Multi-Stage Approach to Fast Face Detection

Duy-Dinh Le [1] and Shin'ichi Satoh [1,2]
[1] Department of Informatics
The Graduate University for Advanced Studies
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, Japan 101-8430
ledduy@grad.nii.ac.jp
[2] National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, Japan 101-8430
satoh@nii.ac.jp

Abstract

This paper describes a multi-stage approach for achieving fast and robust face detection. This approach was motivated by the work of Viola and Jones [7] using a cascade of classifiers yielding a coarse-to-fine strategy to significantly reduce detection time while maintaining high detection rate. However, it is distinguished from the previous work by two facts: *(i) First, a new stage is added to more quickly estimate face candidate regions by using a larger window size and a larger moving step size. (ii) Second, we propose using SVM classifiers instead of AdaBoost classifiers in the last stage and study how to efficiently reuse Haar wavelet features selected by AdaBoost in the previous stage for SVM classifiers.* By combining AdaBoost and SVM classifiers, the final system can obtain both fast and robust detection because most of the non-face patterns are quickly rejected in earlier layers, while only a small number of promising face patterns are robustly classified in the later layers. Extensive experimental results demonstrated that our proposed system can achieve promising results.

1 Introduction

Face detection is one of the most active research areas in computer vision because of the many interesting applications in fields such as security, surveillance, multimedia retrieval, and human computer interaction. Although it has been studied for more than 30 years, a fast and robust face detection system that can handle the variations found in different faces in real applications, such as facial expressions, pose changes, illumination changes, complex backgrounds, and low resolutions, is still challenging [8].

In a typical face detector that is scale-free and location-free, the system uses a face classifier whose windows are swept over the input image at multiple scales. This leads the number of analyzed non-face patterns being very large. Face detectors based on Support Vector Machines (SVM) [4] or Neural Networks [6] are usually slow because they process non-face regions and face regions in the input image equally.

BMVC 2005 doi:10.5244/C.19.61

Viola and Jones [7] proposed a framework to improve the detection speed while maintaining a high detection rate. Its success mainly comes from a cascaded structure of simple-to-complex classifiers based on AdaBoost and Haar wavelet features that allow for a dramatic reduction in computation time. With this type of structure, simple classifiers in the earliest layers are used to quickly reject a large number of non-face patterns, while more accurate and slower classifiers focus on a smaller number of face-like patterns. However, this approach still has the following problems:

First, the layer classifiers that use AdaBoost and Haar wavelet features are only efficient in quickly rejecting simple non-face patterns. In order to robustly classify complex patterns, the system needs to use a larger number of features and layer classifiers. This is apparent in the later layers when face and non-face patterns become hard to distinguish; weak classifiers are too weak to boost [7, 9]. With the first several layers in our experiment using some 500 weak classifiers, more than 99.9% of non-face patterns are rejected. However, to turn later layers into robustly classifying a smaller number of remaining patterns, it requires a lot more weak classifiers, e.g. 5,740 weak classifiers. Recall that the average training time of one weak classifier is about 10 minutes on a PC 2.8 MHz, 512MB RAM; training this detector [7] usually takes a very long training time.

Second, AdaBoost based classifiers are constructed by adding features after each round of boosting until it reaches the training goals, e.g. maximum false positive rate of 0.999 and minimum detection rate of 0.5. To automatically implement a cascade of these classifiers, these target rates must be known in advance. However, in practice, this is impossible because the complexity of the training set varies through layers and leads to training processes not terminating properly. In Viola's implementation [7], the training process for each layer is forced to terminate if a maximum of 200 features are reached. Furthermore, adding more features directly increases the computation time and might cause overfitting.

We propose a multi-stage approach for building a fast and robust face detection system by adopting the advantages of Viola's approach and introducing a method for handling the above problems. Specifically, we use the cascaded structure of simple-to-complex AdaBoost based classifiers to significantly reduce the computation time and propose using SVM classifiers in the later layers of the cascade, both to maintain a high detection rate and to control the balance between the training and running times. By combining AdaBoost and SVM classifiers, the final system can obtain both fast and robust detection because most of the non-face patterns are quickly rejected in earlier layers, while only a small number of promising face patterns are robustly classified in the later layers.

