Boosting for Fast Face Recognition

Guo-Dong Guo and Hong-Jiang Zhang
Microsoft Research China
5F, Beijing Sigma Center
No. 49, Zhichun Road, Haidian District
Beijing 100080, P. R. China
E-mail: $guodong\_guo@yahoo.com$

Abstract

We propose to use the AdaBoost algorithm for face recognition. AdaBoost is a kind of large margin classifiers and is efficient for on-line learning. In order to adapt the AdaBoost algorithm to fast face recognition, the original Adaboost which uses all given features is compared with the boosting along feature dimensions. The comparable results assure the use of the latter, which is faster for classification. The AdaBoost is typically a classification between two classes. To solve the multi-class recognition problem, a majority voting (MV) strategy can be used to combine all the pairwise classification results. However, the number of pairwise comparisons $n(n-1)/2$ is huge, when the number of individuals n is very large in the face database. We propose to use a constrained majority voting (CMV) strategy to largely reduce the number of pairwise comparisons, without losing the recognition accuracy. Experimental results on a large face database of 1079 faces of 137 individuals show the feasibility of our approach for fast face recognition.

Keywords: *Face recognition, large margin classifiers, AdaBoost, constrained majority voting (CMV), principal component analysis (PCA).*

1. Introduction

Face recognition technology can be used in a wide range of applications such as identity authentication, access control, and surveillance. Interests and research activities in face recognition have increased significantly over the past few years [8] [1]. Two issues are central for face recognition, *i.e.*, what features to use to represent a face, and how to classify a new face based on the chosen representation.

Principal Component Analysis (PCA), is a classical technique for signal representation [9]. Turk and Pentland [11] developed a well known face recognition method, the **eigenfaces**, based on the PCA technique for face representation. Some other complex methods such as ICA or non-linear approaches [6] can also be used to extract face features. Here, we focus on the classification problem, and choose to use the simple and efficient PCA technique [11] for face feature extraction.

In the standard eigenfaces approach [11], the nearest center (NC) criterion is used to recognize a new face. In [5], a probabilistic visual learning (PVL) method is developed for face recognition. Another way of Bayesian classification of faces is proposed in [4], called probabilistic reasoning models (PRM), based on some assumptions of the class distributions. More recently, the support vector machine (SVM) [12] is popular for visual object recognition [7]. The SVM constructs a hyperplane between two classes of examples based on the criterion of large margin. The face recognition accuracy based on SVM is relatively high [3]. However, in SVM, both the training and testing process is a little time consuming if the face database is very large. Recently, Freund and Schapire [2] proposed another kind of large margin classifiers, AdaBoost, to tackle the machine learning problems, which is fast and efficient for online learning. AdaBoost algorithm has the potential of fast training and testing for real-time face recognition. Hence, we concentrate on the AdaBoost algorithm and evaluate its performance for face recognition.

In the next Section, we describe the AdaBoost algorithm and give our strategies to adapt it for fast face recognition. Then, the constrained majority voting is presented in Section 3 to tackle the multi-class recognition problems. Section 4 shows the experimental evaluations of AdaBoost in face recognition. Finally, conclusions and discussions are given in Section 5.

2. AdaBoost

Boosting is a method to combine a collection of weak classification functions (weak learner) to form a stronger classifier. AdaBoost is an adaptive algorithm to boost a sequence of classifiers, in that the weights are updated dynamically according to the errors in previous learning [2]. AdaBoost is a kind of large margin classifiers. Tieu and Viola [10] adapted the AdaBoost algorithm for natural image retrieval. They made the weak learner work in a single feature each time. So after T rounds of boosting, T features are selected together with the T weak classifiers. If Tieu and Viola's version can get comparable results with the original Freund and Schapire's AdaBoost [2], it will be a better choice for face recognition because of the reduced computation of T comparisons instead of $T \times D$ in the original AdaBoost [2], where D is the feature dimension. To make it clear, we denote the original AdaBoost [2] as Boost.0. Because of the space limit, we do not give the original AdaBoost algorithm [2] here. Readers can refer to [2] for a detailed explanation. Tieu and Viola's version [10] is briefly described as below:

AdaBoost Algorithm

Input: 1) n training examples $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$ with $y_i = 1$ or 0; 2) the number of iterations T.

Initialize weights $w_{1,i} = \frac{1}{2l}$ or $\frac{1}{2m}$ for $y_i = 1$ or 0, respectively, with $l + m = n$.

Do for $t = 1, \ldots, T$:

1. Train one hypothesis h_j for each feature j with w_t, and error $\epsilon_j = Pr_i^{w_t}[h_j(x_i) \neq y_i]$.

2. Choose $h_t(\cdot) = h_k(\cdot)$ such that $\forall j \neq k, \epsilon_k < \epsilon_j$. Let $\epsilon_t = \epsilon_k$.

