Face Recognition by Combining Several Algorithms

Jie Zhou
Department of Automation, Tsinghua University, Beijing 100084, China
zhoujie@mail.au.tsinghua.edu.cn

David Zhang
Department of Computing, the Hong Kong Polytechnic University, Hong Kong
csdzhang@comp.polyu.edu.hk

Abstract: *There are many algorithms for human face recognition proposed in recent years. Here, we will study how to combine these algorithms to perform better. A fusion framework combining the existing recognition algorithms based on Support Vector Machine is developed. The experimental results demonstrate its effectiveness.*

1. Introduction

Face recognition is one of most important biometrics technologies because it is convenient and easy to be accepted by people [1]. In recent ten years, a large number of algorithms for automatic human face recognition have been proposed in the literature, including geometric feature-based matching, template matching, Eigenfaces, local feature analysis, elastic graph matching and neural network [2]. They vary in performances because of different principles and structures. It has been shown that combining different modalities enables to achieve a better overall performance [3], and then it is natural for us to develop one framework for integrating the existing face recognition algorithms in order to obtain a better result.

Approaches for combining classifier outputs include the Bayesian method, the sum and the product rules, the Borda count, logistic regression, to assign weights to the ranks produced by each classifier, the majority vote, the behavior-knowledge space method and Dempster-Shafer theory for weighted voting [4-7].

In most cases, face recognition need to compute the similarity between each two faces and further judge which two faces are most similar. We hope that the output of the fusion framework is also a continuous value, which can be used for comparison. Unfortunately, the above approaches cannot meet such a need.

Support Vector Machine (SVM) [8] is considered as a good candidate of pattern classification because of its high generalization performance without the need to add a priori knowledge. It is based on the principle of *structural risk minimization* (SRM), which states that better generalization capabilities are achieved through a minimization of the bound on the generalization error. Since there are always a small number of training samples in face recognition applications, SVM is very suitable to be used there.

In this paper, a novel fusion framework using SVM for human face recognition is proposed, which can take the recognition results of some existing algorithms as inputs and obtain a continuous output value. To illustrate its effectiveness, three typical algorithms are chosen to test our framework and the experimental results are rather satisfying.

2. SVM-based fusion framework

2.1 SVM definition

Let $(\mathrm{x}_i, y_i)_{1\le i\le N}$ be a set of training examples, each $\mathrm{x}_i \in R^n$, where n is the dimension of the input space, belongs to a class labeled by $y_i \in \{+1,-1\}$. Our aim is to define a hyper-plane that divides the set of examples such that all the points with the same label are on the same side of the hyper-plane [5]. The optimal separating function can be expressed as

$$f(\mathrm{x}) = \mathrm{sgn}\left(\sum_{i=1}^{N} \alpha_i y_i K(\mathrm{x}_i, \mathrm{x}) + b\right), \quad (1)$$

where $K(\mathrm{x},\mathrm{y})$ is a positive symmetric function, called kernel function, and b is a bias estimated on the training set. The parameters, $\{\alpha_i\}$, can be achieved by maximizing

$$W(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j K(\mathrm{x}_i, \mathrm{x}_j), \quad (2)$$

1051-4651/02 $17.00 © 2002 IEEE

under constraints $\sum_{i=1}^{N} y_i\alpha_i = 0$ and $0 \le \alpha_i \le C$. In the non-separable case, the constant, C, must be set to a given value. We can adopt an empirical approach for its choice and set it to an arbitrary value. In our experiments, it is chosen as 1000. Also, the kernel functions, $K(\mathrm{x}, \mathrm{y})$, must satisfy Mercer's conditions. Two kinds of possible kernel functions can be selected as

— **Polynomial Kernels**: $\mathrm{K}(\mathrm{x}, \mathrm{y}) = (\mathrm{x}^t \mathrm{y} + 1)^d$, where d is a positive integer that defines a polynomial decision surface; and

— **Gaussian Kernels**: $\mathrm{K}(\mathrm{x}, \mathrm{y}) = e^{-g\|\mathrm{x}-\mathrm{y}\|^2}$, where g is a positive real value.

