Object Detection Using Generalization and Efficiency Balanced Co-occurrence Features

Anonymous ICCV submission

Paper ID 131

Abstract

In this paper, we propose a high-accuracy object detector based on co-occurrence features. Firstly, we introduce three kinds of local co-occurrence features constructed by the traditional Haar, LBP, and HOG respectively. Then the boosted detectors are learned, where each weak classifier corresponds to a local image region with a co-occurrence feature. In addition, we propose a Generalization and Efficiency Balanced (GEB) framework for boosting training. In the feature selection procedure, the discrimination ability, the generalization power, and the computation cost of the candidate features are all evaluated for decision. As a result, the boosted detector achieves both high accuracy and good efficiency. It also shows the performance competitive with the state-of-the-art methods for pedestrian detection and general object detection tasks.

1. Introduction

Object detection is an indispensable technology in many applications such as artificial intelligence, multimedia systems, and video surveillance. The major problem is that the object appearances vary greatly because of different illuminations, view points, poses, and the presence of occlusions. This has motivated the invention of various approaches. Among them, a commonly used paradigm is to train a boosted classifier based on local features [14][26]. For example, Viola et al. [26] used AdaBoost algorithm to train a cascade classifier based on the Haar feature. Zhang et al. [36] proposed an improved version of the Haar feature based on up-right human body to construct a cascade pedestrian detector. Dollar et al. [7] enabled neighbouring detectors to communicate by a proposed crosstalk cascade for pedestrian detection.

Boosting family algorithms achieve considerable performance for some object detection tasks. However, since the difficulty of the training samples increases stage by stage, it gets more and more difficult to find appropriate features to describe the object characteristic effectively. For those more complicated objects such as multi-view and multi-pose pedestrian, the problem becomes much more serious that in later training rounds the classification task may be beyond the descriptive ability of traditional features [31]. As a result, many researchers propose to use more powerful features, such as high-order gradient features, heterogeneous features and feature fusion. Recently, the co-occurrence features [18][20][35] have become a hot topic. The co-occurrence information extracted by these features are able to capture some complicate object characteristics. Unfortunately, they also lead to heavy computation cost because the dense feature vector is time-consuming to calculate. In addition, we know that the performance of an object detector is decided not only by the discrimination ability of the features, but also by their generalization power, which is defined as the ability to deal with the cases that are not part of the training process. Due to the fact that some diverse co-occurrence patterns are sensitive to background noise, most of the co-occurrence features have poor generalization power. The detector succeeding in one scene might fail in another scene with different conditions, such as pose, illumination, etc. This is actually a trade-off problem. Although using stronger features may contribute to the training accuracy, it will increase the risk of both low generalization power and high computation cost.

The major contributions of this paper are two folds. Firstly, we design a set of localized co-occurrence features which can be computed efficiently. Three kinds of co-occurrence features, CoHaar, CoLBP, and CoHOG are constructed. In addition, a new Generalization and Efficiency Balanced (GEB) framework is proposed, which is utilized to evaluate the accuracy, efficiency, and robustness of different weak classifiers at the same time. As a result, the boosted detector based on GEB not only achieves high accuracy, but also has good generalization power and considerable efficiency. The experiments on public datasets show that our method achieves competitive performance with the state-of-the-art approaches in both pedestrian and general object detection tasks.

2. Related Work

In object detection, utilizing appropriate machine learning methods with discriminative local features is a commonly-used framework. Many local features were proposed for various object detection tasks. Most of them reflect the characteristic of some pre-defined local patterns, for example, Haar [26][36], SIFT [12][13], HOG [5], and covariance matrix [17][23]. Lowe [13] designed the Scale Invariant Feature Transform (SIFT) for object recognition. Dalal & Triggs [5] proposed the basic form of the HOG with 2×2 cells. Multi-size versions were developed in [1][37], and further extended to pyramid structure [4][6][15][33]. Tuzel et al. [25] utilized the covariance matrix projected on Riemann manifolds for detection. Sometimes these features are combined with each other to enhance the discriminative power. For instance, Levi et al. [11] utilized an accelerated version of the feature synthesis method. Paisitkriangkrai et al. [21] built features on the basis of low-level visual features combination and spatial pooling, which improved the translational invariance and thus the robustness of the detection process.

Recently, many co-occurrence features are proposed. According to whether the spatial neighbouring relationship among features is used in computing the co-occurrence statistics, existing co-occurrence features can be sorted into two categories: global co-occurrence features and local co-occurrence features. In [35], Yuan et al. proposed to mine the co-occurrence statistics of SIFT words for visual recognition. Rasiwasia et al. [19] calculated the co-occurrence information for every pixel in the whole image. These works fall into the category of global co-occurrence features. In [18][20][34][38], the spatial co-occurrences are computed within locally adjacent neighbours instead of on the whole image. Mita et al. [38] designed a face detector based on co-occurrence of multiple Haar-like features. Ren et al. [20] utilized the local co-occurrence of gradient orientation to build a co-occurrence HOG histogram for object detection. Xu et al. [32] designed a co-occurrence LBP feature which detected co-occurrence orientation through gradient magnitude calculation. A rotation invariant version was proposed by Nosaka et al. [16] for texture classification and face recognition, and further improved by Qi. et al. [18]. Most of the above co-occurrence features are designed for specific object categories. Dense co-occurrence patterns and high dimensional vectors are utilized, so that the generalization power and efficiency of the resulting detectors may be relatively low. Few of them work well in the general object detection task.

