Using Support Vector Machines to Enhance the Performance of Bayesian Face Recognition

Zhifeng Li, *Member, IEEE*, and Xiaoou Tang, *Senior Member, IEEE*

***Abstract*—In this paper, we first develop a direct Bayesian-based support vector machine (SVM) by combining the Bayesian analysis with the SVM. Unlike traditional SVM-based face recognition methods that require one to train a large number of SVMs, the direct Bayesian SVM needs only one SVM trained to classify the face difference between intrapersonal variation and extrapersonal variation. However, the additional simplicity means that the method has to separate two complex subspaces by one hyperplane thus affecting the recognition accuracy. In order to improve the recognition performance, we develop three more Bayesian-based SVMs, including the one-versus-all method, the hierarchical agglomerative clustering-based method, and the adaptive clustering method. Finally, we combine the adaptive clustering method with multilevel subspace analysis to further improve the recognition performance. We show the improvement of the new algorithms over traditional subspace methods through experiments on two face databases—the FERET database and the XM2VTS database.**

***Index Terms*—Bayesian analysis, face recognition, support vector machine (SVM).**

I. Introduction

FACE recognition has been one of the most challenging computer vision research topics over the past three decades. A number of face recognition algorithms have been developed in recent years [1], [2]. Among the existing face recognition techniques, subspace methods are widely used in order to reduce the high dimensionality of the raw face image. The Eigenface method [3]–[6] was a first breakthrough for the subspace techniques. The method uses the Karhunen–Loeve Transform (KLT) to produce the most expressive subspace for face representation and recognition. Linear discriminant analysis (LDA) or Fisherface [7]–[12] is an example of the most discriminating subspace methods. It seeks a set of features that best separates face classes. Another important subspace method is the Bayesian algorithm using probabilistic subspace [13], [35]. Different from other subspace techniques, which classify the test face image into M classes of M individuals, the Bayesian algorithm casts the face recognition problem into a binary pattern classification problem with each of the two classes—intrapersonal variation and extrapersonal variation—modeled by Gaussian distribution. In addition to directly processing the original image, subspace methods can also be applied to other features, such as shape and wavelet features. Cootes and Taylor developed the active appearance model (AAM) [28] to explicitly model both shape and texture. Liu and Wechsler applied the Enhanced Fisher Classifier on face recognition based on integrated shape and texture [29] and on Gabor features [30].

After subspace features are computed, most methods use the simple Euclidian distance of the subspace features to classify the face images. Recently, more sophisticated classifiers, such as support vector machines (SVM), have been shown to further improve the classification performance of the PCA and LDA subspace features [14]–[17], [34]. The SVM method is based on the principal of maximal margin bound. Intuitively, given any two classes of vectors, the aim of SVMs is to find one hyperplane to separate the two classes of vectors so that the distance from the hyperplane to the closest vectors of both classes is maximized. The hyperplane is known as the optimal separating hyperplane. SVMs excel at two-class recognition problems and outperform many other linear and nonlinear classifiers.

Since SVM is basically a binary classifier, to apply it to face recognition, which is a typical multiclass recognition problem, we have to reduce the multiclass classification to a combination of SVMs. There are several strategies to solve this problem, among which one-versus-all strategy and pairwise strategy are often used [15], [18]–[20]. Although both approaches can achieve high recognition accuracy, the latter is much simpler than the former. Studies have shown similar face-classification performance for the two approaches [15].

Since the number of classes in face recognition is often very large, for both the one-versus-all strategy and the pairwise strategy, a large number of SVMs have to be trained. In order to alleviate this problem, besides a one-versus-all Bayesian SVM algorithm, we also develop a direct Bayesian SVM by combining the Bayesian analysis method with the SVM directly [32]. The Bayesian method effectively converts the multiclass face recognition problem into a two-class classification problem, which is suitable for using the SVM directly. Therefore, the Bayesian SVM needs only one SVM trained to classify the face difference between within-class variation and between-class variation. Phillips [34] suggested such a combined framework for face recognition. However, it is accomplished in a heuristic manner.

Using only one hyperplane may not be enough to separate the entire within-class space and between-class space given the large number of samples. From experimental comparison, we see that the simplicity of the direct Bayesian SVM comes at a

Manuscript received October 3, 2005; revised October 9, 2007. This work was supported in part by grants from the Research Grants Council of the Hong Kong Special Administrative Region (Projects N_CUHK 409/03, CUHK 4224/03E, and 415105), and in part by the National Science Foundation of China under Grants 60318003 and 60572057. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Davide Maltoni.

