Hierarchical PSO-Adaboost Based Classifiers for Fast and Robust Face Detection

Hong Pan, Yaping Zhu, and Liang Zheng Xia

Abstract—We propose a fast and robust hierarchical face detection system which finds and localizes face images with a cascade of classifiers. Three modules contribute to the efficiency of our detector. First, heterogeneous feature descriptors are exploited to enrich feature types and feature numbers for face representation. Second, a PSO-Adaboost algorithm is proposed to efficiently select discriminative features from a large pool of available features and reinforce them into the final ensemble classifier. Compared with the standard exhaustive Adaboost for feature selection, the new PSO-Adaboost algorithm reduces the training time up to 20 times. Finally, a three-stage hierarchical classifier framework is developed for rapid background removal. In particular, candidate face regions are detected more quickly by using a large size window in the first stage. Nonlinear SVM classifiers are used instead of decision stump functions in the last stage to remove those remaining complex non-face patterns that can not be rejected in the previous two stages. Experimental results show our detector achieves superior performance on the CMU+MIT frontal face dataset.

Keywords—Adaboost, Face detection, Feature selection, PSO

I. INTRODUCTION

HUMAN faces are difficult to model as it is necessary to account for all possible appearance variations caused by changes in scale, location, orientation (in-plane rotation), pose (out-of-plane rotation), facial expression, lighting conditions and partial occlusions, etc. In spite of all these difficulties, tremendous progress has been made in the latest several decades and many systems have shown impressive performance [1]-[12]. In particular, the remarkable breakthrough was achieved by Viola and Jones. In [2], they developed the first real-time face detection system by building a cascade of simple classifiers, each of which is based only on the Haar-like features over fixed image regions. The classifiers themselves are simple threshold functions, but their ensemble allows to learn complex appearance variations. The amazing real-time speed and high detection accuracy of Viola and Jones' face detector can be attributed to three factors: the integral image representation, the cascade framework, and the use of Adaboost to train cascade nodes. The integral image representation enables the calculation of Haar-like features extremely fast. The cascade framework allows non-face background patches to be filtered away quickly. Adaboost

Hong Pan is with School of Automation, Southeast University, Nanjing, China, 210096 (phone: 86-25-83794625; fax: 86-25-83794625; e-mail: enhpan@seu.edu.cn).

Yaping Zhu is with Department of Communication Engineering, School of Information Engineering, Communication University of China, Beijing, China, 100024 (e-mail: zhuyaping@cuc.edu.cn).

Liangzheng Xia is with School of Automation, Southeast University, Nanjing, China, 210096 (e-mail: lzxia@seu.edu.cn).

algorithm selects Haar-like features and combines them into an ensemble classifier in a cascade node. Currently, many face detection systems follow Viola and Jones' cascade-based framework which computes a great number of weak classifiers formed by Haar-like features at all possible positions and scales in a sliding window and then boosts these weak classifiers into a strong classifier to predict whether or not a face is present in the window. However, training such a boosting ensemble classifier is rather lengthy and computationally expensive. We argue this training inefficiency of the boosting ensemble classifier comes from two aspects: (1). poor discriminability and classification performance due to the weak classifier formed by Haar-like features and simple decision stump function; (2). the exhaustive search method used in Adaboost algorithm. Haar-like features are too simple to characterize face patterns in real-world situations. Hence the ensemble classifier derived from Haar-like features is insufficient to achieve accurate face detection. Such 'too weak' classifiers do not contribute to improving detector's generalization performance, so that more feature types as well as unreasonably large number of weak classifiers have to be exploited to boost the performance of the ensemble classifier. However, expansion of feature numbers and types automatically increases the size of feature set and storage memory. Since feature space enlarges dramatically, obviously, exhaustive search mechanism used in the standard Adaboost algorithm can not effectively manage the search process. This in turn makes the training time longer, which is by far one of the main reasons that stop many methods from exploring other feature types.

To cope with the poor discriminability of Haar-like features and the lengthy training process due to the exhaustive Adaboost method, in this paper, we propose an efficient hierarchical face detection system which finds and localizes face images with a cascade of classifiers formed with a group of heterogeneous feature descriptors. To further speed up the training and detection process of our system, we also propose to integrate Particle Swarm Optimization (PSO) algorithm into the Adaboost framework, replacing the exhaustive search used in original Adaboost for efficient feature selection. As shown by the result in Section V, this PSO-Adaboost method not only speeds up the training process but also improves generalization ability of the face detector.

