPCA – SVM have no significant difference.

In the future, we plan to continue our research by investigating other possible methods (both feature extractors and classifiers) to achieve a better system performance, and combining our face recognition method with a face detection method to form a complete face recognition system.

ACKNOWLEDGEMENT

This work is supported by DP2M DIKTI under the research grant 213/SP2H/PP/DP2M/III/ 2010.

REFERENCES

[1] S. Z. Li and A. K. Jain (2005) Handbook of Face Recognition, Springer Science+Business Media, Inc.

[2] M. Gonen, G. Tahugur, and E. Alpaydin (2008) Multiclass Posterior Probability Support Vector Machines, IEEE Transactions on Neural Networks, vol. 19, no. 1, pp. 130 – 139.

[3] Y. H. Liu and Y. T. Chen (2007) Face recognition using total margin-based adaptive fuzzy support vector machines, IEEE Trans. on Neural Networks, vol. 18, no. 1, pp. 178 – 192.

[4] D. Ginhac, F. Yang, X. Liu, and J. Dang (2008) Robust Face Recognition System Based on a Multi-Views Face Database, in Recent Advances in Face Recognition (K. Delac, et.al., Eds), Croatia: In-Tech.

[5] I. K. Timotius, I. Setyawan, and A. A. Febrianto (2010) Face Recognition between Two Person using Kernel Principal Component Analysis and Support Vector Machines, International Journal on Electrical Engineering and Informatics, vol. 2, no. 1, pp. 53 – 61.

[6] R. O. Duda, P. E. Hart, and D. G. Stork (2001) Pattern Classification, New York: Wiley.

[7] S. Haykin (1999) Neural Network: A Comprehensive Foundation, New Jersey: Prentice-Hall.

[8] A. K. Jain, R. P. W. Duin, and J. Mao (2000) Statistical Pattern Recognition: A Review, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 1.

[9] K. R. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf (2001) An Introduction to Kernel-Based Learning Algorithms, IEEE Transactions on Neural Networks, vol. 12, no. 2.

[10] B. B. Scholkopf and A. Smola (1998) Nonlinear Component Analysis as a Kernel Eigenvalue Problem, Neural Computation, vol. 10, pp. 1299-1319.

[11] V. Vapnik (1998) Statistical learning theory, Springer, New York: Berlin Heidelberg.

[12] R. Fletcher (1987) Practical Methods of Optimization, New York: Wiley.