BOOSTED LOCAL BINARIES FOR OBJECT DETECTION

Anonymous ICME submission

ABSTRACT

We propose a novel binary feature for object detection encoding local neighbor patterns of different sizes and locations. Each region pair of the proposed feature is selected by RealAdaBoost algorithm with a penalty term on the structure diversity. As a result, useful features that are good at describing specific objects will be chosen to build the classifier. Moreover, the encoding scheme is applied in both the gradient domain and the intensity domain, which is complementary to standard binary features (e.g. LBP and LAB).

The proposed method was tested using the CMU-MIT frontal face dataset, INRIA pedestrian dataset, and UIUC car dataset respectively. Experimental results show that the proposed method outperforms traditional binary features LBP and LAB, which contributes to a significant improvement on detection accuracy and converges 2 times faster. It also achieves comparable performance with the state-of-the-art algorithms.

***Index Terms*—**Binary Feature, Object Detection, LBP, RealAdaBoost, Structure-Aware

1. INTRODUCTION

In object detection, the high accuracy and efficiency are general targets. On one hand, to achieve better accuracy, some researchers propose descriptors with strong discriminative ability, e.g., Histogram of Oriented Gradient (HOG) feature [1], covariance matrix [2] or their combinations, e.g., heterogeneous feature [3], HOG-LBP [4]. Others focus on designing more effective classifiers, e.g., vector boosted tree [5], multiple kernels SVM [6] and multiple instance learning [7], etc. On the other hand, the efficiency issue is an essential requirement for real-time applications. The cascade structure proposed by Viola and Jones [8] shows considerable result on face detection. Recent works have proved its efficiency and effectiveness on other object categories such as pedestrians [2][9] and cars [3]. In most of the cases, the performance of a boosted classifier mainly depends on the features. To address the accuracy and efficiency issue simultaneously, boosting with appropriate features to construct the cascade classifier is the key step.

Due to the high efficiency, binary feature is one of the most common-used features. The Local Binary Pattern (LBP) [11] is a local descriptor based on binary coding of adjacent region pairs. It is proved to be effective on texture analysis [12], object detection [13], and face recognition [14]. Despite its simplicity, a number of LBP modifications and extensions have been proposed. Some of them focus on the post-processing steps which improve the discrimination ability of binary coding [15]. But the computation cost will also be greatly increased. Others focus on the definition of the location where gray value measurement are taken [13][16]. The improvement of the discrimination ability is relatively limited because the locations are artificially designed, and using intensity information might not be sufficient to solve some complex object detection problems.

This paper presents a method with the following contributions. First, we introduce a Boosted Local Binary (BLB) feature, where the local region pairs are selected by RealAdaBoost algorithm considering both the discrimination ability and the feature structure diversity. In addition, we identify that using the gradient image and gray image together for binary coding is more effective than only using the gray image. As a result, the proposed BLB feature is more discriminative and robust compared to common-used binary features, LBP and LAB [16]. We evaluate it by employing it to detect frontal faces, pedestrian, and side-view cars in static images. Experimental results show that BLB has a superior performance in comparison with LBP and LAB. The accuracy is also comparable with some gradient features (e.g., HOG) and state-of-the-art methods.

The rest of the paper is organized as follows: we first introduce the proposed BLB feature in Section 2. RealAdaBoost algorithm with BLB is described in Section 3. The next section presents the experimental results on frontal face detection, pedestrian detection, and side-view car detection. Conclusions are given in Section 5.

