Fast Discrimination by Early Judgment Using Linear Classifier

Takato Kurokawa, Yuji Yamauchi, Takayoshi Yamashita, Hironobu Hujiyoshi
Chubu University
Kasugai, Aichi, Japan
{kuro@vision., yuu@vision., yamashita@, hf@}cs.chubu.ac.jp

Abstract

Object detection involves classification of a huge number of detection windows obtained by raster scanning of the input image. For each detection window, a classifier trained with local features and a statistical learning method outputs a value for the target class. In this paper, we investigated the introduction of linear SVM approximate computation to object detection to increase the speed of raster scanning. We propose a method of fast discrimination by early judgment using linear classifier based approximation calculation. Doing so enables high-speed linear SVM classification by adaptively determining the number of bases required in the approximation calculations for the input detection window. Also, higher accuracy is attained in the object detection by representing the co-occurrence of binary-coded (B-HOG) forms of the HOG features that are used when doing the linear SVM approximating calculations. Evaluation experiments on human detection show that the proposed method is faster than using HOG features and linear SVM by a factor of 17 and improves the classification accuracy by about 6.1%.

1 Introduction

Object detection is implemented by classifiers that are trained by statistical learning methods applied to local features extracted from training samples. A typical object detection technique is the method proposed in 2005 by Dalal *et al.*, which uses Histograms of Oriented Gradients(HOG) features and linear Support vector machine(SVM)[1] to detect human forms in images [2]. The combination of HOG features and linear SVM has been used as a basic method for research on identifying obscured detection targets [11][12] and techniques for higher object detection accuracy such as the parts model [6]. Making such object detection practical involves the problems of faster processing and reduced memory use as well as higher accuracy in classification performance. The detection process requires processing of the huge number of detection windows generated by raster scanning of the entire input image, so faster object detection requires faster feature extraction and classification, which are the two processes that constitute detection window processing. For faster features extraction, methods using integral histograms [14][5] and computation with a GPU [9] have been proposed. Zha *et al.* proposed a method of fast HOG feature extraction using integral histograms [14]. Dollár *et al.* proposed a gradient-based feature using integral histograms [5] and a feature based on edge using the relationship image scale [4]. The GPU method for fast HOG [9] extraction achieved a speed increase of a factor of 95 compared to use of the CPU. For faster classification using Adaboost [7], Viola *et al.* proposed a method of early judgment of non-target areas by cascaded Adaboost [10]. Dollar *et al.* [3] proposed a method called crosstalk cascade by enabling neighboring detectors to communicate. However, there have been no studies on speeding up the linear SVM processing that is often used in object detection.

We therefore investigated a faster classifier in which linear SVM is used to reach an early judgment on the classification result. The proposed method achieves high-speed raster scanning by performing linear SVM approximation calculations according to the feature vector extracted from the detection window for early determination of the classification results. The proposed method introduces a co-occurrence representation that uses bit operations on the binary code of the B-HOG features[13] to increase classification accuracy while minimizing computational cost by expressing the relationships of features between cells. In this way, we realize object detection with both high accuracy and high speed by early classifier judgment in raster scanning and co-occurrence expression of B-HOG features.

2 Proposed method

To increase the speed of raster scanning, we introduced early judgment by using linear SVM approximation calculations. To solve the problem of reduced accuracy due to quantization of the input feature vector, which is necessary for faster processing based on linear SVM approximation calculations, we increase the accuracy of object detection by representing the co-occurrence of B-HOG features (binary coded HOG features). We describe the faster raster scanning and the binary code co-occurrence representation in the following sections.

2.1 Linear SVM approximation calculations

The calculation of the classifier $F(\mathbf{x})$ trained by linear SVM involves taking the inner product of the feature vector $\mathbf{x}$ and the weight vector $\mathbf{w}$ as shown in Eq. (1), so the time required to process the huge number of detection windows is a problem.

$$F(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\mathbf{x} = \sum_{i=1}^{D} w_i x_i \tag{1}$$

To address that problem, we used the approximate computation method proposed by Hare *et al.* [8] which is to replace the inner product of real-number vectors with the inner product of binary codes, thus achieving high-speed linear SVM classification.