Boosting framework is widely used in training the cascade classifier for fast object detection. The cascade classifier is well performed on the object classes with small intra-class variation, e.g., the frontal-view faces or side-view cars. To strengthen the classification ability, some previous approaches follow the divide-and-conquer strategy to build strong classifiers with more complicate structures. For example, Wu et al. [30] proposed the cluster boosted tree method, in which the sample space is divided by unsupervised clustering based on discriminative image features. Heng et al. [10] proposed a shrink boost method solving a sparse regularization problem with the boosting step for weak classifier construction, and the shrinkage step for feature dimension reduction. These algorithms emphasize the discrimination ability more than other factors, so that they will increase the computation complexity of both the training and testing procedure. Designing an effective framework to solve this trade-off problem is necessary.

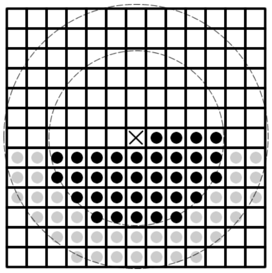

Figure 1. The pixel pairs in co-occurrence patterns. Each black pixel and center pixel correspond to a pixel pair. The highlight parts show the pairs with offsets $U \leq 4, V \leq 4$.

3. Co-occurrence features

3.1. Co-occurrence patterns

The co-occurrence features can be constructed based on the statistics information of several pre-defined *co-occurrence patterns*. Each co-occurrence pattern $\{U, V, F_1, F_2\}$ is a comparison between *pixel pair* $a = \{x_1, y_1, f_1\}$ and $b = \{x_2, y_2, f_2\}$ satisfying the following constraint

$$|x_1 - x_2| = U, y_1 - y_2 = V, f_1 = F_1, f_2 = F_2. \quad (1)$$

In (1), the $(x_1, y_1), (x_2, y_2)$ are the coordinates of a and b. The *offset* $U, V \geq 0$ show the spatial distance of pixel a and b. f_1, f_2 are scores of a and b generated by feature extraction algorithms. F_1, F_2 are constants in the score space F. As shown in Fig. 1, each black pixel and the center pixel correspond to the pixel pair of a co-occurrence pattern with $U \leq 4, V \leq 4$.

To compute a stable distribution that is robust against noise, we utilize the histogram based on the division of score space F as the co-occurrence feature vector. Given an input window R, the offset U, V, and an extraction method to generate F, we divide F into n bins $\{F_1, \ldots, F_n\}$. The co-occurrence feature C is a n^2 dimension vector, where each dimension $c_{i,j}$ is the number of the pixel pairs in R satisfying the co-occurrence pattern (U, V, F_i, F_j)

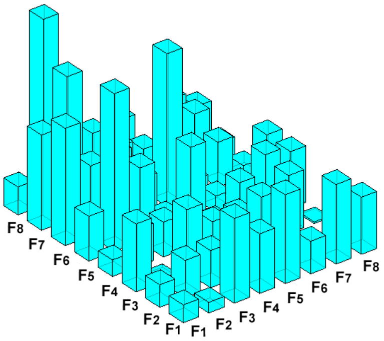

Figure 2. 8 × 8 co-occurrence histograms.

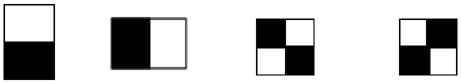

Figure 3. Haar features. Black regions have -1 weight, and white regions have +1 weight.

$$C(U,V) = [c_{1,1}, c_{1,2}, \dots, c_{1,n}, \dots, c_{n,n}], \qquad (2)$$
$$c_{i,j} = Count(U,V,F_i,F_j) \; in \; R, \; 1 \le i,j \le n.$$

As shown in Fig. 2, the two axes correspond to the divided score space F, and the co-occurrence feature vector has $8 \times 8 = 64$ dimensions.

Compared to the covariance matrix, the co-occurrence features utilize the co-occurrence histogram to describe the distribution of object characteristics instead of the covariance. Our co-occurrence features are based on single co-occurrence pattern, so the extraction will be relatively fast if we adopt efficient methods to generate F. In our case, we utilize the methods inherited from Haar, LBP, and HOG to construct the CoHaar, CoLBP, and CoHOG features respectively.

3.2. CoHaar feature

Haar-like features [26], shown in Fig. 3, consist of two or more rectangular regions enclosed in a template. Such features produce a feature value as

$$F = \sum_{t=1}^{l} w_t I_t, \qquad (3)$$

where t iterates through all l rectangles, the I_t represents the mean intensity of the pixels enclosed within the tth rectangle. Every rectangle in the Haar feature is assigned a weight that is represented by w_t. The weights are set such that $\sum_{t=1}^{l} w_t = 0$ is satisfied. The computation of Haar feature is quite efficient because the intensity sum in any rectangles can be easily calculated by the integral image [26].

To construct the CoHaar feature, we extend the Haar feature extraction to the gradient domain. In consideration of the efficiency, we utilize the x and y directional gradient image respectively. The F of CoHaar feature in (2) is replaced with the Haar feature extraction (3) on the intensity domain, gradient-x domain or gradient-y domain. We quantize F to $n = 8$ bins, so the CoHaar feature dimension is $8 \times 8 = 64$. Given an input window R and an indicator k, the CoHaar feature is formulated as

$$CoHaar(U,V,k) = [c_{1,1}, c_{1,2}, \dots, c_{1,8}, \dots, c_{8,8}] \qquad (4)$$
$$c_{i,j} = Count(U,V,F_i,F_j) \; in \; R$$
$$F = \sum_{t=1}^{l} w_t I_t(k), 1 \le i,j \le 8,$$

where F_i, F_j are the quantized Haar feature response, k ranges from 0 to 2, $I(k)$ is the intensity sum when $k = 0$, the gradient sum on gradient-x image when $k = 1$, and on gradient-y image when $k = 2$.