1.1 Related Work

Generally, face detection approaches can be divided into two categories: feature-based approaches and appearance-based approaches [8]. In the appearance-based approach, a combination of simple-to-complex classifiers is proposed [5, 7, 2] to deal with the problem of processing a large number of windows. In the combination proposed by Romdhani et al., non-linear SVM classifiers using pixel-based features are arranged into a sequence with an increase in the number of support vectors or in [2], linear SVM classifiers trained at different resolutions are used for the rejection stage, and a reduced set of PCA-based features are used with the non-linear SVMs in the classification stage, to reduce computation time. The main drawback of using the SVM with pixel-based features or PCA

features is the expensive kernel computation, especially when the number of support vectors is large. Furthermore, the normalization steps, such as lighting correction, histogram equalization, or PCA projection, also usually take time.

In [7], a cascade of classifiers, which is formed by a combination of a different number of simple weak classifiers with increasing complexity, leads to a real-time face detection system. Recently, many improvements in the original boosting cascade ideas have also been proposed [9, 3].

Our system is distinguished from the previous systems by the following two facts: First, a new stage is added to estimate the face candidate regions by using a larger window size and a larger moving step size. We use 36 x 36 pixel window-based classifiers with a moving step size of 12 pixels, to quickly estimate the candidate face regions. The idea of using larger windows and moving the step size was used in [6], but it severely degrades performance. Here we take the advantages from the combination of the Haar wavelet features and the AdaBoost learning for fast evaluation, to improve the speed, while maintaining comparable performance.

Second, we studied how to reuse the features selected by AdaBoost in the previous stage, for the SVM classifiers of the last stage. There were two advantages: (*i) Haar wavelet features are very fast in evaluating and normalizing [7]. Furthermore, it is unnecessary to re-evaluate these features because they have been previously evaluated. (ii) By using SVM classifiers with powerful generalization, using too many features in the cascade is avoided, therefore importantly saving training time and avoiding overfitting.*

2 System Overview

The proposed face detection system consists of three stages that classify a 24 x 24 pixel window as either a face or a non-face. To detect faces of different sizes and locations, we apply a detector at every location and scale in the input image with a scale factor of 1.25. An outline of this system is shown in Figure 1.

The first stage is a cascade of classifiers used to estimate face candidate regions by evaluating 36 x 36 input windows, with a moving step of 12 pixels. If a 36 x 36 window is detected as the existence of a face, 144 (=12 x 12) likely face positions are collected and passed to the next stage. The second stage is a cascade of classifiers that is used to investigate 24 x 24 window face candidate locations returned from the previous stage. It tries to filter out as many non-face patterns as possible before passing hard patterns to the final stage classifier.

Building these two first stages is similar to [7], which uses Haar wavelet features and AdaBoost learning to form a cascade of classifiers. They are very fast to reject non-face patterns because the complexity of classifiers is adapted corresponding to the difficulty in the input patterns. Furthermore, with a larger moving step size, 12 pixels in the first stage, the speed of rejection increases significantly.

The last stage is a cascade of non-linear SVM classifiers that reuse features that have been selected by AdaBoost learning in the second stage classifier. These feature values are evaluated and normalized to be between 0 and 1 to form a feature vector. In our experiments, only 125 features are used and hence it is faster than using any pixel-based SVM classifiers.

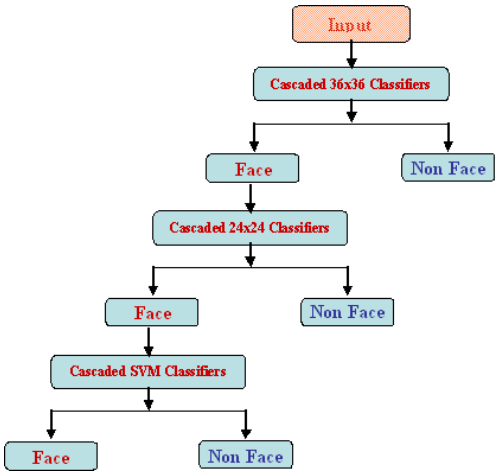

Figure 1: Three-stage face detection system

3 Training Cascaded Classifiers

In this section, training cascaded classifiers is briefly introduced. Because we use the same techniques proposed by Viola [7], readers can refer to that paper for more details.

