Fast Learning for Statistical Face Detection

Zhi-Gang Fan and Bao-Liang Lu

Department of Computer Science and Engineering, Shanghai Jiao Tong University,
1954 Hua Shan Road, Shanghai 200030, China
zgfan@sjtu.edu.cn, blu@cs.sjtu.edu.cn

Abstract. In this paper, we propose a novel learning method for face detection using discriminative feature selection. The main deficiency of the boosting algorithm for face detection is its long training time. Through statistical learning theory, our discriminative feature selection method can make the training process for face detection much faster than the boosting algorithm without degrading the generalization performance. Being different from the boosting algorithm which works in an iterative learning way, our method can directly solve the learning problem of face detection. Our method is a novel ensemble learning method for combining multiple weak classifiers. The most discriminative component classifiers are selected for the ensemble. Our experiments show that the proposed discriminative feature selection method is more efficient than the boosting algorithm for face detection.

1 Introduction

Face recognition techniques have been developed over the past few decades. A first step of any face recognition system is detecting the locations in images where faces are present. Face detection has long been an important and active area in vision research. However, face detection from a single image is a challenging task because of variability in scale, location, orientation (up-right, rotated), and pose (frontal, profile). Facial expression, occlusion, and lighting conditions also change the overall appearance of faces. Furthermore, most of the applications of face detection now demand not only accuracy but also real-time response. Viola and Jones proposed an effective coarse-to-fine scheme using boosting algorithm and cascade structure for face detection [17]. Their framework has prompted considerable interest in further investigating the use of boosting algorithm and cascade structure for face detection, e.g., [4], [14], [18], [6], [19], [5], [7].

Sung and Poggio [15] established a face detection approach based on a mixture of Gaussian model. Rowley and Kanade [12] designed a neural network based face detection approach that uses a small set of simple image features. In [9], Osuna *et al.* described an SVM-based method for face detection. Romdhani *et al.* [11] presented another SVM-based face detection system by introducing the concept of reduced set vectors and the sequential evaluation strategy. The SNoW (sparse network of winnows) face detection system by Yang *et al.* [20] is a sparse network of linear functions that utilizes winnows update rules. In

I. King et al. (Eds.): ICONIP 2006, Part II, LNCS 4233, pp. 187–196, 2006.

[10], Papageorgiou and Poggio established a trainable system for face detection using SVMs and overcomplete Haar wavelet transform. Using an energy-based loss function, Osadchy *et al.* [8] designed convolutional networks for real-time simultaneous face detection and pose estimation. Schneiderman and Kanade [13] established an object detection system using boosting algorithm and wavelet transform.

The excellent work of Viola and Jones [17] has redefined what can be achieved by an efficient implementation of a face detection system. They formulated the detection task as a series of non-face rejection problems. Since then, a number of systems have been proposed to extend the idea of detecting faces through the boosting algorithm. For example, Li *et al.* [4] developed a face detection method through FloatBoost learning. The work by Lienhart and Maydt [5] focused on extending the set of Haar-like features. In [7], Liu and Shum introduced a Kullback-Leibler boosting to derive weak learners by maximizing projected KL distances.

The boosting algorithm is a milestone of the research on face detection. However, the main deficiency of the boosting algorithm for face detection is that a very long training time is required. Using statistical learning theory, we propose a discriminative feature selection method, which can make the training process for face detection much faster than the boosting algorithm without degrading the generalization performance. The boosting algorithm is an iterative learning method, and our discriminative feature selection method can directly solve the learning problem of face detection.

2 Related Work

Viola and Jones [17] have made three key contributions to face detection: Haar-like feature, boosting algorithm and cascade structure. All the three contributions are very important. Haar-like feature is good foundation for image representation in face detection. There are many motivations for using Haar-like features rather than the pixels directly. The most common reason is that Haar-like features can act to encode ad-hoc domain knowledge that is difficult to learn using a finite quantity of training data. Unlike the Haar basis, a set of Haar-like features is overcomplete. So the Haar-like feature can more efficiently represent image in detail than the raw pixel data. Another advantage of using Haar-like feature is that the feature can be rapid calculated using so-called ‘integral image’. The integral image is an intermediate representation for the image which is very similar to the summed area table used in computer graphics for texture mapping. The integral image can be computed from an image using a few operations per pixel. Once computed, any one of these Haar-like features can be computed at any scale or location in constant time.