2. BOOSTED LOCAL BINARYS

2.1. Local Binary Feature and Local Assembly Binaries

The traditional LBP is developed for texture classification [12] and the success is due to its robustness under illumination variations, computational simplicity and discriminative power on specific patterns. Fig. 1 represents an example of traditional LBP, which is a binary coding of the intensity contrast of the center pixel and 8 neighboring pixels. If the intensity of neighboring pixels are higher than

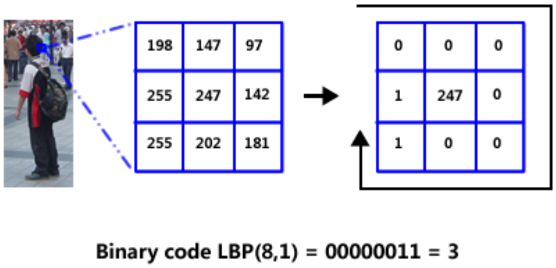

Fig.1. Traditional LBP feature

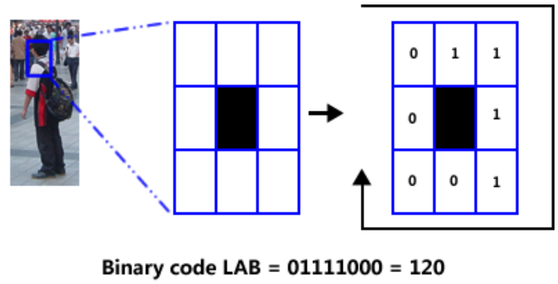

Fig. 2. LAB feature

the center pixel, the corresponding bit will be assigned 1, otherwise assigned 0. Given a center pixel, the LBP feature value is defined by equation (1)

$$LBP_{d,r} = \sum_{i=1}^{d} sign(I_i - I_{center}) \times 2^{i-1} \quad ...(1),$$

where d is the number of neighboring pixels, r is the distance between the neighboring pixels and the center pixel, I is the intensities, and

$$sign(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}.$$

LBP reflects the intensity pattern of single pixels. LAB [16] utilizes rectangles instead of pixels, as shown in Fig. 2. It combines 8 locally adjacent 2-rectangle binary Haar features with the same size. These Haar features share a common center rectangle. The encoding scheme is similar to LBP: if the intensity sum of the adjacent rectangle is higher than the center rectangle, the corresponding bit will be assigned 1; otherwise it will be assigned 0. Given a center rectangle C_0, the LAB feature value is defined by equation (2)

$$LAB_C = \sum_{i=1}^{8} sign(I_{C_i} - I_{C_0}) \times 2^{i-1} \quad ...(2),$$

where C_i is the adjacent rectangle. The sign function is the same as the one in LBP.

The calculation of LBP is efficient because the feature value in equation (1) is based on binary comparisons and bit-shift operations. Although LAB extends pixels to regions, the computation cost will not increase much because the intensity sum of a rectangle in equation (2) could be efficiently calculated using the integral image [8].

2.2. Boosted Local Binaries

Although the robustness of the LBP and LAB has been demonstrated in some applications, it has certain drawbacks when employed to encode general object's appearance. A notable disadvantage is the insufficient discrimination ability. The feature value of LBP and LAB depend on the intensities of particular locations and thus varies by object's appearance. It will be easily influenced by illumination, occlusion, and noises. In addition, although the size of the rectangles of LAB is flexible, the patterns of local pixel and adjacent rectangles are fixed. It might not have sufficient ability to describe the objects in some complicate detection tasks, e.g, pedestrian.

Now we will introduce the proposed Boosted Local Binaries (BLB). First we define the local neighbor patterns, as shown in Fig. 3. BLB comprises of 8 surrounding rectangles C_1, C_2,…,C_8 and the center rectangle C_0 with the same size. As shown in equation (3), the (x_i, y_i) is the left-top corner of the rectangle, and (w_i, h_i) is the width and height of the rectangles

$$\begin{aligned} &C_i = \{x_i, y_i, w_i, h_i\} \\ &w_1 = w_2 = ... = w_8 \quad i = 0,1,...,8 \quad ...(3). \\ &h_1 = h_2 = ... = h_8 \end{aligned}$$