3.3. CoLBP feature

The traditional LBP is developed for texture classification and the success is due to its robustness under illumination variations, computational simplicity and discriminative power on specific patterns. Fig. 4 represents an example of the traditional LBP and its extension. LBP is a binary coding of the intensity contrast of the center pixel/region and 8 neighbouring pixels/regions. If the intensity of the neighbouring pixels/regions are higher than the center one, the corresponding bit will be assigned 1, otherwise it will be assigned 0. Given a center pixel e, the LBP feature response is defined by

$$LBP_{d,r} = \sum_{i=1}^{d} sign(I_i - I_e) \times 2^{i-1}, \qquad (5)$$

where d is the number of neighbouring pixels/regions, r is the distance between the neighbouring pixels/regions and the center one, I is the sum of intensity.

Uniform LBP is a subset of LBP, defined by Δ in (6), which shows the number of bitwise transitions from 0 to 1 or vice versa when the bit pattern is considered circular

$$\Delta(LBP_{d,r}) = |sign(I_{d-1} - I_e) - sign(I_0 - I_e)|$$
$$+ \sum_{i=1}^{d-1} |sign(I_i - I_e|) - sign(I_{i-1} - I_e). \qquad (6)$$

Fig. 5 shows all uniform patterns for $LBP_{8,1}$. The binary patterns are reduced to 59, where all the non-uniform patterns are merged into another pattern.

Ojala et al. [24] has shown that over 90% local structures belong to uniform patterns when using the parameter

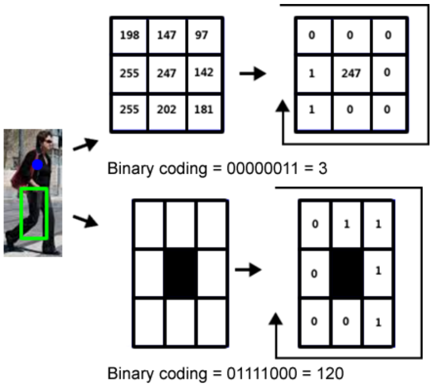

Figure 4. Traditional $LBP_{8,1}$ feature and its variant.

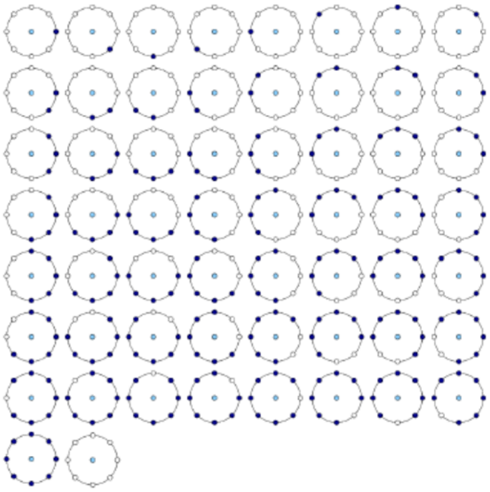

Figure 5. 58 uniform patterns of $LBP_{8,1}$ feature. Each row corresponds to a cluster in the CoLBP extraction.

of $d = 8, r = 1$. Thus, the value calculated by uniform LBP is more stable and less prone to noise. Then we propose the CoLBP based on these uniform patterns. Similarly, the LBP extraction (5) is applied on both the intensity and gradient domain. In consideration of the rotation invariance, we merge the 58 uniform $LBP_{8,1}$ patterns to 8 clusters based on the number of '1' values, shown as the 8 rows in Fig. 5. All the non-uniform patterns construct another cluster. As a result, the LBP response space is divided into $n = 9$ bins, so the CoLBP histogram consists of $9 \times 9 = 81$ dimensions. Given an input window R and an indicator k, the CoLBP feature vector is generated by

$$\begin{gathered} CoLBP(U,V,k) = [c_{1,1}, c_{1,2}, \ldots, c_{1,9}, \ldots, c_{9,9}] \qquad (7) \\ c_{i,j} = Count(U,V,F_i,F_j) \ in \ R \\ F = LBP_{d,r,k}, 1 \le i,j \le 9, \end{gathered}$$

where F_i, F_j are the cluster number of LBP response F, $LBP_{d,r,k}$ is the LBP response (5) on intensity image when $k = 0$, on gradient-x image when $k = 1$, and on gradient-y image when $k = 2$.

3.4. CoHOG feature

Histogram of Oriented Gradient (HOG) breaks the image region into a cell-block structure and generates histogram based on the gradient orientation and spatial location. Watanabe et al. [29] proposed a dense version extracting all possible co-occurrence patterns of the gradient orientation in the whole image, which is rather time consuming. Instead, we build our CoHOG based on single co-occurrence pattern. The gradient orientation on both the intensity domain and the gradient domain are utilized as the F in CoHOG feature and further quantized to 8 bins. Therefore, there are $8 \times 8 = 64$ elements in the co-occurrence histogram. Given an input window R and an indicator k, the CoHOG histogram is formulated as

$$\begin{gathered} CoHOG(U,V,k) = [c_{1,1}, c_{1,2}, \ldots, c_{1,8}, \ldots, c_{8,8}] \qquad (8) \\ c_{i,j} = Count(U,V,F_i,F_j) \ in \ R \\ F = GradientOrientation_k, 1 \le i,j \le 8, \end{gathered}$$

where F_i, F_j are the quantized gradient orientation, F is the gradient orientation on original image when $k = 0$, the gradient orientation on gradient-x image when $k = 1$, and on gradient-y image when $k = 2$.