The authors are with the Department of Information Engineering, The Chinese University of Hong Kong, Shatin, N. T., Hong Kong (e-mail: zli0@ie.cuhk.edu.hk; xtang@ie.cuhk.edu.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIFS.2007.897247

cost of accuracy. We can see that the two methods are at two extremes—one needs too many classifiers and the other has too few classifiers. In order to balance the two extremes, we further develop a two-stage Bayesian SVM. In the first stage, we estimate a similarity matrix to measure the degree of similarity between each pair of faces using the direct Bayesian SVM. Then using the similarity matrix and the hierarchical agglomerative clustering (HAC) algorithm [21]–[25], we group all face classes into clusters of similar faces. In the second stage, we perform the one-versus-all SVMs on the small number of classes within each cluster. During testing, we first use the original Bayesian method to classify the probe face to a cluster, and then the final classification is obtained within this cluster by the second-stage SVM. The method is shown to be as effective as the one-versus-all approach but is more efficient in computation. Notice that the clustering is based on the training data and, thus, stays the same in the testing stage. In order to cluster the data adaptively for each test face, we finally develop an adaptive clustering Bayesian SVM algorithm. We first use a simple Bayesian algorithm to find a cluster of faces that are most similar to the test face, then use a one-versus-all algorithm to reclassify the face in this cluster to find the final result. We use experiments on two face databases—the FERET face database [26] and the XM2VTS face database [27] to compare the four new algorithms with traditional subspace methods.

II. Bayesian SVM

In this section, we first provide a brief review of the SVM and Bayesian face recognition. We then develop the direct Bayesian SVM and the one-versus-all Bayesian SVM.

A. SVMs

In this section, we give a brief description of the basic idea of SVM. Systematic analysis and discussion on SVM can be found in [14].

Consider N points that belong to two different classes

$$\{(x_i, y_i)\}_{i=1}^{N} \quad \text{and} \quad y_i = \{+1, -1\} \tag{1}$$

where x_i is an n-dimension vector and y_i is the label of the class that the vector belongs to. SVM separates the two classes of points by a hyperplane

$$w^T x + b = 0 \tag{2}$$

where x is an input vector, w is an adaptive weight vector, and b is a bias. The goal of SVM is to find the optimal separating hyperplane, to maximize the margin (i.e., the distance between the hyperplane and the closest point of both classes). By Lagrangian formulation, the prediction of the SVM is given by

$$f_l(x) = \sum_{i=1}^{m} y_i \alpha_i \langle x, x_{si} \rangle + b \tag{3}$$

where m is the number of support vectors, each x_{si} representing a support vector and α_i is the corresponding Lagrange multiplier. Each test vector x is then classified by the sign of $f(x)$.

The solution can be extended to the case of nonlinear separating hyperplanes by a mapping of the input space into a high dimensional space $x \rightarrow \phi(x)$. The key property of this mapping is that the function ϕ is subject to the condition that the dot product of the two functions $\phi(x_i) \cdot \phi(x_j)$ can be rewritten as a kernel function $K(x_i, x_j)$. The decision function in (3) then becomes

$$f(x) = \sum_{i=1}^{m} y_i \alpha_i K(x, x_{si}) + b. \tag{4}$$

We use the popular Gaussian kernel in our study.

B. Bayesian Analysis

The Bayesian face recognition method converts the multiclass face recognition problem into a two-class classification problem by classifying the face difference as intrapersonal variations for the same person and interpersonal variations for different persons [12]. Letting Ω_I represent the intrapersonal variations and Ω_E represent the extra-personal variations, the ML similarity between any two images can be defined as

$$S(I_1, I_2) = P(\Delta | \Omega_I) \tag{5}$$

where Δ is the difference between the two images.

To estimate $P(\Delta|\Omega_I)$, we perform PCA on the face difference set $\{\Delta | \Delta \in \Omega_I\}$ to decompose the image difference space into two orthogonal and complementary subspaces: the principle subspace F, called intrapersonal eigenspace with K eigenvectors, and its complementary space $\bar{F}$ with $N - K$ eigenvectors. The likelihood can be estimated as

$$\hat{P}(\Delta|\Omega_I) = \left[\frac{\exp\left(-\frac{1}{2} d_F(\Delta)\right)}{(2\pi)^{K/2} \prod_{i=1}^{K} \lambda_i^{1/2}} \right] \left[\frac{\exp\left(\frac{-\varepsilon^2(\Delta)}{2\rho}\right)}{(2\pi\rho)^{(N-K)/2}} \right] \tag{6}$$

where $d_F(\Delta)$ is the so-called distance-in-feature-space (DIFS)

