Basis Mapping Based Boosting for Object Detection

Haoyu Ren, Ze-Nian Li
Vision and Media Lab
School of Computing Science
Simon Fraser University
Vancouver, BC, Canada
{hra15, li}@sfu.ca

Abstract

We propose a novel mapping method to improve the training accuracy and efficiency of boosted classifiers for object detection. The key step of the proposed method is a non-linear mapping on original samples by referring to the basis samples before feeding into the weak classifiers, where the basis samples correspond to the hard samples in the current training stage. We show that the basis mapping based weak classifier is an approximation of kernel weak classifiers while keeping the same computation cost as linear weak classifiers. As a result, boosting with such weak classifiers is more effective. In this paper, two different non-linear mappings are shown to work well. We adopt the LogitBoost algorithm to train the weak classifiers based on the Histogram of Oriented Gradient descriptor (HOG). Experimental results show that the proposed approach significantly improves the detection accuracy and training efficiency of the boosted classifier. It also achieves high performance on public datasets for both pedestrian detection and general object detection tasks.

1. Introduction

Object detection of a special class is a fundamental problem of computer vision. One of the major challenges in this field is that the object appearances may vary greatly due to many factors, such as different illuminations, view points, poses, etc. This has motivated inventions of various approaches. Among them, a widely used paradigm is to train a classifier on local features [17][30] or descriptors [5][23] using algorithms of boosting family. For example, Viola et al. [30] build an efficient face detector using AdaBoost based on Haar-like feature. Zhang et al. [36] propose an improved version of Haar feature based on up-right human body and further select them to construct a cascade pedestrian detector. Cabrera et al. [25] investigate the use of boosted dominant orientation templates to learn a binary mask that allows to remove background clutter and include relevant context information. Dollar et al. [11] exploit the correlation of neighboring detection windows in boosted classifier and use fast feature pyramids on pedestrian detection [10].

Boosting family algorithms achieve considerable performance for some object detection tasks. However, since the boosting procedure focuses on the hard samples gradually, it gets more and more difficult to find the weak classifiers that can efficiently improve the classification power of the strong classifier. For those more complicated objects such as the multi-view and multi-pose pedestrian, the problem becomes much more serious that in later training rounds the current classification task might be beyond the ability of the weak classifier [34]. As a result, the training may converge very slowly or can not converge at all. In this paper, we propose a novel basis mapping approach in the boosting framework to solve the above problem. The basis mapping maps the original samples into a constrained region referring to the current hard-to-classify samples (namely *basis sample*) in each boosting round, which makes positive patterns with less inner-class variation and easier to be discriminated from negative patterns. As a result, boosting on such mapped region is much more effective than that on the original sample space. In addition, we show that the weak classifier based on basis mapping is an approximation of using kernel methods, while keeping the computation cost same as the linear methods, so that both the detection accuracy and the training efficiency of the boosted classifier will be improved. In our case, the mapping is realized by the proposed Histogram Intersection Mapping (HIM) and the CHi-square Mapping (CHM). The LogitBoost algorithm is adopted to train the cascade classifier with the mapped HOG descriptor. Several experiments on public datasets are used to evaluate our method. The results show that our method improves both the accuracy and the training efficiency of the boosted classifier. It also achieves comparable performance with the commonly-used approaches in both pedestrian and

978-1-4673-6964-0/15/$31.00 ©2015 IEEE

general object detection tasks.

The rest of our paper is structured as follows. Section 2 gives the related works on object detection. Section 3 describes the proposed basis mapping method and two different non-linear mappings to enhance the boosting. Section 4 shows the relationship of the basis mapping and kernel method. Section 5 gives details of the LogitBoost algorithm with the proposed basis mapping. Section 6 shows the experimental results on INRIA pedestrian, Caltech pedestrian and PASCAL VOC 2007 datasets. Conclusions are in the last section.

2. Related Work

There have been a wide variety of approaches developed for object detection. Most of them focus on designing more discriminative local descriptors and using appropriate machine learning methods.

There are many local features and descriptors proposed for various object detection tasks. Most of them reflect the characteristic of some pre-defined local patterns, e.g., Haar [30][36], covariance matrix [28][29], and contour-based descriptor [21]. In these years, HOG descriptor [9][10][12][22][35][37] becomes one of the most popular local descriptors in object detection due to its high discrimination ability. Dalal & Triggs [9] propose the basic form of the HOG descriptor with 2×2 cells. Multi-size versions are developed in [3][12][37], and further extended to pyramid structure [4][8][10][22][24][35]. HOG descriptor is also combined with other low-level features. Levi et al. [20] utilize an accelerated version of the Feature Synthesis method on the low-level description of multiple object parts. Bar-Hillel et al. [2] design an iterative process including feature generation and pruning using multiple operators for part localization. Chen et al. [6] propose Multi-Order Contextual co-Occurrence (MOCO), to implicitly model the high level context using solely detection responses from the object detection based on the combination of HOG and LBP. Paisitkriangkrai et al. [26] utilize new features built on the basis of low-level visual features combination and spatial pooling, which improves the translational invariance and thus the robustness of the detection process.