3.1 Haar Wavelet Features

As in Viola's proposal [7], there are three kinds of Haar wavelet features modeled from adjacent rectangles with the same size and shape. The feature value is defined as the difference of the sum of the pixels within the rectangles. By using integral image definition [7], these rectangle feature values can be computed very fast.

3.2 AdaBoost Learning

The aim of boosting is to improve the classification performance of any given simple learning algorithm [1]. Given T weak classifiers $h_t(x)$ learned through the T round of boosting, the strong classifier is formed by a linear combination: $H(x) = \sum_{t=1}^{T} \alpha_t h_t(x)$ where α_i are the coefficients found in the boosting process.

Each weak classifier h_j is associated with a feature f_j and a threshold θ_j, such that the number of incorrect classified examples corresponding to this weak classifier is minimized: $h_j(x) = \begin{cases} 1 & if\ p_j f_j(x) < p_j \theta_j \\ 0 & \text{otherwise} \end{cases}$, where polarity p_j indicates the direction of the inequality sign. Each round of boosting, the best weak classifier h_t with the lowest error ε_t is chosen.

3.3 Cascade of Classifiers

The main idea of building a cascade of classifiers is to reduce the computation time by giving different treatments to different complexities of input windows (Figure 2). Only

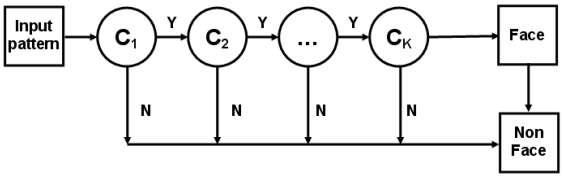

Figure 2: Cascade of classifiers for object detection

input windows that have passed through all the layers of the cascade are classified as faces.

Training cascaded classifiers that can achieve both good detection rate and less computation time is quite complex, because a higher detection rate requires more features, but more features are correspondent to more time for evaluation. To simplify this, the detection rate goal and the false positive rate goal for each layer are set beforehand. Viola [7] stated that, if the layer classifier could achieve the predefined target goals after 200 features are used, the training process will stop and a new layer will be added.

4 SVM classifier

The Support Vector Machine (SVM) is a statistical learning method based on the structure risk minimization principle. It has been very efficiently proved in many pattern recognition applications [4, 5, 2]. In the binary classification case, the objective of the SVM is to find a best separating hyperplane with a maximum margin.

The form of SVM classifiers is: $y = sign(\sum_{i=1}^{N} y_i \alpha_i K(x, x_i) + b)$, where x is the d-dimensional vector of an observation example, $y \in \{-1, +1\}$ is a class label, x_i is the vector of the i^{th} training example, N is the number of training examples, and $K(x, x_i)$ is a kernel function. $\alpha = \{\alpha_1, \alpha_2, ..., \alpha_n\}$ is learned through the learning process.

Compared to AdaBoost classifiers, SVM classifiers are much slower in the evaluation because of the large number of support vectors and heavy kernel computation.

5 Experiments

5.1 Experiment Setup

For training, we collected 8,000, 24 x 24 size face patterns on the Internet. Non-face patterns are generated at different locations and scales from 6,278 images with various subjects, such as rocks, trees, buildings, scenery, and flowers, which contain no faces.

Face patterns for training the 36 x 36 classifiers are generated by randomly translating the 24 x 24 face patterns above within 36 x 36 windows. Some examples are shown in Figure 3. Only Haar wavelet features which belong to the 24 x 24 window located right in the middle of the 36 x 36 window are used, because a uniform background was used for the translated face patterns. In this way, the effects of the uniform backgrounds are removed. Only pixels that belong to the original 24 x 24 face patterns are used by the learn classifiers.