$$d_F(\Delta) = \sum_{i=1}^{K} \frac{y_i^2}{\lambda_i}. \tag{7}$$

In (6) and (7), y_i is the principle component of the principle subspace F, λ_i is the corresponding eigenvalue, $\varepsilon^2(\Delta)$ is the PCA residual error in $\bar{F}$, also called the "distance-from-feature-space" (DFFS), and ρ is the average of all the eigenvalues of $\bar{F}$

$$\rho = \frac{1}{N - K} \sum_{i=K+1}^{N} \lambda_i. \tag{8}$$

From (6), we can see that the estimation of $P(\Delta|\Omega_I)$ is equivalent to computing the distance measure in the intrapersonal subspace

$$D_I = d_F(\Delta) + \frac{\varepsilon^2(\Delta)}{\rho}. \tag{9}$$

For simplicity, we only use the DIFS in our study, since DFFS and MAP are costlier to compute. Our purpose is not to improve

the Bayesian method itself but to show how SVM can be combined with Bayesian to achieve better performance.

C. Bayesian SVM

As discussed before, SVM is a binary classifier. For the face recognition problem, we need to extend it to a multiclass classifier. The pair-wise strategy and the one-versus-all strategy are the two most popular methods. For the pair-wise strategy, one SVM is trained to separate each pair of classes. So the method needs $c*(c-1)/2$ SVMs trained, where c is the number of classes. During testing, each SVM votes for one class, and the winning class is the one that has the largest number of votes. For the one-versus-all strategy, each SVM is trained to separate a single class from the remaining classes. In other words, each class is associated with one hyperplane. So it needs c SVMs trained. Each test vector is assigned to the class whose hyperplane is farthest from it. Since the one-versus-all method is simpler and is as effective as the pair-wise method, we first adopt it to implement a straightforward one-versus-all Bayesian SVM.

However, for face recognition, the number of classes is often very large. The one-versus-all method needs to train a large number of SVMs. In order to alleviate this problem, we develop a simple Bayesian SVM for face classification. The method is straightforward since the traditional Bayesian algorithm already converts the face recognition problem into a two-class problem for the intrapersonal and the extrapersonal variation. We therefore only need to train one SVM for the two-class features.

For the training data, we first compute the image difference between images of the same person to construct the intrapersonal variation set $\{\Delta_I|\Delta_I \in \Omega_I\}$. We then compute the image difference between images of different persons to construct the extrapersonal variation set $\{\Delta_E|\Delta_E \in \Omega_E\}$. The eigenvalue matrix Λ_I and eigenvector matrix V_I of the intrapersonal subspace are then computed from the intrapersonal variation set $\{\Delta_I|\Delta_I \in \Omega_I\}$. Finally, all of the image difference vectors are projected and whitened in the intrapersonal subspace

$$\Delta_I' = \Lambda_I^{-1/2} V_I^T \Delta_I \tag{10}$$

$$\Delta_E' = \Lambda_I^{-1/2} V_I^T \Delta_E. \tag{11}$$

These two sets of image difference vectors are used to train the SVM to generate the decision function $f(\Delta)$. For the testing process, we again compute the face difference vector Δ_i between the probe vector x and each gallery vector x_i^g, and then project and whiten the difference vector in the intrapersonal subspace

$$\Delta_i' = \Lambda_I^{-1/2} V_I^T \Delta_i. \tag{12}$$

The final classification decision is made by

$$d(x) = \arg \max_{1 \le i \le c} (f(\Delta_i')) \tag{13}$$

where c is the number of people in the gallery. The larger the value of d is, the more reliable the result is.

The direct Bayesian SVM is simpler than the one-versus-all Bayesian SVM since it only needs one SVM trained. However, this new method may have oversimplified the problem since it uses one hyperplane to separate the intrapersonal variation and the extrapersonal variation. To balance the tradeoff between the two methods, we develop a two-stage SVM method in the next section.

III. Two-Stage Clustering-Based Classification

The problem with the one-versus-all approach is that too many SVMs need to be trained. On the contrary, the problem with the direct Bayesian SVM is too many samples for just one SVM. In this section, we try to find a solution that balances the two extremes.

When we train an SVM, the most important region in the training data space is around the decision hyperplane, since that is where mistakes often occur. Samples that are further away from the hyperplane play less significant roles in the training process. Therefore, it is reasonable to train an SVM for samples that are near the hyperplane. Toward this, we first partition the gallery data into clusters, with each cluster containing only similar images.

We first use the Bayesian SVM to quickly estimate the similarity matrix of the gallery set, and then use the HAC technique [21]–[25] to group the similar face clusters in order to reduce the number of binary SVMs in the second stage.

A. Hierarchical Agglomerative Clustering (HAC)

In the HAC process, clusters are constructed by combining existing clusters based on their proximity. The basic process of the HAC can be summarized by the following steps.