Boosting framework is widely used in training the cascade classifier for fast object detection. Conventional boosting algorithms such as AdaBoost is well performed on the object classes with small intra-class variation, e.g., the frontal-view faces [30]. However, it shows poor result for more complicated objects with large inner-class variations such as multi-view and multi-pose pedestrians. In order to solve this problem, some previous approaches use more powerful weak classifiers. For example, Zhu et al. [37] apply linear SVM on HOG descriptor as the weak classifier to build a cascade detector. Laptev [19] utilizes Fisher Linear Discriminative Analysis (FLDA) to project the histogram feature onto one principal direction for AdaBoost training. In consideration of the efficiency, most of these weak classifiers are linear weak classifiers, so the performance improvement is still limited. Other approaches follow the divide-and-conquer strategy to build strong classifiers with more complex structures. For example, Huang et al. [18] utilize the vector boosting to train the predictors for the branching nodes of the tree that have multi-components outputs as vectors for face detection. Wu et al. [34] propose the cluster boosted tree method, in which the sample space is divided by unsupervised clustering based on discriminative image features selected by boosting algorithm. Heng et al. [16] design a shrink boost method solving a sparse regularization problem with two iterative steps. First, a boosting step uses weighted training samples to learn a full high dimensional classifier on all features. Next, a shrinkage step shrinks least discriminative classifier dimension to zero to remove the redundant features. Unfortunately, these algorithms increase the computation complexity of both the training and testing procedure. It is also relatively difficult to tune the parameters in order to achieve a better performance. Compare to these algorithms, the proposed basis mapping is relatively efficient. It is also more intuitive and easier to be implemented.

3. Basis Mapping

3.1. Basis Mapping

During the boosting procedure, the weak learner is forced to focus on the hard samples in the training set. This makes it more and more difficult to find weak classifiers. Therefore, the key issue of improving the effectiveness of boosting is how to facilitate the weak classifier training on these hard samples.

A reasonable method is to restrict the current classification problem within a constrained region, rather than considering the whole sample space. In this paper, we propose to map the original samples into the constraint region subject to the hard samples. Intuitively, such mapping "moves" the original samples into a region "around" the hard samples, so that the weak learner can specifically learn the classification hyperplane within the region. This learning task should be much easier than that in the whole sample space.

We define the basis mapping as a process that condenses the sample space into a region by referring to a hard sample, namely *basis sample*. Formally, we formulate the basis mapping which maps the original space $\mathbf{R}^m$ to a new space at the same dimension

$$\Phi(\mathbf{x}) = \varphi(\mathbf{x}, \mathbf{x}_{basis}) \quad \varphi : \mathbf{R}^m \times \mathbf{R}^m \rightarrow \mathbf{R}^m, \quad (1)$$

where $\mathbf{x} \in \mathbf{R}^m$ is an original sample vector and $\mathbf{x}_{basis} \in \mathbf{R}^m$ is a basis sample vector.

3.2. Construction of Basis Mapping

Based on equation (1), we present a kind of mapping that restricts the mapped samples to a "hypersphere" around the $\Phi(\mathbf{x}_{basis})$ with radius $2||\Phi(\mathbf{x}_{basis})||$ as

$$\forall \mathbf{x} \in \mathbf{R}^m : ||\Phi(\mathbf{x}) - \Phi(\mathbf{x}_{basis})|| \leq 2||\Phi(\mathbf{x}_{basis})||, \quad (2)$$

where $||\bullet||$ represents the sum of absolute $\mathbf{x}^{(i)}$ as (3), and $\mathbf{x}^{(i)}$ is the ith dimension of $\mathbf{x}$

$$\forall \mathbf{x} \in \mathbf{R}^m : ||\mathbf{x}|| \equiv \sum_{i=1}^{m} ||\mathbf{x}^{(i)}||. \quad (3)$$

We further give (4), which is a sufficient condition of (2) to constrain the mapping function

$$\begin{aligned} ||\Phi(\mathbf{x}) - \Phi(\mathbf{x}_{basis})|| &\leq ||\Phi(\mathbf{x})|| + ||\Phi(\mathbf{x}_{basis})|| \\ &\leq ||2\Phi(\mathbf{x}_{basis})||. \end{aligned} \quad (4)$$

Therefore, we use (5) as a constraint of the mapping function

$$\forall \mathbf{x} \in \mathbf{R}^m : ||\Phi(\mathbf{x})|| \leq ||\Phi(\mathbf{x}_{basis})||. \quad (5)$$

Substituting φ into (5), it can be seen that $||\varphi(\bullet, \bullet)||$ is a kind of similarity measure for vectors in $\mathbf{R}^m$. In our case, the histogram features are used. Therefore, we adopt two similarity metrics of histograms, the Histogram Intersection and the Chi-Square Distance respectively to conduct the function $||\varphi(\bullet, \bullet)||$.

Histogram intersection [22] is usually used as a similarity metric for histogram-based representations of images. It can be calculated as

$$S_{HI}(\mathbf{x}, \mathbf{x}_{basis}) = \sum_{i=1}^{m} min(\mathbf{x}^{(i)}, \mathbf{x}_{basis}^{(i)}). \quad (6)$$

Defining the basis mapping based on this measurement has certain advantage because the L-1 distance based measurement is more robust to outliers compared to L-2 distance. According to (6), we define the Histogram Intersection Mapping (HIM). Each dimension of the mapped vector can be calculated as

$$\Phi^{(i)}(\mathbf{x}) = \varphi_{HIM}^{(i)}(\mathbf{x}, \mathbf{x}_{basis}) = min(\mathbf{x}^{(i)}, \mathbf{x}_{basis}^{(i)}). \quad (7)$$

To evaluate the effectiveness of the HIM, we train a classifier using HOG descriptor and LogitBoost algorithm (See details in Section 5) on INRIA pedestrian dataset (See details in Section 6.1). The sample distributions on the first selected HOG descriptor are plotted in Fig. 1. The X-axis and the Y-axis represent the two most important dimensions