In the linear SVM approximate computation method, weight vector $\mathbf{w}$ is decomposed into a real vector $\mathbf{c}$ and a binary code $\mathbf{M} \in \{-1, 1\}^{N_b \times D}$ by [8]. Here N_b denotes the number of basis. Using the real vector $\mathbf{c}$ and base binary code $\mathbf{M}$ obtained from the

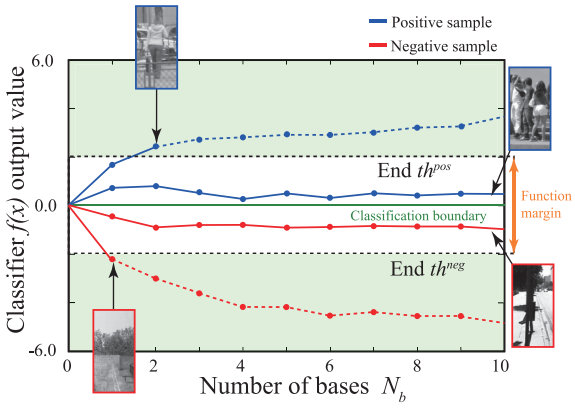

Figure 1. Output value of approximation calculation by number of bases.

weight vector $\mathbf{w}$ in the linear SVM based classifier $F(\mathbf{x})$ makes it possible to do the approximation calculation $F(\mathbf{x}) \approx f(\mathbf{x}) = \sum_{i=1}^{N_b} c_i \mathbf{m}_i^{\mathrm{T}} \mathbf{x}$. Here, by decomposing the base binary code $\mathbf{M}$ to $\mathbf{M}^+ \in \{0,1\}^{N_b \times D}$ and $\bar{\mathbf{M}}^+ (\mathbf{M} = \mathbf{M}^+ - \bar{\mathbf{M}}^+)$, it is possible to calculate the linear SVM approximation from the inner product $< \mathbf{m}_i^+, \mathbf{x} >$ and the norm $|\mathbf{x}|$ as shown in Eq. (2).

$$f(\boldsymbol{x}) = \sum_{i=1}^{N_b} c_i {\mathbf{m}_i}^{\mathrm{T}} \mathbf{x} = \sum_{i=1}^{N_b} c_i (2 < \mathbf{m}_i^+, \mathbf{x} > - |\mathbf{x}|) \quad (2)$$

Here, if the input feature vector $\mathbf{x}$ is a binary code $\mathbf{x} \in \{0,1\}^D$ like the B-HOG features, taking the conjunction of the inner product $< \mathbf{m}_i^+, \mathbf{x} >$ and the norm $|\mathbf{x}|$ can be calculated with bitwise operator AND and bit count operations, which is faster than taking the product of real numbers. Furthermore, the bit count can performed at high speed by using the POPCNT function that is implemented directly on the CPU from the Streaming SIMD Extensions(SSE)4.2.

2.2 Number of bases and approximation calculation results

An example of the results of the linear SVM approximation calculations for various number of bases for a sample of the data set is shown in Fig. 1. From the graph in Fig. 1, we can see that when the number of bases is small, the linear SVM approximation calculation roughly calculates the classifier output value. As the number of bases increases, the classifier output changes so as to reduce the error with respect to the linear SVM. Furthermore, the output values leave the linear SVM function margin region when the number of bases is small for the two samples of Fig. 1. For good accuracy in object detection, the classifier threshold, th, is set so that the two classes y are known within the function margin for the most cases. Thus, for samples that have output values outside the function margin for low number of bases in the linear SVM approximation calculations, the number of bases is increased and there is no change in the classification results for threshold th.

