# Improving Detector of Viola and Jones through SVM

Zhenchao Xu, Li Song, Jia Wang, and Yi Xu

Institute of Image Communication and Information Processing
Shanghai Jiao Tong University, Shanghai, PRC 200240
{xuzhenchao,song_li,jiawang,xuyi}@sjtu.edu.cn

**Abstract.** Boosted cascade proposed by Viola and Jones is applied to many object detection problems. In their cascade, the confidence value of each stage can only be used in the current stage so that interstage information is not utilized to enhance classification performance. In this paper, we present a new cascading structure added SVM stages which employ the confidence values of multiple preceding Adaboost stages as input. Specifically, a rejection hyperplane and a promotion hyperplane are learned for each added SVM stage. During detection process, negative detection windows are discarded earier by the rejection SVM hyperplane, and positive windows with high confidence value are boosted by promotion hyperplane to bypass the next stage of cascade. In order to construct the two distinct hyperplanes, different cost coefficients for training samples are chosen in SVM learning. Experiment results in UIUC data set demonstrate that the proposed method achieve high detection accuracy and better efficiency.

## 1 Introduction

Object detection is popular and significant issue in computer vision and pattern recognition. Examples include vehicle, face, and pedestrian detection. Many approaches have been proposed to solve detection problem in different circumstance. The majority of them use machine learning to construct a detector from a large number of training examples. Then the detector is scanned over the entire input image in order to find a pattern of intensities which is consistent with the target object. In smart video surveillance systems, object detection are usually integrated with object tracking and the methods for the two tasks can be facilitated by each other. In order to provide a real-time assistance for tracking process, a both accurate and rapid detection method is essential in integration object detector into a tracking algorithm.

A great number of algorithms have been proposed to address the problem of object detection. At the beginning, some researchers present models based on background subtraction to solve detection task, but it is difficult for them to identify a special class object from a crowd foreground. In [1], Viola and Jones describe a boosted cascade based on haar features for rapid face detection. Rotated haar-like features is introduced by Lienhart and Maydt [2] for

R. Koch et al. (Eds.): ACCV 2010 Workshops, Part I, LNCS 6468, pp. 64–73, 2011.

better detection. Viola and Jones also improve their proposal by integrating image intensity information with motion information. Wang and Jia [3] propose a cascading structure using boosted HOG features. There work focus on boosting classification performance by improving feature pool, but optimization in cascade structure is ignored. Wu and Brubaker [4] present a asymmetric learning for cascade to reduce the training time of detector. Chen [5] speed up detection process by combining cascade Adaboost with linear SVM, their framework improve the efficiency of negative detection windows but take no action for positive windows.

In this paper, we describe a new boost cascaded classifier added SVM stages which can reduce the detection time. Each efficient SVM stage is composed of a rejection hyperplane and a promotion hyperplane, which are both learned by SVM, but with different cost coefficients for positive examples and negative examples. Some negative detection windows are discarded earlier by the rejection SVM hyperplane to save the time in rejecting negatives. With the help of promotion hyperplane, those positive windows with high confidence don't enter the next stage as normal, but jump to the following one of the next stage so that they can be detected faster through bypassing some stages. The experiment results show that the proposed method can get better efficiency and achieve approximate accuracy in detection.

The paper is organized as follows. In section 2, we review the Viola and Jones's cascading structure and SVM learning. In Section 3, we introduce our improving cascading structure and discuss the training process of rejection hyperplane and promotion hyperplane. The experiment results are reported in section 4. Finally, we summarize and conclude the paper in Section 5.

## 2 Preliminaries

### 2.1 Basic Cascading Classifier

Viola and Jones's detector based on cascade structure (see Fig. 1) is extensively used in many researcher's work. The cascade consists of several Adaboost classifiers which are arranged in order of complexity. In this cascade structure, the output of previous stage classifiers is used as the input of the subsequent stages of cascade, and each successive classifier is trained only those samples which pass through the preceding stages. Detection windows are thought to be positive only when they can pass all the stages of cascade. While those windows do not contain object are rejected in the early stage of cascade.