1) Initialize a set of clusters.

2) Find the nearest pair of clusters that have the largest similarity measure, and then merge them into a new cluster. Estimate the similarity measure between the new cluster and all the other clusters.

3) Repeat step 2 until the stopping rule is satisfied.

In each of the three steps of the basic algorithm, different strategies can be used to lead to different designs of the HAC algorithm. For example, in the first step, we can either assign each data point as a distinct cluster or form some initial small clusters for seeding. For face recognition, we can simply assign each image in the gallery as a cluster (assuming only one image per person in the gallery). In the third step, the stopping rule could either be that clustering has reached its root, or the clustering has reached the number of clusters specified by the user, or the similarity measure between the two nearest clusters is above a preset threshold. In our study, we will use the cluster number as stopping criteria. One key design issue for the HAC algorithm is the similarity measure between clusters in the second step. In the new algorithm described in the following section, we use the direct Bayesian SVM to estimate the similarity measure between face clusters. The output of the HAC will be a dendrogram, in which the similarity measure between any two clusters is the mean values of all the similarity values of image pairs

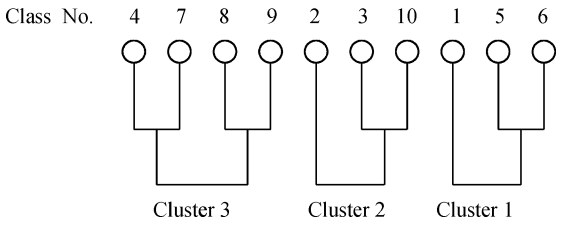


Fig. 1. Dendrogram example.

across the two clusters. An example is shown in Fig. 1, where ten classes are merged into three clusters.

B. Two-Stage SVM

In order to use HAC to partition the gallery face data into clusters, we first need to compute the similarity among the face images. For a pair of face images i and j in the gallery, we first compute the image difference Δ_E^{ij}, then project and whiten it in the intrapersonal subspace

$$\Delta_E^{ij\prime} = \Lambda_I^{-1/2} V_I^T \Delta_E^{ij}. \tag{14}$$

The similarity measure between the two images is then defined as

$$S_{ij} = f(\Delta_E^{ij\prime}) \tag{15}$$

where f is the SVM decision function in (3). The further away the image difference is from the decision hyperplane, the closer the image difference is to the intrapersonal variation. This means the two images are more similar to each other. The similarity values for all of the image pairs form the similarity matrix for the image gallery set. Using the similarity matrix, we then group the gallery dataset into clusters of similar images through the HAC.

After the similar images are clustered, in the second stage, we perform the one-versus-all Bayesian SVM within each cluster. Since the image number is much smaller in each cluster, the training complexity is significantly reduced. In addition, the SVM needs to only focus on a small number of similar images within each cluster. These data points are closer to the decision surface; thus, they are more likely to become support vectors.

During testing, we first compute the whitened face difference vector Δ_i' between the probe vector x and each gallery vector, and then simply find the face class that gives the smallest Δ_i'. This is equivalent to the original Bayesian method. If the output is probe class k, we find the face cluster $C(k)$ that contains class k. A second stage one-versus-all SVM is then performed on the cluster $C(k)$ to obtain the final classification result. Since the original Bayesian method only requires computation between two short feature vectors, it is much faster and is used in the first stage to rank all of the data. Then, the more costly one-versus-all Bayesian SVM is only needed to process one small cluster. So the complexity of the HAC clustering-based algorithm is much less than the one-versus-all approach.

However, since the clustering is based on the training data only, the face clusters will stay the same in the testing stage. They are tuned to the training data without any adaptation to the test data. In order to cluster the data adaptively for each test face, we further develop an adaptive clustering Bayesian SVM algorithm. We first use the original Bayesian algorithm to find a cluster of faces that are the most similar to the test face. We then use a one-versus-all algorithm to reclassify the face in this cluster to find the final result. Unlike the HAC clustering approach that only needs to train SVM classifiers in the training stage, if we have to retrain the one-versus-all classifier for each new cluster in the testing stage, the cost of computation will be simply too high. Instead, we train the one-versus-all Bayesian SVM in the training stage for all of the training data just like the original one-versus-all Bayesian SVM. We then use this one-versus-all Bayesian SVM to reclassify only the faces in the new cluster. So for training the complexity, it is the same as the one-versus-all, but for testing, the new cluster method will be much faster since it only needs to focus on a small cluster and the first-step original Bayesian algorithm is much faster. In experiments, we will see that this algorithm improves the recognition accuracy over all other methods. We will explain the reason in the experiment section.