4. Generalization and Efficiency Balanced Framework

In this section, we will introduce the proposed Generalization and Efficiency Balanced (GEB) framework based on RealAdaBoost algorithm [22]. For the binary object/background classification problem, denote the input 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. Each co-occurrence feature can be seen as a function from the image space to a real valued range $f : \mathbf{x} \rightarrow [f_{min}, f_{max}]$. We divide the sample space into N_b equal sized sub-ranges B_j, the weak classifier is defined as a piecewise function

$$h(\mathbf{x}) = \frac{1}{2} ln(\frac{W_+^j + \epsilon}{W_-^j + \epsilon}), \qquad (9)$$

where ϵ is the smoothing factor, $W_\pm$ is the probability distribution of the feature response for positive/negative samples, implemented as a histogram

$$W_\pm^j = P(\mathbf{x} \in B_j, y \in \{-1, 1\}), j = 1, \ldots, N_b. \qquad (10)$$

The best feature is selected according to the classification error Z of the piecewise function (11). Better features lead to lower Z

$$Z = 2\sum_{j}\sqrt{W_{+}^{j}W_{-}^{j}}. \tag{11}$$

If the discriminative ability is the only objective, using the features minimizing (11) seems to be a good idea. In our case, the generalization power and computation cost are also considered. Firstly we discuss the influence of the generalization power. The classification margin of the weak classifier h on $\mathbf{x}$ is $y \cdot h(\mathbf{x})$, where the h is normalized to $[-1, 1]$. This margin represents the classification ability of the classifier. Larger margins imply lower generalization error [22]. In addition, using the co-occurrence features, if the pixel pair in the co-occurrence patterns lay far away from each other, the feature response will be influenced by noises because these two pixels might have few contextual relationship. So we define the term used to evaluate the influence of the generalization power as

$$S(h, \mathbf{x}) = \frac{y \cdot h(\mathbf{x})}{s}, \tag{12}$$

where s is a parameter related with the offsets (U, V) in co-occurrence pattern, calculated as

$$s = \begin{cases} 1 & max(U,V) \le \delta \\ 1.5 - \dfrac{1}{1+\exp^{\delta - max(U,V)}} & max(U,V) > \delta \end{cases}.$$

This equation means that we believe the co-occurrence patterns within δ pixels offset are confident. We set $\delta = 4$ because in our experiments, the accuracy of the detectors trained on larger offset features are lower. (refer to Section 5.2 for details). Balancing the generalization power and the discrimination ability requires us to evaluate both (12) and (11). So we add a generalization penalty term into (11), where α is the generalization-aware factor

$$Z = 2\sum_{j}\sqrt{W_{+}^{j}W_{-}^{j}} - \frac{\alpha}{n \cdot s}\sum_{i=1}^{n} y \cdot h(\mathbf{x}_i). \tag{13}$$

If the confidence of the selected feature is lower, which corresponds to a smaller margin and larger s. Then the second term will be smaller, and Z will be larger. So this feature will have less probability to be selected.

Then we discuss the influence of the computation cost. In real object detection, an object detector will go around the input image to check every candidate detection window. The number of the false positive windows is far larger compared to the true positive windows, especially at the beginning stages. As a result, the execution time of the whole detection procedure mainly depends on the number of false positive windows

$$T \approx \sum_{i=1}^{l} N_{neg,i}t_i, \tag{14}$$

where l is the stage number, N_{neg} is the number of false positive windows, t is the computation cost of the weak classifiers. Because N_{neg} depends on the current false positive rate, (14) is equal to (15), where N is the total window number, fp_i is the false positive rate of the ith stage

$$T \approx \sum_{i=1}^{l} N fp_i t_i = N\sum_{i=1}^{l} fp_i t_i. \tag{15}$$

Then we add another term into (13) as

$$Z = 2\sum_{j}\sqrt{W_{+}^{j}W_{-}^{j}} - \frac{\alpha}{n \cdot s}\sum_{i=1}^{n} y \cdot h(\mathbf{x}_i) + \beta \cdot fp \cdot t, \tag{16}$$

where the β is the efficiency-aware factor. The trade-off of discrimination ability and efficiency in (16) can be explained as follows: in the beginning stages of RealAdaBoost, because the false positive rate is larger, and the target object is still easy to be classified with the background, so RealAdaBoost will refer to efficient features. In the following stages when the false positive rate is smaller and the problem becomes more difficult, the features with higher computation cost will be considered. This strategy makes sense, because the overall efficiency of a cascade boosted detector is mainly influenced by the beginning stages, which filter most of the negative windows. Using efficiency features in the beginning stages clearly contributes to the overall efficiency.

With the GEB framework, the training procedure is illustrated in Fig. 6. To learn the best feature, the most intuitive way is to look through the whole feature pool, which is rather time consuming. So we sample $M = 60$ windows per iteration to speed up the feature selection process. The offsets (U, V) range from (1,1) to (15,30). $O = 15$ offsets are sampled per window, while at least 5 of them are within $(4, 4)$. For CoHaar feature, the four patterns illustrated in Fig. 3 are utilized, and the block size of single Haar feature ranges from 4×4 to 20×20. For CoLBP feature, the block size is set from 1×1 (traditional LBP) to 8×8 (LBP variant). After picking a window and an offset, a random co-occurrence feature is generated and evaluated according to (16). We set the computation cost of CoHaar to 2, CoLBP to 4, and CoHOG to 10. The generalization-aware factor α is set to 0.1, and the efficiency-aware factor β is set to 0.15, which is decided by the experimental results of several detectors with different parameters trained on Caltech database.