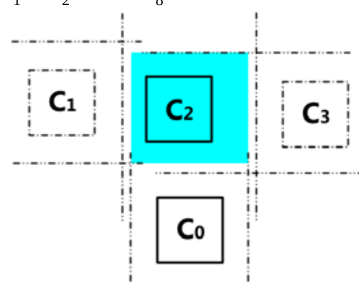

Fig. 3. Pattern of surrounding rectangles in BLB

In BLB, the centers of surrounding rectangles are arranged in a similar order to LBP and LAB, and none of them are overlapped. Fig. 3 shows the enter rectangle C_0, left-top rectangle C_1, top rectangle C_2 and right-top rectangle C_3. C_2 is located at the left side of C_3, and C_1 is located at the left side of C_2. All of these three rectangles lay at the top of C_0. In addition, to keep the neighboring patterns, each neighboring pair (e.g., C_1 and C_2, C_2 and C_0, C_2 and C_3) should have at least 50% overlap on the range of either width or height. In Fig. 3, the possible region for C_2 is the blue region because C_2's width overlaps higher than 50% with C_0's width, and its height also overlaps with C_1's height and C_3's height.

If the surrounding rectangles lay far away from the center rectangle, the feature value will be easily influenced by noises. It means that the classification confidence based on this feature might be lower. So we define the structure diversity of the BLB as

$$D = \sum_{i=1}^{8} \frac{dis(C_i, C_0)}{w_i} \quad ...(4).$$

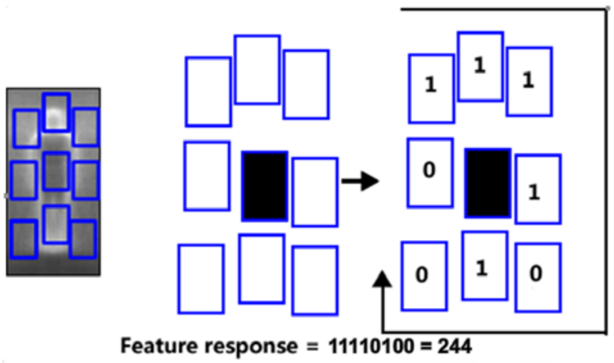

Fig. 4. BLB in gradient image

This factor will be used in the penalty term of the RealAdaBoost procedure.

Besides using these patterns on intensity image, we also apply them on gradient images. In consideration of the efficiency, we generate the x-direction gradient image and y-direction gradient image respectively. Given a BLB feature $F\{C_i = \{x_i, y_i, w_i, h_i\}, i = 0,1,...,8\}$, the feature response is

$$f(F, index) = \sum_{i=1}^{8} sign(g(C_i, index_i) - g(C_0, index_i)) \times 2^{i-1}$$

$$g(C_i, index) = \begin{cases} IntensitySum(C_i) & if \quad index = 0 \\ gradientxSum(C_i) & if \quad index = 1 \\ gradientySum(C_i) & if \quad index = 2 \end{cases} \quad ...(5)$$

The computation cost of BLB is similar to LBP and LAB because the sum of a region in both intensity image and gradient image could be calculated by integral images. An example of the BLB in gradient image is illustrated in Fig. 4.

3. REALADABOOST WITH BLB

In this section, we will introduce how to select the effective BLB features using RealAdaBoost algorithm. In RealAdaBoost, the weak classifier is a function from the feature vector **x** to a real valued object/non-object classification confidence $f(\mathbf{x})$. For the binary classification problem, suppose the training data as $(\mathbf{x}_1, y_1),\ldots,(\mathbf{x}_N, y_N)$, where $\mathbf{x}_i$ is the training sample and $y_i \in \{-1,1\}$ is the class label, we first divide the sample space into N_b equal sized sub-ranges R_j

$$X_j = \{\mathbf{x} \mid f(\mathbf{x}) \in R_j\},\ j = 1,..., N_b \quad ...(6).$$

The weak classifier is defined as a piecewise function

$$h(\mathbf{x}) = \frac{1}{2}\ln(\frac{W_+^j + \varepsilon}{W_-^j + \varepsilon}),\ j = 1,..., N_b \quad ...(7),$$

where ε is the smoothing factor, and $W_\pm$ is the probability distribution of the feature value for positive/negative samples

$$W_\pm^j = P(\mathbf{x} \in X_j, y = \pm 1),\ j = 1,..., N_b \quad ...(8).$$

Parameters:

- N number of training samples
- N_f number of randomly evaluated features in each iteration
- T maximum number of weak classifiers
- a penalty factor on the structure diversity