AdaBoost algorithm was used to select a small number of important features from a huge library of potential Haar-like features [17]. Within any image subwindow the total number of Haar-like features is very large, far larger than the number of pixels. In order to ensure fast classification, the learning process must exclude a large majority of the available features, and focus on a small set of critical features. The goal of feature selection is achieved using AdaBoost learning algorithm by constraining each weak classifier to depend on only a single feature. As a result each stage of the boosting process, which selects a new weak classifier, can be viewed as a feature selection process. The weak learning algorithm is designed to select the single Haar-like feature which best separates the positive and negative examples. For each feature, the weak learner determines the optimal threshold classification function, such that the minimum number of examples are misclassified. A weak classifier $h(x, f, p, \theta)$ thus consists of a feature (f), a threshold (θ) and a polarity (p) indicating the direction of the inequality [17]:

$$h(x, f, p, \theta) = \begin{cases} 1 & \text{if } pf(x) < p\theta \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Here x is a fixed size pixel sub-window of an image.

3 Discriminative Feature Selection

The discriminative feature selection approach proposed in this paper consists of two main steps. The first step is to extract Haar-like features and train single feature weak classifiers, and the second step is to search out a small set of critical features (namely critical weak classifiers) and build classifiers for face detection.

3.1 Feature Extraction

Our feature extraction process uses the Haar-like features as used by Viola and Jones [17]. Being similar to [17], the Haar-like features to be extracted have five prototypes. We also use the weak classifier $h(x, f, p, \theta)$ as shown in equation (1) in our feature extraction process. Unfortunately, as showed in Figure.1, the boosting algorithm for face detection requires all weak classifiers be retrained in each iteration step because the training data have been re-weighted. This is a computationally demanding task which is in the inner loop of the boosting algorithm. Therefore, the boosting algorithm for face detection has very long training time.

As showed in Figure.2, we train all weak classifiers once in advance without retraining the weak classifiers in the afterward discriminative feature selection process. In [19], the same strategy was used and a forward feature selection (FFS) method was proposed for face detection. All weak classifiers $h(x, f, p, \theta)$ are trained on single Haar-like feature after Haar-like feature extraction and the thresholds for every single feature are obtained. By thresholding every single Haar-like feature with these weak classifiers, we set each feature to binary value, zero or one. As a result, the data space becomes a binary value space after feature extraction. Feature selection and classifier construction will be finished within this binary value data space.

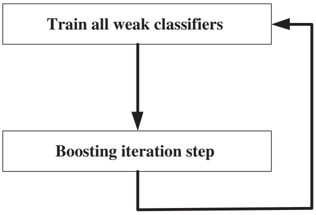

Fig. 1. Weak classifiers training and boosting algorithm

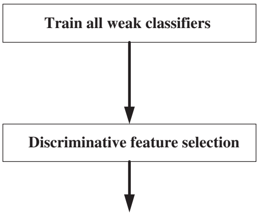

Fig. 2. Weak classifiers training and discriminative feature selection

3.2 Learning and Feature Selection

After feature extraction and thresholding on every single feature by weak classifiers, learning is carried out using statistical learning theory [16] for feature selection and classifier construction in the binary value feature space. So our method is a novel ensemble learning method for combining multiple weak classifiers. Every single feature is a weak classifier in this specific environment. The most discriminative weak classifiers (namely discriminative features) are selected for the ensemble. We use the optimal separating hyperplane in the output space of all the weak classifiers as the combining mechanism for classifier ensemble learning using the statistical learning theory. Statistical learning theory is not only a tool for the theoretical analysis but also a tool for creating practical algorithms for pattern recognition. This abstract theoretical analysis allows us to discover a general model of generalization. On the basis of the VC dimension concept, constructive distribution-independent bounds on the rate of convergence of learning processes can be obtained and the structural risk minimization principle has been found. Optimal separating hyperplane and support vector machines (SVMs) [16] are machine learning techniques which are well-founded in statistical learning theory. As an application of the theoretical breakthrough, SVMs have high generalization ability and are capable of learning in high-dimensional spaces with a small number of training examples. It accomplishes this by minimizing a bound on the empirical error and the complexity of the classifier, at the same time. This controlling of both the training set error and the classifier's complexity has allowed SVMs to be successfully applied to very high dimensional learning tasks.