In the training process, the first bootstrap will be called when 50% of the negative samples are filtered by current

```
Parameters
  N    number of training samples
  M    number of evaluated windows each iteration
  O    number of evaluated offsets each iteration
  L    maximum number of weak classifiers
Input: Training set $\{(\mathbf{x}_i, y_i)\}, y_i \in \{-1, 1\}$
1. Initialize sample weight and classifier output
   $w_i = 1/N, H(\mathbf{x}_i) = 0$
2. Repeat for $l = 1, 2, \ldots, L$
  2.1 Update the sample weight $w_i$ using the $l^{th}$ weak
   classifier output $w_i = w_i e^{-y_i h_l(\mathbf{x}_i)}$
  2.2 For $m = 1$ to M
      For $o = 1$ to O
       For $k = 0$ to 2
        2.2.1 Generate a random $R$ and $(U, V)$
        2.2.2 Calculate feature response $C(U, V, k)$ on $R$
        2.2.3 Build the $W_+$ and $W_-$ (10)
        2.2.4 Select the best feature minimizing $Z$ (16)
        2.2.5 If the fp is lower than $10^{-6}$, break
  2.3 Update weak classifier $h_l(x)$ using (9)
  2.4 Update strong classifier $H_{l+1}(\mathbf{x}_i)$
3. Output classifier $H(\mathbf{x}_i) = sign[\sum_{j=1}^{l} h_j(\mathbf{x})]$
```

Figure 6. Selecting co-features using RealAdaBoost with GEB.

strong classifier. Then new samples are bootstrapped to replace the filtered negative samples, and the training is ongoing. Every time 50% of the negative samples are filtered, the bootstrap will be called. This procedure is repeated until the overall fp is lower than 10^{-6} or the number of weak classifiers exceeds L.

5. Experimental Results

5.1. Datasets

We evaluate the proposed method on pedestrian detection and general object detection tasks. For pedestrian detection, the INRIA dataset and Caltech dataset are utilized. The INRIA dataset [5] contains 1,774 human annotations (3,548 with reflections) and 1,671 person free images, while the Caltech dataset [8] consists of about 250,000 frames with a total of 350,000 bounding boxes and 2,300 unique pedestrians are annotated. The individuals in these datasets appear in many positions, orientations, and background variety. We use 64×128 pedestrians and co-occurrence windows from 6×6 to 56×112. The locations of the window centers are sampled every 4 pixels. As a result, this will generate $7,997$ different windows. The evaluation is based on the detection rate versus False Positive rate Per Image (FPPI) [27].

For general object detection, the standard benchmark dataset PASCAL VOC 2007, is employed. This dataset contains images from 20 different categories with about 5,000 images for training and validation, and a test set of size about 5,000 images. For the aeroplane, bird, bottle, chair, diningtable, person, pottedplant, sofa, and TV monitor category, all samples are used together to train a single detector. For all other categories, the training samples are divided into the front/rear view samples and side-view samples according to the aspect ratio. Then two detectors are trained respectively. The final detection result is based on merging the outputs of these two detectors . The object size (w, h) used to train these detectors are listed in the second column of Table 2. The co-occurrence window size ranges from 4×4 to $w' \times h'$, where w', h' are the maximum multiple of 4 smaller than $w - 8$ and $h - 8$. As a result, the number of the co-occurrence windows ranges from 2,546 to 8,022. The detection performance are measured using the average precision (AP). A detection result is considered as correct if it has an intersection-over-union ratio of at least 50% with a ground-truth object instance.

5.2. Comparison with different co-occurrence features and feature combinations

We first evaluate the performance of different co-occurrence features with conventional RealAdaBoost on INRIA dataset. Fig. 7(a) illustrates the performance of the boosted detectors with different co-occurrence features and traditional features. It can be seen that all the detectors with co-occurrence features clearly outperform the detectors with corresponding traditional features. The CoHOG detector performs better than CoLBP and CoHaar detector, which shows that gradient orientation co-occurrence is more discriminative compared to intensity and gradient magnitude co-occurrence. In addition, we notice that the accuracy of the detectors trained on larger offset (> 4) CoFeatures are lower, which explains why we use $\delta = 4$ in the generalization penalty term of GEB. In fact, 90% offsets of the selected CoFeatures in 'CoX (All offsets)' curves are within (4,4). Compared to the existing co-occurrence features, our co-occurrence features are histograms of quantized feature response on selected co-occurrence patterns, which is more discriminative than the combination of Haar responses (JointHaar [38]). In addition, using dense feature vector (CMLBP [16], dense CoHOG [29]) might lead to the dimension redundant. So the accuracy of our co-occurrence features are better.

Next, we compare the combination of the proposed 3 co-occurrence features with other feature combinations. From Fig. 7(b) we could find that the combination of the co-occurrence features clearly shows better accuracy compared to the combination of low-level features 'Haar+LBP+HOG'. The groups of 'X+Co-X' achieve slightly better compared to using 'Co-X', which is still lower than the combination of CoFeatures. The best one among these groups, which combines all co-occurrence feature together, achieves 15% average miss rate. It is a signif-

