CLASSIFICATION OF GENDER AND FACE BASED ON GRADIENT FACES

Len Bui, Dat Tran, Xu Huang and Girija Chetty

University of Canberra
Faculty of Information Sciences and Engineering
ACT 2601 Australia

ABSTRACT

This paper presents a new method for solving face gender identification and face classification problems. The proposed method uses gradient features for feature extraction and support vector machine for classification. Experiments for the proposed method have been conducted on two public data sets CalTech and AT&T. The results show that the proposed method could improve the classification rates.

1. INTRODUCTION

Human faces contain a lot of important biometric information. Therefore, face recognition has a variety of potential applications in public security and law enforcement such as verifying a person from credit card or driving license, identifying a person from video surveillance. The structure of face recognition system has two main modules: feature extraction module and training/classification module (depending on which task is training or testing) (see Fig. 1).

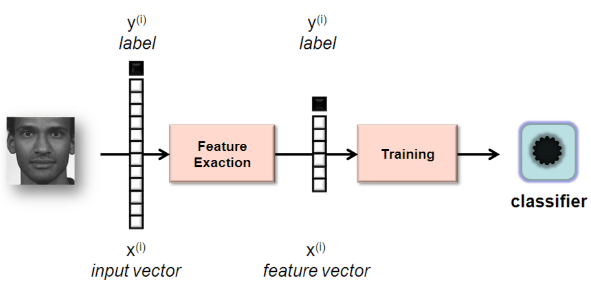

Fig. 1. Structure of face recognition system

One of the most difficulties in face recognition is facial feature extraction. Since Matthew Turk and Alex Pentland [1] used Principal Component Analysis (PCA) to deal with the face recognition problem, PCA has become the major mathematical tool to extract feature vectors. Nevertheless, PCA could not capture even the simplest invariance unless this information is explicitly provided in the training data. To deal with this problem, some researchers proposed other approaches. For example, Wiskott et al. [2] suggested a technique known as elastic bunch graph matching. Bartlett et al. [3] proposed using independent component analysis (ICA) for face representation and reported that it performed better than PCA. Ming-HsuanYang [4] suggested Kernel PCA for face feature extraction and recognition and described that his method outperformed the classical method. However, the performance costs of ICA and Kernel PCA are higher than PCA. Jian Yang [5] proposed a new method called 2D Principal Component Analysis (2DPCA). In conventional PCA, face images have been represented in vectors by some technique like concatenation. As opposed to PCA, 2DPCA represents face images by using matrices or 2D images instead of vectors. He also reported the method achieved the good results. However, the approach need a lot of computing resources; especially, it took a significant time performance in the recognition task. In our study, we suggests using PCA face and PCA gradient face as the features. Clearly, using gradient images is easily computed and still keep local information of the original images, which may bring more important features for classification.

There have been a few of classification methods such as k-Nearest Neighbors (k-NN), Fuzzy C-Means (FCM) and Artificial Neural Network (ANN). Recently, most of researchers have focused on the classification method, Support Vector Machine (SVM). In 1995, Vapnik and Cortes [6] presented the foundations for SVM. Since then, it has become the prominent method to solve problems in pattern classification and regression. The basic idea behind SVM is finding the optimal linear hyperplane such that the expected classification error for future test samples is minimized, i.e., good generalization performance. Obviously, the goal of all classifiers is not to get the lowest training error. For example, a k-NN classifier can achieve the accuracy rate 100% with k=1. However, in practice, it is the worst classifier because it has high structural risk.

They suggested the formula testing error = training error + risk of model. To achieve the goal to get the lowest testing error, they proposed the structural risk minimization inductive principle. It means that a discriminative function that classifies the training data accurately and belongs to a set of functions with the lowest VC dimension will generalize best results regardless of the dimensionality of the input space. Based on this principle, an optimal linear discriminative function has been found. For linearly non-separable data, SVM maps the input to a higher dimensional feature space where a linear hyperplane can be found. Although there is no warranty that a linear solution will always exist in the higher dimensional space, it is able to find effective solutions in practice. To deal with the face classification, many researchers [7, 8, 9, 10] have applied SVM in their studies and stated that the experiment results are very positive. In our research, we have combined the power of each method, gradient face and SVM, to solve the problem.