3. Update: $w_{t+1,i} = w_{t,i}\beta_t^{e_i}$, where $e_i = 1$ or 0 for example x_i classified correctly or incorrectly respectively, with $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$ and $\alpha_t = \log\frac{1}{\beta_t}$.

4. Normalize the weights so that they are a distribution, $w_{t+1,i} \longleftarrow \frac{w_{t+1,i}}{\sum_{j=1}^{n} w_{t+1,j}}$.

Output the final hypothesis,

$$h_f(x) = \begin{cases} 1 & if \ \sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2}\sum_{t=1}^{T} \alpha_t \\ 0 & otherwise \end{cases} \quad (1)$$

It should be noted, however, a problem emerges when Tieu and Viola's boosting is used for face recognition. Since it starts with the most discriminative feature and adds another one in next round of boosting, the algorithm may begins with a feature having zero classification error, i.e., $\epsilon_t = 0$, then $\beta_t = \frac{\epsilon_t}{1-\epsilon_t} = 0$. So $\alpha_t = \log\frac{1}{\beta_t}$ can not be defined, and the boosting should stop there [2]. Because boosting is based on the classification error in previous round [2]. It is explicit that very few rounds of boosting and hence very small number of features are not sufficient for the complicated task of face recognition. In fact, we find the phenomenon of zero boosting error in quite a few cases. To solve this problem and make the boosting process go forward, we define a small value for β_t instead of zero in case of $\epsilon_t = 0$. We let $\beta_t = 0.01$ and 0.1 and compare their effects on the recognition results. We call them Boost.1 and Boost.2 respectively corresponding to different settings of β_t values.

One suspicion still exits. That is, whether we need to weight the features with the distribution w_t in step 1 of AdaBoost. It should be clarified by experiments. For this purpose, we do not weight the features in Boost.1 and Boost.2, and call it Boost.3 if the distribution w_t is used to weight the features (and simultaneously set $\beta_t = 0.1$, if $\epsilon_t = 0$, which is experimentally better than $\beta_t = 0.01$). The weak learner is the simple nearest center classifier, as that in [10].

3. Multi-class Recognition

AdaBoost is typically used to solve two-class classification problems. In a multi-class scenario, we can use a majority voting (MV) strategy to combine all pair-wise classification results. However, it needs $\frac{n(n-1)}{2}$ pairwise comparisons, where n is the number of classes. In order to speed up the process for fast face recognition, we first use the nearest center criterion to rank all classes with respect to a given query. The class labels appear on the top list if the class centers are nearest to the query. Then, top m classes are selected and used for voting. We call it Constrained Majority Voting (CMV), which can largely reduce the number of comparisons. We compare the performance of CMV with the majority voting which uses all pairs of classes.

We also show the face recognition results with the method of probabilistic reasoning models (PRM) [4], which is an approximation to the Bayesian classifier with the assumption that the covariance matrix is diagonal. The recognition accuracy of the standard eigenface is also shown for comparison.

4. Experiments

Different versions of AdaBoost from Boost.0 to Boost.3 are evaluated on a compound face database with 1079 face images of 137 persons.

4.1. Face Database

The face database is a collection of five databases: (1). The Cambridge ORL face database which contains 40 distinct persons. Each person has ten different images. (2). The Bern database contains frontal views of 30 persons, each with 10 images. (3). The Yale database contains 15 persons. For each person, ten of its 11 frontal view images

Figure 1. Examples of face images in our face database.

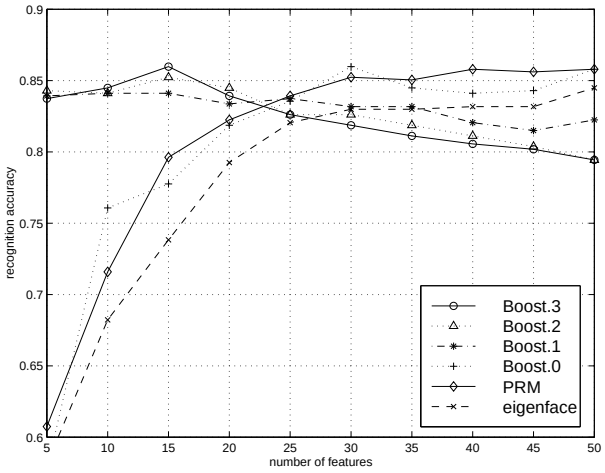

Figure 2. Face recognition performance with respect to rounds of boosting (or number of features). Only a small number of boosting is sufficient for AdaBoost.

are randomly selected. (4). Five persons are selected from the Harvard database, each with 10 images. (5). A database composed of 179 images of 47 Asian students, each with three or four images. The face images are cropped and scaled to the same size of 128×128 pixels in our database. The face images have large variations in facial expressions and facial details, and changes in light, face size and pose. Some face examples in the database are shown in Fig. 1.