Figure 3: Face patterns used for training the 36x36 window based classifier

To train the cascade of classifiers of the rejection stage, the same 8,000 face patterns are used for all layers. Non-face patterns of the training and validating sets of the first layer in the cascade are selected randomly. Non-face patterns of the subsequent layer classifiers are false positives collected by the partial cascade on the set of non-face images. For each layer classifier, 8,000 non-face patterns are used for training and 8,000 other non-face patterns are used for validating.

To compare performance of the classifiers, we have implemented a fully cascaded classifier trained by AdaBoost, similar to what Viola used [7]. This detector consists of 39 layers, using 6,240 features. For the comparison of the SVM classifiers' performance, 2,450 face patterns and 7,500 non-face patterns, which are different from the training set, were used.

All experiments were run on a PC Pentium 4, 2.8 MHz, 512 MB RAM. The training process is terminated when no more false positives are found in the non-face images of the data set.

5.2 Features Selected by AdaBoost for SVM

There are two questions surrounding the reuse features selected by AdaBoost: *(i) Which layer that its features will be reused for SVM is the best? and (ii) How many features should be used?*

In Figure 4a, we show a comparison of the performance of the classifiers trained on 200-feature sets selected by different layers in the cascade (layers 7, 11, and 15). These comparable performances suggest that we can switch from the second stage (using AdaBoost) to the final stage (using SVM) at any time. As a result, total training time of the system can be easily controlled.

To answer the second question *(ii)*, we used the 200-feature set selected in layer 15 to generate different sub-feature sets having different numbers of features. The features in each set were selected in the order that they were added in the training process. For example, 25-feature sets consist of the first 25 features selected by AdaBoost when training layer 15. The results shown in Figure 4b indicate that with more than 100 features, classifiers' performance is comparable. Basically, in terms of speed, the higher number of features, the slower the classifier. Therefore, using about 125 features might obtain satisfactory results.

5.3 Efficiency of SVM Classifiers

Efficiency of a single SVM classifier over cascaded AdaBoost classifiers on hard classified patterns is shown in Figure 5. To gather hard non-face patterns in this experiment, a cascade of 15 AdaBoost classifiers was used to collect false positives. These false positives then used non-face patterns to train two classifiers: a single RBF SVM classifier and a cascade of 15 AdaBoost classifiers, with the training goal being a maximum hit rate of

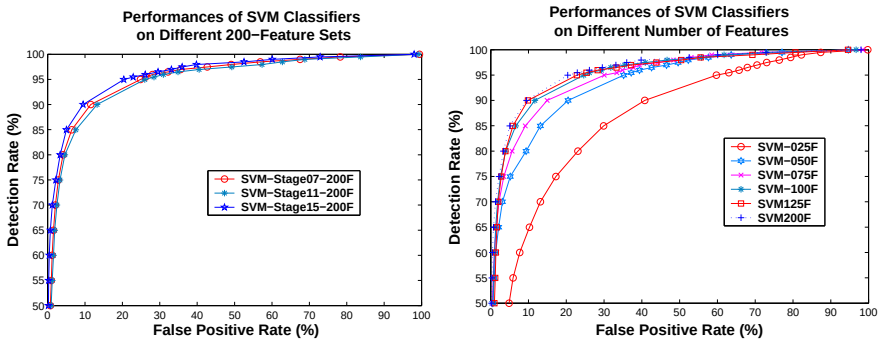

Figure 4: Performance of non-linear SVM classifiers on (a) different 200-feature sets and (b) different number of features

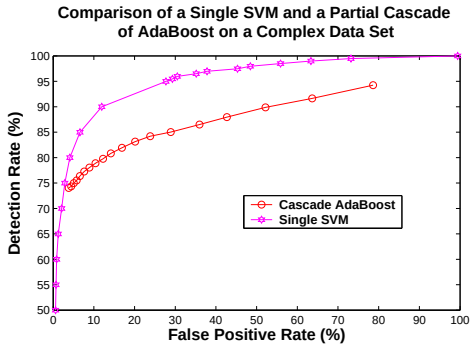

Figure 5: Comparison of single SVM classifier and cascaded AdaBoost classifiers on hard classified patterns

0.999, a minimum false positive rate of 0.50, and a maximum of 200 features. The result demonstrates that with hard classified patterns, a single SVM classifier is more robust than the cascaded AdaBoost classifiers.