The remaining sections of our paper will discuss the implementation of our approach, related theory, and experiments. Section 2 gives details of feature extraction. Section 3 discusses how to use SVM in face classification. In Section 4, we will describe the implementation and experiments. Finally, Section 5 is our conclusion.

2. FEATURE EXTRACTION

2.0.1. *Preprocessing*

The first preprocessing step is to normalize face images of datasets by cropping and resizing to the resolution of 80-by-80. This task was done by using the object detection algorithm developed by Viola and Jones [11, 12].

2.0.2. *Global vectors*

We then represented normalized face images as global vectors $\mathbf{x}_{global} = (x_1, x_2, ..., x_{6400})^T$, where x_i is a gray value of each pixel (see Fig. 2a).

2.0.3. *Component vectors*

We continued to use Viola and Jones algorithm to detect regions of images containing left eye, right eye, nose and mouth. After locating them, we also scale them to the conventional sizes. After that we reshaped them into column vectors $\mathbf{x}_{leye}$ for left eye, $\mathbf{x}_{reye}$ for right eye, $\mathbf{x}_{nose}$ for nose and $\mathbf{x}_{mouth}$ for mouth. These vectors are called component vectors (see Fig. 2b).

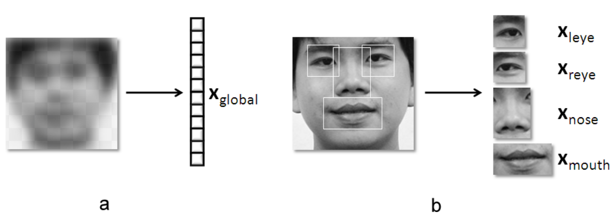

Fig. 2. a) Global vector b) Component vectors

2.0.4. *Gradient vectors*

In image analysis, gradients are used in edge detection. Furthermore, image gradients can be used to extract important information from images because image gradients are less susceptible to lighting. First, we used Eq. (1) to normalize image histograms.

$$\mathbf{I}'[r,c] = \frac{\mathbf{I}'_{\max} - \mathbf{I}'_{\min}}{\mathbf{I}_{\max} - \mathbf{I}_{\min}} \left(\mathbf{I}[r,c] - \mathbf{I}_{\min}\right) + \mathbf{I}'_{\min} \qquad (1)$$

Next, we applied Eq. (2) to compute two gradient images $\mathbf{G}_x$ and $\mathbf{G}_y$. We also normalized their histograms and reshaped them into column vectors $\mathbf{x}_{gradx}$ and $\mathbf{x}_{grady}$

$$\begin{pmatrix} \mathbf{G}_x & \mathbf{G}_y \end{pmatrix}^T = \begin{pmatrix} \frac{\partial \mathbf{I}}{\partial x} & \frac{\partial \mathbf{I}}{\partial y} \end{pmatrix}^T \qquad (2)$$

Fig. 3. Gradient vector

2.0.5. *Proposed input vectors*

We proposed an input vector as combination of global vector, component vectors and gradient vectors.

$$\mathbf{x} = (\mathbf{x}^T_{global}\, \mathbf{x}^T_{leye}\, \mathbf{x}^T_{reye}\, \mathbf{x}^T_{nose}\, \mathbf{x}^T_{mouth}\, \mathbf{x}^T_{gradx}\, \mathbf{x}^T_{grady}) \quad (3)$$

2.0.6. *Feature extraction*

One of well-known methods for extracting facial feature is PCA as mentioned above. It was first applied in face classification by Sirovich and Kirby and then Matthew Turk and Alex Pentland. Now it has become the standard method in this field.

The input is a dataset $\mathcal{D} = \{\mathbf{x}^{(i)}; i = 1, ..., N\}$

Algorithm 1: Building feature extractor

Step 1: Compute the mean of data

$$\overline{\mathbf{x}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}^{(i)} \qquad (4)$$

Step 2: Compute the covariance matrix of data

$$\mathbf{A} = \frac{1}{N} \sum_{i=1}^{N} \left(\mathbf{x}^{(i)} - \overline{\mathbf{x}}\right)^T \left(\mathbf{x}^{(i)} - \overline{\mathbf{x}}\right) \qquad (5)$$

Step 3: Compute the eigenvectors ϕ_i and eigenvalues λ_i of matrix A

Finally, statistical feature extractor built from data set $\mathcal{D}$ is $\mathbf{x} = \overline{\mathbf{x}} + \mathbf{M}\mathbf{x}'$, where $\mathbf{M} = \{\phi_1, \phi_2, ..., \phi_k\}$ and feature vector $\mathbf{x}' = (x'_1, x'_2, ..., x'_k)$. To calculate feature vector $\mathbf{x}'$ corresponding to input vector $\mathbf{x}$, we can use the formula $\mathbf{x}' = \mathbf{M}^{-1}(\mathbf{x} - \overline{\mathbf{x}})$.