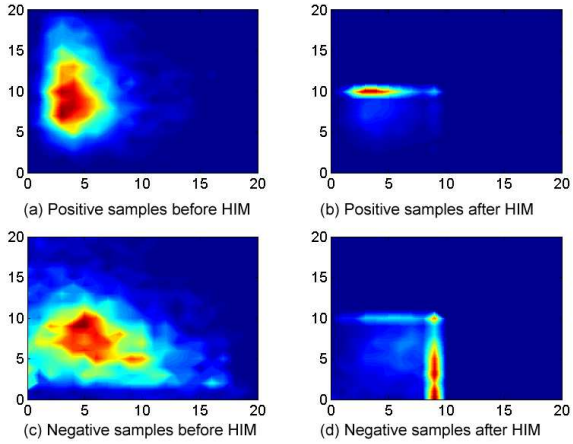

Figure 1. Sample distribution before and after HIM

of the HOG descriptor respectively. They correspond to the two largest weights in the linear regression. Fig. 1(a) and 1(c) show the positive and negative sample distributions before the HIM. Fig. 1(b) and 1(d) show the distributions after the HIM referring a basis sample at (9, 10) in original space. The HIM maps the original samples into a condensed space, where the pattern distributions become much more separable. So that it is easier to learn a classification hyperplane in the mapped space.

Besides the HIM, we also propose another mapping inspired by the Chi-Square Distance. The Chi-Square Distance is another distance metric between histograms [1], as formulated in (8)

$$S_{CHI}(\mathbf{x}, \mathbf{x}_{basis}) = C - \sum_{i=1}^{m} w^{(m)} \frac{(\mathbf{x}^{(i)} - \mathbf{x}_{basis}^{(i)})^2}{\mathbf{x}^{(i)} + \mathbf{x}_{basis}^{(i)}}. \quad (8)$$

We construct the CHi-square Mapping (CHM) by setting all the $w^{(m)}$ to 1 and omitting the constant C

$$\Phi^{(i)}(\mathbf{x}) = \varphi_{CHM}^{(i)}(\mathbf{x}, \mathbf{x}_{basis}) = \frac{(\mathbf{x}^{(i)} - \mathbf{x}_{basis}^{(i)})^2}{\mathbf{x}^{(i)} + \mathbf{x}_{basis}^{(i)}}. \quad (9)$$

Similarly, the sample distribution of CHM on INRIA dataset is shown in Fig. 2, where the basis sample is (2, 2). It can be seen that the CHM also maps the original samples into a more condense space around the basis sample, which makes the mapped space more appropriate for learning a classification hyperplane. These results show that the CHM also plays the similar role as the HIM.

4. Basis Mapping and Kernel Method

Kernel methods map the original samples into an implicit high-dimension space, where the linear classification is subsequently applied. As a result, boosted classifier using kernel weak classifiers achieves better performance compared

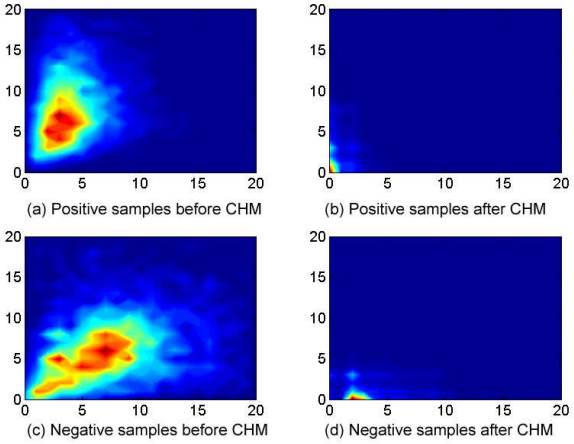

Figure 2. Sample distribution before and after CHM

to traditional boosted classifier based on linear methods. In this section, we will show that the basis mapping defined in section 3 is an approximation of applying additive kernels methods as weak classifiers in the boosting algorithm.

Generally, linear classification in the implicit space can be implemented in the original space through the *kernel trick*. Given two m-dimension samples $\mathbf{x}, \mathbf{z}$ in the original space and a kernel function $K(\mathbf{x}, \mathbf{z})$ that satisfies the Mercer's Condition, there exists a function ψ

$$K(\mathbf{x}, \mathbf{z}) = \psi(\mathbf{x}) \bullet \psi(\mathbf{z}), \tag{10}$$

where $\bullet$ is the dot product of two vectors.

In boosting training, learning weak classifier f could be considered as a finding an optimal classification hyper-plane based on the training samples in original m-dimensional space. If the kernel method is applied, denote the optimal classification hyper-plane in the implicit n-dimension space by $\mathbf{w}^*$, given a sample in the original m-dimension space by $\mathbf{x} = [\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}]$, the optimal classification function $f^*(\mathbf{x})$ is the dot product of the $\psi(\mathbf{x})$ and $\mathbf{w}^*$

$$f^*(\mathbf{x}) = \mathbf{w}^* \bullet \psi(\mathbf{x}). \tag{11}$$

In the extreme case, if there is a vector $\mathbf{x}^* \in \mathbf{R}^m$ satisfies $\psi(\mathbf{x}^*) = \mathbf{w}^*$, equation (11) can be implemented by (12)

$$f^*(\mathbf{x}) = \mathbf{w}^* \bullet \psi(\mathbf{x}) = \psi(\mathbf{x}^*) \bullet \psi(\mathbf{x}) = K(\mathbf{x}, \mathbf{x}^*). \tag{12}$$

Using (12) for classification is relatively convenient. So the only problem is to find out such an $\mathbf{x}^*$. Unfortunately, in most of the cases, ψ is not invertible or even ψ itself could not be explicitly described, so it seems to be impossible to find such an $\mathbf{x}^*$. But in boosting framework, we could approximate $\mathbf{x}^*$ by selecting one of the current training samples $\mathbf{x}'$. The optimal f^* is then approximated using the classification function f in (13)

$$f^*(\mathbf{x}) \approx f(\mathbf{x}) = \mathbf{w}' \bullet \psi(\mathbf{x}) = K(\mathbf{x}, \mathbf{x}'), \tag{13}$$

where $\mathbf{w}' = \psi(\mathbf{x}')$. This implies that by referring to an appropriate sample $\mathbf{x}'$, the linear classification in the implicit space could be approximated by the above kernel function.