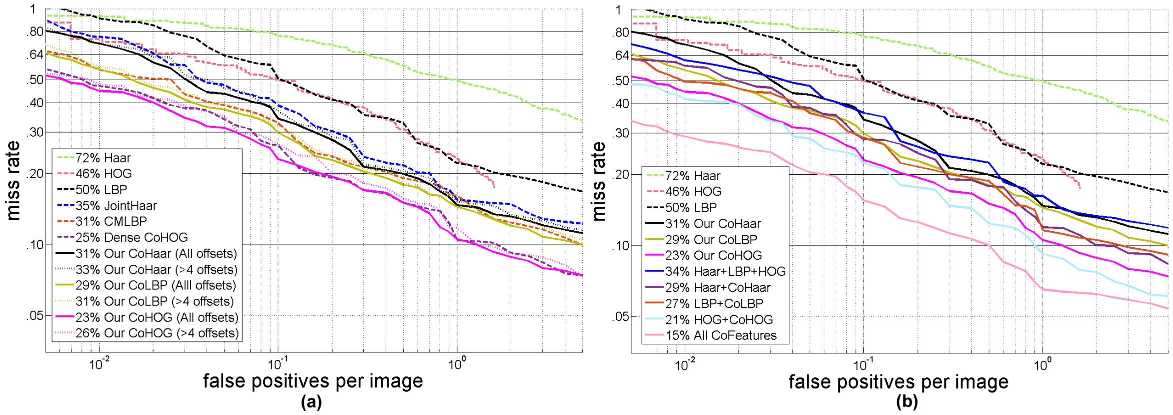

Figure 7. Comparison of different co-occurrence features and feature combinations on INRIA pedestrian dataset.

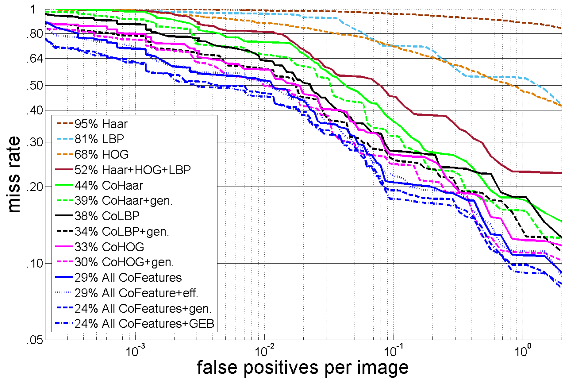

Figure 8. Experiments of GEB framework on Caltech dataset.

icant improvement compared to using single co-occurrence feature. These results reflect the advantange of using multiple co-occurrence features in pedestrian detection.

5.3. Experiments on the GEB framework

Then we conduct the experiments on Caltech dataset to show the effectiveness of the GEB framework. Fig. 8 gives the results of the conventional detectors, the detectors with the generalization penalty (gen.) and efficiency penalty (eff.) respectively, and using the whole GEB. Firstly we notice that the proposed co-occurrence features also work better compared to traditional features and their combinations on Caltech database. Using the generalization penalty, the accuracy is improved at least 3% for both single co-occurrence feature and its combinations. Using the efficiency penalty term will not influence the accuracy very much. We know that in image-based object detection, the overall accuracy is decided not only by the discriminative ability of the detector, but also by the generalization power. So using the GEB framework to balance them could contribute to the accuracy of the resulting classifier.

Moreover, We test the resulting classifiers on a desktop dual core I7 PC with 8 GB memory. The average execution speed for 640×480 images is shown in Table 1. It could be seen that although CoHOG has better discrimination power compared to CoHaar and CoLBP, but the computation is relatively slow because there is no efficient implementation on the algorithm level to get the gradient orientation co-occurrence. If we combine these features together and use the GEB framework, the execution time is significantly reduced from 82.2ms per frame to 50.8ms per frame. These results show the effectiveness of the GEB framework on improving the efficiency.

Table 1. Execution speed of the detectors on Caltech database.

Approach	Detection time(ms) per frame
CoHaar	27.5
CoLBP	36.5
CoHOG	89.4
All features	82.2
All features+GEB	50.8

5.4. Comparison with the state-of-the-art

Furthermore, we compare our results with the state-of-the-art on pedestrian detection in Fig. 9. We notice that using the combination of all 3 co-occurrence features, the accuracy is much better than some boosting family methods (Multiftr, FPDW, pAUCBoost, FisherBoost, Crosstalk), but the detectors with single co-occurrence feature achieve lower accuracy compared to the state-of-the-art (SketchTokens, Spatialpooling, LDCF, ACF-Caltech+). The reason is that the proposed co-occurrence features are extracted on single scale, so that the discriminative ability might be lower compared to the evolution of channel features, such as the dense sampled multi-scale feature in ACF-Caltech+ [15], or the decorrelated features in LDCF [15]. But this gap can be compensated by the combination of multiple co-occurrence features selected by GEB framework. As a result, the 'All CoFeatures+GEB' detector achieves competitive accuracy with the state-of-the-art on both two datasets. In addition, our detector is also better than the MOCO

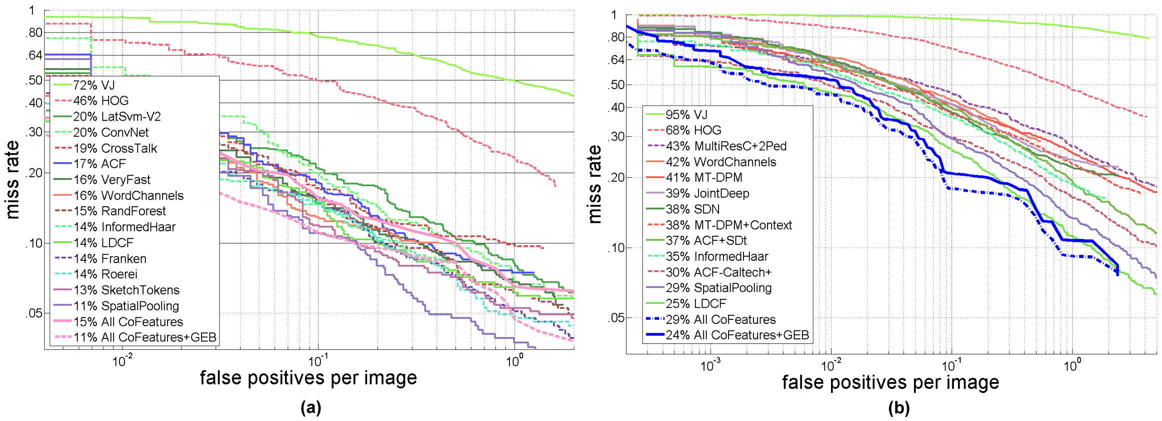

Figure 9. Comparison with the state-of-the-art algorithm. (a) INRIA. (b) Caltech.