3. CLASSIFICATION

3.1. Two-class Classification

The goal of SVM classifiers is to find a hyperplane that separates the largest fraction of a labeled data set The most important requirement, which the classifiers must have, is that it has to maximize the distance or the margin between each class and the hyperplane (see Fig. 4).

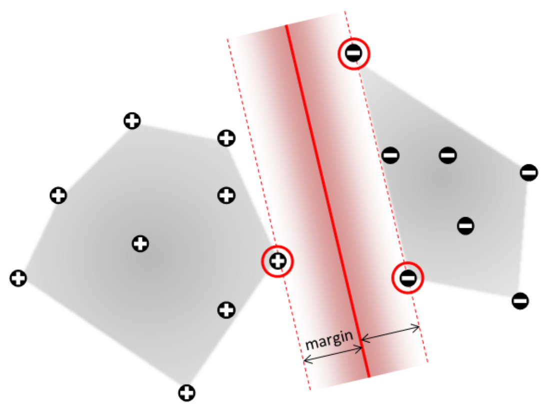

Fig. 4. A Typical SVM classifier

In most of real applications, the data could not be linearly classified. To deal with this problem, we transform data into a higher dimensional feature space and assume that our data in this space can be linearly classified. In fact, determining the optimal hyperplane is a constrained optimization problem and can be solved using quadratic programming techniques. The discriminant hyperplane is defined by

$$y(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i y^{(i)} K\left(\mathbf{x}^{(i)}, \mathbf{x}\right) + b \tag{6}$$

where $K(\mathbf{x}_1, \mathbf{x}_2)$ is the kernel function.

Algorithm 2: Construct classifier {

Step 1: Compute $\mathbf{H}$

$$H_{ij} = y^{(i)} y^{(j)} K\left(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}\right) \tag{7}$$

Step 2: This is a quadratic optimization problem with objective function

$$\alpha = \underset{\alpha}{\operatorname{argmin}} \left(\frac{1}{2}\alpha^T \mathbf{H}\alpha - \sum_{i=1}^{N} \alpha_i\right) \tag{8}$$

where $0 \leq \alpha_i \leq C$ and $\sum_{i=1}^{N} \alpha_i y^{(i)} = 0$.

Step 3: Compute b

$$b = \frac{1}{N_{idx}} \sum_{i \in idx} \left(y^{(i)} - \sum_{j \in idx} \alpha_j y^{(j)} K\left(\mathbf{x}^{(j)}, \mathbf{x}^{(i)}\right)\right) \tag{9}$$

where

$$\begin{cases} idx = \{i | \alpha_i > 0\} \\ N_{idx} = |idx| \end{cases}$$

Algorithm 3: Classify

Step 1: Compute

$$y = \operatorname{sgn}\left(\sum_{i=1}^{N} \alpha_i y^{(i)} K\left(\mathbf{x}^{(i)}, \mathbf{x}\right) + b\right) \tag{10}$$

Step 2: Classify

$$\begin{cases} \text{if } y = +1 \text{ then } \mathbf{x} \text{ belongs to positive class} \\ \text{if } y = -1 \text{ then } \mathbf{x} \text{ belongs to negative class} \end{cases}$$

3.2. Multi-class Classification

To apply SVM to face recognition, we use One-Against-All decomposition to transform multi-class problem to a set of two-class problems.

