A Rapid Pretreatment Method For Object Detection in Dynamic Scenes

Zhenming Nong, Yuesheng Zhu, and Hao Lai

***Abstract*—In this paper, an efficient approach is proposed to improve detection efficiency of sliding window based detection methods by setting adaptive thresholds for regular object detection in the moving environment. In the proposed approach, the symmetry and variance (SYM-VAR) information of targets is learned from current frame and historical frames, and the information is used to filter out the sub-windows which may not contain the targets in the next frame. Our experimental results have demonstrated that the proposed approach can reduce nearly 50% of the average detection time with a small tradeoff of accuracy compared to typical HOG-based (histogram of oriented gradient) methods.**

***Index Terms*—Adaptive online learning, variance, symmetry, sliding window detection.**

I. Introduction

Detecting objects in moving environment is an important modular in video surveillance. Sliding window based methods are good detection approaches [1]–[7]. However, they suffer from expensive computational cost for scanning too many sub-windows. For example in Fig. 1, for a 64×128 scanning window and 8 sliding-stride, an image with size 640×480 will generate 3285 sub-windows, and when the scale size of the input image decreases with the factor of 1/1.05, it will generate 27 multi-scale images and 25132 sub-windows. In order to detect the regular objects fast in moving environment, an effective object detection approach based on symmetry (*SYM*) and variance (*VAR*) information is proposed in this paper to eliminate the uninterested scanning sub-windows by using an online adaptive thresholds update tactic.

The rest of the article is organized as follows. Related work is discussed in Section II. Section III presents the proposed approach. Experimental results are presented and discussed in Section IV. Finally we conclude in Section V.

II. Related Work

There are many algorithms which are proposed for object detection. Most of them are sliding window based and use features to detect objects. Reference [8] shows the local binary patterns (LBP), which is widely used in the face or people detection. Haar wave-let [9] has been used as the feature for general object detection. Reference [1] proposed the histogram of oriented gradient (HOG) which has been demonstrated to be very effective for object detection. The HOG-SVM classification is a representative sliding window based method which is widely used in the objects detection area, especially for the pedestrian detection [1], [5], [6].

The disadvantage of the sliding window based methods is that they are time-consuming for detecting too many sub-windows. One way to reduce the sub-windows is to shrink the detect region by background modeling/subtraction [10], [11]. However, background modeling/subtraction may not always be effective especially when the background changes frequently. In this paper we present a method use the symmetry and variance information to detect objects in the moving environment. Experiment show that the proposed method can filter out the uninterested sub-windows effectively.

Fig. 1. Multi-scale sliding window detection method.

III. Proposed Approach

The framework of the proposed method is shown in Fig. 2. For each scanning window, it will be filtered by the learned variance and symmetry information. The scanning window which is not filtered out will be sent to the HOG-SVM step for target detection. Finally the detected targets will be used to update the thresholds.

We found that most of the regular targets are approximate symmetry, such as Fig. 3 (a), (b), (f). The symmetry information can be used to filter out most of the scanning windows when the targets are not in the center of the window (generally have larger *SYM* value), such as Fig. 3 (c), (d), (g), (h). And the variance information can be used to eliminate most the scanning windows which are single pure backgrounds (generally have smaller *VAR* value), such as the road, the sky, and the piazza, for example in Fig. 3 (e), (i), (j).

Each image has its best thresholds for the symmetry (*SYM*) and variance (*VAR*) information, and the continuous frames have similar thresholds, the discontinuous frames may have difference thresholds. So constant thresholds will not be effective to filter the sub-windows, in our method, an adaptive threshold update method is proposed to find the thresholds which are most appropriate for each image.

Manuscript received July 15, 2013; revised September 20, 2013.

The authors are with the Communication and Information Security Lab, Shenzhen Graduate School, Peking University, China (e-mail: zhuys@pkusz.edu.cn).