2.3 Faster Distinction by early judgment of the approximation calculation result

Given the trade-off between classification accuracy and speed for the number of bases, we would like to make the number of bases as low as possible for fast processing of the huge number of detection windows. Therefore, in the proposed method, we introduce early judgment from the calculation results using the function margin in the approximation calculation process to achieve faster raster scanning while maintaining classifier performance (**Algorithm 1**). Here, th^{pos} and th^{neg} are the support vectors(classification boundary ± 1.0) for the detection target class $y = 1$ and the non-target class $y = -1$ and , $th^{pos} < f(\mathbf{x}) < th^{neg}$ represents that the output value is within the function margin. As we see in Fig. 1, the linear SVM approximation calculation is halted when large values in the positive and negative orientations are output and an early judgment of the classification result is possible. The input samples for which classification within the function margin is difficult are judged by performing approximation calculations using additional bases to reduce the error with respect to linear SVM. In this way, it is possible to achieve high-speed raster scanning by early judgment according to the detection window while maintaining the classification accuracy of linear SVM.

Algorithm 1 Faster Distinction by early judgment.

Require: input image I

1. Raster scan the detection windows for image I

for $k = 1$ to K **do** //K: total number of detection windows

2. Extract binary coded feature vectors $\mathbf{x}_k$ from detection window $I(k)$.
3. Initialize $f(\mathbf{x}_k) \leftarrow 0$
4. Obtain the classifier output value with the approximation calculation.

for $i = 1$ to N_b **do** //N_b: Maximum number of bases

4.1 Linear SVM approximation calculation: $c_i(2 < \mathbf{m}_i^+, \mathbf{x}_k > - |\mathbf{x}_k|)$

$f(\mathbf{x}_k)$ += $c_i(2\ \mathrm{POPCNT}(\mathbf{m}_i^+ \& \mathbf{x}_k) - \mathrm{POPCNT}(\mathbf{x}_k))$

4.2 Judge the calculation result.

if $f(\mathbf{x}_k) > th^{pos}$ **or** $f(\mathbf{x}_k) < th^{neg}$ **then**
break //End the approximation calculation.
end if

end for //Approximation calculation up to N_b

5. Judge label y for the target from threshold th.

$$y_k = \begin{cases} 1 & \textbf{if} \quad f(\mathbf{x}_k) > th \\ -1 & \textbf{otherwise} \end{cases}$$

end for //End raster scanning.

return $y_1, y_2, ..., y_K$

2.4 Co-occurrence representation in binary code

As described in section 2.1, we use B-HOG features so that we can use bit operations for faster approximation calculations. Because B-HOG features introduce quantization error that reduces classification accuracy in object detection. Our method represents co-occurrence with bit operations between B-HOG features to increase classification accuracy when using binary coded feature vectors without adding much to the computational cost.

The proposed method adds binary code that represents co-occurrence for combinations of cells within a block region (Fig. 2) to the binary code of B-HOG features. We represent the relationships between the

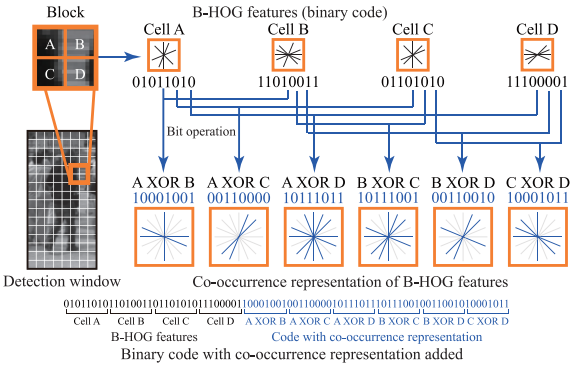

Figure 2. Co-occurrence representation of B-HOG features.

binary codes in histograms of oriented gradients from B-HOG features $\mathbf{P}_{c_i}$ and $\mathbf{P}_{c_j}$ of two cells c_i, c_j of a block with the bit operations shown in Eq. (3). The bitwise operators AND, OR, or XOR may be used.