II. THE PROPOSED HIERARCHICAL FACE DETECTOR

Considering the task-specific characteristics for face detection, it is sensible to apply a hierarchical face classifier, in which simple and fast classifiers remove large parts of the
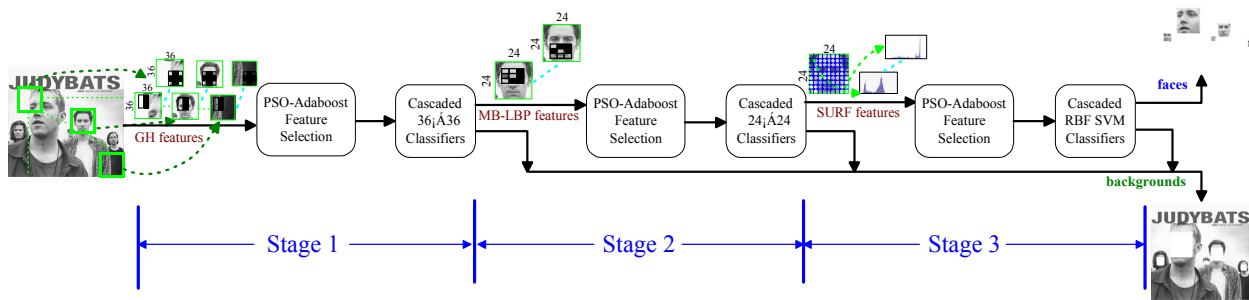
Fig. 1 Framework of the proposed hierarchical Hetero-PSO-Adaboost-SVM face detector

background in the beginning and intermediate stages of the hierarchy and more accurate but slower classifiers perform the final detection in the last level. Based on this observation, we propose a fast hierarchical face detector (i.e. the Hetero-PSO-Adaboost-SVM face detector) that boosts a set of weak classifiers formed with heterogeneous feature descriptors including Generalized Haar-like (GH) descriptor, Multi-Block Local Binary Patterns (MB-LBP) descriptor and Speeded Up Robust Feature (SURF) descriptor. In such a hierarchical architecture where the complexity of classifiers increases as the stage goes further, only a small percentage of non-face patterns that look similar to face patterns require an accurate ensemble classifier to avoid false classification. By propagating those patterns that were not classified as background, we can quickly decrease the amount of data to process. To further lessen the training burden of the final face classifier, a feature selection scheme using PSO algorithm searches the entire feature space and filters out a minimum number of discriminative features that give the highest classification rate, and then Adaboost algorithm is carried out to boost these carefully selected weak classifiers (features) into a strong classifier in each hierarchical layer.

Fig.1 depicts framework of the proposed Hetero-Adaboost-SVM face detector. More specially, the whole system consists of three stages that classify and localize all face regions in an input image using a 24×24 pixel window. To detect faces of different sizes and positions, we apply the detector at every position and scale in the input image with a scale factor of 1.25. The first stage is a cascade of GH feature-based classifiers which are used to estimate face candidate positions approximately in a 36×36 pixel window with a moving step of 12 pixels. If a face pattern is found inside a 36×36 pixel window, a total of 144 (i.e. 12×12) possible face positions are collected and passed to the next stage. The second stage is a cascade of 24×24 pixel window-based classifiers constructed from MB-LBP features to explore the accurate face candidate locations retrieved from the previous stage. The main purpose of applying these two hierarchical cascade classifiers is trying to filter out as many as simple non-face (background) patterns quickly before passing the remaining complex patterns to the final stage. Although the cascade of 24×24 classifiers rejects non-face patterns rapidly, it still causes a great deal of computation due to the large

