SVMs for few examples-based face recognition

Guoqin Cui, Wen Gao
Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080,China

***Abstract*—We present an extensive study of the support vector machine (SVM) to few examples–based face recognition problem. The few examples can't express many conditions the test data combine such as the changes of poses etc., so we use a simple method to generalize the examples to others. Then principal component analysis (PCA) has been applied to feature extraction from all the images, after that we use SVM to train and test the data. In the ICT-YCNC face gallery, the proposed system obtains competitive results: a correct recognition rate of 91.59% for all the 350 persons.**

***Index Terms*--face recognition, support vector machine, principal component analysis.**

I. INTRODUCTION

Face recognition has attracted much research report in the past thirty years. Although it has proven to be a difficult task even for frontal faces, certain algorithms can perform well under constrained conditions but the shortcoming is that the constraints always very cruel. Two of the most prominent works were [1], which introduced the eigenfaces that is very likely to Principal component analysis (PCA), and [2] [3], that introduce the Elastic Bunch Graph Matching, have been widely used as references. Recently, One of the most widely accepted methods of classification is Support Vector Machines [4], which have made it possible to obtain high accuracies for processing the high dimensional data in the pattern recognition problem.

Face recognition, different from other classical pattern recognition problems such as character recognition, there are many individuals (classes), and only a few images (samples) for a person, and algorithms must recognize faces by deducing from the training samples. To generalize new samples methods maybe a good thought, [6][7][8][9] talks about how to get other modes from the known examples, but they all very complexity and cannot easy to be used.

This paper we first briefly introduce the SVM algorithm and to extend SVM to process multi-class problem we have used, then describe how to get the training and test data from the face database with few examples of one class, at last we give the experiment results and summarize the conclusion.

II. SVM LEARNING AND CLASSIFIER METHOD

This section introduced the theory behind Support vector machines, [5] or [4] provided a detailed discussion.

A. Support Vector Machines

Given a labeled set of training samples (x_i, y_i), where $x_i \in R^N$ and $y_i \in \{-1,1\}$ is the associated label, then the classes of x_i can be partitioned into P for $y_i = +1$, and N for $y_i = -1$ two sets.

If the training data are linearly separable then there exists a pair (w,b) such that

$$y_i(w^T x_i + b) \geq 1, \forall\, x_i \in P \cup N \quad (1)$$

with the decision rule given by

$$f_{w,b}(x) = w^T x + b \quad (2)$$

w is expressed as the weight vector and b the threshold. The learning problem is hence reformulated as:

$$\min_{w,b} \Phi(w) = \frac{1}{2}\|w\|^2$$
$$s.t. \quad y_i(w \cdot x_i + b) \geq 1, \quad i = 1,2,\cdots,l \quad (3)$$

This is equivalent to maximizing the distance between the two classes P and N. Introducing slack variables $\xi_1,\ldots,\xi_l \geq 0$ to allow for the possibility of examples violating (1), and using the optimization theory, the problem (3) can then be changed to the following problem.

$$\max_{\mathrm{A}} L_{\mathrm{D}}(\Omega) = \sum_{i=1}^{l}\alpha_i - \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j x_i \cdot x_j$$
$$\text{s.t.} \quad 0 \leq \alpha_i \leq C, i = 1,\cdots,l \quad (4)$$
$$\sum_{i=1}^{l}\alpha_i y_i = 0$$

The decision function (2) can be given by

$$w = \sum_{i=1}^{l}\alpha_i y_i x_i \,, b = y_i - w \cdot x_i$$

Where x_i is called support vector for which $0 < \alpha_i < C$.

In order to generalize the linear decision function to non-linear decision surface, we first mapped the data to some higher dimensional feature space denoted by

$$R^N \rightarrow H, \quad x \rightarrow \phi(x)$$

and introduce kernel function K: $K(x,z) \equiv \phi(x) \cdot \phi(z)$.

Training is the same as (4), but $x_i \cdot x_j$ must be changed by $K(x_i, x_j)$. The decision function (2) becomes

$$f(x) = \sum_{i=1}^{l} y_i \alpha_i K(x, x_i) + b \quad (5)$$

In our system we use the following three kernel functions as learning machines: Linear function, Polynomial function, Gaussian radial basis function (RBF). $k(x,x') = x \cdot x'$, $k(x,x') = (x \cdot x'+1)^p$, $k(x,x') = \exp\left(-|x-x'|^2 / 2\sigma^2\right)$, Here the parameters p, σ are given before training and test.