Input: Training set

$\{(\mathbf{x}_i, y_i)\}\ i = 1,..., N,\ y_i \in \{-1,1\}\}$

1. Initialization

   $\varpi_i = 1/N,\ H(\mathbf{x}_i) = 0,\ i = 1,..., N$

2. Repeat for t = 1,…,T

   2.1 Update the sample weight w_i using the t^{th} weak classifier output $h_t(\mathbf{x}_i)$ (7) as $w_i = w_i e^{-y_i h_t(\mathbf{x}_i)}$

   2.2 For m = 1 to N_f

   For n = 0 to 2

   2.2.1 Generate a random BLB feature F with structure diversity D

   2.2.2 Calculate the feature value $f(F, n)$ according to equation (5)

   2.2.3 Choose the best block based on minimizing Z value in equation (10)

   2.3 Calculate the weak classifier response $h(\mathbf{x}_i)$, update $H(\mathbf{x}_i)$ as $H_{t+1}(\mathbf{x}_i) = H_t(\mathbf{x}_i) + h_t(\mathbf{x}_i)$

3. Output classifier $H(\mathbf{x}) = sign[\sum_{j=1}^{T} h_j(\mathbf{x})]$

Fig. 5. RealBoost training with BLB features

The best weak classifier is selected according to the classification error Z of the piecewise function (7).

$$Z = 2\sqrt{\sum_{j=1}^{N_b} W_+^j W_-^j} \quad ...(9),$$

In consideration of the structure diversity, we add a structure-aware criterion into RealAdaBoost, which is similar to the complexity factor used in selecting the image strip features [17]. The discriminative criterion of RealAdaBoost is shown in equation (10)

$$Z = 2\sum_j \sqrt{W_+^j W_-^j} + a \cdot fp \cdot D \quad ...(10),$$

where D is the structure diversity of the features in equation (4), a is the structure-aware factor to balance the discriminative capability and the feature diversity, fp is the false positive rate of current stage.

The equation (10) could be explained as follows: in the beginning stages of RealAdaBoost, the samples are still easy to be classified, so RealAdaBoost will refer to the features with more confident patterns. In the following stages, the training samples are complicated, so features with diverse patterns might be utilized. The above strategy makes sense, because the overall performance and robustness of a cascade boosted classifier is strongly influenced by beginning stages which filter most of the candidate windows. In our experiment, we set the structure-aware factor a to 0.15.