$$\begin{cases} \mathbf{P}^{\mathrm{AND}}_{c_i,c_j} = \mathbf{P}_{c_i} \& \mathbf{P}_{c_j} \\ \mathbf{P}^{\mathrm{OR}}_{c_i,c_j} = \mathbf{P}_{c_i} | \mathbf{P}_{c_j} \\ \mathbf{P}^{\mathrm{XOR}}_{c_i,c_j} = \mathbf{P}_{c_i} \hat{} \, \mathbf{P}_{c_j} \end{cases} \quad (3)$$

For block size of 2×2 cells as shown in Fig. 2, there are six patterns of binary code co-occurrence according to the cell combinations: $\mathbf{P}^{operator} = \{\mathbf{P}_{c_1,c_2}, \mathbf{P}_{c_1,c_3}, \mathbf{P}_{c_1,c_4}, \mathbf{P}_{c_2,c_3}, \mathbf{P}_{c_2,c_4}, \mathbf{P}_{c_3,c_4}\}$. These bit operations are used in determining co-occurrence between binary codes, co-occurrence can be represented with low computational cost.

3 Evaluation experiments

To evaluate the effectiveness of the proposed method, we did comparison experiments on classification accuracy and classification processing time.

3.1 Overview of the experiment

To evaluate the effectiveness of our method, we compared the classification accuracy for the features and classifier combinations listed below.

- HOG features and linear SVM
- B-HOG features and linear SVM
- Proposed method

We compared the performance for the three bitwise operators used for representing the co-occurrence of B-HOG features in the proposed method (AND, OR, and XOR). The system used for the processing time comparison was Intel Xeon CPU X7542 at 2.67 GHz. All of the experiments used SVM Light to train the linear SVM. We chose the number of bases $N_b = 16$ for the linear SVM approximation calculations based on the results of preliminary experiments. Our evaluation experiments used the INRIA Person Dataset [2], which is widely used as a human detection benchmark. The training samples were a positive sample of 2,416 images and a negative sample of 12,180 images. The test sample for evaluating the classification accuracy comprised 1,306,029 images obtained by complete raster scanning of a positive sample of 1,126 images and a negative sample of 453 background images.

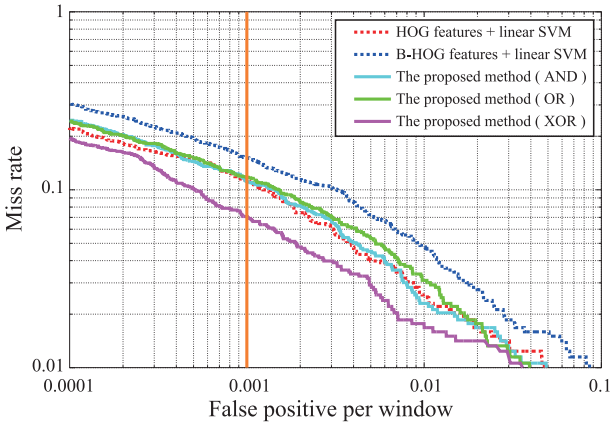

Figure 3. DET curve.

Table 1. Detection rate(DR)[%] and processing time(PT)[ms] for each classifier.

Classifier	DR	PT
SVM without appr. comp.	94.16	0.034
SVM with appr. comp. (N_b=2)	91.07	0.002
SVM with appr. comp. (N_b=16)	94.08	0.013
SVM with appr. comp. and early judg.	94.08	0.002

3.2 Evaluation of classification accuracy

The Detection Error Tradeoff(DET) curve of the experiment results (Fig. 3) shows a decrease in classification accuracy when the B-HOG features are used in the linear SVM compared to using the HOG features. When our method was applied with the AND operator or the OR operator, the classification rate was on the same level as when the HOG features were used. Furthermore, the proposed method with the XOR operator produced an improvement of about 6.1% compared to "HOG features + linear SVM" at a false detection rate of 0.1%. For the AND operator and the OR operator, there is a bias on the probability of occurrence with "0" and "1". Therefore, the same binary code from positive samples and negative samples is sometimes calculated, when the AND operator and the OR operator are used for co-occurrence representation. On the other hand, the XOR operator makes to produce a unique binary code because the probability of occurrence with "0" and "1" is same.