B. Multiple Class Support Vector Machine for face images

Multi-class pattern recognition systems can be obtained by combining two-class SVMs. Different possibilities maybe two ways: one is to combine several binary classifiers by one against one, or one against the others that compares a given class with all the others put together; another is to modify the design of the SVM in order to incorporate the multi-class learning directly in the quadratic solving algorithm.

According to a study of the comparison of the above methods [10], the accuracies are almost the same. As a consequence, we chose "one against the others" method.

In order to train the data, we labeled the different person in the different label such as $0,1,\cdots,m-1$, and the face data are from the eigenface technique that we will narrate in *next chapter*. An *m-class* SVM algorithm will generate m different decision surfaces. For this m-$class$ (one for each algorithm a_k) we can get the binary SVM classifier $u_k(x)$ and can separate a_k to other classes. Thus we get the classifier of multi-class problem $a_{L(x)}$, for an input sample $x: L(x) = \arg\max_l \{u_l(x)\}$.

the $u_l(x)$ express the classifier function from SVM (5) :

$$u_l(x) = \sum_{i=1}^{l_k} y_{ki} a_{ki} K(x, x_{ki}) + b_k$$

When we test a sample z to decide the class it belong to we calculate the valued of $u_l(z)$, l is from 1 to m, after that we can get $L(z)$ and the correspondent class label l_0, since we can't get all the first one as the right choice, we also get other classes that are the nearest to the value of l_0 within the set $\{u_l(x)\}$, otherwise the claim is not the class label we want.

Sometimes we also use "one against one " method to confirm some results. In "one against one" method, each machine is trained as a classifier for one class against another class. In order to classify test data, pair-wise competition between all the machines is performed, and the final winner determines the class of the test data.

III. Application of SVM to face recognition

A. Face Database

We have used the ICT-YCNC face database, which contains a set of faces taken between August 1999 and April 2002 in our Face Recognition Laboratory. There are now more than 20,000 different images of 350 distinct subjects. For some of the subjects the images were taken at different poses. There are variations in facial expression and facial details. Figure 1 shows some variation in scale of up to about 15% thumbnails of part of the images in the gallery.

Figure 1.Part of The ICT-YCNC face database.

B. Preprocess the image

1) Preprocessing

Certainly at first we must preprocess the raw image to the gray level and normalize geometry shape, and remove background and hair. The procedure can be expressed as *figure 2*, consisted of auto locating the centers of the eyes,

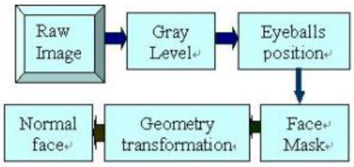
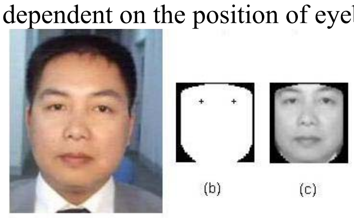


Figure 2.process of getting the normal face

translating, scaling, and rotating the faces to place the center of the eyes on specific pixels; use facemask to mask the faces to remove background and hair; histogram equalizing the non-masked facial pixels; and scaling the non-masked facial pixels to have zero mean and unit variance, then we get the normal face (*figure 3(c)*).

Since the eyeballs are the only features that are salient and have strong invariant property, the distance between them will be used to normalize faces for recognition. Motivated from this, with the face detected and the structural information extracted, we consider it to be the key to give the accurate position of eyeballs on face image. The facemask position is also dependent on the position of eyeball location.



(a) (b) (c)

Figure 3. (a) Original image (b) Face mask (c) Image after preprocessing.

2) Automatic location to the center of eyeballs

At first to a given face area, the image is two-binary, and some pair of connected areas are founded. By using the face structural knowledge, we can find a particular pair of connected area left and right where the two eyes locate.

In the binary image B and binary edge image E of the area L and R, the two eyes area can be described by:

$$Rng = \left\{(x,y) \mid B(x,y) = 0,\ \sum_{(i,j)\in X} E_{(i,j)} > \theta\right\}$$

X is the eight pixels set around the point (x, y),that can be moved in the face image field, θ is a threshold to decide whether the position is eyeball.

Then we get the cursory sets of the two eyes. The barycenter of the points set is calculated and we get some candidates of the irises (the center of the eyes). For each candidate, we calculate the following support function in the edge image E.

$$S_p = \frac{1}{N} \sum_{(x,y)\in A} E(x,y)$$

A is the annulus area with $R_1 \le r \le R_2$, and the one, which has maximum support function, is chosen as the iris.

$$P_{left-iris} = \arg(\max_{p \in Rng_{left}} S_p)$$

$$P_{right-iris} = \arg(\max_{p \in Rng_{left}} S_p)$$

After the two irises is located, we need rotated the image in order to fit with the facemask. We rotate the image based on the barycenter of the face rather than the total image. The barycenter of the face is calculated use the position of the two irises center position and the face structural knowledge.