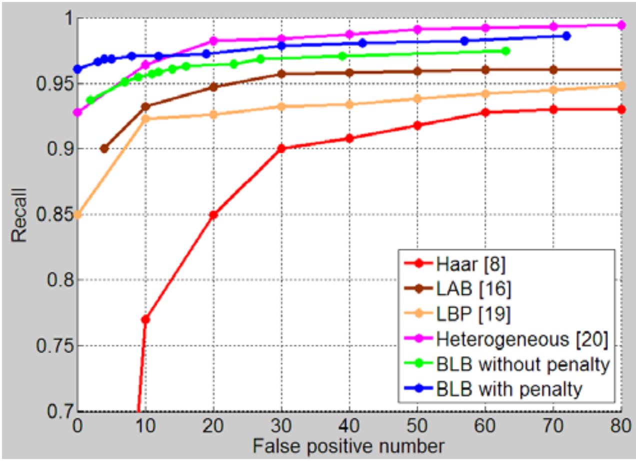

Fig. 6. Experimental results on CMU-MIT frontal face dataset

4. EXPERIMENTS

4.1. Frontal face detection

We train a face detector utilizing the proposed BLB feature and evaluate its performance on CMU + MIT frontal face dataset [18]. This dataset consists of 130 images containing 507 frontal faces with various conditions such as facial expression, occlusion, pose, and scale variations. Most images include more than one face on various backgrounds. Fig. 6 plots the Receiver Operating Characteristics (ROC) curves of our method as well as other popular binary features and the state-of-the-art face detection algorithms, including Haar [8], LBP [19], LAB [16], and heterogeneous features [20], in terms of the number of false positives with respect to the detection rate. As shown in Fig. 6, our detector achieves 96.0% detection rate at 0 false positive. To our knowledge, it is the best result for 0 false positive on the CMU + MIT frontal face dataset. Obviously, compared with other algorithms using single feature, the detection rate of our method is improved dramatically, especially for cases at low false alarms. The performance of our method is also similar to the classifier using heterogeneous features [20], which is trained by kernel SVM. In addition, we find that the boosted classifier with the penalty term (blue curve) on feature diversity achieves better accuracy than the one without penalty term (green curve), especially for lower false positive rate. The reason is that features with larger structure diversity might be easily influenced by the noise, so that the corresponding classifier performs poor on object detection in real images. Some examples of the detection results are shown in the first row of Fig. 11.

4.2. Pedestrian detection

We evaluate the proposed BLB feature using the INRIA pedestrian dataset [1]. The INRIA database contains 1,774 human annotations (3,548 with reflections) and 1,671 person free images. Detection on INRIA pedestrian dataset is challenging since it includes subjects with a wide range of variations in pose, clothing, illumination, background and partial occlusions. In the experiments, we firstly follow the training and testing protocols at patch level proposed by Dalal & Triggs [1]. In Fig. 7, we plot the miss rate tradeoff False Positive rate Per Window (FPPW) curves on a log-log scale by tuning the rejection threshold of the classifiers. We compare the BLB with common used features, HOG (Histogram of Oriented Gradient) [1], and the covariance matrices mapped to Riemann Manifold [2]. It can be seen that the BLB achieves ~2% less miss rates at 10e-4 FPPW comparing to HOG. The accuracy is also comparable with covariance matrix. But BLB's computation cost is much less compared to HOG and covariance matrix. In addition, the performance of the classifiers with penalty term (blue curve) is similar to the one without penalty (green curve) at lower FP. The reason is that FPPW evaluation utilizes patch windows, which is quite different from real detection.

Furthermore, we evaluate our method under the criteria of the detection rate versus False Positive rate Per Image

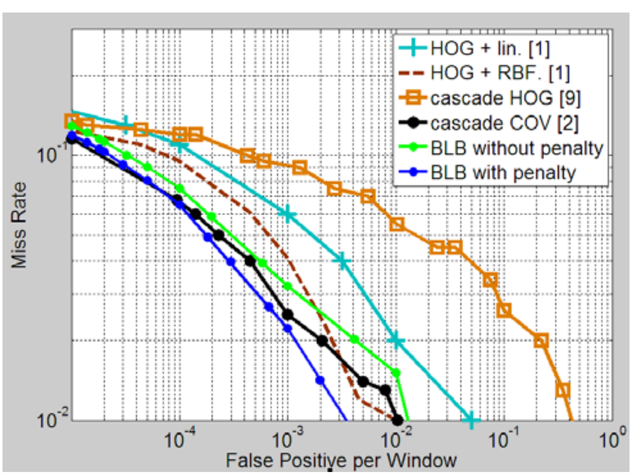

Fig. 7. FPPW evaluation on INRIA pedestrian dataset

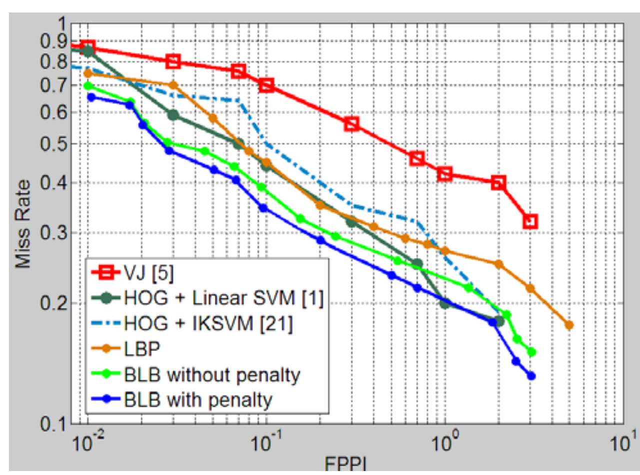

Fig. 8. FPPI evaluation on INRIA pedestrian dataset

(FPPI) [4]. Fig. 8 gives the comparison with the algorithms based on single feature [1, 5, 21] on INRIA dataset. Our algorithm shows competitive result with Haar [5], LBP, and HOG [1][21] feature. Some examples of the detection results are shown in the second row of Fig. 11.