The cascade can achieve real-time in detection, which is because that if at any point in the cascade a classifier rejects the sub-window under inspection, no further processing is performed and the search moves on to the next sub-window. The cascade therefore has the form of a degenerate decision tree. The performance of the entire cascade is closely related with each individual stage classifier, because the activation of each stage depends completely on the behavior of its predecessor. The overall detection rate D and false positive rate F for an entire cascade can be estimated as follows:

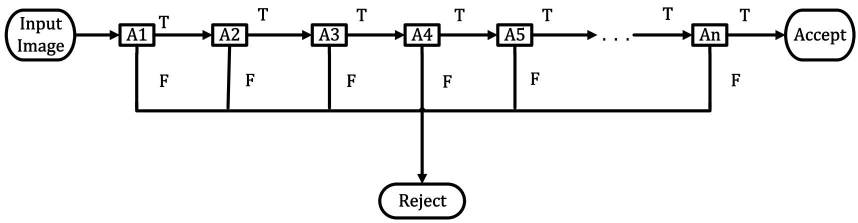


**Fig. 1.** Schematic depiction of basic cascading classifier, where "A" denotes the Adaboost stages. The output of previous stage classifiers is used as the input of the subsequent stages of cascade.

$$D = \prod_{i}^{n} d_i \tag{1}$$

$$D = \prod_{i}^{n} f_i \tag{2}$$

where n denotes the number of stages in this cascade, $d_i$ and $f_i$ denote the detection rate and false positive rate of the ith stage, respectively.

## 2.2 Adaboost Learning

Generally, the stage classifier of cascade is constructed by strong learning algorithms, which are used to select a small set of features and enhance the performance of classifier. Adaboost learning algorithm is proposed by Freund and Schapire [6] and is proved that the learning error of the strong Adaboost classifier approaches zero exponentially in the number of training rounds.

Adaboost learning is an adaptive machine learning algorithm in the sense that subsequent classifiers built are tweaked in favor of those examples misclassified by previous classifiers. A few weak classifiers are selected by Adaboost learner in a series rounds. Given example images $\{\mathbf{x_i}, y_i\}$, i=1,. . . ,n, where n is the number of examples, yi= -1,1 for negative and positive examples respectively. Weights of each example $w_{1,i}$ is initialized to be $\frac{1}{n}$. For week classifier j, the threshold classification function is $h_j(\mathbf{x_i})$, and the error is evaluated as follows:

$$e_j = \sum_{i} \frac{1}{2} w_i \mid h_j(\mathbf{x_i}) - y_i \mid \tag{3}$$

On every round, the classifier with lowest error $e_k$ is extracted from weak classifier set. Then update and normalize the weights of each examples as follows:

$$w_{k+1,i} = w_{k,i} \gamma_k^{1-a_i} \tag{4}$$

$$w_{k+1,i} = \frac{w_{k+1,i}}{\sum_{i=1}^{n} w_{k+1,i}} \tag{5}$$

where k denotes the kth round, $a_i = 0$ if example $x_i$ is classified correctly, $a_i = 1$ otherwise, and $\gamma_k = \frac{e_k}{1-e_k}$. According to the adaptive process of weights adjustment, the week classifier selected in the next round focuses more on incorrect samples. Hence, each round of the boosting process, which selects a new weak classifier, can be viewed as a feature selection process. Adaboost provides an effective learning algorithm and strong bounds on generalization performance [7][8].

## 3 Improving Cascade through SVM

In Viola and Jones's cascade structure, the classification result relies on the confidence value of the stage classifier for data. During detection process, each stage classifier compares the confidence value for a detection window with its threshold, and then decides to accept or reject the detection window. Normally, the decision of the current stage is only related with its corresponding confidence value, which can not be used by other stages. In that case, interstage information is not utilized by cascade to make decision of classification.

Actually, it is feasible to exploit both stage-wise information and cross-stage information to boost the performance of detector, which is implemented by creating some new stages for original cascade, and the input vector of each new stage is composed of the confidence values of multiple preceding Adaboost stages. In order to own ability to make a further decision to those detection windows which have passed the preceding stages, the new stage added after several Adaboost stages in original cascade will be trained to be high precision based on SVM. In the following of this paper, the new high precision stage is called "H" stage.