When we finished to find the position of eyeballs, we can use facemask (figure 3(b)) to normalize the face image. The result is as *figure 3(c)* shows.

Since the ICT-YCNC database have simple background, the above method can get good result to gain the right label of the iris. The rate is about 97%. All the following algorithm and process are based on the image after the preprocessing.

After the process the true face database that we will use can be see as figure 4.

Figure 4.Examples of the ICT-YCNC train database after processing.

C. *Generalization of the Face Database*

A few face images can't show enough information of the poses or lighting conditions to a person as figure 5shows. For this reason we have to get felicitous multi-face images from the normal one in order to express more conditions.

Among papers directly related to the method, [6][7][8][9] addressed the problem of modeling human faces in order to generate virtual faces to be used in a face recognition system, these methods all must find the similar reference model and need cost too much time to change the model. We use the method only from the normal image, and very simple to get the images under different poses of a person.

In chapter 3 we have talked about how to get the normal face image by the position of eyes. We emphasize here that the following images are based on the normal face like *figure 3(c)*. In order to get the different images, we first estimate the rectangle that left iris is in the center as *figure 6.* From adjust the center position of the left iris to P_0, P1, ···, P_7, and keep the right iris fixedly. By the facemask (see *figure 3(b)*) we get eight different face images as the different poses from the normal face. Figure 7 shows the result of some examples.

D. *Face Representations*

It is natural to pursue dimensionality reduction schemes because great amounts of storage are difficult to process for its large dimension, for instance of the images we used which width and height all are 64,thus if we have not process the image we must process 4,096 dimension data. In our system we use the standard principal components analysis (PCA) technique also called eigenface method [1,11,12][13] to reduce the dimensions face vectors. Figure 8 shows the first 60 principal components that reconstructed from the 120 eigenfaces in our system.

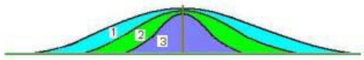


Figure 5.The estimator variance of different train data set.1 express fitting too more; 2 express fitting well; 3 express fitting insufficient.

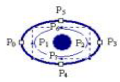


Figure 6. The different position of the pseudo left iris

Figure 7. Generalized views under variable pose from the first normal image

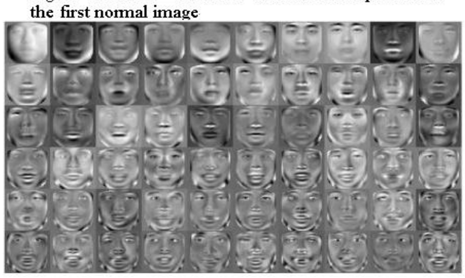


Figure 8.Some PCA basis images for the ICT-YCNC Face Database.

IV. EXPERIMENTS

In order to establish the performance of SVM, in comparison with other schemes, we carried out experiments on two independent face databases, the ICT-YCNC Face Database and the famous Yale Face Database. The Yale Face Database contains 165 images (11 per individual), with changes in facial expression, occlusion, and illumination conditions. From the ICT-YCNC Face Database we used 4795 face images with total 350 people. All the images were firstly processed according to what we have referred at first. From the comparison between Table 1 and Table 2, we can see that if we generalize the images in the face database as new training database, we can get notable rise of the recognition rates to the same database.

We use only one normal image as training data each one class, the other 10 images as the test images for the YALE database. Table 3 shows the different results when we use and not use the generalized method to the face database, which correspond to the labels in table 3 is Recognition Rate (2) and (1). The rank sequence we only shows from No. one to No. seven since after the seventh the rate all 100%. We use one to rest Multi-SVM and RBF function as the training and test kernel. From table 3 or figure 9,we observe that it if we don't use the generalized method, Multi-SVM does not give satisfactory performance (only 80.0% for rank =1, but 90.0% rank = 2), which is expected considering we only use a very small amount of information from the one face image. In this situation the classification is based only on the match of a single automatically extracted feature vector in the image to a stored one in the gallery. But when we use the generalized method, the recognition rates of rank 1 enhance 14%. That shows that the one sample image combined much other information that can be excavated.