Then we turn back to the basis mapping proposed in Section 3. In HIM and CHM, each dimension is independent with each other, so equation (1) could be written as

$$\begin{aligned} \Phi(\mathbf{x}) &= \varphi(\mathbf{x}, \mathbf{x}_{basis}) \\ &= [\varphi(\mathbf{x}^{(1)}, \mathbf{x}^{(1)}_{basis}), \ldots, \varphi(\mathbf{x}^{(m)}, \mathbf{x}^{(m)}_{basis})]. \end{aligned} \tag{14}$$

Notice that the HIM corresponds to the histogram intersection kernel, and CHM corresponds to the Chi-Square kernel. Both of these kernels are additive kernels (15)

$$K(\mathbf{x}, \mathbf{x}') = \sum_{i=1}^{m} k(\mathbf{x}^{(i)}, \mathbf{x}^{'(i)}). \tag{15}$$

So the φ in equation (14) is exactly the same as the k in (15) for HIM and CHM. Then the kernel classification function (13) could be written as

$$f(\mathbf{x}) = K(\mathbf{x}, \mathbf{x}') = \sum_{i=1}^{m} \varphi(\mathbf{x}^{(i)}, \mathbf{x}^{'(i)}). \tag{16}$$

As mentioned above, in the boosting framework, we could use $\mathbf{x}'$ to approximate $\mathbf{x}^*$. This is achieved by evaluating different hard samples in current training stage to get the best one $\mathbf{x}_{basis}$. Then (16) is achieved by (17)

$$f(\mathbf{x}) = \sum_{i=1}^{m} \varphi(\mathbf{x}^{(i)}, \mathbf{x}^{'(i)}) = \sum_{i=1}^{m} \varphi(\mathbf{x}^{(i)}, \mathbf{x}^{(i)}_{basis}). \tag{17}$$

We further fit a linear classifier based on (17) as the final weak classifier

$$f(\mathbf{x}) = \sum_{i=1}^{m} a^{(i)} \varphi(\mathbf{x}^{(i)}, \mathbf{x}^{(i)}_{basis}) + b. \tag{18}$$

According to (14), (18) is the linear classification on the mapped space $\Phi(\mathbf{x})$ around the basis sample. The kernel classification (13) is finally transformed to a linear classification. So we get the conclusion that the basis mapping $\Phi : \mathbf{R}^m \rightarrow \mathbf{R}^m$ is an approximation of additive kernel classification in the original space, which significantly has better discrimination power than simple decision stump or linear weak classifiers. In general, the performance of a boosted classifier mainly depends on the weak classifiers [27]. So the proposed basis mapping will contribute to the overall accuracy of the boosted classifier. Because the basis mapping does not increase the feature dimension, the computation cost will not increase much compared to the linear

weak classifiers. In our implementation, the a and b in (18) are learned by regularized least square.

5. LogitBoost based on Basis Mapping

In this section, we start with a brief description of the LogitBoost algorithm [15] and then introduce how to use the basis mapping under LogitBoost framework.

For a binary classification problem, suppose the training data as $(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_N, \mathbf{y}_N)$, where $\mathbf{x}_i$ is the ith training sample and $\mathbf{y}_i \in \{0, 1\}$ is the class label. The probability of sample $\mathbf{x}_i$ being a member of class 1 is represented by

$$p(\mathbf{x}_i) = \frac{e^{F(\mathbf{x}_i)}}{e^{F(\mathbf{x}_i)} + e^{-F(\mathbf{x}_i)}}, \tag{19}$$

where F is the classification function

$$F(\mathbf{x}) = \frac{1}{2} \sum_{j=1}^{T} f_j(\mathbf{x}). \tag{20}$$

The LogitBoost algorithm treats the weak classifiers as a set of regression functions $f_j(\mathbf{x}_i)_{j=1,2,,T}$ by minimizing the negative binomial log-likelihood of the training samples $l(y, p(\mathbf{x}_i))$ through Newton iterations

$$\begin{aligned} l(y, p(\mathbf{x}_i)) = -\sum_{i=1}^{N} [y_i log(p(\mathbf{x}_i)) + \\ (1 - y_i) log(1 - p(\mathbf{x}_i))]. \end{aligned} \tag{21}$$

This regression function $f_j(\mathbf{x}_i)_{j=1,2,,T}$ fits the training samples $\mathbf{x}_i$ to response values z_i

$$z_i = \frac{y - p(\mathbf{x}_i)}{p(\mathbf{x}_i)(1 - p(\mathbf{x}_i))}. \tag{22}$$