[2] (46% on Caltech) , which is a combination of zero-order, first-order, and second-order co-occurrence information. The major advantage of our method is that we select the meaningful co-occurrence patterns by RealAdaBoost, while MOCO uses all possible patterns with Latent SVM. There will be some redundant information in the dense feature vector.

In Table 2, we compare our method with the state-of-the-art using local features [2][3][9][28] on PASCAL VOC 2007 dataset, in terms of detection AP on the test set. Firstly, it could be seen that the mAP of using all CoFeatures is 0.408, which is better compared to using single co-occurrence feature. So combining boosted co-occurrence features are also effective for general object detection. In addition, we notice that the co-occurrence features based on binary information (CoHaar, CoLBP) work relatively well on some object categories with specific structural information, such as the pottedplant with a consistent base, or the chair which consists of several rigid parts. In this case, such co-occurrence features are easier to capture the binary information. In contrast, the gradient information based CoHOG works better on the object categories with complicate appearance such as sheep or sofa. Compared to the state-of-the-art, the combination of the CoFeatures are more effective than pyramid HOG [9], MOCO [2] and SIFT fisher vectors [3]. Using the GEB process, the mAP is further improved to 43.7%, which is better than heterogeneous features [28] including multi-scale HOG and covariance matrix. It also implies that using the combination of co-occurrence features is better compared to the combination of traditional features.

6. Conclusion

In this paper, we show that using the co-occurrence features for object detection is effective. Three kinds of co-occurrence features are proposed based on the traditional Haar, LBP, and HOG, and further combined to train an object detector. In addition, we design a GEB framework which balances the discriminative ability, generalization power, and computation cost for boosted detector. As a result, the boosted detector not only achieves high accuracy, but also is computed efficiently. The experimental results on INRIA, Caltech, and PASCAL VOC 2007 dataset show the effectiveness of our method.

References

[1] R. Benenson, M. Mathias, T. Tuytelaars, and L. Van Gool. Seeking the strongest rigid detector. In *CVPR*, 2013. 2

[2] G. Chen, Y. Ding, J. Xiao, and T. X. Han. Detection evolution with multi-order contextual co-occurrence. In *CVPR*, 2013. 7, 8, 9

[3] R. G. Cinbis, J. Verbeek, and C. Schmid. Segmentation driven object detection with fisher vectors. In *ICCV*, 2013. 8, 9

[4] A. Costea and S. Nedevschi. Word channel based multiscale pedestrian detection without image resizing and using only one classifier. In *CVPR*, 2014. 2

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005. 2, 6

[6] P. Dollár, R. Appel, S. Belongie, and P. Perona. Fast feature pyramids for object detection. In *PAMI*, 36:1532–1545, 2014. 2

[7] P. Dollár, R. Appel, and W. Kienzle. Crosstalk cascades for frame-rate pedestrian detection. In *ECCV*. 2012. 1

[8] P. Dollar, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: An evaluation of the state of the art. In *PAMI*, 34:743–761, 2012. 6

[9] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. In *PAMI*, 32:1627–1645, 2010. 8, 9

[10] C. K. Heng, S. Yokomitsu, Y. Matsumoto, and H. Tamura. Shrink boost for selecting multi-lbp histogram features in object detection. In *CVPR*, 2012. 2

[11] D. Levi, S. Silberstein, and A. Bar-Hillel. Fast multiple-part based object detection using kd-ferns. In *CVPR*, 2013. 2

Table 2. Average precision(%) on PASCAL VOC 2007 dataset. In the second column, the 'f' means the sample size for front/rear images.

	Sample size	CoHaar+GEB	CoLBP+GEB	CoHOG+GEB	**All**	**All+GEB**	[2]	[3]	[28]	[9]
aerop.	128x64	30.1	39.7	42.3	51.5	**56.7**	41.0	56.1	54.2	36.6
bicycle	128x64 40x100(f)	45.5	51.2	59.9	59.7	61.6	**64.3**	56.4	52.0	62.2
bird	100x100	22.3	**26.3**	25.4	24.8	26.0	15.1	21.8	20.3	12.1
boat	100x100	16.3	21.6	**27.0**	24.2	26.6	19.5	26.8	24.0	17.6
bottle	40x120	18.9	25.9	25.1	28.1	30.9	**33.0**	19.9	20.1	28.7
bus	128x64 100x100(f)	31.3	43.9	52.2	55.9	**58.2**	57.9	49.5	55.5	54.6
car	100x60 100x100(f)	46.2	58.4	55.9	62.6	66.5	63.2	57.9	**68.7**	60.4
cat	100x60 100x100(f)	27.2	38.6	38.0	41.9	44.2	43.8	**46.2**	42.6	25.5
chair	100x100	19.1	**24.6**	23.0	23.7	24.5	23.2	16.4	19.2	21.1
cow	100x60 60x100(f)	22.5	29.5	36.9	37.7	40.9	28.2	41.4	**44.2**	25.6
dt.	100x60	26.0	33.4	41.7	43.8	44.0	29.1	47.1	**49.1**	26.6
dog	100x60 80x100(f)	18.6	21.8	24.9	26.7	**29.3**	16.9	29.2	26.6	14.6
horse	100x100 60x100(f)	39.1	46.8	49.7	51.2	55.3	**63.7**	51.3	57.0	60.9
motor.	128x64 40x100(f)	44.2	48.5	48.3	49.1	**54.5**	53.8	53.6	**54.5**	50.7
person	60x100	29.4	44.7	**48.6**	47.2	48.4	47.1	28.6	43.4	44.7
plant	60x100	15.5	19.6	17.7	20.5	**21.2**	18.3	20.3	16.4	14.3
sheep	100x100 60x100(f)	32.3	30.9	35.4	34.9	37.7	28.1	**40.5**	36.6	21.5
sofa	100x100	22.9	30.9	37.9	39.7	**42.7**	42.2	39.6	37.7	38.2
train	128x64 100x100(f)	39.6	49.2	48.9	50.0	53.8	53.1	53.5	**59.4**	49.3
tv	100x100	28.4	38.2	42.3	43.8	47.5	49.3	**54.3**	52.3	43.6
mAP	-	28.5	36.2	39.1	40.8	**43.7**	38.7	40.5	41.7	35.4