Dataset $\mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)}); \mathbf{x}^{(i)} \in \mathbf{R}^n; y^{(i)} \in \{1, 2, ..., C\}, i = 1, ..., N\}$ is transformed to C subsets $\mathcal{D}_c = \{(\mathbf{x}^{(i)}, y_c^{(i)}), y_c^{(i)} \in \{-1, +1\}\}$, where

$$y_c^{(i)} = \begin{cases} +1 & y^{(i)} = c \\ -1 & y^{(i)} \neq c \end{cases}$$

Algorithm 1 is used to compute the discriminant functions corresponding to $\mathcal{D}_c$

$$f_c(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i y_c^{(i)} K\left(\mathbf{x}^{(i)}, \mathbf{x}\right) + b \tag{11}$$

In classification task, we use the following rule to identify the class for input $\mathbf{x}$.

$$c = \arg\max_{c} \left(f_c(\mathbf{x})\right) \tag{12}$$

4. IMPLEMENTATION AND EXPERIMENTS

We used AT&T and CalTech databases to evaluate our approach. The AT&T database was taken at AT&T Laboratories. It contains 400 images (92-by-112) of 40 individuals (36 males and 4 females). Each person has ten images. In preprocessing step, face regions of AT&T images were cropped and scaled to 80-by-80 resolution (see Fig. 5a). CalTech database was collected by Markus Weber at California Institute of Technology. It contains 450 color images (896-by-592) of frontal faces of 26 individuals (17 males and 9 females). Each person has from 5 to 29 images. In preprocessing task, we converted all images into grayscale format. Next, face regions of images were identified and extracted from the background of the input images using Viola and Jones algorithm [11]. Then, they were scaled to 80-by-80 resolution (see Fig. 5b). To implement our system, we used Matlab and OpenCV to build image processing module and RapidMiner to build classification module.

Fig. 5. a) Some images from dataset AT&T b) Some images from dataset CalTech

Table 1. Experimental results on AT&T database

Test	Feature		Classification		Problem	Accuracy (%)
	Method	Parameters	Method	Parameters		
1	GLO-PCA	n=72	k-NN	k=1	gender	98.8
2	GLO-PCA	n=72	MLP	sigmoid	gender	85.7
3	GLO-PCA	n=72	SVM	rbf, C=4.0	gender	98.9
4	GRA-PCA	n=69	k-NN	k=1	gender	78.5
5	GRA-PCA	n=69	MLP	sigmoid	gender	65.4
6	GRA-PCA	n=69	SVM	rbf, C=4.0	gender	78.9
7	MY-PCA	n=79	k-NN	k=1	gender	98.7
8	MY-PCA	n=79	MLP	sigmoid	gender	85.1
9	MY-PCA	n=79	SVM	rbf, C=4.0	gender	98.2
10	GLO-PCA	n=72	k-NN	k=1	id	98.0
11	GLO-PCA	n=72	MLP	sigmoid	id	81.3
12	GLO-PCA	n=72	SVM	rbf, C=4.0	id	98.2
13	GRA-PCA	n=69	k-NN	k=1	id	75.6
14	GRA-PCA	n=69	MLP	sigmoid	id	62.3
15	GRA-PCA	n=69	SVM	rbf, C=4.0	id	76.8
16	MY-PCA	n=79	k-NN	k=1	id	97.9
17	MY-PCA	n=79	MLP	sigmoid	id	84.8
18	MY-PCA	n=79	SVM	rbf, C=4.0	id	98.0

4.1. Experiments on AT&T database

In our experiments, we used three kinds of input vectors corresponding to three types of feature vectors. They are global vectors (GLO-PCA), gradient vectors (GRA-PCA) and our proposed vectors (MY-PCA). We used the algorithm 1 to create PCA feature extractor. The dimension of feature vectors was selected such that they could keep at least 95% of the variance of the training sets.

We implemented three classification methods to conduct experiments on the AT&T database to solve two problems of face gender classification and identification recognition:

- k-Nearest Neighbor (k-NN): It uses distance metric L2 for classification; with the value of k set to 1, 3, 5, 7 and 9
- Multi Layer Perceptron (MLP): The MLP has three layers: input layer, hidden layer and output layer. It uses Gradient Back-Propagation algorithm
- Support Vector Machine (SVM): It applies nonlinear SVM with the kernel function Radial Basis Functions and the value of C set to $2^{-10}, 2^{-9}, ..., 2^{0}, ..., 2^{9}, 2^{10}$.

The reported results were obtained with Cross-Validation analysis with 5-folds on the dataset. They are given in Table 1.