4.3. Side-view car detection

The side-view car detection performance is evaluated on the UIUC car dataset [10]. This dataset contains a single scale test set (170 images with 200 cars), a multi-scale test set (108 images with 139 cars), and a training set of 550 side-view car images. The car patches from the training images are resized to 100×40 pixels and horizontally flipped, so that there are totally 1,100 car patches in the positive training set. We also collect 1,000 images without any cars on the internet as the negative training set.

We compare our method with previous approaches following the Equal Precision and Recall rate (EPR) method. The results are listed in Table 1. It can be seen that the proposed method has high performance competitive to other state-of-the-art methods on both single scale and multi-scale testing sets. Some examples of the detection results are shown in the third row of Fig. 11.

Table 1. Experimental results on UIUC car dataset

Method	Single Scale	Multi Scale
Leibe [22]	97.5%	95.0%
Lowe [23]	99.9%	90.6%
Wu [24]	97.5%	93.5%
Zheng [17]	98.0%	96.0%
Lampert [25]	98.5%	98.6%
Ours no penalty	**99.0%**	**98.6%**
Ours with penalty	**99.3%**	**98.6%**

4.4. Feature analysis

We compare the convergence speed of the training process in INRIA pedestrian dataset. Fig. 9 plots the FPPW against the number of weak classifiers for different methods. The RealAdaBoost with penalty term on the feature diversity is utilized. This figure shows that BLB converges faster, at the rate of approximately two times faster than LAB and LBP. In addition, the performance of boosted classifiers is shown to be positively proportional to the convergence speed in training. This signifies that the proposed BLB performs better on the training accuracy and the training speed of boosted classifiers. Moreover, we test the resulting face detector on a desktop PC with 2.5 GHz I3 CPU and 2 GB memory. It only takes 10ms to detect all faces in a 640×480 input image if the minimum face size is 30×30.

Fig. 10 shows the first selected two features of the faces, pedestrians and cars. These features could well capture the typical structures of the objects. Some of them could not be covered by traditional binary features. For example, the two features in Fig. 10(c) reflect the car wheel and body pattern.

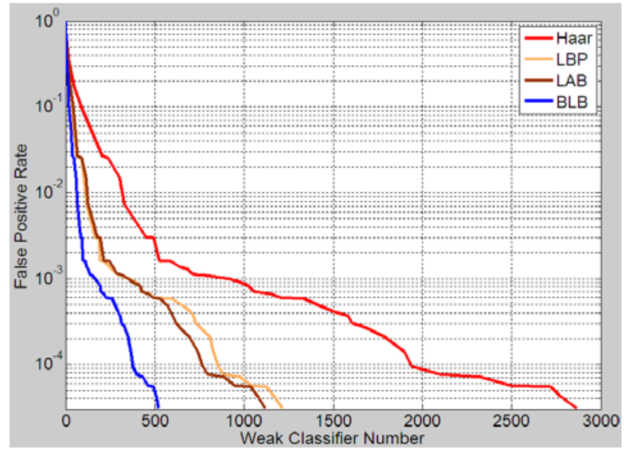

Fig. 9. Convergence speed of the trained classifiers in INRIA

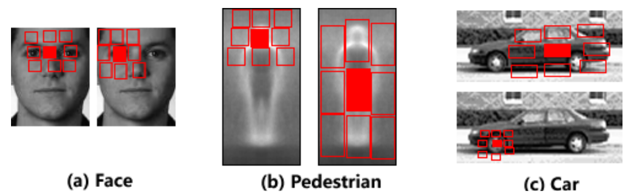

Fig. 10. Selected first two features of faces, pedestrians, and cars

4. CONCLUSION

In this paper, we proposed an object detector that achieves both high accuracy and fast speed. We built a large binary feature pool with variable-location and variable-size blocks for both the intensity domain and the gradient domain. We further utilized RealAdaBoost algorithm to select informative features by evaluating different blocks pairs while considering the structure diversities. Experimental results show that our approach achieves high accuracy on face, pedestrian and car detection tasks. It also speeds up the convergence compared to standard binary features.