### 3.1 Improving Cascading Structure

An efficient cascade added one "H" stage after every two Adboost stages is illustrated in Fig. 2. The cascade structure is defined as "AAHAHAH. . . AH". In this structure, we employ the interstage cross-reference information of neighboring stages to boost the detection performance. The confidence values of the preceding two Adaboost classifiers are used as the input of the "H" classifier.

In our algorithm, "H" stage makes a decision with three choices by learning two SVM hyperplanes, which are formulated as $H_- : \mathbf{w}_- \cdot \mathbf{x} + b_- = 0$ , and $H_+ : \mathbf{w}_+ \cdot \mathbf{x} + b_+ = 0$, respectively. We add a new "jump" choice besides simply accepting and rejecting windows. The decision of "H" stage is based on the SVM confidence value of "H" stage for detection window x. We will give a detailed discussion about cost coefficients selection strategy in next section.

### 3.2 Optimization of "H" Stage

For each stage of our cascade, we need train two different SVM hyperplanes to further reject negative samples and accelerate positives to pass cascade classifier. Unbalanced cost coefficients for positive and negative training examples are used

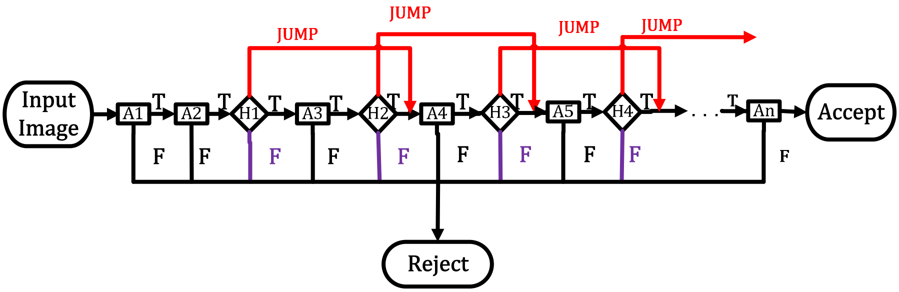


**Fig. 2.** Illustration of a efficient cascade structure. In the cascade, "A" denotes the Adaboost stages and "H" denotes the high efficiency SVM stages. The confidence value of the preceding two Adaboost stages are as the input of "H"stage. Red arrows show that detection window is promoted to bypass the next stage, and purple lines demonstrate that detection window is rejected in advance.

for finding most efficient negative rejection hyperplanes and positive promotion hyperplane. The training of this two hyperplanes can be formulated as:

$$\min_{w,b,x_i} \quad \frac{1}{2}\| \mathbf{w} \|^2 + C_+ \sum_{k=1}^{n_+} \xi_i + C_- \sum_{k=1}^{n_-} \xi_i \tag{6}$$

$$s.t.: \quad y_i[K(\mathbf{w}, \mathbf{x_i}) + b] - 1 \geq -\xi_i \tag{7}$$

$$\xi_i \geq 0, \ \ for \ \ i = 1 \ldots n \tag{8}$$

In our implementation, we expect that the rejection hyperplane to allow all the positive training samples to be classified correctly and get highest rejection rate for negatives. In order to achieve the expectation, training positives examples are given a large cost coefficient $C_+$, while negative examples are given a quite small cost coefficient $C_-$, which means $C_+ \gg C_-$. Therefor, any training positives located in incorrect side of rejection hyperplane will bring a bigger penalty for objective function Eq. 6, while negatives located in wrong side bring minor penalty. As show in Fig. 3 (a), all positive data are above the rejection hyperplane trained in this condition of unbalanced cost coefficients, while negatives are distributed in both sides of the rejection hyperplane.

Accelerating positives examples to go through cascade classifier can save time in detecting positive windows. Because of only positives with quite high confidence value have potential to bypass the next stage. In training of promotion hyperplane, incorrect decision for negative will be penalized greatly but misclassification of positives makes a little sense. The unequally importance of samples motivate us to set the small cost coefficient $C_+$ for positives but give negatives a quite big cost coefficient $C_-$ , which means $C_- \gg C_+$. As show Fig. 3 (b), all negatives examples are below the promotion hyperplane, while positives are distributed in both sides of the hyperplane.