4.2. Experiments on CalTech database

We used only two feature vectors (GLO-PCA and MY-PCA) and two classification methods (k-NN and SVM) to conduct experiments on the CalTech database: The experiments were conducted using Cross-Validation analysis with 5-folds. The results are given in Table 2. In addition, The ROCs of Test 4 is shown in Fig. 6

Table 2. Experimental results on CalTech database

Test	Feature		Classification		Problem	Accuracy
	Method	Parameters	Method	Parameters		
1	GLO-PCA	n=76	k-NN	k=1	gender	95.8
2	GLO-PCA	n=76	SVM	rbf, C=4.0	gender	96.2
3	MY-PCA	n=84	k-NN	k=1	gender	97.3
4	MY-PCA	n=84	SVM	rbf, C=4.0	gender	98.4
5	GLO-PCA	n=76	k-NN	k=1	id	93.5
6	GLO-PCA	n=76	SVM	rbf, C=4.0	id	93.8
7	MY-PCA	n=84	k-NN	k=1	id	97.2
8	MY-PCA	n=84	SVM	rbf, C=4.0	id	97.8

4.3. Discussions

The results for both datasets show that the classification method having the best result is SVM. It always give the highest accuracy in all experiments. In contrast, the worst is MLP. It always give the lowest classification rate.

Although our proposed vectors did not provide better classification rate for the AT&T dataset, it gives significant improvements for the CalTech dataset. The face images in AT&T are very good so the information from gradient images is unnecessary. However for other databases like CalTech, the information is very important and hence provides more good features for classification.

5. CONCLUSION

We have proposed a new approach for face gender classification and face recognition. It is using gradient face for extracting feature vectors and combining them with SVM for classification. We have compared our method with conventional methods.

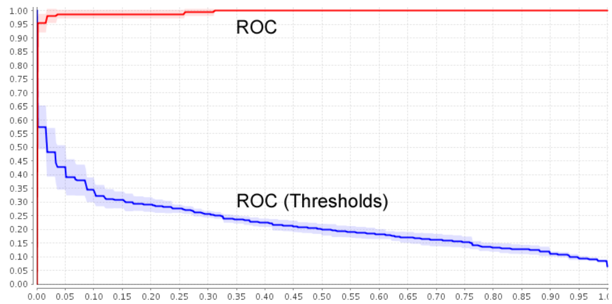

Fig. 6. ROC and ROC (Thresholds) for Test 4

6. REFERENCES

[1] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR '91., IEEE Computer Society Conference on*, 1991, pp. 586–591.

[2] L. Wiskott, J. M. Fellous, N. Kuiger, and C. von der Malsburg, "Face recognition by elastic bunch graph matching," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 19, no. 7, pp. 775–779, 1997.

[3] M. S. Bartlett, J. R. Movellan, and T. J. Sejnowski, "Face recognition by independent component analysis," *Neural Networks, IEEE Transactions on*, vol. 13, no. 6, pp. 1450–1464, 2002.

[4] Yang Ming-Hsuan, "Kernel eigenfaces vs. kernel fisherfaces: Face recognition using kernel methods," in *Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on*, 2002, pp. 215–220.

[5] Yang Jian, D. Zhang, A. F. Frangi, and Yang Jing-yu, "Two-dimensional pca: a new approach to appearance-based face representation and recognition," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 1, pp. 131–137, 2004.

[6] C Cortes and V Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[7] Chen Huajie and Wei Wei, "Pseudo-example based iterative svm learning approach for gender classification," in *Intelligent Control and Automation, 2006. WCICA 2006. The Sixth World Congress on*, 2006, vol. 2, pp. 9528–9532.

[8] B. Moghaddam and Yang Ming-Hsuan, "Gender classification with support vector machines," in *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, 2000, pp. 306–311.

[9] Han Xia, H. Ugail, and I. Palmer, "Gender classification based on 3d face geometry features using svm," in *CyberWorlds, 2009. CW '09. International Conference on*, 2009, pp. 114–118.

[10] Leng Xue-Ming and Wang Yi-Ding, "Gender classification based on fuzzy svm," in *Machine Learning and Cybernetics, 2008 International Conference on*, 2008, vol. 3, pp. 1260–1264.

[11] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[12] Le Thai Hoang and Bui Len Tien, "A hybrid approach of adaboost and artificial neural network for detecting human faces," in *Research, Innovation and Vision for the Future, 2008. RIVF 2008. IEEE International Conference on*, 2008, pp. 79–85.