number of 24×24 patterns that it needs to process. For this reason, in the first stage, a cascade of 36×36 classifiers is added to decrease the number of analyzed patterns. To this end, this stage is trained specially to make the classifiers invariant to small face translations. These classifiers can detect faces that are off-center by up to six pixels in vertical and/or horizontal directions. The 36×36 pixel window is chosen in accordance with the observation in [4] that the classifier can be trained to be invariant to translation by up to 25% of the original window size. With this flexible classifier, the moving step size can be increased by up to 12 pixels to dramatically reduce the number of analyzed patterns. This simple to complex matching idea is inspired from Viola and Jones' approach [2], in which Haar wavelet features form the weak classifiers and the cascade Adaboost classifiers enable extremely fast computation. Whereas, our system builds the ensemble stage classifier with a group of specially selected heterogeneous feature descriptors (i.e. GH descriptor and MB-LBP descriptor) based on a PSO-Adaboost feature selection method, which guarantees a fast derivation of robust and effective stage classifiers. In the last stage, a cascade of complex and powerful nonlinear SVM classifiers is utilized to carefully remove those remaining complex non-face patterns which look so similar to face patterns that can not be rejected during the previous two stages. In particular, radial basis function (RBF) is used as the kernel function in these SVM classifiers. For each detected 24×24 region yielded from the second stage classifiers, discriminative SURF descriptors at fixed points in the locations of the nodes on an 8×8 regular grid, overlapping the region, are evaluated for these nonlinear RBF kernel-based SVM classifiers in the last stage.

III. FACE REPRESENTATION WITH HETEROGENEOUS FEATURES

Three types of heterogeneous feature descriptors with increasing discriminability, i.e. GH descriptor, MB-LBP descriptor and SURF descriptor, are used in the proposed detector to represent face patterns. In particular, GH features are suitable for catching edge structures, whereas MB-LBP features are texture-based descriptors and SURF features are interest point-based high dimensional descriptors on gradient orientations, so these complementary feature descriptors, with varying distinctiveness and complexity, characterize a face
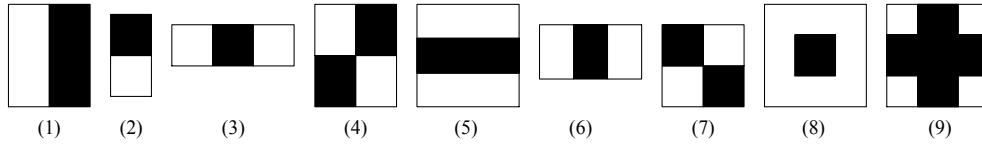
Fig. 2 Generalized Haar-like features used in our face detector. (1)-(4) Original Haar-like features used by Viola and Jones, (5)-(9) Extended Haar-like features proposed by Pham and Cham [11]

pattern from different aspects. Furthermore, it is worth noting that all these three types of descriptors can be efficiently evaluated via integral images.

In [2], Viola and Jones proposed a basic set of four types of Haar-like features for detecting frontal faces. In our approach, we adopt a total of 9 generalized Haar-like features, including a group of extended Haar-like features proposed in [11] and four basic Haar-like features, to increase the detector's performance. Fig.2 compares the four basic Haar-like features applied by Viola and Jones as well as the generalized Haar-like features used in our approach.

The basic idea of MB-LBP operator [12] is originated from the Local Binary Patterns (LBP) descriptor [13]. Formally, the MB-LBP operator is defined by comparing the central rectangle's average intensity with those of its neighboring rectangles in a 3×3 neighborhood. Fig.3 demonstrates the evaluation of MB-LBP operator, which results an output of MB-LBP value (00011110). Interested readers may refer to [12] for more details on MB-LBP. MB-LBP is more discriminative since it can capture diverse image structures in a large scale, which may be the dominant features of an image. Another advantage of MB-LBP is that the number of comprehensive set of MB-LBP features is much smaller than Haar-like features, which makes the implementation of feature selection significantly easy. Similar to the LBP descriptor, by using circular neighborhoods and bilinearly interpolating the pixel values, MB-LBP operator can also be extended to MB-LBP$_{P,R}$ with local neighborhood of P equally-spaced sampling points on a circle of radius of R to capture significant features with at different sizes and scales.
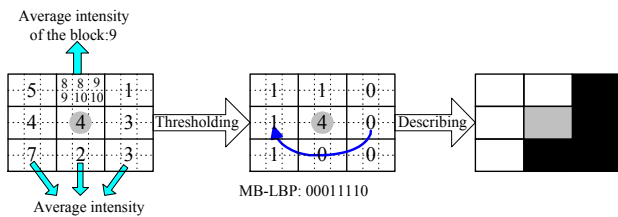