5. REFERENCES

[1] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *CVPR*, 2005.
[2] O. Tuzel, F. Porikli, and P. Meer. Pedestrian Detection via Classification on Riemannian Manifolds. In *PAMI*, Vol. 30, Page(s):1713-1727, 2008.
[3] B. Wu and R. Nevatia. Optimizing Discrimination-Efficiency Tradeoff in Integrating Heterogeneous Local Features for Object Detection. In *CVPR*, 2008.
[4] X. Wang, T. Han, and S. Yan. An HOG-LBP Human Detector with Partial Occlusion Handling. In *ICCV*, 2009.
[5] C. Huang, H. Ai, Y. Li, and S. Lao. Vector Boosting for Rotation Invariant Multi-View Face Detection. In *ICCV*, 2005.

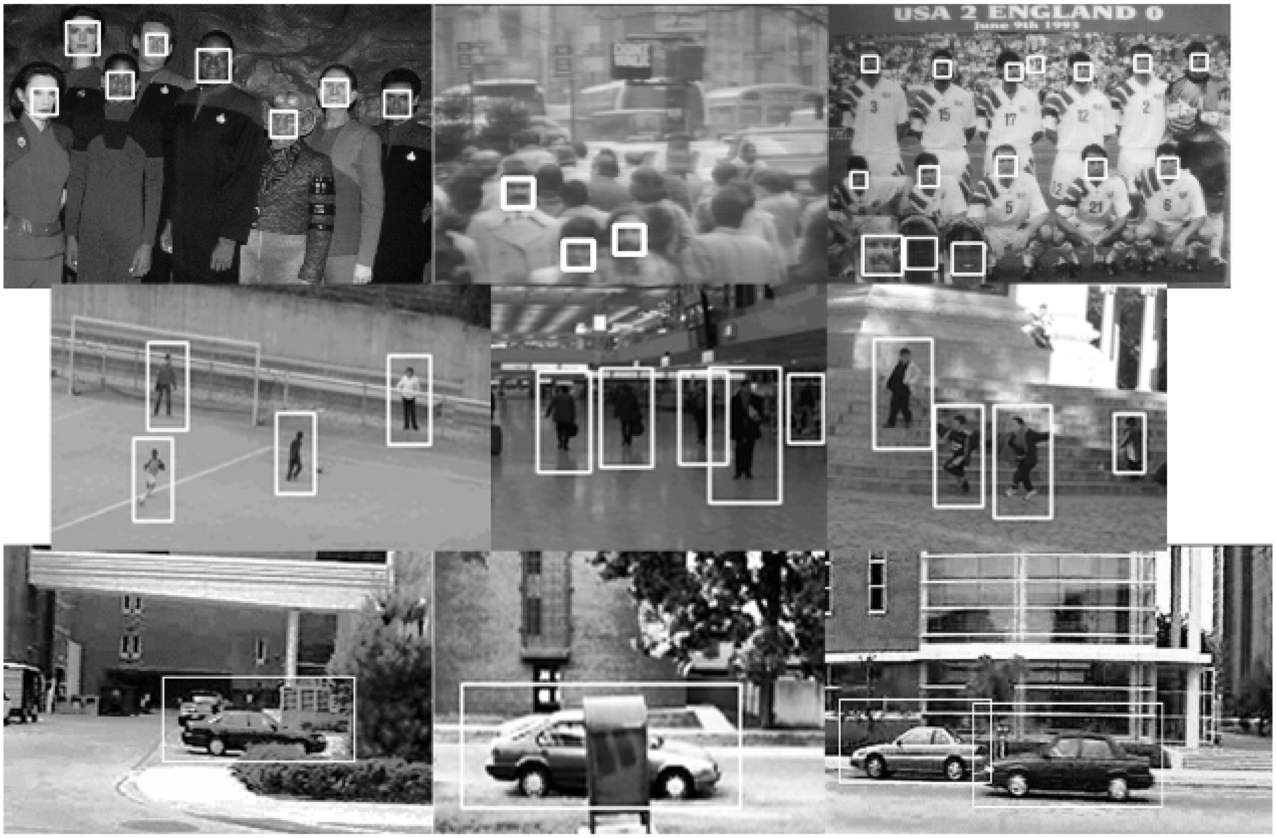

Fig. 11. Detection results. First row – CMU face dataset. Second row – INRIA pedestrian dataset. Third row – UIUC car dataset

[6] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple Kernels for Object Detection. In *ICCV*, 2009.

[7] P. Viola, J. Platt, and C. Zhang. Multiple Instance Boosting for Object Detection. In *NIPS*, 2005.

[8] P. Viola and M. Jones. Rapid Object Detection Using a Boosted Cascade of Simple Features. In *CVPR*, 2001.

[9] Q. Zhu, S. Avidan, M. Yeh, and K. Cheng. Fast Human Detection using a Cascade of Histograms of Oriented Gradients. In *CVPR,* 2006.

[10] S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation. In PAMI, Vol. 26, Page(s) 1475-1490, 2004.

[11] T. Ojala, M. Pietikäinen, and T. Mäenpää. Multi-resolution Gray-scale and Rotation Invariant Texture Classification with Local Binary Patterns. In PAMI, Vol. 24, Page(s): 971–987, 2002.

[12] G. Zhao and M. Pietikäinen. Dynamic Texture Recognition using Local Binary Patterns with an Application to Facial Expressions. In PAMI, Vol. 29, Page(s): 915–928, 2007.