The face database is divided into two non-overlapping sets for training and testing. The training data consist of 544 images: five images per person are randomly chosen from the Cambridge, Bern, Yale, and Harvard databases, and two images per person are randomly selected from the Asian students database. The remaining 535 images are used for testing.

4.2. Experimental Results

Firstly, the principal components are calculated from the face images in the training set. The projection coefficients of face images to these principal components are computed and used as the features. Then, different algorithms are used for face recognition with respect to the number of features or rounds of boosting in AdaBoost. We compare the original Adaboost (Boost.0) with Boost.1 ($\beta = 0.01$, if $\epsilon = 0$, without using the distribution w_t to weight a new coming feature), Boost.2 which is the same as Boost.1 except for setting $\beta = 0.1$, if $\epsilon = 0$, and Boost.3 (using previous distribution w_t to weight a new feature in boosting, and setting $\beta = 0.1$, if $\epsilon = 0$). It is shown in Fig. 2 the recognition rates of each algorithm with respect to rounds of boosting or the number of features (for PRM and eigenfaces). We can observe several results: 1). The four versions of boosting can give comparable results. This guarantee to use the simple boostings instead of the original complex one. In detail, the best recognition rate of Boost.0 is 85.98% ($T = 10$, $dim = 30$), while Boost.1 is 84.11% ($T = 10$ or 15), Boost.2 is 85.23% ($T = 15$), and Boost.3 is 85.98% ($T = 15$). The results of Boost.2 and Boost.3 are slightly better than Boost.1, and comparable to Boost.0; 2). The different behavior of Boost.1 to Boost.3 shows the effects of various settings of interior parameters on the recognition performance. Boost.3 is preferable in small number of boosting rounds ($T \leq 15$); 3). The recognition accuracy of Boost.3 is not lower than the approximate Bayesian classification, PRM [4], which gives 85.79% recognition accuracy with 40 features. This demonstrates the acceptability of boosting for face recognition; 4). Both boosting and PRM can improve the recognition rates over the standard eigenfaces, however, the boosting algorithms select less features to use (15 features are sufficient); 5). The problem of over-fitting is serious for boosting on face data. When $T > 15$, the performance deteriorates obviously. It is interesting to observe that, when the round of boosting is small ($T \leq 15$), the recognition performance is Boost.3 $>$ Boost.2 $>$ Boost.1, where "$>$" represents "better than". While Boost.3 degenerates more rapidly if the number of boosting rounds becomes larger ($T > 15$).

In above, we use majority voting to solve the multi-class recognition problem. Thus it should do 9316 pairwise comparisons for a single query. In order to speed up the process,

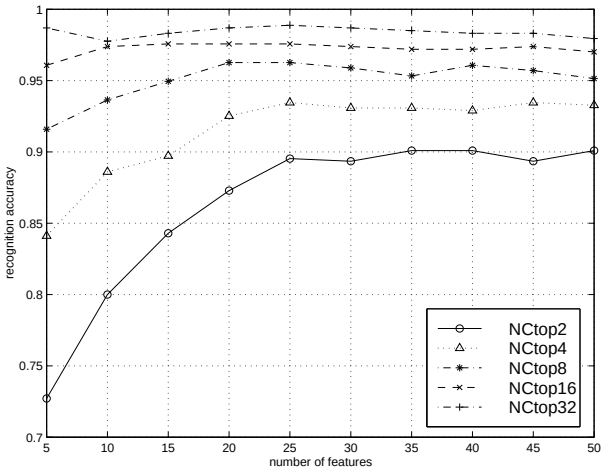

Figure 3. The recognition accuracy of top m classes ranked by NC criterion with respect to the number of features for $m = 2, 4, 8, 16, 32$.