[12] L. Li, H. Su, Y. Lim, and L. Fei-Fei. Object bank: An object-level image representation for high-level visual recognition. In *IJCV*, 107(1):20–39, 2014. 2

[13] D. G. Lowe. Object recognition from local scale-invariant features. In *ICCV*, 1999. 2

[14] M. Mathias, R. Benenson, R. Timofte, and L. V. Gool. Handling occlusions with franken-classifiers. In *ICCV*, 2013. 1

[15] W. Nam, P. Dollár, and J. H. Han. Local decorrelation for improved pedestrian detection. In *NIPS*, 2014. 2, 7

[16] R. Nosaka, Y. Ohkawa, and K. Fukui. Feature extraction based on co-occurrence of adjacent local binary patterns. In *Advances in Image and Video Technology*. 2012. 2, 6

[17] S. Paisitkriangkrai, C. Shen, and A. v. d. Hengel. Efficient pedestrian detection by directly optimize the partial area under the roc curve. In *ICCV*, 2013. 2

[18] X. Qi, R. Xiao, J. Guo, and L. Zhang. Pairwise rotation invariant co-occurrence local binary pattern. In *ECCV*. 2012. 1, 2

[19] N. Rasiwasia and N. Vasconcelos. Holistic context modeling using semantic co-occurrences. In *CVPR*, 2009. 2

[20] H. Ren, C.-K. Heng, W. Zheng, L. Liang, and X. Chen. Fast object detection using boosted co-occurrence histograms of oriented gradients. In *ICIP*, 2010. 1, 2

[21] C. S. Sakrapee Paisitkriangkrai and A. van den Hengel. strengthen the effectiveness of pedestrian detection. In *ECCV*, 2014. 2

[22] R. E. Schapire, Y. Freund, P. Bartlett, and W. S. Lee. Boosting the margin: A new explanation for the effectiveness of voting methods. *Annals of statistics*, 1998. 4, 5

[23] C. Shen, P. Wang, S. Paisitkriangkrai, and A. van den Hengel. Training effective node classifiers for cascade classification. In *IJCV*, 103:326–347, 2013. 2

[24] M. Topi, O. Timo, P. Matti, and S. Maricor. Robust texture classification by subsets of local binary patterns. In *ICPR*, 2000. 3

[25] O. Tuzel, F. Porikli, and P. Meer. Human detection via classification on riemannian manifolds. In *CVPR*, 2007. 2

[26] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR*, 2001. 1, 2, 3

[27] X. Wang, T. X. Han, and S. Yan. An hog-lbp human detector with partial occlusion handling. In *ICCV*, 2009. 6

[28] X. Wang, M. Yang, S. Zhu, and Y. Lin. Regionlets for generic object detection. In *ICCV*, 2013. 8, 9

[29] T. Watanabe, S. Ito, and K. Yokoi. Co-occurrence histograms of oriented gradients for pedestrian detection. In *Advances in Image and Video Technology*. 2009. 4, 6

[30] B. Wu and R. Nevatia. Cluster boosted tree classifier for multi-view, multi-pose object detection. In *ICCV*, 2007. 2

[31] B. Wu and R. Nevatia. Optimizing discrimination-efficiency tradeoff in integrating heterogeneous local features for object detection. In *CVPR*, 2008. 1

[32] J. Xu, Q. Wu, J. Zhang, and Z. Tang. Object detection based on co-occurrence gmulbp features. In *ICME*, 2012. 2

[33] J. Yan, X. Zhang, Z. Lei, S. Liao, and S. Z. Li. Robust multi-resolution pedestrian detection in traffic scenes. In *CVPR*, 2013. 2

[34] Y. Yang and S. Newsam. Spatial pyramid co-occurrence for image classification. In *ICCV*, 2011. 2

[35] J. Yuan, M. Yang, and Y. Wu. Mining discriminative co-occurrence patterns for visual recognition. In *CVPR*, 2011. 1, 2

[36] S. Zhang, C. Bauckhage, and A. Cremers. Informed haar-like features improve pedestrian detection. In *CVPR*, 2013. 1, 2

[37] Q. Zhu, M.-C. Yeh, K.-T. Cheng, and S. Avidan. Fast human detection using a cascade of histograms of oriented gradients. In *CVPR*, 2006. 2

[38] Mita, Takeshi and Kaneko, Toshimitsu and Hori, Osamu. Joint haar-like features for face detection. In *ICCV*, 2005 2, 6