Fig. 3 Evaluation of the Multi-block LBP (MB-LBP) feature

SURF [14-15] is a robust scale-invariant and rotation-invariant interest point detector and descriptor. Conceptually similar to the SIFT descriptor, SURF descriptor also focuses on the spatial distribution of gradient information within the interest point neighborhood, where interest points itself can be localized by interest point detection approaches or in a regular grid. In this paper, both the standard 64-dimensional version and the extended 128-dimensional version of SURF descriptor with the 4×4 sub-region division solution are used to provide the best performance for face detection.

IV. FEATURE SELECTION AND LEARNING USING PSO-ADABOOST

Recall that in most face detection methods a huge quantity of Haar-like features are built as weak classifiers, among which a set of the most discriminative weak classifiers are selected via Adaboost algorithm to derive a final strong classifier that can accurately distinguish face patterns from those complicated non-face patterns. However, the exhaustive search mechanism used in Adaboost algorithm significantly increases the selection complexity and prolongs the selection process. In addition, the weak learning ability of the simple decision stump classifier also worsens the efficiency of these face detection methods. Taking the above two factors into account, two improvements are proposed in our face detector to lessen the computational burden of feature selection and speed up the feature selection. First, powerful nonlinear SVM classifiers trained on discriminative SURF descriptors take the role of decision stump functions to strengthen the feature learning ability as well as the classification capability for weak classifiers at the last stage of the hierarchical face detector. Second, PSO technique is integrated with Adaboost algorithm, so that it can efficiently search the whole feature space and select a minimum number of optimal feature sets to shorten the training processing.

A. Adaboost for Feature Selection

Adaboost [16] is an ensemble learning method that trains multiple weak classifiers and strengthens them into a more powerful strong classifier. In the iterations of Adaboost, the weak classifier that minimizes classification error on a set of training examples is chosen and added into the final ensemble classifier. Therefore, the learned ensemble classifier effectively obtains a perfect classification power for given training samples. More specially, given a set of training examples $\{(x_n, y_n), y_n= \{-1,1\}\}(n=1,\cdots, N)$, weighted by ω_n uniformly, Adaboost algorithm iterates over a number of T rounds. In every round t, and for each feature f_j, a weak classifier $h_j(x)$ that consists of the feature, a decision threshold $\theta_j = 0.5(M_{-j} + M_{+j})$ where M_{-j} and M_{+j} are the mean value of feature responses on negative and positive examples respectively, and a parity parameter p_j indicating the direction of the inequality sign, is trained to assist the classification of positive patterns from negative patterns.

$$h_j(x) = \begin{cases} 1 & p_j f_j(x) < p_j \theta_j \\ 0 & otherwise \end{cases} \quad (1)$$

Exhaustive search is used to select the classifier $h_t(x)$ that generates the minimum classification error, among available classifiers. Classification error $\varepsilon_t = \sum_{n=1}^{N} \omega_n |h_t(x_n) - y_n|$ is

defined as the total weights of the misclassified examples. At the end of each round, weights of the training examples misclassified by $h_t()$ are increased, so that the learning algorithm will focus more on these misclassified examples in the next round. The classifier $h_j()$ with the minimum classification error at round t is taken to be the winner among all available classifiers. The final strong classifier $H(x)$ at the end of T rounds is constructed as a linear combination of the weighted weak classifiers, where each classifier is weighted by a weight parameter proportional to its error rate.