DOI: 10.7763/IJFCC.2014.V3.281

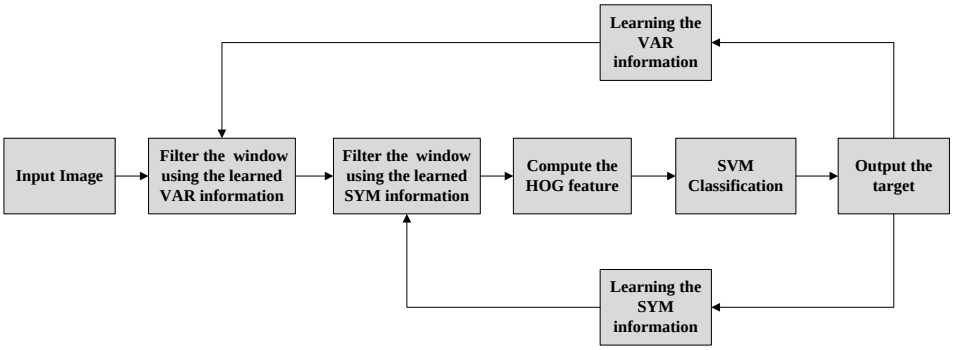


Fig. 2. The framework of the proposed method based on HOG features.

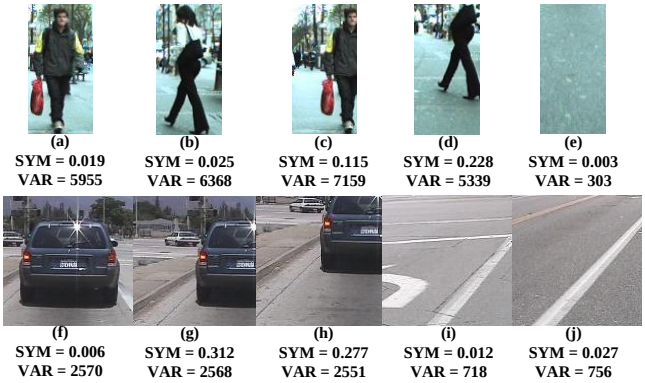


Fig. 3. Examples of sub-window with its *SYM* and *VAR* value under it. (a), (b), (f) are the interested sub-windows, they have smaller *SYM* value and bigger *VAR* value; (c), (d), (e), (g), (h), (i), (j) are the uninterested sub-windows, (c), (d), (g), (h) have bigger *SYM* value and (e), (i), (j) have smaller *VAR* value.

A. Variance Information Extract

While scanning the sub-windows, the variance value of each sub-window is computed. Suppose region D is one of the sub-windows with width w and height h in the input image as shown in Fig. 4.

Let $Sum_{x,y}$ represent the sum of the pixel intensity in the region where $0 \le w_r \le x$ and $0 \le h_r \le y$ (w_r and h_r are the width and height of the region). And $Sq_{x,y}$ represents the sum of squares of the pixels.

Then the sum of sub-window D (Sum_D) and the sum of squares of sub-window D (Sq_D) can be calculated as follows:

$$Sum_D = Sum_{x,y} - Sum_{x-w,y} - Sum_{x,y-h} + Sum_{x-w,y-h} \quad (1)$$

$$Sq_D = Sq_{x,y} - Sq_{x-w,y} - Sq_{x,y-h} + Sq_{x-w,y-h} \quad (2)$$

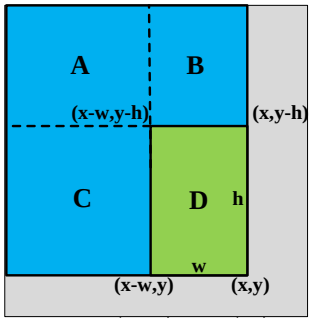


Fig. 4. The model to calculate *VAR* value.

Now we can get the value of the sub-window D's variance information (VAR_D) by (3):

$$VAR_D = Sq_D/(w \times h) - (Sum_D/(w \times h))^2 \quad (3)$$

B. Symmetry Information Extract

For each class of objects, a symmetry model which is most appropriate for them is designed as in Fig. 5. The specific model can help us to filter out the uninterested sub-windows more effectively.

The pedestrian is approximate bilateral symmetry, the model for pedestrian can be divided into three parts respectively correspond to the head, body and legs as in Fig. 5 (a), (b). In each sub-window such as Fig. 5 (a), the symmetry value can be calculated by (4):

$$SYM_V = |S_R - S_L| \quad (4)$$

where SYM_V is the value of the pixel difference, S_R and S_L are the sum of the pixel intensity in the right and left regions. S_R and S_L can be computed by (5), (6):

$$S_R = S_{RT} + S_{RM} + S_{RB} \quad (5)$$

$$S_L = S_{LT} + S_{LM} + S_{LB} \quad (6)$$

S_{RT}, S_{RM}, S_{RB}, S_{LT}, S_{LM}, and S_{LB} are the sum of the pixel intensity in the corresponding rectangle regions in Fig. 5 (b) and they can be computed quickly by the integral image method [12].