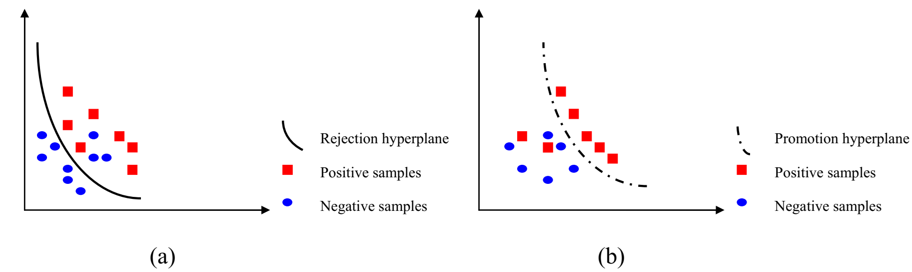


**Fig. 3.** SVM hyperplane learning in "H" stage. (a) Rejection hyperplane, red box denote positive samples, and blue box denote negatives. (b) Promotion hyperplane, red box denote positive samples, and blue box denote negatives.

### 3.3 Training Process of Improving Cascading Structure

In our proposal, there are two steps in training the improving cascading structure. The first step is training of original Viola and Jones's boosted cascade classifier. The second step is construction of novel "H" stage based on SVM. Table. 1 is the input parameters for the whole training process. The pseudo-code for learning the improving cascading structure is given in algorithm 1 and algorithm 2.

**Table 1.** Input parameters for training original cascade and "H" stage

| notation | definition or explanation |
|---|---|
| $d_{min}$ | Minimal desired hit rate of Adaboost stage |
| $f_{max}$ | Maximal desired false alarm rate of Adaboost stage |
| $F_{overall}$ | Overall false positive rate of cascade |
| $f_i$ | Current alarm rate of Adaboost stage |
| $\varphi_i$ | Current threshold of Adaboost stage |
| $N$ | The number of cascading stages |
| $V_{i,j}$ | Confidence value of Adaboost stage |
| $x_{i,j}$ | Input data of "H" stage |
| $d^H_{min}$ | Minimum detection rate of "H" stage |
| $f^H_{max}$ | Maximum false alarm rate of "H" stage |
| $d^H_i$ | Detection rate of "H" stage |
| $f^H_i$ | False alarm rate of "H" stage |
| $\alpha_i$ | Ratio of cost coefficient in rejection hyperplane |
| $\beta_i$ | Ratio of cost coefficient in promotion hyperplane |

**Algorithm 1.** Training of Original Cascade

**Initilization**: $N = log_{f_{max}}^{F_{overall}}, f_i = 1.$
**for** i=1:N
◇ **while**$(f_i > f_{max})$
•Add a week learner to the Adaboost classifier, and make sure it have lowest error as Eq. 3.
•Update the threshold $\varphi_i$ to guarantee detection rate $d_{min}$ is satisfied.
•Calculate the false alarm rate $f_i$.
•Modify the weights of each training samples and normalize them according to Eq. 4 and Eq. 5 respectively.
**end for**

**Algorithm 2.** Training of "H" Stage

(1) Train Reject SVM separating hyperplane
**Initialization**: $C_i^+ = \alpha_1 C_i^-,\ \alpha_1 = 5.$
**for** i=1:N-1
◇Use the confidence value of preceding two stages as the input of current SVM stage, $x_{i,j} = (V_{i,j}, V_{i+1,j})$.
◇ **while**$(d_i^H > d_{min}^H)$
•Training SVM separating hyperplane $H_i^+$ using optimization function.
•Recalculate detection rate $d_i^H$ for $H_i^+$.
•Modify $\alpha_i$: $\alpha_i \leftarrow 1.5\alpha_i$.
(2) Train Reject SVM separating hyperplane
**Initialization**: $C_i^- = \beta_1 C_i^+,\ \beta_1 = 5.$
**for** i=1:N-1
◇Use the confidence value of preceding two stages as the input of current SVM stage, $x_{i,j} = (V_{i,j}, V_{i+1,j})$.
◇ **while**$(f_i^H > f_{max}^H)$
•Training SVM separating hyperplane $H_i^-$ using optimization function.
•Recalculate false positive rate $f_i^H$ for $H_i^-$.
•Modify $\beta_i$: $\beta_i \leftarrow 1.5\beta_i$.
**end for**

## 4 Experiment Results

In order to evaluate our efficient cascade classifier, we applied it in a challenging data set, the UIUC Image Database for car detection in our experiment. In total, we use 550 positive car samples and 550 non-car samples in training process. The size of all training images is $50 \times 20$. In addition, a test dataset which contain car images or non-car images is used to analyze performance of our classifier.Both the training dataset and test dataset are appropriate for our experiment, they contain cars in distinct backgrounds and different categories negative samples.