B. Particle Swarm Optimization (PSO) Algorithm

PSO [17] is a kind of population-based stochastic optimization technique that simulates the social behavior of bird flocking or fish schooling to describe an evolving system. In PSO, each candidate solution to the problem at hand is called a particle, and the population consists of all potential solutions which create flocks of birds. Each particle has a fitness value that indicates the goodness of the solution represented by the particle. The swarm is initialized so that these particles are randomly distributed over the search space and then fly freely across the multi-dimensional search space. During the flying, each particle updates its velocity and position states based on two best values, i.e. the personal best position it experienced, and the global best position the entire population experienced. The updating policy guided by the fitness function drives the particle swarm towards the region with higher objective value iteratively, and finally all particles gather around the point with the highest objective value that produces the optimum solution. In particular, if the number of particles in the population is I, for the ith particle, let $X_i^l = \left[x_{i1}^l, x_{i2}^l, ..., x_{iD}^l \right]^T$ and $V_i^l = \left[v_{i1}^l, v_{i2}^l, ..., v_{iD}^l \right]^T$ denote its position and velocity vector in a D-dimensional solution space at the lth iteration, respectively. Let $P^s = \left[p_1^s, p_2^s, ..., p_D^s \right]^T$ and $P^g = \left[p_1^g, p_2^g, ..., p_D^g \right]^T$ denote the personal best point and the global best point respectively. In PSO, during each iteration, with the knowledge of P^s and P^g, each particle updates its velocity and position according to the individual (local) and social (global) information, as shown in (2) and (3).

$$V_i^{l+1} = wV_i^l + c_s r_s (P_i^s - X_i^l) + c_g r_g (P_i^g - X_i^l) \quad (2)$$

$$X_i^{l+1} = X_i^l + V_i^{l+1} \quad (3)$$

where $i = 1, 2, ...I$; w is the inertia weight that controls the impact of the past velocity over the current velocity; r_s and r_g are random variables within the range of [0,1]; c_s and c_g are positive weighting factors that adjust the influence of personal and global best solutions, respectively. Equation (2) and (3) describe the flying trajectory of a population of particles. As shown in (2), since particles fly through the search space with velocities dynamically adjusted according to their and the global historical behaviors, thus particles have a tendency to fly towards better and better search area. Finally, the global best particle (with its position state P^g) found so far is taken as the optimal solution to the problem.

C. PSO-Adaboost for Feature Selection

To tackle the high training complexity due to exhaustive search in the original Adaboost algorithm, we apply the PSO algorithm for all possible feature location, size and type combinations to find a set of discriminative features. Then these features are combined with Adaboost algorithm to construct an effective ensemble classifier. We call this feature selection algorithm as PSO-Adaboost based feature selection.

Algorithm *Pseudocode of PSO-Adaboost algorithm for feature selection*

Give N labeled examples $\{x_n, y_n\}_{n=1}^N$ ($x_n \in R^D$: example, $y_n \in \{-1,1\}$: class label)

Initialize examples weights: if $y_n=1$, then $\omega_n = \frac{1}{2p}$; otherwise $\omega_n = \frac{1}{2q}$ (p: # of positive examples, q: # of negative examples)

for $t = 1, ..., T$

    (1). Normalize examples weights: $\omega_{t,n} = \frac{\omega_{t,n}}{\sum_{n=1}^N \omega_{t,n}}$, and for each feature j, train a weak classifier $h_j()$

    (2). Optimize weak-classifiers $\{h_j()\}_{j=1}^J$ using the PSO algorithm: $\{h_t(), \varepsilon_t\} = \boldsymbol{PSO}\left(\{h_j()\}_{j=1}^J, \{x_n, y_n, \omega_n\}_{n=1}^N \right)$

      ▪ Evaluate the fitness function of all classifiers
      ▪ Choose the classifier $h_t()$ with the lowest fitness value ε_t

    (3). Evaluate the classifier weight $\alpha_t = \log \frac{1-\varepsilon_t}{\varepsilon_t}$

    (4). Update examples weights: $\omega_{t+1,n} = \omega_{t,n} \beta_t^{1-b_n}$, where $b_n = \begin{cases} 0, & h_t(x_n) = y_n \\ 1, & otherwise \end{cases}$, and $\beta_t = \varepsilon_t / (1-\varepsilon_t)$

end for

Output strong classifier: $H(x) = \begin{cases} 1 & \sum_{t=1}^T \alpha_t h_t() \geq 0.5 \sum_{t=1}^T \alpha_t \\ 0 & otherwise \end{cases}$

Optimization function $\boldsymbol{PSO()}$ for PSO-Adaboost algorithm

Input arguments $\left\{ \{h_j()\}_{j=1}^J, \{x_n, y_n, \omega_n\}_{n=1}^N \right\}$