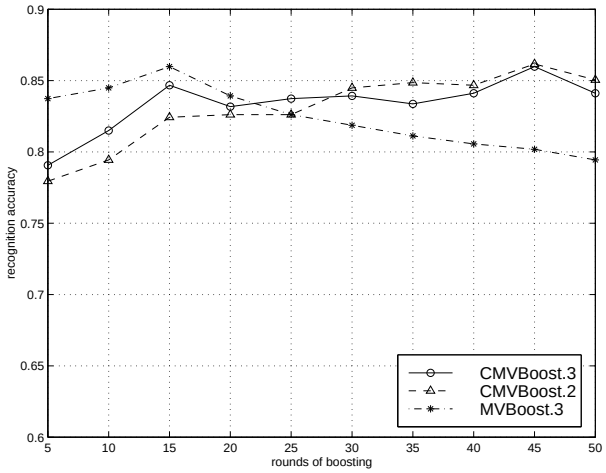

Figure 4. The recognition accuracy of boosting by using the constrained majority voting (CMV) strategy, with respect to rounds of boosting.

we propose to use a constrained majority voting (CMV) strategy. To do it, we must firstly demonstrate the efficiency of class ranking with the nearest center (NC) criterion. It is shown in Fig. 3 the recognition rates of top m classes, for $m = 2, 4, 8, 16, 32$. The selection of m is arbitrary here, and we just set m as the power of 2. We can find that top 32 classes (with 25 features) can cover 98.88% correct classes. Hence it is safe to use only a small number of classes to feed into the multi-class solver – CMV.

In our experiments with CMV, we only use top 4 classes, ranked by NC with 25 features. The number of pairwise comparisons is largely reduced from 9316 to 6. The recognition performance with CMV is shown in Fig. 4. We try both Boost.2 and Boost.3 using CMV, denoted as CMVBoost.2 and CMVBoost.3 respectively, and compare their results with Boost.3 in Fig. 2, but here denoted as MVBoost.3. It is interesting to observe that the boosting behavior has changed somewhat with respect to boosting rounds. The best results of boosting with CMV now correspond to 45 rounds of boosting. But the computation time is still reduced explicitly as compared with MVBoost.3. The best recognition rate of CMVBoost.2 is 86.17%, and CMVBoost.3 is 85.98%, comparable to the MVBoost.3 of 85.98% (15 rounds of boosting).

5. Conclusions and Discussions

We have evaluated the AdaBoost algorithm for face recognition. Boosting along dimensions can give comparable results as that using all features in each round. Hence both learning and testing processes can be largely sped up. To overcome the problem of $\epsilon_t = 0$ in some beginning rounds of boosting, two small values are tried for β_t. From the experiments, it is better to set $\beta_t = 0.1$ than 0.01. Further more, it makes little difference to weight the features or not in the boosting process along the feature dimensions. To further speed up the multi-class face recognition process, the constrained majority voting (CMV) strategy can be used, which is faster than the traditional majority voting strategy using all pairs, without explicitly losing the recognition accuracy. As a result, both CMVBoost.2 and CMVBoost.3 can be used for fast face recognition. Additional observation is that over-fitting is a serious problem for boosting on face data. Our experimental evaluations should stimulate more research on boosting method itself for face recognition, which can be expected to further improve the face recognition accuracy.

More recently, a new web site on AdaBoost, **http://www.boosting.org**, was just opened for researchers to exchange their research results or do discussions. This is useful to stimulate and speed up the research on boosting methods and their applications.

6. Acknowledgements

The authors would like to thank Kinh Tieu and Gunnar Ratsch for their helpful discussions on AdaBoost algorithm.

References

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

[2] Y. Freund and R. E. Schapire, A decision-theoretic generalization of online learning and an application to boosting. *J. Comp. & Sys. Sci.*, 55(1):119-139, 1997.

[3] G. Guo, S. Li, and K. Chan, Face recognition by support vector machines, *Proc. of the International Conferences on Automatic Face and Gesture Recognitionz*, 196-201, 2000.

[4] C. Liu and H. Wechsler, Probablistic reasoning models for face recognition, in *Proc. of Computer Vision and Pattern Recognition*, 827-832, 1998.

[5] B. Moghaddam and A. Pentland, Probablistic visual learning for object representation, *IEEE Trans. Pattern Anal. Machine Intell.*, v. 19, 696-710, 1997.

[6] B. Moghaddam, Principal manifolds and bayesian subspaces for visual recognition, *Proc. of IEEE conf. Compter Vison*, 1131-1136, 1999.

[7] M. Pontil and A. Verri, "Support Vector Machines for 3-D Object Recognition", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 20, 637-646, 1998.

[8] A. Samal and P. A. Iyengar, "Automatic recognition and analysis of human faces and facial expressions: A survey", *Pattern Recognition*, vol. 25, 65-77, 1992.

[9] L. Sirovich and M. Kirby, "Low-dimensional procedure for the characterization of human faces", *J. Opt. Soc. Amer.* A, vol. 4, no. 3, 519-524, 1987.

[10] K. Tieu and P. Viola, Boosting image retrieval, in *Proc. of Computer Vision and Pattern Recognition*, v. 1, 228-235, 2000.

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

[12] V. N. Vapnik, *Statistical learning theory*, John Wiley & Sons, New York, 1998.