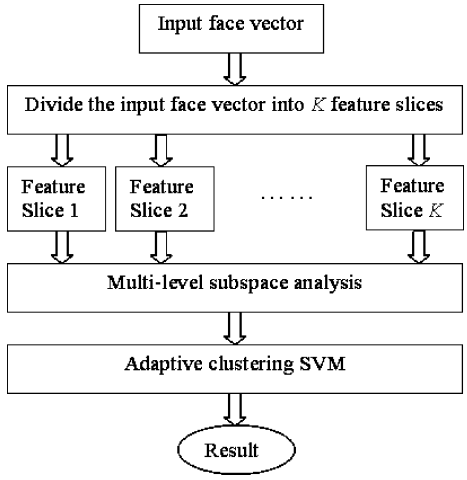


Fig. 2. Illustration of the adaptive clustering multilevel subspace SVM algorithm.

IV. Adaptive Clustering Multilevel Subspace SVM Algorithm

As discussed before, our proposed adaptive clustering SVM algorithm is indeed a two-stage algorithm. The classifier in the first stage helps to select a small subset of gallery classes used for the following SVM-based classification in the second stage. One thing to note is that the classier in the first stage can be any other classifier. This is another advantage of our adaptive clustering algorithm. The users can freely select the appropriate classifier in the first stage for convenience.

In order to achieve better recognition performance, we further develop an adaptive clustering multilevel subspace SVM algorithm by integrating the multilevel subspace analysis [31] in the first stage with the adaptive clustering algorithm in the second stage, as illustrated in Fig. 2.

The detailed algorithm is as follows.

In the first stage:

1) Divide the original face vector into K feature slices. Project each feature slice to its PCA subspace computed

from the training set of the slice and adjust the PCA dimension to reduce the most noise.
2) Compute the intrapersonal subspace using the within-class scatter matrix in the reduced PCA subspace and adjust the dimension of intrapersonal subspace to reduce the intrapersonal variation.
3) For the L individuals in the gallery, compute their training data class centers. Project all of the class centers onto the intrapersonal subspace, and then normalize the projections by intrapersonal eigenvalues to compute the whitened feature vectors.
4) Apply PCA on the whitened feature vector centers to compute the final discriminant feature vector.
5) Combine the extracted discriminant feature vectors from each slice into a new feature vector.
6) Apply PCA on the new feature vector to remove redundant information in multiple slices. The features with large eigenvalues are selected to form the final feature vector for recognition.

Steps 1 to 4 are the first level of the multilevel subspace analysis and steps 5 to 6 are the second level of the multilevel subspace analysis. That is why we call this algorithm multilevel subspace analysis [31]. The second stage of the adaptive clustering multilevel subspace analysis SVM algorithm is similar to that of the adaptive clustering Bayesian SVM.

The adaptive clustering multilevel subspace analysis SVM algorithm takes full advantage of the multilevel subspace analysis and adaptive clustering SVM and, thus, further improves the recognition performance. Furthermore, when the adaptive clustering multilevel subspace analysis SVM algorithm is applied to local features, such as elastic graph gabor features [33], we achieve the best recognition results in the experiment.

V. Experiments

In this section, we conduct experiments on two face databases—the FERET face database [26] and the XM2VTS face database [27]. To better evaluate the recognition performance, we preprocess the face images through the following steps: 1) rotate the face images to align the vertical face orientation; 2) scale the face images so that the distances between the two eyes are the same for all images; 3) crop the face images to remove the background and the hair region; and 4) apply histogram equalization to the face images for photometric normalization.

We compare all the four new algorithms with the three traditional subspace methods—PCA, LDA, Bayesian method and the conventional One-Versus-All LDA-based SVM algorithm.

A. Experiment on the FERET Face Database

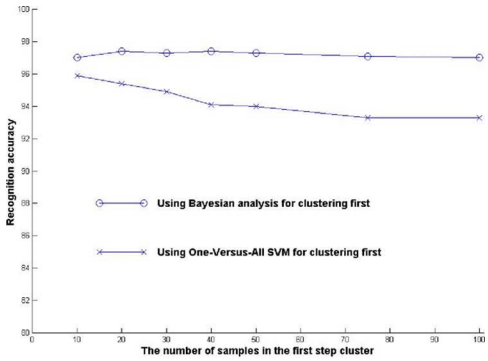
For the FERET face database (fa/fb) [26], we use 495*2 images of 495 people as training data, and use images of the other 700 people as test data. Therefore, the gallery set is composed of 700 images of 700 people. The probe set is composed of 700 images of the same 700 people.