We are interesting in the optimal separating hyperplane which can also be called linear SVMs because of the nature of the data sets under investigation. Linear SVMs use the optimal hyperplane

$$(w \cdot x) + b = 0 \tag{2}$$

which can separate the training vectors without error and has maximum distance to the closest vectors. In our method, the input vector x is in the output space of all the weak classifiers. We use this optimal separating hyperplane in the output space of all the weak classifiers to combine multiple weak classifiers. To find the optimal hyperplane one has to solve the following quadratic programming problem: minimize the functional

$$\Phi(w) = \frac{1}{2}(w \cdot w) \tag{3}$$

under the inequality constraints

$$y_i[(x_i \cdot w) + b] \geq 1, \quad i = 1, 2, \ldots, l. \tag{4}$$

where $y_i \in \{-1, 1\}$ is class label [16].

According to the hyperplane as shown in equation (2), the linear discriminant function can be constructed for SVMs classifier as follows:

$$f(x) = \text{sign}\{(w \cdot x) + b\} \tag{5}$$

The inner product of weight vector $w = (w_1, w_2, \ldots, w_n)$ and input vector $x = (x_1, x_2, \ldots, x_n)$ determines the value of $f(x)$. Intuitively, the input features in a subset of $(x_1, x_2, \ldots, x_n)$ that are weighted by the largest absolute value subset of $(w_1, w_2, \ldots, w_n)$ influence most the classification decision. If the classifier performs well, the input feature subset with the largest weights should correspond to the most informative features . Therefore, the weights $|w_k|$ of the linear discriminant function can be used as feature ranking coefficients [2], [3], [1]. However, this way for feature ranking is a greedy method and we should look for more evidences for feature selection. In [3] and [1], support vectors have been used as evidence.

Assume the distance between the optimal hyperplane and the support vectors is Δ, the optimal hyperplane can be viewed as a kind of Δ-margin separating hyperplane which is located in the center of margin $(-\Delta, \Delta)$. According to [16], the set of Δ-margin separating hyperplanes has the VC dimension h bounded by the inequality

$$h \leq \min\left(\left[\frac{R^2}{\Delta^2}\right], n\right) + 1 \tag{6}$$

where R is the radius of a sphere which can bound the training vectors $x \in X$ and n is the dimension of the space.

Inequality (6) points out the relationship between margin Δ and VC dimension: a larger Δ means a smaller VC dimension. Therefore, in order to obtain high generalization ability, we should still maintain margin large after feature selection. However, because the dimensionality of original input space has been reduced after feature selection, the margin is usually to shrink and what we can do is trying our best to make the shrink small to some extent. Therefore, in feature selection process, we should preferentially select the features which make more contribution to maintaining the margin large. This is another evidence for feature ranking. To realize this idea, a coefficient is given by

$$c_k = \left| \frac{1}{l_+} \sum_{i \in SV_+} x_{i,k} - \frac{1}{l_-} \sum_{j \in SV_-} x_{j,k} \right| \tag{7}$$

where SV_+ denotes the support vectors belong to positive samples, SV_- denotes the support vectors belong to negative samples, l_+ denotes the number of SV_+, l_- denotes the number of SV_-, and $x_{i,k}$ denotes the kth feature of support vector i in input space R^n.