Then, we normalize the symmetry value by (7):

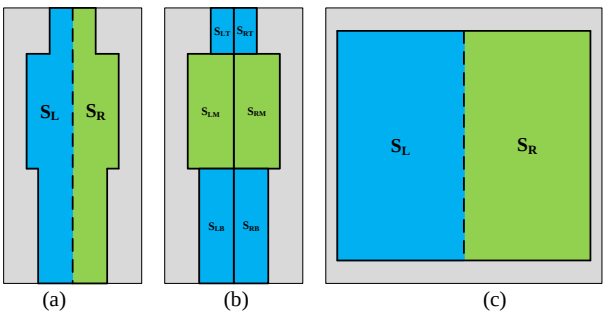


Fig. 5. The model to calculate *SYM* value. (a), (b) are the pedestrian models; (c) is the rear of car model.

$$SYM = SYM_V/(S_R + S_L) \quad (7)$$

where *SYM* is the value of the sub-window's symmetry information. For cars (the rear) detection we use the model

like Fig. 5 (c). For other objects, we can use the models which are most suitable for them, the principle is the same.

C. HOG-SVM Detection

In the detection stage, HOG feature [1] is applied in the proposed algorithm. Note that we do not contribute to the feature extraction, thus, any local feature descriptors other than HOG can also be applied to our detection model.

HOG has been accepted as one of the best features to capture the edge or local shape information. It splits the image into small squared or circular cells, computes the histogram of oriented gradients in each cell, normalizes the result using a block-wise pattern, and returns a descriptor for each cell. Stacking the cells into a squared image block can be used as an image window descriptor for object detection, combining with the SVM classifier, it can detect objects effectively.

D. Filter Criteria and Adaptive Update Method

Note that the *SYM* value is used to eliminate most of the scanning windows when the object locates near the boundary of the windows or the object is not in the center of the windows (generally have larger *SYM* value). And the *VAR* value is used to eliminate most of the scanning windows which are single pure backgrounds (generally have smaller *VAR* value), such as the road and the sky. The eliminate rule is as (8):

$$Object = \begin{cases} 1, & SYM < TH_S \text{ and } \mathrm{VAR} > \mathrm{TH_V} \\ 0, & otherwise \end{cases} \quad (8)$$

where TH_S and TH_V are the thresholds of *SYM* and *VAR* value. The scanning window may be considered to contain target when it fulfills $SYM < TH_S$ and $VAR > TH_V$, then it will be send to the HOG-SVM detection step to verify whether it has target. Other window which does not meet the condition will be eliminated.

The *SYM* and *VAR* value of a target will change while the appearance of the object or the illumination changes. For example, the front and side view of the same people. In the moving environment, object changes not much in the continuous frames, but changes a lot in the frames which have large interval.

In order to adapt the appearance variations caused by pose or illumination changes. The thresholds need to be updated properly.

Assume there are k targets being detected from frame i, for each target there are corresponding values of SYM_k and VAR_k. The best thresholds of this image can be obtained by (9), (10):

$$THB_{S,i} = \max\{SYM_1, SYM_2, \cdots, SYM_k\} \quad (9)$$

$$THB_{V,i} = \min\{VAR_1, VAR_2, \cdots, VAR_k\} \quad (10)$$

where $THB_{S,i}$ and $THB_{V,i}$ represent the best thresholds for image i. If the thresholds are learned from n continuous frames, when detecting the $(j+1)-th$ frame, the detection thresholds TH_S and TH_V can be calculated as follows:

$$TH_S = \max\{THB_{S,j-n}, THB_{S,j-n+1}, \cdots, THB_{S,j}\} \quad (11)$$

$$TH_V = \min\{THB_{V,j-n}, THB_{V,j-n+1}, \cdots, THB_{V,j}\} \quad (12)$$

The above strategy shows us how to obtain the thresholds, the outline of the adaptive thresholds update method is described in Algorithm I.

Algorithm I: Thresholds Update

Initialization:

$TH_S = 1, TH_V = 0, THB_{S,f} = 0, THB_{V,f} = \infty. (f \in (-\infty, +\infty))$

Input: Image frame i, learning parameter n and reset parameter m.