When we train the SVM with one-versus-all strategy, the training samples are unbalanced (i.e., the number of samples for the positive (the same class) is often very small while the number of samples for the negative (the different classes) on the other hand is very large). Hence, it is crucial to balance between them. Considering that not all samples contribute to the discriminative learning, it is reasonable to select a portion of negative samples which contain the most discriminant information. As discussed before, the samples that are near the boundary usually play a more significant role in the training process and, thus, deserve emphasis. This mechanism is explicitly implemented in the training process of the one-versus-all SVM, in which only a portion of selected negative samples near the boundary together with the positive samples are used to train the one-versus-all SVM.

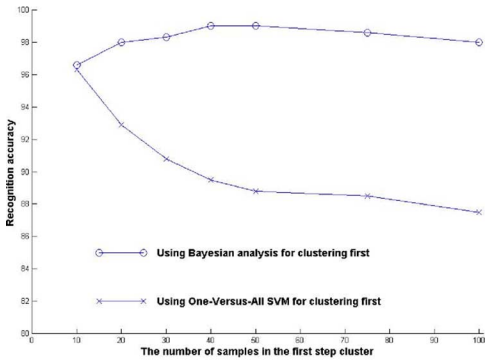
TABLE I
Recognition Error Rate on the FERET Database and the XM2VTS Database

Methods	Recognition Error Rate (%)	
	FERET	XM2VTS
PCA	15.4	33.9
LDA	9.7	11.9
Bayesian	6.7	11.5
Direct Bayesian SVM	3.9	6.8
One-Versus-All LDA-based SVM	6.1	10.8
One-Versus-All Bayesian SVM	4.0	2.7
HAC Bayesian SVM	3.7	2.7
Adaptive Clustering Bayesian SVM	2.6	1.0
Adaptive Clustering Multi-Level Subspace SVM	1.3	1.0
Gabor Features With One-versus-all SVM	3.4	3.1
Gabor-based Adaptive Clustering Multi-Level Subspace SVM	0.6	0.7

The recognition results of all the tested methods are summarized in Table I. From the results, we can see that the Bayesian SVM is only slightly better than the original Bayesian algorithm. On one hand, this lack of significant improvement confirms that using only one hyperplane is not enough to separate the intrapersonal and extrapersonal subspaces. On the other hand, the result the Bayesian SVM achieves is still very encouraging with a 3.9% error rate achieved even though only one simple hyperplane is used. This clearly shows the power



(a)



(b)

Fig. 3. Comparison of the recognition results for adaptive clustering using a different number of samples in the first step cluster. (a) FERET database. (b) XM2VTS database.

of the Bayesian framework. For the HAC clustering SVM, we set the cluster number to six. Both the one-versus-all method and the HAC-based method improve the recognition accuracy significantly. Compared to the original Bayesian method, the recognition error rate is reduced by 45%. One thing to note is that even though the training data are unbalanced for each individual SVM with each one having only one positive sample while many negative samples, the accumulated effect of all the classifiers can still perform very well. Finally, the adaptive clustering method gives the best accuracy of 97.4% among all Bayesian-based methods. This is very high accuracy for the FERET database. Both the HAC clustering and adaptive clustering methods are more efficient in computational cost since they only need to compute a small number of SVMs in the testing stage.

The good result for the adaptive clustering method is particularly interesting. Given that we use a regular one-versus-all method to reclassify the cluster of images selected by the original Bayesian method in the first step, instead of retraining the SVM, the method is effectively the same as combining the two classifier in a series operation. For the sake of comparison, we can also use the one-versus-all method first and then use the original Bayesian method. Of course, this is not a good approach since the former method is more expensive to compute. We select a different number of samples in the first step clustering for the two methods and compute the recognition accuracy. Fig. 3(a) shows the results for both methods. Clearly, using the first approach is much better. This can be explained by the complementary properties of the two classifiers. The Bayesian method is more stable but less accurate. The one-versus-all Bayesian, on the other hand, is more accurate but less stable since it is possible that one or a few of the large number of SVMs may produce a larger than normal distance measure outlier that happens to overshadow the real face class. When a stable Bayesian classifier is used first, it will help to remove these outliers from the selected cluster of candidates to help improve the performance of the one-versus-all Bayesian classifier. In the experiment, the algorithm reaches the best performance with only 20 images in the cluster. If we use the less stable one-versus-all method first and then use the original Bayesian, the performance is actually worse than using the one-versus-all method alone, since the Bayesian method is less accurate. As the number of images in the cluster increases, the combined method actually gets closer to the second algorithm with decreasing influence of the first.

Furthermore, when using the adaptive clustering method on the multilevel subspace analysis method, the recognition error rate is further reduced by 50%.

Finally, when this algorithm is applied to local features, such as Gabor features, we achieve the best accuracy of 99.4% on the FERET database. Compared with the results of using only the Gabor features with the one-versus-all SVM, the error rate is reduced by at least 80%.