Set $c_s = c_g = 2$, $w_{min} = 0.2$, $w = w_{max} = 1.2$
Set random parameters: $r_s, r_g \in [0,1]$
Set state vector: $X_i^l \in R^D$ and $V_i^l \in R^D$ with random value

for $l = 1, \cdots, L$

    for $i = 1, \cdots, I$

      (1). Fit a classifier $h(X_i^l; x)$ to the training example using weights ω_n

      (2). Evaluate $\varepsilon_i^l = \frac{\sum_{n=1}^N \omega_n \, \delta\left[y_n - h(X_i^l; x_n) \right]}{\sum_{n=1}^N \omega_n}$

      (3). Update state of particles using (2) and (3)

    end for

    Update momentum: $w \leftarrow w_{max} - \frac{l}{L}(w_{max} - w_{min})$

end for

Return $\{h_g(),\varepsilon_{h_g 0}\}$

Fig. 4 Pseudocode of the proposed PSO-Adaboost algorithm for feature selection

According to this paradigm, the weak classifier now consists of a feature parameter set (*type*, x_s, y_s, *width*, *height*, *sampling_points*, *radius*), where *type* denotes the feature descriptor type which takes value in 1~12 (1~9 for GH, 10 for MB-LBP, 11 and 12 for SURF-64 and SURF-128 features, respectively); (x_s, y_s) denotes the feature position in the detection window; (*width, height*) denotes the size of the GH features in 36×36 cascade classifiers in the first stage; (*sampling_points*, *radius*) denote the number of neighboring sample points (*P*) and the radius of sample circle (*R*) for MB-$LBP_{P,R}$ features in 24×24 cascade classifiers in the second stage. All these seven parameters are integers, and their values are constrained by the number of feature types or the dimension of the detection window. Hence, the whole problem is turned to a constrained optimization problem.

In our PSO-Adaboost based feature selection method, considering the position vector of a particle as $X = [type, x_s, y_s, width, height, sampling\_points, radius]^T$, the best feature descriptors are searched over the entire feature space using the PSO algorithm, according to the fitness function defined in (4). The fitness function is to minimize the weighted classification error rate ε, which takes the same formula as the original Adaboost algorithm.

$$fitness = \varepsilon = \sum_{n=1}^{N} \omega_{t,n} b_n \ , \ where \ b_n = \begin{cases} 0 & h_t(X_n) = y_n \\ 1 & otherwise \end{cases} \quad (4)$$

Fig.4 shows the pseudocode of PSO-Adaboost algorithm for feature selection. Following the above procedure, the Haar features, MB-LBP features and SURF features with high accuracy are selected efficiently in our face detector.

V. EXPERIMENTAL RESULTS

For training the face detector, a set of 10000 frontal face images from various sources were collected. These face images cover ±15° in-plane rotation and ±45° out-of-plane rotation from the exact upright face. All face images were scaled and aligned to a base resolution of 24×24 pixels, then histogram equalization was performed to decrease the variation caused by illumination changes. Another set of 20000 images containing no faces were collected from various subjects to form the non-face dataset. For each aligned face example, four synthesized face examples were generated considering the following random transformations: mirroring, random shifting to ±1 pixel, in-plane rotation between ±180° and scaling within 20% variations. Finally, we got a total number of 40000 face training examples. Face patterns for training the 36×36 classifiers were generated by selecting 36×36 windows that contain the 24×24 face window of the input image. To train the cascade of 36×36 and 24×24 Adaboost classifiers used in the first two stages, all face images were used for all layers and all non-face examples were used to train the first layer classifier in the cascade structure.

For the subsequent layer classifiers, non-face training set was updated in such a way that the non-face examples which were correctly classified by the current stage classifier were removed from the training set, whereas false positive examples produced by the stage classifier were supplemented into the training set. To keep a balance between complexity and classifier's accuracy, the maximum number of layers for 36×36 classifiers in the first stage was set to three, since using more layers would degrade the overall detection rate dramatically. For the SVM classifiers used in the last stage, 10000 face patterns separated from the training set together with 10000 non-face patterns were used to learn the best classification boundary for face and non-face patterns. SVM classifiers were trained using a RBF kernel function $K(x,y) = \exp(-\gamma\|x-y\|^2)$ with the width parameter $\gamma = 0.0625$ and the penalty parameter $C = 8$. These SVM parameters were decided in a 10-fold cross-validation manner.