2.2 Fusion framework

Suppose we have already developed n face recognition algorithms. So, for the given face image and any registered person, each algorithm can deliver a matching score between them. Assume that the matching score is negative and the smaller the score, the given face is more similar to that person. When combining these n algorithms, the similarity between the given face and that person can be represented by n matching scores. This means that the fusion scheme will process an n-dimensional vector, where its ith component is the matching score delivered by the ith algorithm. The fusion module will take the vector as input and produce a new confidence value.

SVM is originally defined for two-category classification, which becomes the foundation of nearly all published papers applying SVM to pattern recognition tasks. In [9], SVM was also used to give a binary decision by fusing two different data, i.e. the fusion module produced a binary value. However, combining different modalities to represent a binary decision is not enough for most face recognition applications and it is better to define a confidence value to compare the similarity of face images. We hope that a continuous confidence measure based on integrating these modalities can be developed. That is, given a vector $\mathrm{x} = (x_1, x_2, \cdots, x_n)$, where $x_i (\ge 0)$ is the matching score delivered by the ith face recognition algorithm, there should exist a function, $q(\mathrm{x})$, satisfying

(a) The value of $q(\mathrm{x})$ should be continuous;

(b) In order to compare the confidence measurement for face recognition, the smaller value of $q(\mathrm{x})$ means the larger similarity between the given face and the registered person.

Since this new measurement is based on matching scores of different algorithms, it should also satisfy the below constraint: When face A is compared with two persons, B and C, if A is more similar to B than C using each algorithm, A should be more similar to B other than C by using the new measurement. This means that the new function, $q(\mathrm{x})$, should also satisfy

(c) $\frac{\partial}{\partial \mathrm{x}} q(\mathrm{x}) \ge 0$.

Then, a number of vector samples are needed to train SVM. These vectors are divided into two subsets: the set of positive vector samples (i.e. the given face image is of that person) and the set of negative vector samples (i.e. it is not belonging to that person). Using the training algorithm, we can get an optimal separating hyper-plane between the positive samples and negative samples as

$$\sum_{i=1}^{N} \alpha_i y_i K(\mathrm{x}_i, \mathrm{x}) + b = 0. \quad (3)$$

The distance between one arbitrary vector, x, and this hyper-plane can be measured by

$$p(\mathrm{x}) = \sum_{i=1}^{N} \alpha_i y_i K(\mathrm{x}_i, \mathrm{x}) + b, \quad (4)$$

where $\alpha_i \in [0, C]$. The $p(\mathrm{x}) > 0$ means that the vector, x, is on the positive side of the hyper-plane and $p(\mathrm{x}) < 0$ on the negative side.

Obviously, the value of $p(\mathrm{x})$ can be thought as one measurement of the degree for the vector, x, belonging to "positive". The larger $p(\mathrm{x})$ means a higher degree of vector x being "positive". As stated above, the word "positive" represents the given face belonging to the registered person. Then $p(\mathrm{x})$ can be used to compare the similarity degree between the given face and that person. The larger value corresponds to the stronger similarity and on the contrary, the smaller value (including negative value) means a weaker similarity.

To satisfy the conditions (a) and (b), we can define the fusion function as

$$q(\mathrm{x}) = -\sum_{i=1}^{N} \alpha_i y_i K(\mathrm{x}_i \cdot \mathrm{x}). \quad (5)$$

Because the distribution of samples may be complex, it is difficult to prove that $q(\mathrm{x})$ satisfies the condition (c), i.e. $\frac{\partial}{\partial \mathrm{x}} q(\mathrm{x}) \ge 0$, in strict mathematical sense. But, a lot of experimental result has showed that this conclusion can hold in most real cases, where the positive examples always lie near the point, $(0, 0, \cdots, 0)$; at the same time, the negative examples distribute far from this point.

After the training step, the above confidence measurement can be obtained. For any testing face images, when it is compared with any registered person's face images in the database, a vector of matching scores is produced. Then Eq. (5) is utilized to give a confidence value so that the similarity between the given face and that person can be stated.