B. Experiment on the XM2VTS Face Database

For the XM2VST database, we select all 295 people with four face images from four different sessions for each person. For the training data, we select 295*3 images of 295 people from the first three sessions. The gallery set is composed of 295 images of 295 people from the first session. The probe set is composed of 295 images of 295 people from the fourth session.

We implement the comparative experiments similar to the FERET face database experiment. Although the data size is smaller than the FERET database, the fact that the probe set and the gallery set in this experiment are from different sessions makes the recognition task also very challenging. This can be seen from the poor results of the PCA method, which is similar to direct matching of face images. The recognition results of the eight tested methods are summarized in Table I. The adaptive clustering recognition results for a different number of images are shown in Fig. 3(b). The results further confirm our observation in the FERET data experiments.

VI. Conclusion

In this paper, we first develop a direct Bayesian-based SVM by combining the Bayesian analysis with the SVM. The direct Bayesian SVM needs only one SVM to be trained to classify the face difference between within-class variation and between-class variation. However, with additional simplicity, the new method also has an inherent drawback. It tries to separate two complex subspaces by just one hyperplane. In order to improve the recognition performance, we further develop three more Bayesian-based SVMs, including the one-versus-all method, the HAC-based method, and the adaptive clustering method. We compare the new algorithm with traditional subspace methods—PCA, LDA, and Bayesian method through experiments on two face databases—the FERET face database

and the XM2VTS face database. The results clearly demonstrate the superiority of the new algorithm over traditional subspace methods. In addition, the clustering strategy is also extended to the multilevel subspace analysis [31] and elastic graph Gabor features [33] to further improve recognition performance.

REFERENCES

[1] R. Chellappa, C. L. Wilson, and S. Sirohey, "Human and machine recognition of faces: A survey," *Proc. IEEE*, vol. 83, no. 5, pp. 705–741, May 1995.

[2] W. Zhao, R. Chellappa, and P. Philips, Face Recognition: A Literature Survey UMD CAR, 2000, Tech. Rep. CAR-TR-948.

[3] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cognitive Neurosci.*, vol. 3, no. 1, pp. 71–86, 1991.

[4] M. Turk and A. Pentland, "Face recognition using eigenfaces," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, 1991, pp. 586–591.

[5] A. Pentland, B. Moghaddam, and T. Starner, "View-Based and modular eigenspaces for face recognition," presented at the IEEE Conf. CVPR, 1994.

[6] M. Kirby and L. Sirovich, "Application of the Karhunen-Loeve procedure for the characterization of human faces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 1, pp. 103–108, Jan. 1990.

[7] P. Belhumeur, J. Hespanha, and D. Kriegman, "Eigenface vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.

[8] K. Etemad and R. Chellappa, "Discriminant analysis for recognition of human face images," *J. Opt. Soc. Amer. A*, vol. 14, no. 8, pp. 1724–1733, 1997.

[9] W. Zhao and R. Chellappa, "Discriminant analysis of principal components for face recognition," in *Proc. IEEE Conf. Automatic Face and Gesture Recognition*, 1998, pp. 336–341.

[10] K. Etemad and R. Chellappa, "Face recognition using discriminant eigenvectors," in *Proc. ICASSP*, 1996, vol. 4, pp. 2148–2151.

[11] X. Wang and X. Tang, "Dual-space linear discriminant analysis for face recognition," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, 2004, vol. 2, pp. 564–569.

[12] Z. Li, W. Liu, D. Lin, and X. Tang, "Nonparametric subspace analysis for face recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Jun. 2005, vol. 2, pp. 961–966.

[13] B. Moghaddam, "Principle manifolds and probabilistic subspace for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 6, pp. 780–788, Jun. 2002.

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

[15] B. Heisele, P. Ho, and T. Poggio, "Face recognition with support vector machines: Global versus component-based approach," in *Proc. ICCV*, 2001, vol. 2, pp. 688–694.

[16] G. Guo, S. Z. Li, and K. Chan, "Face recognition by support vector machines," in *Proc. IEEE Int. Conf. Automatic Face and Gesture Recognition*, 2000, pp. 196–201.

[17] D. Xi, I. T. Podolak, and S. Lee, "Facial component extraction and face recognition with support vector machines," in *Proc. IEEE Int. Conf. Automatic Face and Gesture Recognition*, 2002, pp. 76–81.

[18] E. Allwein, R. Schapire, and Y. Singer, "Reducing multiclass to binary: A unifying approach for margin classifier," in *Proc. ICML*, 2000, pp. 113–141.