After regression, $F(\mathbf{x}_i)$ and $p(\mathbf{x}_i)$ are updated according to (20) and (19). Then the sample weights are updated as

$$w_i = p(\mathbf{x}_i)(1 - p(\mathbf{x}_i)). \tag{23}$$

The proposed basis mapping is integrated into the LogitBoost framework as an independent module in each training round. The key process is how to select a hard sample as the basis sample. We know that the hard sample should be the sample close to the classification hyperplane, i.e. $F(\mathbf{x}_i) \approx 0$. It can be seen from (19) that those samples with $p(\mathbf{x}_i) \approx 0.5$ result in $F(\mathbf{x}_i) \approx 0$, which can be considered as the "hard samples". According to (23), the weight w_i gets the maximum value when $p(\mathbf{x}_i)$ is 0.5, so we consider the samples with top 20% weights as candidate basis samples and randomly select one at each time for the basis mapping. This procedure repeats for $N_b = 10$ times and the best basis mapping is selected. To learn the best feature, the most intuitive way is to look through the whole feature pool, which is rather time consuming. So we resort to a sampling method to speed up the feature selection process. More specifically, a random sub-sample of size $N_f = log0.05/log0.95 = 59$ will guarantee that we can find the best 5% features with a probability of 95%. Fig. 3 illustrates the detailed algorithm.

Parameters
- N number of training samples
- N_b number of basis samples per iteration
- N_f number of features per iteration
- T maximum number of weak classifiers
- θ threshold of false positive rate (fpr)

Input: Training set $\{(\mathbf{x}_i, y_i)\}, \mathbf{x}_i \in R^m, y_i \in \{0, 1\}$

1. Initialization $w_i = 1/N, F(\mathbf{x}_i) = 0, p(\mathbf{x}_i) = 0.5$
2. Repeat for $t = 1, 2, \ldots, T$
   - 2.1 Compute z_i (22) and w_i (23)
   - 2.2 For $m = 1$ to N_f
     - For $n = 1$ to N_b
       - 2.2.1 Randomly select a basis sample $\mathbf{x}_{basis}$ with top 20% weights
       - 2.2.2 Calculate the original feature vectors $\mathbf{x}_i$
       - 2.2.3 Calculate the mapped vectors $\Phi(\mathbf{x}_i) = \varphi(\mathbf{x}_i, \mathbf{x}_{basis})$
       - 2.2.4 Fit the function f (18) by weighted least square regression from $\Phi(\mathbf{x}_i)$ to z_i
       - 2.2.5 Select the best feature and basis sample with minimum regression error
       - 2.2.6 Calculate the fpr. If it is lower than θ, break
   - 2.3 Update $F(\mathbf{x}_i)$ and $p(\mathbf{x}_i)$ using (20) and (19)
3. Output classifier $F(\mathbf{x}) = sign[\sum_{j=1}^{T} f_j(\mathbf{x})]$

Figure 3. LogitBoost training with basis mapping

6. Experiments

In this section, we show the effectiveness of the proposed method on INRIA pedestrian, Caltech pedestrian and PASCAL VOC 2007 dataset.

6.1. Experiment on INRIA pedestrian dataset

We first evaluate the basis mapping using the INRIA pedestrian dataset [9]. Detection on INRIA dataset is challenging since it includes subjects with a wide range of variations in pose, clothing, illumination, background and partial occlusions. HOG descriptors from 4×4 to 28×56 cell size and 2×2 cell arrangement are utilized. 5,672 HOG descriptors are generated for 64×128 scanning window.

In the experiments, we first follow the training and testing protocols proposed by Dalal & Triggs [9]. In Fig. 4, the performances of the boosted classifiers with different basis mappings and without the basis mappings using the same HOG feature are compared. All classifiers are trained

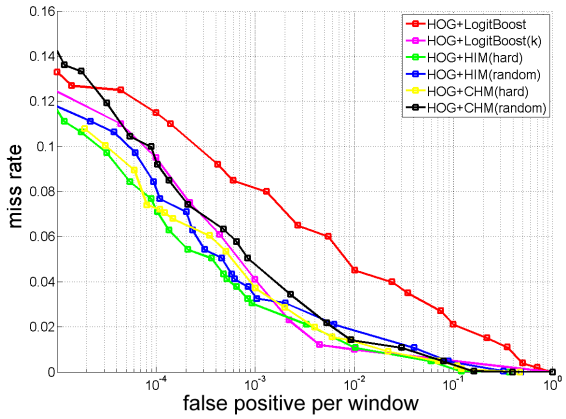

Figure 4. FPPW evaluation on INRIA pedestrian dataset

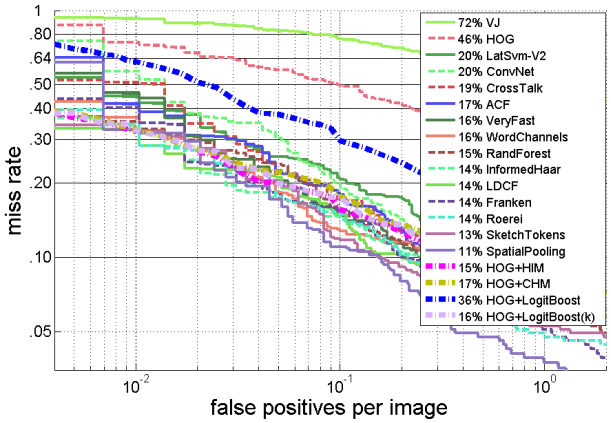

Figure 5. FPPI evaluation on INRIA pedestrian dataset

to 10^{-6} false positive rate. The number of weak classifiers and convergence issue will be discussed in section 6.4. It can be seen that all the algorithms with basis mappings clearly outperform the algorithms without basis mapping. The HIM and CHM consistently achieve about 3% less miss rates at all FPPW, which is similar to the curve 'HOG+LogbitBoost(k)' with HIKSVM as weak classifier. This result shows that the weak classifier based on basis mapping is a good approximation of kernel weak classifier. In addition, we also notice that the HIM using hard samples as basis samples achieves better accuracy than randomly picking basis samples. This also happens on CHM. So it shows the effectiveness of concentrating the classification on the hard samples.