Output: New thresholds

1: **if** ($i \bmod m = 0$ or no hit targets), that means the $i-th$ frame fulfills the reset requirement.

then

2: Reset the thresholds:

$TH_S = 1, \quad TH_V = 0$

end

3:Detect image i by TH_S and TH_V, use the detected k targets to update the best thresholds of image i:

$THB_{S,i} = \max\{SYM_1, SYM_2, \cdots, SYM_k\}$

$THB_{V,i} = \min\{VAR_1, VAR_2, \cdots, VAR_k\}$

If there are no hit targets, keep the best thresholds of image i as the default value.

4: Update the thresholds based on the historical frames and frame i for the next frame:

$TH_S = \max\{THB_{S,j-n}, THB_{S,j-n+1}, \cdots, THB_{S,j}\}$

$TH_V = \min\{THB_{V,j-n}, THB_{V,j-n+1}, \cdots, THB_{V,j}\}$

The reset method is used to avoid missing the new targets which do not meet the thresholds or the background has big changes.

IV. Experimental Results

In order to evaluate the performace of our proposed approach, we test it on the Caltech cars datasets and ETH pedestrian datasets with HOG features. These datasets contain pedestrians and cars (the rear) in the mutative backgrounds. All of the experiments are tested on the PC with Pentium dual core 2.8 GHz, and 2 GB memory.

A. Caltech Cars Experiment

The training datasets contains 516 positive samples (80x80) from the MIT cars datasets and 10039 negative samples (80x80) cut out from the INRIA pedestrian negative images.

The test sets are 448 images from the Caltech cars datasets, the images are taken from a vehicle driving in the road environment which contains 519 cars (the rear) with 50 pixels high at least. The parameters of our algorithm are adjusted as follows: the learning parameters is set to 5 and the reset parameters is set to 10.

The detection results based on HOG, HOG+VAR, HOG+SYM, and HOG+VAR+SYM (the proposed mehtod) are compared. The DET Curve results of FPPI (false positive per image)-Miss Rate are shown in Fig. 6. The detection accuracy and average detection time comparisons are shown in Table I.

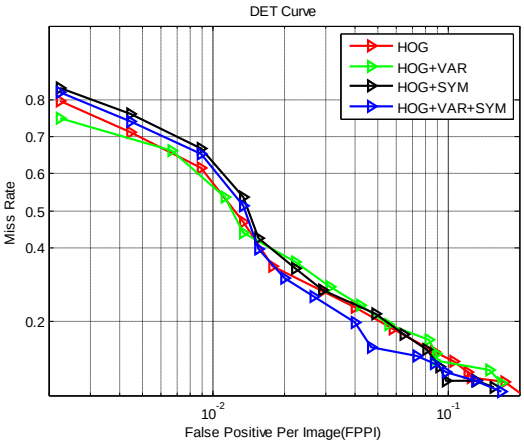


Fig. 6. The DET Curve results of the Caltech cars datasets.

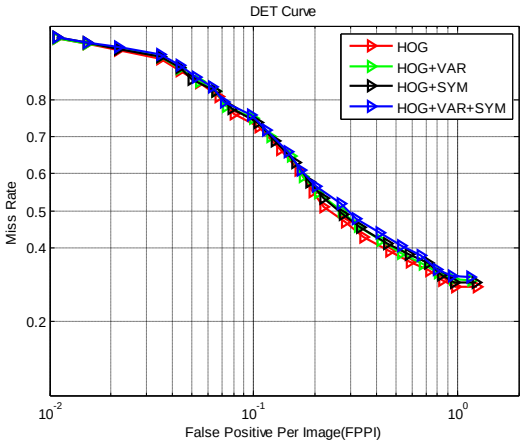


Fig. 7. The DET Curve results of the ETH pedestrian datasets.