The larger c_k indicates that the kth feature of feature space can make more contribution to maintaining the margin large. Therefore, c_k can assist $|w_k|$ for feature ranking. The solution is that, combining the two evidences, we can order the features by ranking $c_k|w_k|$ and select the features which have larger value of $c_k|w_k|$. We present below an outline of the discriminative feature selection and classifier training algorithm.

- Input:
  Training examples (using binary Haar-like features)
  $$X_0 = \{x_1, x_2, \dots x_l\}^T$$
- Initialize:
  Indices for selected features: $s = [1, 2, \dots n]$
  Train the SVM classifier using samples X_0
- For $t = 1, \dots, T$:
  1. Compute the ranking criteria $c_k|w_k|$ according to the trained SVMs
  2. Order the features by decreasing $c_k|w_k|$, select the top M_t features, and eliminate the other features
  3. Update s by eliminating the indices which not belong to the selected features
  4. Restrict training examples to selected feature indices
     $$X = X_0(:, s)$$
  5. Train the SVM classifier using samples X
- Outputs:
  The small set of critical features and the final SVM classifier

Usually, the iterative loop in the algorithm can be terminated before the training samples can not be separated by a hyperplane. Clearly, this algorithm can integrate the two tasks, feature selection and classifier training, into a single consistent framework and make the feature selection process more effective. Using this discriminative feature selection method, we can search out the small set of critical features and build classifiers for face detection.

4 Experiments

We have made several sets of experiments to illustrate the effectiveness of the proposed discriminative feature selection algorithm for face detection. In all experiments reported here, we use the MIT-CBCL face database [3] , a database of faces and non-faces that have been used extensively at the Center for Biological and Computational Learning at MIT. All input gray-scale images are of size

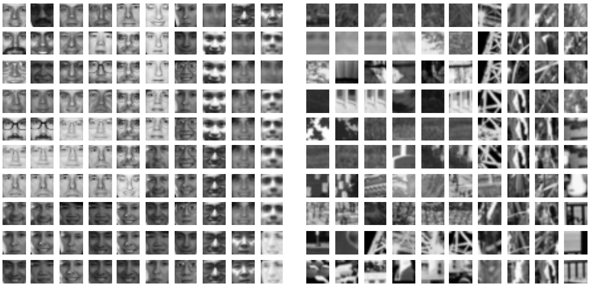

Fig. 3. Some face and non-face sample images in the MIT-CBCL database

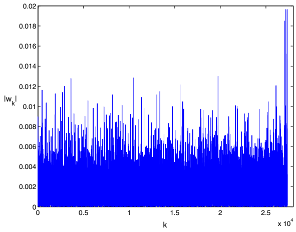

Fig. 4. The diversity of $|w_k|$

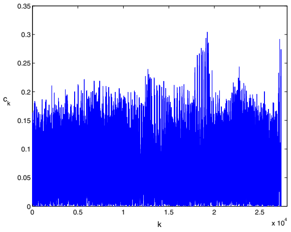

Fig. 5. The diversity of c_k

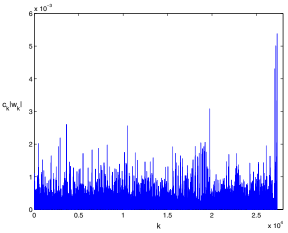

Fig. 6. The diversity of $c_k|w_k|$

19×19 and the dimensionality of the resulting input vectors is $N = 361$. Figure 3 depicts some face and non-face sample images in the MIT-CBCL database. The overall database is partitioned into two subsets: the training set and test set. The training set is composed of 2429 face images and 4548 non-face images. The test set is composed of 472 face images and 23573 non-face images. All the image data have been histogram equalized . All of the experiments were performed on a 3.0GHz Pentium 4 PC with 2.0GB RAM.