Using the above experimental setting and applying the PSO-Adaboost based feature selection scheme, we derive the final structure of the proposed Hetero-PSO-Adaboost-SVM face detector. In particular, in the first stage, the cascaded 36×36 classifiers contains three layers among which a total of 110 GH features are selected via the PSO-Adaboost method. The second stage includes 6 layers with a total of 440 selected MB-LBP features from the 36×36 face candidate windows yielded from the previous stage. The final stage is a cascade of three-layer RBF SVM classifiers that captures 64 SURF features from the remaining 24×24 potential face regions. As a comparison, we also trained a full cascade of Haar-Exhaustive-Adaboost-DS classifier that was built upon homogeneous Haar features and decision stump functions using exhaustive Adaboost algorithm. Similar to Viola and Jones' approach [2], this Haar-Exhaustive-Adaboost-DS classifier came with a cascade structure of 30 layers with 6450 features. Apparently, our system uses much fewer distinctive features (only 614 features in total), so it is more efficient and can reduce both training and detection time.

We tested detection performance of our Hetero-PSO-Adaboost-SVM face detector on the CMU+MIT frontal face dataset [4]. This dataset consists of 130 images containing 507 frontal faces with various conditions. Fig.5 plots the Receiver Operating Characteristics (ROC) curves of our method as well as other popular state-of-the-art face detection algorithms including Féraud et al. [5], Garcia and Delakis [6], Huang et al. [7], Li and Zhang [8], Yan et al. [9], Schneiderman [10] and Viola and Jones [3], in terms of the number of false positives with respect to the detection rate. As shown in Fig.5, our detector achieves a detection rate of 96.50% at ten false positives. As far as we know, this is the best detection result for ten false positives on the CMU+MIT frontal face dataset. Obviously, compared with other algorithms, detection rate of our method is also improved dramatically, especially for cases at low false alarms. Fig.6 gives some detection results from our face detector.

Regarding the training time, thanks to the PSO-Adaboost algorithm, we significantly improve the efficiency for feature selection. Training a final hierarchical Hetero-PSO-Adaboost-SVM detector requires approximately 49900 seconds on a

platform with PIV 3.0 GHz CPU and 2GB RAM, which is about 20 times faster than training using original exhaustive Adaboost algorithm. To detect faces in a 320×240 pixel image, our detector spends an average time of 70ms.
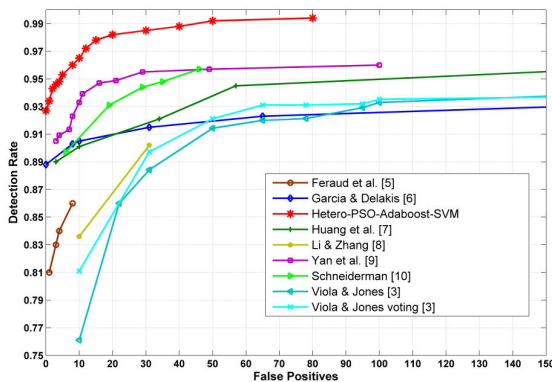

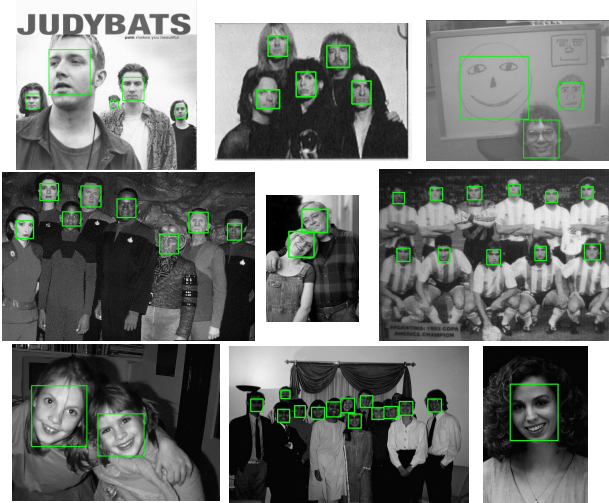Fig. 5 ROC curves for all detection algorithms on the CMU+MIT frontal face dataset