TABLE I: THE DETECTION ACCURACY AND AVERAGE DETECTION TIME COMPARISONS OF CALTECH CARS RESULTS AT FPPI = 0.05

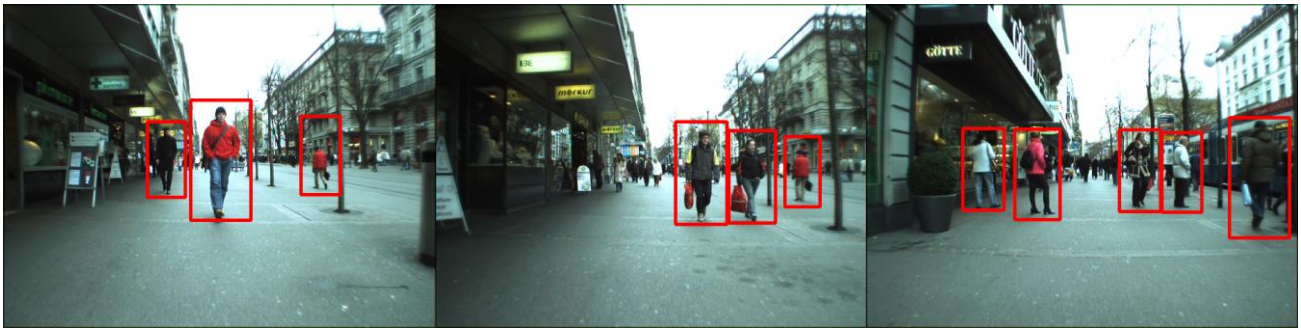
Algorithm	True positive / Total targets	Accuracy	Average Time
HOG	420/519	81%	587ms
HOG+VAR	436/519	84%	425ms
HOG+SYM	425/519	82%	379ms
HOG+VAR+SYM	457/519	88%	293ms

TABLE II: THE DETECTION ACCURACY AND AVERAGE DETECTION TIME COMPARISONS OF ETH PEDESTRIAN RESULTS AT FPPI = 0.5

Algorithm	True positive / Total targets	Accuracy	Average Time
HOG	1103/1809	61%	1552ms
HOG+VAR	1085/1809	60%	1057ms
HOG+SYM	1092/1809	60%	1243ms
HOG+VAR+SYM	1067/1809	59%	857ms

$TH_S = 0.007$ $THB_S = 0.006$ $TH_V = 2098$ $THB_V = 2098$ | $TH_S = 0.01$ $THB_S = 0.01$ $TH_V = 2098$ $THB_V = 2227$ | $TH_S = 0.02$ $THB_S = 0.02$ $TH_V = 2098$ $THB_V = 2392$ | $TH_S = 0.04$ $THB_S = 0.04$ $TH_V = 3889$ $THB_V = 3933$ | $TH_S = 0.06$ $THB_S = 0.05$ $TH_V = 1529$ $THB_V = 1959$ | $TH_S = 0.06$ $THB_S = 0.06$ $TH_V = 2391$ $THB_V = 2462$

(a)



$TH_S = 0.05$ $THB_S = 0.05$ $TH_V = 3608$ $THB_V = 3608$ | $TH_S = 0.04$ $THB_S = 0.03$ $TH_V = 3515$ $THB_V = 3720$ | $TH_S = 0.04$ $THB_S = 0.04$ $TH_V = 2521$ $THB_V = 2659$

(b)

Fig. 8. Some experiment results with their detection thresholds and best thresholds under them. (a) the Caltech cars results, the previous three images are continuous and they have similar best thresholds, the later three images are discontinuous and they may have different best thresholds; (b) the ETH pedestrian results, the images are discontinuous and they have different best thresholds, but the learned detection threshold of each image is close to its best thresholds.

Fig. 6 and Table I show that the detection result of our approach is better than the original method or at least as good as it and when we use the VAR or SYM information we can reduce the average detection time. The average detection time of our method is 50% faster than the origin method.

B. ETH Pedestrian Experiment

We take 18758 negative samples (64x128) cut out from the INRIA pedestrian negative datasets and 2417 positive samples (64x128) from the INRIA pedestrian positive datasets to train ours HOG descriptor for pedestrian detection.

The test sets are 636 images from the ETH datasets, all the images are got by a moving camera in the street, there contian 1809 pedestrians which are 100 pixels hight at least. The parameters of our algorithm are adjusted as follows: the learning parameters is set to 5 and the reset parameters is set to 10.

The DET Curve results of FPPI (false positive per image)-Miss Rate results based on HOG, HOG+VAR, HOG+SYM, and HOG+VAR+SYM (the proposed method) are shown in Fig. 7. And the detection accuracy and average detection time comparisons are shown in Table II.

Fig. 7 and Table II demonstrate that the detection result of the proposed method is as good as the origin method and when we use the VAR or SYM information we can reduce the

average detection time with a small tradeoff of accuracy. The proposed method can decrease nearly 50% of the average detection time compare to the origin method with a small tradeoff of accuracy.

Some of the experiment results are shown in Fig. 8. Fig. 8 (a) are the Caltech cars results, the previous three images are continuous and the later three images are discontinuous. It is visible that the continuous images have similar best thresholds and the discontinuous images may have different best thresholds. This phenomenon can be utilized by the proposed method effectively. The ETH pedestrian results are shown in Fig. 8 (b). The images are discontinuous and they have different best thresholds, but the detection threshold which is learned with the proposed method of each image is close to its best thresholds. We can use the detection threshold to filter out the uninterested sub-windows.