After Haar-like feature extraction, the dimensionality of the feature vectors without feature selection is $N = 27348$. In the binary value feature space of the dimensionality $N = 27348$, we train linear SVMs and obtain the coefficients c_k and $|w_k|$. The diversities of c_k, $|w_k|$ and $c_k|w_k|$ have been showed in Figures.4 through 6, respectively. Figures 7 through 9 show, respectively, c_k, $|w_k|$ and $c_k|w_k|$ being ordered increasingly. From these figures, we can see that $c_k|w_k|$ has the steepest variability curve which is useful for feature selection. To evaluate the different impacts of the three coefficients on feature selection, we use the three coefficients respectively to select features. We use four iterative steps (T= 4) and the parameter M_t is set as: $M_1 = 5000, M_2 = 1000, M_3 = 500, M_4 = 200$. After feature selection, the classification accuracy is examined on the test data set. The test results are showed in Table 1, where 'FS-W', 'FS-C', and 'FS-CW' denote the feature selection using coefficient $|w_k|$, the feature selection using coefficient c_k, and the feature selection using coefficient $c_k|w_k|$, respectively. The

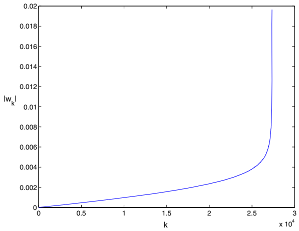

Fig. 7. $|w_k|$ ordered increasingly

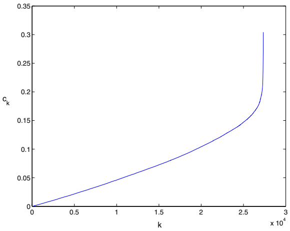

Fig. 8. c_k ordered increasingly

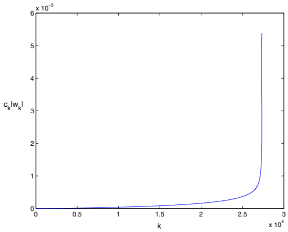

Fig. 9. $c_k|w_k|$ ordered increasingly

Table 1. Test results using three coefficients respectively

Methods	No. features	True positive rate (%)	True negative rate (%)
FS-W	5000	42.3729	98.8080
	1000	41.3136	98.7231
	500	41.1017	98.7274
	200	41.1017	98.4643
FS-C	5000	42.1610	98.5110
	1000	41.1017	94.8119
	500	40.4661	93.3271
	200	39.6186	95.3167
FS-CW	5000	42.5847	98.9098
	1000	41.9492	98.7443
	500	41.5254	98.8589
	200	41.5254	98.5025

linear SVMs are used as classifier in the three cases. Through Table 1 we can see that the FS-CW approach is the best one among the three methods.

Figure 10 shows the ROC (receiver operating characteristic) curves for the face detection test. In this set of experiments, we have used four different methods for comparison study. In Figure 10, ‘FFS’, ‘Viola-Boosting’, and ‘Pixel Method’ denote the forward feature selection method [19], the AdaBoost algorithm [17], and the linear SVMs using raw pixel data, respectively. The experimental setting of our method is the same as mentioned above. We used linear SVMs as the weight setting algorithm of the FFS method. In the pixel method, we used the raw image pixel data as input features and didn’t use the Haar-like features. But the Haar-like features have been used for the FFS and Viola-Boosting. The dimensionality of the raw pixel feature vectors is $N = 361$ and the parameter C of the linear SVMs was set to 0.001 for the pixel method. For the other three methods, our discriminative feature selection method, FFS and the Viola-Boosting, the dimensionality of the feature vectors is $N = 200$ after feature

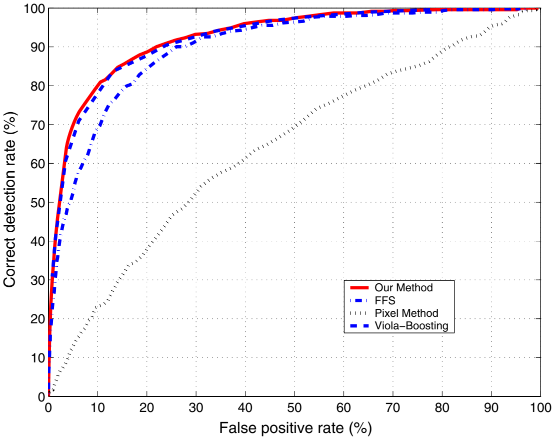

Fig. 10. ROC (receiver operating characteristic) curves for the face detection test

selection. Through Figure 10, we can see that the accuracy of our method is the highest among the four methods. And our method has much shorter training time than the Viola-Boosting algorithm. In our experiments, the training time of our discriminative feature selection method is 15 minutes and the training time of Viola-Boosting is 7 hours. The accuracy of the pixel method is very low because it doesn't use Haar-like features.