[13] D. Nguyen, Z. Zong, P. Ogunbona, and W. Li. Object Detection Using Non-redundant Local Binary Patterns. In ICIP, 2010.

[14] H. Tang, Y. Sun, B. Yin, and Y. Ge. Expression-robust 3D Face Recognition using LBP Representation. In ICME, 2010.

[15] D. Kim, J. Kwak, B. Ko and J. Nam. Human Detection Using Wavelet-Based CS-LBP and a Cascade of Random Forest. In ICME, 2012.

[16] S. Yan, S. Shan, X. Chen, and W. Gao. Locally Assembled Binary (LAB) Feature with Feature-centric Cascade for Fast and Accurate Face Detection. In CVPR, 2008.

[17] W. Zheng and L. Liang. Fast Car Detection using Image Strip Features. In CVPR, 2009.

[18] H.A. Rowley, S. Baluja, T. Kanade, Neural Network Based Face Detection. In PAMI, Vol. 20, Page(s) 23-38, 1998.

[19] J. Trefny and J. Matas. Extended Set of Local Binary Patterns for Rapid Object Detection. In Computer Vision Winter Workshop, 2010.

[20] H. Pan, Y. Zhu, and L. Xia. Efficient and Accurate Face Detection using Heterogeneous Feature Descriptors and Feature Selection. In CVIU, Vol. 117, Page(s) 12-28, 2013.

[21] S. Maji, A. Berg, and J. Malik. Classification Using Intersection Kernel Support Vector Machines is Efficient. In CVPR, 2008.

[22] B. Leibe, A. Leonardis and B. Schiele. Robust Object Detection with Interleaved Categorization and Segmentation. In IJCV, Vol. 77, Page(s): 259-289, 2008.

[23] J. Mutch and D. G. Lowe. Multiclass Object Recognition with Sparse, Localized Features. In CVPR, 2006.

[24] B. Wu and R. Nevatia. Cluster Boosted Tree Classifier for Multi-View, Multi-Pose Object Detection. In ICCV, 2007.

[25] C. H. Lampert, M. B. Blaschko and T. Hofmann. Beyond Sliding Windows: Object Localization by Efficient Subwindow Search. In CVPR, 2008.