[19] R. Rifkin and J. Rennie, "Improving Multi-Class Text Classification with the Support Vector Machines," AI Memo, Mass. Inst. Technol., Cambridge, MA, 2001, AIM-2001-026.

[20] J. Platt, N. Cristianini, and J. Shawe-Taylor, "Large margin DATs for multiclass classification," in *Advance in NIPS*. Cambridge, MA: MIT Press, 2000, vol. 12.

[21] R. R. Sokal and P. H. A. Sneath, *Principles of Numerical Taxonomy*. San Francisco, CA: Freeman, 1963.

[22] R. Duda and P. Hart, *Pattern Classification and Scene Analysis*. New York: Wiley, 1973.

[23] K. Cios, W. Pedrycz, and R. Swiniarski, *Data Mining Methods for Knowledge Discovery*. Norwell, MA: Kluwer, 1998.

[24] J. Bezdek, J. Keller, R. Krisnapuram, and N. Pal, *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*. Norwell, MA: Kluwer, 1999.

[25] R. Yager, "Intelligent control of the hierarchical agglomerative clustering process," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 30, no. 6, pp. 835–845, Dec. 2000.

[26] P. Phillips, H. Moon, P. Rauss, and S. Rizvi, "The FERET evaluation methodology for face-recognition algorithms," in *Proc. IEEE CVPR*, 1997, pp. 137–143.

[27] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Matitre, "XM2VTSDB: The extended M2VTS database," in *2nd Int. Conf. Audio and Video-Based Biometric Person Authentication*, Mar. 1999, pp. 72–77.

[28] A. Lanitis, C. J. Taylor, and T. F. Cootes, "Automatic interpretation and coding of face images using flexible models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 743–756, Jul. 1997.

[29] C. Liu and H. Wechsler, "A shape- and texture-based enhanced Fisher classifier for face recognition," *IEEE Trans. Image Process.*, vol. 10, no. 4, pp. 598–608, Apr. 2001.

[30] C. Liu and H. Wechsler, "Gabor feature based classification using the enhanced Fisher linear discriminant model for face recognition," *IEEE Trans. Image Process.*, vol. 11, no. 4, pp. 467–476, Apr. 2002.

[31] X. Tang and Z. Li, "Frame synchronization and multi-level subspace analysis for video based face recognition," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, 2004, vol. 2, pp. 902–907.

[32] Z. Li and X. Tang, "Bayesian face recognition using support vector machine and face clustering," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, 2004, vol. 2, pp. 374–380.

[33] L. Wiskott, J. M. Fellous, N. Kruger, and C. von der Malsburg, "Face recognition by elastic bunch graph matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 775–779, Jul. 1997.

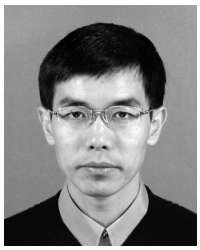
[34] P. J. Phillips, "Support vector machines applied to face recognition," *NIPS*, pp. 803–809, 1998.

[35] B. Moghaddam, T. Jebara, and A. Pentland, "Bayesian modeling of facial similarity," *NIPS*, pp. 910–916, 1998.

Zhifeng Li (M'06) received the B.S. degree (Hons.) from the University of Science and Technology of China (USTC), Hefei, China, in 1999, and the M.Phil. and Ph.D. degrees in information engineering from the Chinese University of Hong Kong (CUHK), Shatin, Hong Kong, in 2003 and 2006, respectively.

Currently, he is a Postdoctoral Fellow in the Department of Systems Engineering and Engineering Management (SEEM), the CUHK. His research interests include computer vision, pattern recognition, and multimodal biometrics.

Dr. Li is a Program Committee Member of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR) 2007 and the IEEE International Conference on Computer Vision (ICCV) 2007.



Xiaoou Tang (S'93–M'96–SM'02) received the B.S. degree from the University of Science and Technology of China, Hefei, China, in 1990, the M.S. degree from the University of Rochester, Rochester, NY, in 1991, and the Ph.D. degree from the Massachusetts Institute of Technology, Cambridge, in 1996.

Currently, he is a Professor in the Department of Information Engineering, the Chinese University of Hong Kong and the Group Manager of the Visual Computing Group at the Microsoft Research Asia, Beijing, China. His research interests include computer vision, pattern recognition, and video processing.

Dr. Tang is a Local Chair of the IEEE International Conference on Computer Vision (ICCV) 2005, Area Chair of CVPR'07, Program Chair of ICCV'09, and General Chair of the IEEE ICCV International Workshop on Analysis and Modeling of Faces and Gestures 2005. He has been a Guest Editor for the *IEEE Journal of Oceanic Engineering* and IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY. He is an Associate Editor of IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE and *Pattern Recognition Journal*.