If the correlation between different modalities is strong, combining them using fusion strategy can outperform little. But if the correlation is small, combining them can have a much better performance. So, when we use the fusion framework for face recognition, it is a good way to select those algorithms with small correlation (i.e. utilizing different kinds of characteristics of human face). From the other side, this fusion framework can also be applied to test the correlation of several face recognition algorithms. If combining these algorithms performs much better than using each single modality, it means small correlation existing between them.

3. Fusion based face recognition

In this section, we want to demonstrate the effectiveness of our fusion framework by combining some popular face recognition algorithms. Suppose all face images have been geometrically normalized to establish the correspondence, and then three algorithms are considered as follows.

A. Geometric feature-based matching [10]: In this algorithm, the coarse positions of mouth and nose are located using template matching strategy. Then, a refined estimate of their real positions is obtained by looking for peaks and valleys of horizontal and vertical projection of edges. The height and width of mouth and nose can be finally computed in smaller windows. Eyebrow position and thickness can be found through a similar analysis. One elliptical curve-fitting program is taken to find the face outline. There are totally 24 geometrical features extracted by this algorithm. A distance can be defined to measure the similarity of one given face image with the registered person:

$$P = (\xi - m)^T \Delta^{-1} (\xi - m), \qquad (7)$$

where ξ is the feature vector of the given face image, m is the average vector representing the registered person, and Δ is the covariance matrix of all examples in the training set. Obviously $P \geq 0$ and the smaller the distance is, the more similar that given face is to that person.

B. Template matching [10]: Similar to geometrical feature-based matching, we should first find the positions of eyes, mouth and nose. Then, three small windows are chosen to contain eyes, mouth and nose, respectively. The image in each window is normalized using gradient information for template correlation. To compare two face images, the similarity score of each small window are added. The distance from one given face image to the registered person can be defined as the minimum of all the matching scores between the given image and each sample image of that person.

C. Eigenfaces [11]: The aim of the Eigenfaces is to identify the subspace of the image space spanned by the training face images and to decorrelate the pixel values. This can be achieved by finding the eigenvectors of the within-class scatter matrix. These eigenvectors with non-zero eigenvalues are called eigenfaces. The classical representation is obtained by projecting it to the coordinate system defined by eigenfaces. It also acts as a feature extraction mapping for face recognition. Two face images can be compared with the correlation value of their mapped vector. In our experiments, eigenfaces corresponding to 16 largest eigenvalues are used. The distance from one given face image to the registered person is defined as the same as in Template Matching. It is a non-negative function and the smaller distance corresponds to the larger similarity between the given face and that person.

In summary, these algorithms aim at different aspects of the face: Geometric Feature-based Matching mainly utilizes the geometrical distances between the key points of face; Template Matching exploits three small templates for the eyes, mouth and nose; Eigenfaces algorithm takes consideration of the whole face in low-dimensional space. They can compensate each other in some sense, so it seems that a better overall performance can be obtained by using our fusion framework.

We use three face recognition algorithms here and a 3-dimensional vector is produced to represent the similarity between a given face and the registered person. Then, each vector is inputted into the fusion framework and a new confidence value will be produced. We can compute the similarity between the given face and that registered person using this new measurement.

4. Experimental results

The experiments have been taken on a real data set, which contains 60 subjects. Every subject is required to take his/her photographs 10 times via CCD cameras in our laboratory. When photographing, the person's face is allowed to have movement and expression change. The lighting condition is intentionally varied someway. In order to employ the three face recognition algorithms directly, all these face images are geometrically normalized ahead.

All images are divided into two parts. Part I acts as the database for the registered users, which include 4 face images for each person. The remains serve as Part II, which is designed for the testing images.

10 subjects are selected for the training. We take three algorithms to produce one 3-dimensional vector between each face image in Part II and each person from Part I. Thus, there are totally $6 \times 10 \times 10$ =600 vectors for the training, in which 60 vectors are positive samples and the

others are negative samples. These samples are utilized to estimate parameters for the fusion scheme in Eq. (5). Note that polynomial kernel ($d = 3$) and Gaussian kernel ($g = 2$) are used in our SVM kernel functions.