5 Conclusions

We have presented a discriminative feature selection method for face detection. This discriminative feature selection method can make the training process for face detection much faster than the boosting algorithm without degrading the generalization performance. The boosting algorithm works in an iterative way, while our discriminative feature selection method can directly solve the learning problem of face detection. Our method is a novel ensemble learning method for combining multiple weak classifiers. We use the optimal separating hyperplane in the output space of all the weak classifiers as the combining mechanism for classifier ensemble learning. The most discriminative component classifiers are selected for the ensemble. Through the experimental results, we can see that our method is more efficient than the boosting algorithm for face detection. We also can see that the Haar-like features are more powerful than the raw pixel features. We can learn more detail of the nature of the learning methods for face detection in this study.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China via the grants NSFC 60375022 and NSFC 60473040.

References

1. Z. G. Fan and B. L. Lu. Fast recognition of multi-view faces with feature selection. *Proc. ICCV 2005*, 1:76–81, 2005.
2. I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46(3):389–422, 2002.
3. B. Heisele, T. Serre, S. Prentice, and T. Poggio. Hierarchical classification and feature reduction for fast face detection with support vector machine. *Pattern Recognition*, 36(9):2007–2017, 2003.
4. S. Z. Li and Z. Zhang. Floatboost learning and statistical face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1112–1123, 2004.
5. R. Lienhart and J. Maydt. An extended set of haar-like features for papid object detection. *Proc. ICIP 2002*, 1:900–903, 2002.
6. Y. Lin and T. Liu. Robust face detection with multi-class boosting. *Proc. CVPR 2005*, 1:680–687, 2005.
7. C. Liu and H. Shum. Kullback-leibler boosting. *Proc. CVPR 2003*, 1:587–594, 2003.
8. R. Osadchy, M. Miller, and Y. LeCun. Synergistic face detection and pose estimation with energy-based model. In *Advances in Neural Information Processing Systems (NIPS 2004)*. MIT Press, 2005.
9. E. Osuna, R. Freund, and F. Girosi. Training support vector machines: An application to face detection. *Proc. CVPR 1997*, 1:130–136, 1997.
10. C. Papageorgiou and T. Poggio. A trainable system for object detection. *International Journal of Computer Vision*, 38(1):15–33, 2000.
11. S. Romdhani, P. Torr, B. Scholkopf, and A. Blake. Computationally efficient face detection. *Proc. ICCV 2001*, 2:695–700, 2001.
12. H. Rowley and T. Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):23–38, 1998.
13. H. Schneiderman and T. Kanade. Object detection using the statistics of patrs. *International Journal of Computer Vision*, 56(3):151–177, 2004.
14. J. Sun, J. M. Rehg, and A. Bobick. Automatic cascade training with perturbation bias. *Proc. CVPR 2004*, 2:276–283, 2004.
15. K. K. Sung and T. Poggio. Example-based learning for view-based human face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):39–51, 1998.
16. V. Vapnik. *The Nature of Statistical Learning Theory.* Springer-Verlag New York, 2000.
17. P. Viola and M. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154, 2004.
18. B. Wu, H. Ai, C. Huang, and S. Lao. Fast rotation invariant multi-view face detection based on real adaboost. *Proc. FGR 2004*, 1:79–84, 2004.
19. J. Wu, J. M. Rehg, and M. D. Mullin. Learning a rare event detection cascade by direct feature selection. In *Advances in Neural Information Processing Systems 16.* MIT Press, Cambridge, MA, 2004.
20. M. H. Yang, D. Roth, and N. Ahuja. A snow-based face detector. In *Advances in Neural Information Processing Systems 12.*