Haar rectangle features, including horizontal-edge, vertical-edge and titled rectangle features, are emplyed for our detector. This is because Haar features can acquire the crucial information of object and be calculated quickly through the integral-image. The feature set for our experiment consists of 344630 features for each $50 \times 20$ detection window.

Intel OpenCV library and Lin Chih-Jen's LIBSVM are employed to construct our experiment system. In the following, we will demonstrate the benefits of the presented approach by comparing our improving cascade with basic cascade. At the beginning, we set the minimum detection rate $d_{min}$ of each Adaboost stage classifier to be 99.95% and the maximum false positive rate $f_{max}$ of each Adaboost classifier to be 50%, and original Adaboost cascade is created using Viola and Jones's method. Ratio $\alpha$ for rejection hyperplane and ratio $\beta$ for promotion hyperplane are initialized to be 5. Then, we adjust $\alpha$ and $\beta$ to construct "H" stages with the optimal rejection hyperplane and promotion hyperplane.

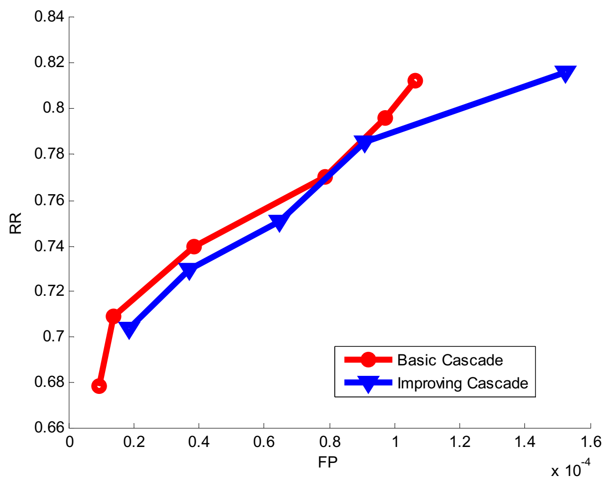


**Fig. 4.** Accuracy Performance contrast of basic cascade structure and improving cascade, where the horizontal axis denotes false positive rate, the vertical axis denotes recall rate. Blue curve and red curve denote the performance of basic cascade and improving cascade, respectively.

We show the superiority of our method by comparing the accuracy and efficiency performance of two kinds of cascade structure. We use the recall rate (RR) versus false positive rate (FP) curve to reflect the accuracy of the detector. The recall rate describes the ratio of the number of positives samples that were correctly classified to the total number of objects, whereas the false positive rate describes the ratio of the negatives that were incorrectly classified to the total number of testing negative windows. The accuracy performance of various detectors is demonstrated in Fig. 4.

**Table 2.** Efficiency performance contrast of basic cascade and improving cascade. BN and IN are the number of desired stages to reject all negative windows for basic cascade and improving cascade, respectively. BP and IP are the number of desired stages to detect all positive windows for basic cascade and improving cascade, respectively. $\Delta 1$ and $\Delta 2$ are the reduced stages for using our cascade.

| Image | BN | IN | $\Delta 1$ | BP | IP | $\Delta 2$ |
|---|---|---|---|---|---|---|
| Image 1 | 640 | 588 | 52 | 140 | 92 | 48 |
| Image 2 | 425 | 388 | 37 | 70 | 46 | 24 |
| Image 3 | 592 | 535 | 57 | 70 | 48 | 22 |
| Image 4 | 467 | 422 | 45 | 98 | 64 | 34 |
| Image 5 | 388 | 363 | 25 | 98 | 67 | 31 |
| Image 6 | 736 | 689 | 47 | 126 | 84 | 42 |