Furthermore, we evaluate our method under the criteria of the detection rate versus False Positive rate Per Image (FPPI) [32]. The average miss rate of these curves are illustrated in Fig. 5. It could be seen that the basis mapping (HOG+HIM, HOG+CHM) clearly improves the detection accuracy of the object detector using the same HOG feature and traditional boosting algorithm (HOG+LogitBoost). The best one, HOG+HIM, reduces the miss rate greatly from 35.7% to 15.3%, to a similar level to using HIKSVM as weak classifier. But the efficiency is much better for both the training and the testing procedure. The accuracy is also comparable with some boosted algorithm with more complicate features (Crosstalk, ACF, Veryfast, Wordchannels), which implies that using kernel weak classifiers with simple features is also effective.

6.2. Experiment on Caltech pedestrian dataset

Next, we evaluate the proposed basis mapping using the Caltech pedestrian dataset [13]. This dataset is one of the largest public available pedestrian dataset. It offers a large number of samples, which consists of approximately 10 hours of 640×480 30Hz video taken from a vehicle driving through regular traffic in an urban environment. About

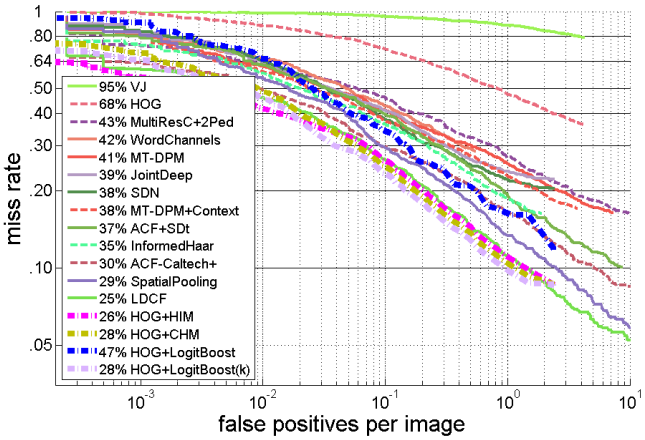

Figure 6. FPPI evaluation on Caltech pedestrian dataset

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 follow the training and evaluation protocol proposed by Dollar et al. [13]. The training sample size and HOG feature pool are the same as used in INRIA dataset. The pedestrians at least 50 pixels tall under no or partial occlusion are evaluated. Fig. 6 illustrates the experimental results of our approach and the state-of-the-art algorithms. It could be seen that the accuracy of HOG feature is significantly improved, where the miss rate is greatly reduced from 46.9% (HOG+LogitBoost) to 28.3% (HOG+HIM) and 30.4% (HOG+CHM), and it is similar to the LogitBoost with HIKSVM as weak classifier (28.4%). This significant improvement proves the effectiveness of the proposed basis mapping. Our method also performs better than the algorithms using HOG feature (MT-DPM) and some complicated features (WordChannels, ACF+SDT).

6.3. Experiment on PASCAL VOC 2007 dataset

Moreover, we employ the standard benchmark object detection dataset, PASCAL VOC 2007, to test our detector. The PASCAL dataset contains images from 20 different categories with about 5,000 images for training and validation and a test set of about 5,000 images. The object detection performance is measured using the standard protocol: average precision (AP) per class, as well as the mean AP (mAP) across all classes. For both measures, we consider that a window is correct if it has an intersection-over-union ratio of at least 50% with a ground-truth object instance.

The training sample size and HOG feature pool are different for different object categories. For the aeroplane, bird, boat, bottle, chair, diningtable, person, pottedplant, sofa, and TV monitor, all the samples are placed 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 based on the aspect ratio, and then trained into two detectors respectively. The final detection result is based on the voting of these two detectors. The sample size (w, h) used to train these classifiers are listed in the second column of Table 1. The size of the HOG cell ranges from 4×4 to $w' \times h'$, where w', h' are the maximum multiple of 4 which is smaller than $w/2$ and $h/2$. As a result, the size of the HOG feature pool ranges from 1,764 to 5,672 for different object categories.

In Table 1, we compare our method with the state-of-the-art algorithms [6][7][14][33] in terms of detection AP on the test set. From the results we could find that the basis mapping significantly improves the mAP at 12% using the same HOG feature, which is similar to directly applying kernel SVM as weak classifier. The HIM achieves the best result on 6 categories, while CHM achieves the best result on 3 categories. This result is reasonable because it is difficult to solve the general object detection problem by linear classifiers. Using kernel classifier clearly contributes to the overall accuracy. In addition, we find that if the difficulty of the detection task is beyond the description ability of HOG descriptor, the HIM and CHM will fail to locate the object. In the experiments, we notice that some of the false positives of HIM and CHM are exact the same as HOG+LogitBoost. This is similar to [31], where the false positives are due to the insufficient description ability of features rather than the classifiers. Our performance is better compare to the SIFT fisher vectors [7], pyramid HOG [14], heterogeneous features [33] and co-occurrence features [6]. Although the features used in these algorithms are stronger than us, this blank could be compensated by the proposed basis mapping which enhances the classification to the kernel level. These results show that the detectors trained on basis mapping are effective for general object detection tasks.