5.4 Efficiency of Cascaded 36 x 36 Classifiers

The first stage is a cascade of classifiers that processes 36 x 36 patterns with a moving step size of 12 pixels. Figure 6 shows the face candidate regions returned by this stage. The rejection speed increased extremely fast, because this stage classifier was very fast in the evaluation and moved with a large step size. As showed in Table 1, if we use 36 x 36 AdaBoost classifiers with 24 x 24 AdaBoost classifiers, the detection speed increases approximately 1.5 times and the rejection rate increases 1.26 times compared to only using 24 x 24 AdaBoost classifiers. However, because 36 x 36 classifiers were trained by shifted 24 x 24 face samples on a uniform background, performance degrades 1.83% when compared to using only 24 x 24 classifiers.

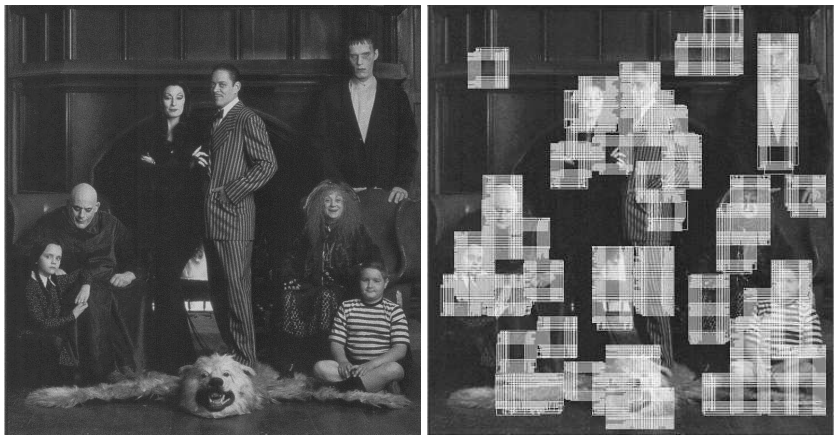

Figure 6: Face regions estimated by 36 x 36 classifiers: (a) Original image and (b) Candidate face regions

Table 1: Rejection performance of cascaded 36x36 classifiers

Performance on MIT+CMU test set[6]	With 36x36 classifier	Without 36x36 classifiers
Detection speed (WPS)	725,975	475,933
Face rejection rate	1.83%	0.0%
Non-face rejection rate	74.22% (cascaded 36x36 classifiers + layer 1 of cascaded 24x24 classifiers)	58.70% (only layer 1 of cascaded 24x24 classifiers)

5.5 Structure of the Final System

The final system consists of three stages. The first stage, which is a cascade of 36 x 36 classifiers, consists of five layers whose total number of features is 92. The second stage, which is a cascade of 24 x 24 classifiers, consists of 15 layers with 1,800 features. Compared with 6,061 features used by Viola [7], our system uses fewer features, therefore saving significant training time. The final stage consists of three SVM classifiers that take 125 features selected from the last layer in the second stage, to form a feature vector. The number of layers used in the first two stages is found by empirical experiments for optimization in both performance and speed.

5.6 Speed Comparison

In Table 2, we show the average running time in terms of the number of windows per second (WPS) of the three systems. The pure SVM has a constant running time of 554 WPS, regardless of the complexity of the input images (242 WPS is the average speed of the cascade of the three SVM classifiers), the pure AdaBoost (39 layers/6,240 features) has a running time of 796,623 WPS, and the fusion of the AdaBoost+SVM (15 layers/1800