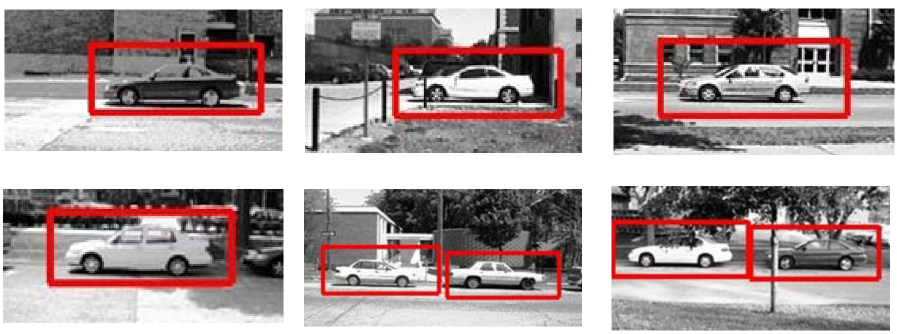

**Fig. 5.** Some detection results in UIUC data set

There are several positive windows and negative windows in each test image in UIUC. We compare the efficiency performance of basic cascade and improving cascade according to the number of desired stage classifiers for all positive and negative windows in one image. In our experiment, we use BN and IN to denote the number of desired stages to reject all negative windows in one image for basic cascade and improving cascade, respectively, and use BP and IP to denote the number of desired stages to detect all positive windows in one image for basic cascade and improving cascade, respectively. $\Delta 1$ and $\Delta 2$ are the reduced stages for using our cascade to reject negatives and detect positives, respectively. The efficiency contrast of our detector with basic cascading structure for several images in UIUC is demonstrated in Table. 2.

In average for totally 170 test images, 6.9% stages are reduced for using rejection hyperplane, and 20.9% stages are reduced for using promotion hyperplane. The experiment results show that our cascade structure with “H” stage achieve approximately the same detection accuracy as basic cascade classifier and better efficiency performance. Some detection results are illustrated in Fig. 5.

## 5 Conclusion

In this paper, we present a new cascaded structure by added high efficient stages, each of them provide a rejection hyperplane and a promotion hyperplane for vehicle detection. The structure can help keep the detection accuracy and have better detection efficiency. The efficient structure can also be applied to other object detection problems. In future, we will try to extend this structure to multiclass classification and integrate our detection method with some tracking algorithms.

## Acknowledgements

This work was supported in part by NSFC(60702044,60625103,60632040),MIIT of China(2010ZX03004-003) and 973(2010CB731401,2010CB731406).

## References

1. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: IEEE Proc. Int. Conf. Computer Vision and Pattern Recognition, pp. 511–518 (2001)
2. Lienhart, R., Maydt, J.: An extended set of haar-like features for rapid object detection. In: IEEE ICIP, pp. 900–903 (2002)
3. Wang, Z.-R.: Pedestrian detection using boosted HOG features. Intelligent Transportation Systems, 1155–1160 (2008)
4. Wu, J., Brubaker, S.C., Mullin, M.D., Rehg, J.M.: Fast asymmetric learning for cascade face detection. Pattern Analysis and Machine Intelligence, 369–382 (2008)
5. Chen, Y.-T., Chen, C.-S.: Fast Human detection using a novel boosted cascading structure with meta stages. IEEE Transactions on Image Processing, 1452–1464 (2008)
6. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting, pp. 23–37 (1995)
7. Papageorgiou, C., Oren, M., Poggio, T.: A general framework for object detection. In: International Conference on Computer Wsion, pp. 555–562 (1998)
8. Schapire, R.E., Freund, Y., Bartlett, P., Lee, W.S.: Boosting the margin: A new explanation for the effectiveness of voting methods. In: Proceedings of the Fourteenth International Conference on Machine Learning, pp. 1651–1686 (1997)
9. Burges, J.C.: A Tutorial on Support Vector Machines for Pattern Recognition. Data Mining and Knowledge Discovery, 121–167 (1998)
10. Tu, P.H., Rittscher, J., Krahnstoever, N.: Simultaneous estimation of segmentation and shape. Computer Vision and Pattern Recognition, 486–493 (2005)