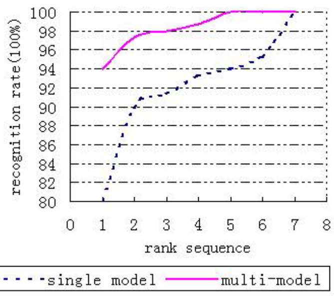


Figure 9.Comparison between two recognition rates of YALE face database

V. CONCLUSIONS AND FUTURE WORK

In this paper we present an extensive study of the support vector machine (SVM) to few examples–based face recognition problem, use a different method as P. J. Phillips has done [14]. Since the images for training can't express many conditions of the test database. We use a simple method to generalize the examples to other conditions and get notable good results. We can draw such a conclusion that we can get better results using SVM if we do more work on the generalization of the face database. The future work we will do research on other methods of generalization new views from a known face database.

REFERENCES

[1] M. Turk and A. Pentland. Eigenfaces for recognition. J. Cognitive Neuroscience, 3(1): 71–86, 1991.

[2] L. Wiskott, et.al. Face recognition by elastic graph matching. IEEE Trans. PAMI, pp.775-779, 1997.

[3] *M. Lades,et.al.* Distortion Invariant Object Recognition in the Dynamic Link Architecture, IEEE Transactions on Computers, 42,300--311,1993.

[4] V. Vapnik. The Nature of Statistical Learning Theory. Springer, N.Y., 1995.

[5] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. Data mining and knowledge discovery, Vol.2, pp.121-167. 1998

[6] *Timothy F. Cootes, Gareth J. Edwards, Christopher J. Taylor,* Active appearance model. IEEE Trans. PAMI, pp. 681-685,June 2001 (Vol. 23, No. 6).

[7] D. Beymer and T. Poggio. Face recognition from one example view. In Proc. 5th Intl. Conf. on Computer Vision, pages 500--507. IEEE, 1995.

[8] Athinodoros S. Georghiades, Peter N. Belhumeur, David J. Kriegman, From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose, IEEE Trans. PAMI, pp. 643-660

[9] W. Zhao and R. Chellappa, "Sfs based view synthesis for robust face recognition," in 4th IEEE Conference on Automatic Face and Gesture Recognition, Grenoble, France, 2000, pp. 285 --292.

[10] J.Weston and C. Watkins, "Multi-class support vector machines", Tech. Rep. CSD-TR-98-04, Royal Holloway, university of London, 1998.

[11] W. Zhao, A. Krishnaswamy, et.al. Discriminant Analysis of Principal components for Face Recognition. In *Face Recognition: From Theory to Applications*. Springer-Verlag, 1998.

[12] Guoqin Cui et.al, "Face Recognition Based on Support Vector Method", ACCV2002: The 5th Asian Conference on Computer Vision, 23--25 January 2002, Melbourne, Australia.

[13] Fernando De la Torre and Michael Black, Robust Principal Component Analysis for Computer Vision, ICCV2001, pp362—369.

[14] P. J. Phillips. Support vector machines applied to face recognition. In M. I. Jordan,M. J. Kearns, and S. A. Solla, editors, Advances in Neural Information Processing Systems 11, 1998.

Table 1: Recognition Rates obtained in ICT YCNC Database Using SVM

Train Num	Test Num	First Rank Recognition		SVM	Eigen Dim.	Kernel	C
		Error Num	Right%				
350x5	3045	514	83.2	One to rest	100	DOT	10,000
350x5	3045	505	83.4	One to rest	100	RBF(σ=1)	10,000
350x5	3045	499	83.62	One to rest	120	RBF(σ=1)	1,000
300x5	300x10	469	84.37	One to rest	120	RBF(σ=1)	1,000
300x5	3045	570	81.28	One to one	100	Poly (p=3)	1,000
300x5	300x10	481	84.20	One to rest	120	Poly (p=3)	1,000

Table 2: Recognition Rates obtained in Generalized ICT_YCNC Database Using SVM.

Train Num	Test Num	First Rank Recognition		SVM	Eigen Dim.	Kernel	C
		Error Num	Right%				
350x5	3045	326	89.29	One to rest	120	DOT	10,000
350x5	3045	315	89.66	One to rest	120	RBF(σ=1)	10,000
350x5	3045	256	91.59	One to rest	120	Poly(p=3)	10,000

Table 3. Test results rate of the face recognition system with each class containing only one normal image as the training set of YALE face base (64 eigenvector).

Selection Rank	1	2	3	4	5	6	7
Recognition Rate (1) (%)	80.0	90.0	91.3	93.3	94.0	95.3	100
Recognition Rate (2) (%)	94.0	97.3	98.0	98.7	100	-	-