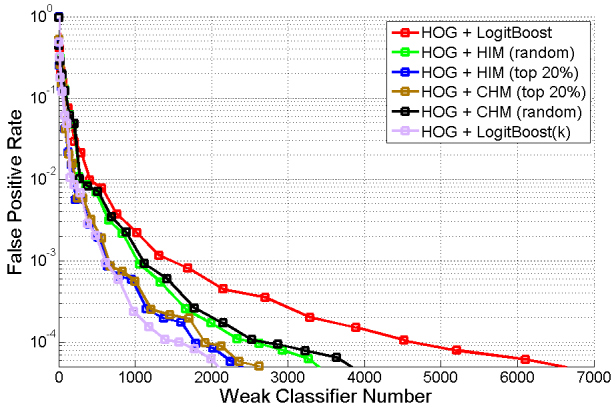

Figure 7. Convergence speed of INRIA pedestrian dataset

6.4. Speed analysis

Finally, we compare the convergence speed of the training process. Fig. 7 plots the false positive rate against the number of weak classifiers for detectors trained on INRIA dataset. This figure shows that it is difficult for the conventional training using LogitBoost with HOG descriptor to converge, especially when the FPPW is lower. On the other hand, the algorithms with HIM and CHM converge faster. Among them, the HIM algorithm is found to be the fastest, at the rate of approximately three times faster than the conventional training without any basis mapping. The convergence speed is a little slower than LogitBoost with HIKSVM, which is due to the fact that the basis mapping is an approximation rather than exact equivalence. We also notice that using hard samples for basis mapping converges faster than using random training samples. In general, the performance of boosted classifier is shown to be positively proportional to the convergence speed in training. This signifies that the proposed basis mapping can consistently enhance the training accuracy and the training speed of boosted classifiers.

We test the training and detection speed of different methods on an Intel I7 dual core PC with 8GB memory. The results are listed in Table 2. It could be seen that the training speed is greatly improved using the basis mapping compared to using conventional linear weak classifiers, while the testing speed is still the same. Compared to the LogitBoost utilizing efficient HIKSVM [22] as weak classifier, both the training speed and testing speed are far better. So we can get the conclusion that the basis mapping contributes to the accuracy and efficiency at the same time.

7. Conclusion

In this paper, we proposed a basis mapping method that is capable of improving accuracy and training speed of boosted classifiers. The original samples are mapped to a

‘

Table 1. Experimental results on PASCAL VOC 2007 dataset. In the second column, the ‘f’ denotes the sample size for front/rear images. There is only one detector for this category without ‘f’ label

	Sample size	**HOG+LogitBoost**	**HOG+HIM**	**HOG+CHM**	Chen [6]	Cinbis [7]	Wang [33]	Felzenszwalb [14]
aeroplane	128x64	27.8	55.5	53.7	41.0	**56.1**	54.2	36.6
bicycle	128x64 40x100(f)	52.3	61.9	54.4	**64.3**	56.4	52.0	62.2
bird	100x100	19.9	**25.6**	22.9	15.1	21.8	20.3	12.1
boat	100x100	17.2	25.3	**26.9**	19.5	26.8	24.0	17.6
bottle	40x120	18.3	31.4	27.2	**33.0**	19.9	20.1	28.7
bus	128x64 100x100(f)	31.0	57.2	56.6	**57.9**	49.5	55.5	54.6
car	100x60 100x100(f)	47.9	62.4	61.2	63.2	57.9	**68.7**	60.4
cat	100x60 100x100(f)	23.5	42.9	43.8	27.8	**46.2**	42.6	25.5
chair	100x100	18.3	**23.7**	21.4	23.2	16.4	19.2	21.1
cow	100x60 60x100(f)	23.2	35.2	34.7	28.2	41.4	**44.2**	25.6
dining table	100x60	22.1	45.1	**49.1**	29.1	47.1	**49.1**	26.6
dog	100x60 80x100(f)	19.2	**29.6**	29.3	16.9	29.2	26.6	14.6
horse	100x100 60x100(f)	39.0	58.4	56.1	**63.7**	51.3	57.0	60.9
motorbike	128x64 40x100(f)	46.0	**55.6**	50.6	53.8	53.6	54.5	50.7
person	60x100	30.1	45.1	**47.2**	47.1	28.6	43.4	44.7
plant	60x100	17.2	**20.9**	20.0	18.3	20.3	16.4	14.3
sheep	100x100 60x100(f)	23.4	36.8	37.1	28.1	**40.5**	36.6	21.5
sofa	100x100	24.4	**42.5**	41.9	42.2	39.6	37.7	38.2
train	128x64 100x100(f)	42.1	49.0	49.0	53.1	53.5	**59.4**	49.3
tv	100x100	38.3	52.8	49.1	49.3	**54.3**	52.3	43.6
mAP	-	29.1	**42.8**	41.6	38.7	40.5	41.7	35.4

Table 2. Training and testing speed of methods w/o basis mapping

Approach	**Training speed**	**Patches/sec**
HOG + LogitBoost	47 hours	167k
HOG + LogitBoost(k)	242 hours	8k
HOG + HIM	14 hours	181k
HOG + CHM	15 hours	174k

closed, restricted local region defined by identified hard-to-classify samples. Such mapping modifies the distribution of the samples so that classification performed on mapped space is enhanced over classification on original un-mapped space. This results in more efficient boosting. In addition, we also show the relationship of the basis mapping and kernel method. The linear classification in the mapped space achieves similar performance with the non-linear classification in the original space. Two basis mappings (namely HIM and CHM) are proposed and shown their effectiveness on INRIA, Caltech, and PASCAL VOC 2007 dataset.