The experiment certifies that proposed method can reduce the uninterested sub-windows effectively, especially in the capacious environment.

V. Conclusion

In this paper, an efficient pretreatment method to modify the sliding window based object detection algorithm is proposed by combining the symmetry (*SYM*) information and variance (*VAR*) information. In addition, an effective online adaptive learning tactic is used to choose the best thresholds to eliminate the uninterested sub-windows. Experimental results demonstrate the effectiveness of the proposed algorithm in the moving environment. The proposed method can be applied to the systems which contain moving cameras, such as the video-based driver assistance systems and the latest Google Glass systems.

References

[1] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Computer Vision and Pattern Recognition*, San Diego, CA, USA, June 20-26, 2005, vol. 1, pp. 886-893.

[2] C. Papageorgiou and T. Poggio, "A trainable system for object detection," *International Journal of Computer Vision*, vol. 38, no. 1, pp. 15-33, June 2000.

[3] P. Dollár, S. Belongie, and P. Perona, "The fastest pedestrian detector in the west," in *Proc. British Machine Vision Conference*, Aberystwyth, Wales, UK, August 30–September 2, 2010, pp. 2895-2902.

[4] Y. Ding and J. Xiao, "Contextual boost for pedestrian detection," in *Proc. IEEE Computer Vision and Pattern Recognition*, Providence, RI, USA , June 16-2, 2012, pp. 2895-2902.

[5] X. Wang, T. X. Han, and S. Yan, "An HOG-LBP human detector with partial occlusion handling," in *Proc. IEEE 12th International Conference on Computer Vision*, Kyoto, Japan, September 27-October 4, 2009, pp. 32-39.

[6] Q. Zhu, M. C. Yeh, K. T. Cheng, and S. Avidan, "Fast human detection using a cascade of histograms of oriented gradients," in *Proc. IEEE Computer Vision and Pattern Recognition*, New York, NY, USA, June 17-22, 2006, vol. 2, pp. 1491-1498.

[7] Y. Zheng, C. Shen, R. Hartley, and X. Huang, "Pyramid center-symmetric local binary/trinary patterns for effective pedestrian detection," in *Proc. Computer Vision - ACCV 2010-10th Asian Conference on Computer Vision*, Queenstown, New Zealand, November 8-12, 2010, pp. 281-292.

[8] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern recognition*, vol. 29, no. 1, pp. 51-59, Aug. 1996.

[9] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, and T. Poggio, "Pedestrian detection using wavelet templates," in *Proc. IEEE Computer Vision and Pattern Recognition*, San Juan, Puerto Rico, June 17-19, 1997, pp. 193-199.

[10] S. Stalder, H. Grabner, and L. V. Gool, "Exploring context to learn scene specific object detectors," presented at the IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, Miami, Florida, USA, June 25, 2009.

[11] P. M. Roth, S. Sternig, H. Grabner, and H. Bischof, "Classifier grids for robust adaptive object detection," in *Proc. IEEE Computer Vision and Pattern Recognition*, Miami, Florida, USA, June 13-18, 2009, pp. 2727-2734.

[12] P. Viola, and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Computer Vision and Pattern Recognitio*n, Kauai, HI, USA, December 8-14, 2001, vol. 1, pp. I-511.

Zhenming Nong received BSc degree in 2011 from South China University of Technology. He has studied in the Communication and Information Security Lab, Shenzhen Graduate School, Peking University, Shenzhen, China since 2011 for MSc degree. His research interests are computer vision, machine learning, and multimedia technology.

Yuesheng Zhu received his B.Eng. degree in Radio Engineering, M.Eng. degree in Circuits and Systems and Ph.D. degree in Electronics Engineering in 1982, 1989 and 1996, respectively. He is currently working as a professor at the Lab of Communication and Information Security, Shenzhen Graduate School, Peking University. He is a senior member of IEEE, fellow of China Institute of Electronics, and senior member of China Institute of Communications. His interests include digital signal processing, multimedia technology, communication and information security.

Hao Lai received BSc degree in 2011 from Beijing University of Posts and Telecommunications. He has studied in the Communication and Information Security Lab, Shenzhen Graduate School, Peking University, Shenzhen, China since 2011 for MSc degree. His research interests are computer vision, multimedia technology, and information security.