Fig. 6 Detection examples using our detector on test images from the CMU+MIT frontal face dataset

VI. CONCLUSION

We propose an efficient and robust face detection system based on heterogeneous feature representation and feature selection using PSO-Adaboost. To quickly remove large background parts in an image, we apply a simple to complex matching strategy and implement it with a three-stage hierarchical cascade classifiers system, built on heterogeneous feature descriptors with increasing discriminability and complexity. In particular, all possible location of face patterns are first approximately detected using a large size sliding window. A cascade of 24×24 pixel classifiers and nonlinear RBF SVM classifiers are then used to further remove non-face patterns and precisely localize face regions in the later two stages. In our system, three types of heterogeneous descriptors, i.e. generalize Haar-like descriptor, MB-LBP descriptor and SURF descriptor form the feature pool, from which an efficient PSO-based Adaboost algorithm is employed to select a minimum set of distinctive features for cascade classifiers at each stage. Combining the merits of face representation with heterogeneous descriptors, PSO-Adaboost based feature selection and nonlinear SVM weak classifier makes the proposed face detector achieve superior detection accuracy while maintaining a low training and detection complexity. Experimental results verified the robustness and efficiency of our system.

REFERENCES

[1] M. H. Yang, D. J. Kreigman, and N. Ahuja, "Detecting faces in images: A survey," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.24, no.1, pp.34-58, Jan. 2002.

[2] P. Viola, and M. Jones, "Rapid object detection using a boosted cascade of simple features," in Proc. of CVPR 2001, vol.1, 2001, pp.I-511- I-518.

[3] P. Viola, and M. Jones, "Robust real-time face detection," International Journal of Computer Vision, vol.57, no.2, pp.137-154, Jan. 2004.

[4] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network based face detection," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.20, no.1, pp.23-38, Jan. 1998.

[5] R. Féraud, O. Bernier, J. Viallet, and M. Collobert, "A fast and accurate face detector based on neural networks," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.23, no.1, pp.42-53, Jan. 2001.

[6] C. Garcia, and M. Delakis, "Convolutional face finder: A neural architecture for fast and robust face detection," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.26, no.11, pp.1408-1423, 2004.

[7] C. Huang, H. Z. Ai, B. Wu, and S. H. Lao, "Boosting nested cascade detector for multi-view face detection," in Proc. of ICPR 2004, vol.2, 2004, pp.415-418.

[8] S. Z. Li, and Z. Zhang, "Floatboost learning and statistical face detection," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.26, no.9, pp.1112–1123, Sep. 2004.

[9] S.Y.Yan, S.G.Shan, X.L.Chen, and W. Gao, "Locally Assembled Binary (LAB) feature with feature-centric cascade for fast and accurate face detection," in Proc.of CVPR 2008, 2008, pp.1-7.

[10] H. Schneiderman, "Feature-centric evaluation for efficient cascaded object detection," in Proc. of CVPR 2004, vol.2, 2004, pp.29-36.

[11] M.T. Pham, and T.J. Cham, "Fast training and selection and Haar features using statistics in boosting-based face detection," in Proc. of ICCV 2007, 2007, pp.1-7.

[12] L.Zhang, R.Chu, S.Xiang, S.Liao, and S.Li, "Face detection based on multi-block LBP representation," in Proc. of ICB 2007, 2007, pp.11-18.

[13] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray scale and rotation invariant texture analysis with local binary patterns," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.24, no.7, pp.971-987, July, 2002.

[14] H. Bay, A. Ess, T. Tuytelaars, and G. L. Van, "SURF: Speeded Up Robust Features, " in Proc. of ECCV, Part(1), 2006, pp.404-417.

[15] H. Bay, A. Ess, T. Tuytelaars, and G. L. Van, "SURF: Speeded Up Robust Features,"Computer Vision and Image Understanding, vol.110, no.3, pp.346-359, June, 2008.

[16] Y. Freund, and R. E. Schapire, "Experiments with a new boosting algorithm," in Proc. of ICML 1996, 1996, pp.148-156.

[17] J. Kennedy, and R.C.Eberhart, "Particle swarm optimization," in Proc. of ICNN 1995, 1995, pp.1942-1948.