The algorithms proposed in this paper are promising and they will be further studied. Histogram-based features are widely used in object detection. The proposed algorithm is effective for other histogram-based descriptors such as LBP histogram and co-occurrence histograms. We will also attempt to utilize the proposed mapping in other machine learning algorithms.

8. Acknowledgement

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada under the Grant RGP36726.

References

[1] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28:2037–2041, 2006.

[2] A. Bar-Hillel, D. Levi, E. Krupka, and C. Goldberg. Part-based feature synthesis for human detection. In *European Conference on Computer Vision*. 2010.

[3] R. Benenson, M. Mathias, R. Timofte, and L. Van Gool. Pedestrian detection at 100 frames per second. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[4] R. Benenson, M. Mathias, T. Tuytelaars, and L. Van Gool. Seeking the strongest rigid detector. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[5] R. Benenson, M. Omran, J. Hosang, , and B. Schiele. Ten years of pedestrian detection, what have we learned? In *European Conference of Computer Vision Workshop*, 2014.

[6] G. Chen, Y. Ding, J. Xiao, and T. X. Han. Detection evolution with multi-order contextual co-occurrence. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[7] R. G. Cinbis, J. Verbeek, and C. Schmid. Segmentation driven object detection with fisher vectors. In *IEEE International Conference on Computer Vision*, 2013.

[8] A. Costea and S. Nedevschi. Word channel based multiscale pedestrian detection without image resizing and using only one classifier. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[9] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[10] P. Dollár, R. Appel, S. Belongie, and P. Perona. Fast feature pyramids for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36:1532–1545, 2014.

[11] P. Dollár, R. Appel, and W. Kienzle. Crosstalk cascades for frame-rate pedestrian detection. In *European Conference on Computer Vision*. 2012.

[12] P. Dollár, S. Belongie, and P. Perona. The fastest pedestrian detector in the west. In *British Machine Vision Conference*, 2010.

[13] P. Dollar, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: An evaluation of the state of the art. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34:743–761, 2012.

[14] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32:1627–1645, 2010.

[15] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The Annals of Statistics*, 28:337–407, 2000.

[16] C. K. Heng, S. Yokomitsu, Y. Matsumoto, and H. Tamura. Shrink boost for selecting multi-lbp histogram features in object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[17] V.-D. Hoang, M.-H. Le, and K.-H. Jo. Hybrid cascade boosting machine using variant scale blocks based hog features for pedestrian detection. *Neurocomputing*, (135):357–366, 2014.

[18] C. Huang, H. Ai, Y. Li, and S. Lao. Vector boosting for rotation invariant multi-view face detection. In *IEEE International Conference on Computer Vision*, 2005.

[19] I. Laptev. Improving object detection with boosted histograms. *Image and Vision Computing*, 27:535–544, 2009.

[20] D. Levi, S. Silberstein, and A. Bar-Hillel. Fast multiple-part based object detection using kd-ferns. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[21] J. J. Lim, C. L. Zitnick, and P. Dollár. Sketch tokens: A learned mid-level representation for contour and object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[22] S. Maji, A. C. Berg, and J. Malik. Classification using intersection kernel support vector machines is efficient. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[23] M. Mathias, R. Benenson, R. Timofte, and L. V. Gool. Handling occlusions with franken-classifiers. In *IEEE International Conference on Computer Vision*, 2013.

[24] W. Nam, P. Dollár, and J. H. Han. Local decorrelation for improved pedestrian detection. In *Neural Information Processing Systems*, 2014.

[25] R. Rios-Cabrera and T. Tuytelaars. Boosting masked dominant orientation templates for efficient object detection. *Computer Vision and Image Understanding*, 120:103–116, 2013.

[26] C. S. Sakrapee Paisitkriangkrai and A. van den Hengel. strengthen the effectiveness of pedestrian detection. In *European Conference of Computer Vision*, 2014.

[27] R. E. Schapire and Y. Singer. Improved boosting algorithms using confidence-rated predictions. *Machine learning*, 37:297–336, 1999.

[28] C. Shen, P. Wang, S. Paisitkriangkrai, and A. van den Hengel. Training effective node classifiers for cascade classification. *International Journal of Computer Vision*, 103:326–347, 2013.

[29] O. Tuzel, F. Porikli, and P. Meer. Pedestrian detection via classification on riemannian manifolds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(10):1713–1727, 2008.

[30] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

[31] C. Vondrick, A. Khosla, T. Malisiewicz, and A. Torralba. Hoggles: Visualizing object detection features. In *IEEE International Conference on Computer Vision*, 2013.

[32] X. Wang, T. X. Han, and S. Yan. An hog-lbp human detector with partial occlusion handling. In *IEEE International Conference on Computer Vision*, 2009.

[33] X. Wang, M. Yang, S. Zhu, and Y. Lin. Regionlets for generic object detection. In *IEEE International Conference on Computer Vision*, 2013.

[34] B. Wu and R. Nevatia. Cluster boosted tree classifier for multi-view, multi-pose object detection. In *IEEE International Conference on Computer Vision*, 2007.

[35] J. Yan, X. Zhang, Z. Lei, S. Liao, and S. Z. Li. Robust multi-resolution pedestrian detection in traffic scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[36] S. Zhang, C. Bauckhage, and A. Cremers. Informed haar-like features improve pedestrian detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[37] Q. Zhu, M.-C. Yeh, K.-T. Cheng, and S. Avidan. Fast human detection using a cascade of histograms of oriented gradients. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2006.