After obtaining the parameters, face images of the remaining 50 subjects are taken for the test of our fusion framework. For any image in Part II, it is compared with each registered person (in Part I) by using the three algorithms, i.e. geometrical feature-based matching, template matching and eigenfaces, respectively. Thus, three kinds of similarity can be obtained to match the given image with the registered person. As a result, we input the three similarity results to Eq. (5) and produce a new confidence value.

The missing ratio corresponding to a given rank is computed to reflect the recognition ability for our framework and each single algorithm, respectively. If the rank is chosen as 3, the 'missing' means we cannot find the given image's true identity from the top 3 most similar persons in the registered database. The experimental results are listed in Table 1. It shows that face recognition using our fusion algorithm has a much better performance than directly using each single algorithm no matter the rank is chosen as 1, 2 or 3.

Table 1. Experimental results: the missing ratios of the algorithms (corresponding to different ranks)

Algorithms	Rank=1	Rank=2	Rank=3
Geometrical feature-based matching	21.7%	14.3%	9.7%
Template matching	11.0%	6.7%	4.0%
Eigenfaces	12.3%	8.3%	3.7%
Fusion (polynomial kernel)	4.3%	2.3%	0.3%
Fusion (Gaussian kernel)	5.7%	3.0%	0.7%

5. Conclusions

A lot of algorithms for human face recognition have been published in recent years. In this paper, we develop a fusion framework based on SVM to give a new confidence value, which can be used to compare the similarity between faces. Our fusion framework is suitable for the training of small-number samples such as in face recognition, because of SVM's good generalization. To demonstrate the framework's effectiveness, we select three popular face recognition algorithms and use the fusion framework to integrate them. The experimental results show that the performance is much better than each single algorithm.

6. Acknowledgements

The authors wish to acknowledge support from Natural Science Foundation of China under grants 69775009 and 69885004.

References

[1] D. Zhang, *Automated Biometrics: Technologies and Systems*, Kluwer Academic Publisher, USA, 2000.

[2] R. Chellappa, C.L. Wilson, and S. Sirohey, "Human and machine recognition of faces: a survey", *Proc. IEEE*, vol. 83, pp. 705-741, 1995.

[3] Y. Zhuravlev. An Algebraic Approach to Recognition and Classification Problem // Pattern Recognition and Image Analysis. Advances in Mathemathical Theory and Applications. vol. 8 (1), pp.59-100, 1998

[4] J. Kittler, M. Hatef, R. Duin, J. Matas, "On combining classifiers". *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 20 (3), pp.226-239, 1998

[5] T. Ho, J. Hull, S. Srihari, "On multiple classifier systems for pattern recognition". *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 16 (1), pp.66-75, 1994.

[6] L. Lam, C. Suen, "Optimal combination of pattern classifiers". *Pattern Recognition Letters*, vol. 16, pp. 945-954, 1995.

[7] L. Xu, A. Krzyzak and C.Y. Suen. "Methods of Combining Multiple Classifiers and Their Applications in Handwritten Character recognition". *IEEE Trans. System, Man and Cybernetics*, vol. 22, pp. 418-435, 1992.

[8] V. Vapnik. *Statistical Learning Theory*. New York: Wiley, 1998.

[9] S. Ben-Yacoub, Y. Abdeljaoued, and E. Mayoraz. "Fusion of Face and Speech Data for Person Identity Verification". *IEEE Trans. on Neural Networks*, vol. 10(5), pp. 1065-1072, 1999.

[10] R. Brunelli and T. Poggio. "Face Recognition: Features versus Templates". *IEEE trans. Pattern Analysis and Machine Intelligence*, vol. 15(10), pp. 1042-1052, 1993.

[11] M.A. Turk and A. Pentland. "Eigenfaces for Recognition." *Journal of Cognitive Neurosience*, vol. 3(1), pp. 71-86, 1994.