Table 2: A comparison with AdaBoost-based systems

	AdaBoost+SVM		Full AdaBoost
Stages	Rejection stage (cascaded 36x36 and 24x24 classifiers)	Classification stage (cascaded 3 SVM classifiers)	39 layers - 6,240 features
Remaining patterns	0.026 %	0.001%	0.001%
Overall Time rate	32.86%	67.14 %	
Average WPS	2,209,133	242	
	725,975		796,623

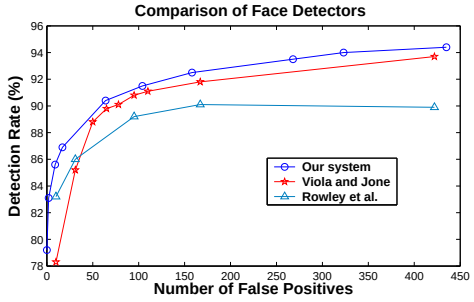

Figure 7: Comparison to other face detection systems

features) has an average running time of 725,975 WPS. Roughly speaking, the fusion of the AdaBoost+SVM is considerably faster than the pure SVM (more than 1,300 times), but it is slightly slower than the pure AdaBoost (approximately 1.09 times).

5.7 Detection Performance Comparison

We tested our system on the MIT+CMU frontal face benchmark test set [6]. This test set consists of 125 images with 480 frontal faces (excluding five images containing hand-drawn and cartoon faces). For each image in this set, we calculated the time that is used in each stage, over the total time needed to detect an image (*Total time= Time of rejection stage + Time of classification stage*). In addition, we also calculated the number of window patterns remaining after each stage. The rejection stage implies that the first two stages and the classification stage are from the SVM classifiers. The values shown in Table 2 indicate that the significant running time (67.14%) is used for classifying a very small number of hard patterns (0.026%).

Compared to other systems, Figure 7 and Table 3 demonstrate that our system outperforms systems [6, 7] and other SVM-based systems [5, 2].

Table 3: A comparison with other SVM-based systems

Performance	Detection rate	False positive rate
Our system	96.5%	1.0E-05
Romdhani et al. [5]	80.7%	1.0E-05
Heisele et al. [2]	70.0%	1.0E-05

6 Conclusion

We have presented a method for building a fast and robust face detection system based on a multi-stage approach. The cascaded structure of the AdaBoost-based classifiers in the first two stages allows for the best adaptation to various complexities of input patterns, while non-linear SVM classifiers in the final stage are robust enough to achieve good results. Extensive experimental results demonstrated that significant computation time is devoted to potential face regions. Almost all non-face patterns are quickly rejected by the first two stages and only a very small number of face-like patterns are processed by slow SVM classifiers. Discriminant Haar wavelet features selected from AdaBoost are used for all stage classifiers to take advantage of their efficient representation and fast evaluation.

References

[1] Y. Freund and R. E. Schapire. A short introduction to boosting. *J. Japan. Soc. for Artif. Intel.*, 14(5):771–780, Sep 1999.

[2] B. Heisele, T. Serre, S. Prentice, and T. Poggio. Hierarchical classification and feature reduction for fast face detection with support vector machines. *Pattern Recognition*, 36(9):2007–2017, Sep 2003.

[3] S.Z. Li and Z. Zhang. Floatboost learning and statistical face detection. *IEEE TPAMI*, 26(9):23–38, Sep 2004.

[4] E. Osuna, R. Freund, and F. Girosi. Training support vector machines: An application to face dectection. In *Proc. CVPR*, pages 130–136, Jun 1997.

[5] S. Romdhani, P. H. S. Torr, B. Schlkopf, and A. Blake. Computationally efficient face detection. In *Proc. ICCV*, volume 1, pages 695–700, Jul 2001.

[6] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE TPAMI*, 20(1):23–38, Jan 1998.

[7] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. CVPR*, volume 1, pages 511–518, Jun 2001.

[8] M-H. Yang, D. Kriegman, and N. Ahuja. Detecting faces in images: A survey. *IEEE TPAMI*, 24(1):34–58, Jan 2002.

[9] D. Zhang, S.Z. Li, and G. Perez. Real-time face detectioin using boosting in hierarchical feature spaces. In *Proc. ICPR*, volume 2, pages 411–414, Aug 2004.