3.3 Evaluation of processing speed

The detection rates for using feature vectors represented by XOR co-occurrence and the processing time required to classify one detection window are listed in Table 1. The approximate computation method with $N_b = 16$ bases has the same level of classification accuracy as linear SVM, but the processing for the approximate computation method was about three times faster than for linear SVM. Because the proposed method performs early judgment, it is faster than the approximate computation method and about 17 times as fast as linear SVM. In that case, the early judgment was based on the results of approximation calculations in which the average number of bases was 7.78 for the positive sample and 1.46 for the negative sample. The approximate computation method using two bases had the same processing time as the proposed method, but the detection rate was lower. For the case of VGA

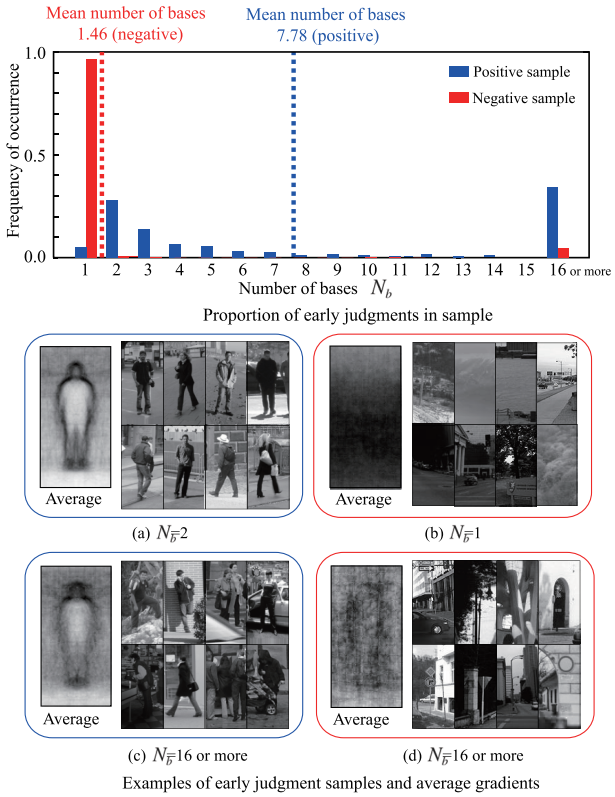

Figure 4. Samples for which early judgment was made in the classification results.

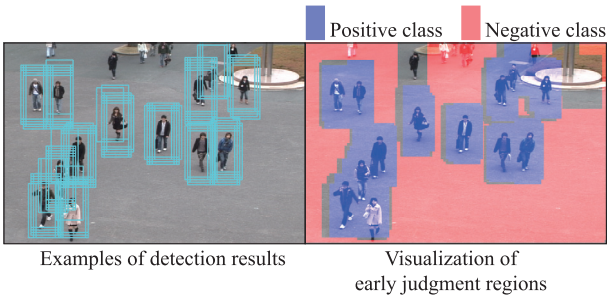

Figure 5. Examples of detection results and visualization of early judgment regions.

size (640 by 480 pixel) input images the total time required for detection window classification processing was 674.2 ms (1.48 fps) for linear SVM, but only 39.66 ms (25.21 fps) for the proposed method, demonstrating that high-speed raster scanning was achieved.

The proportions of samples for which early judgment was made in the classification results by the proposed method and example samples are shown in Fig. 4. We can see that the samples for when the number of bases is low contain human figures that are easily distinguished from the background, as can be confirmed from the mean gradient image of the early judgment samples (a) and (b). For the difficult samples (c) and (d), on the other hand, the approximation calculations continued until the number of bases was large ($N_b = 16$). When the proportion of negative samples was 90% or more, high-speed raster scanning was possible because the judgment could be made with a single base. Because ordinary object detection scenes (input images) contain much background (non-target areas), the proposed method applies a low number of bases in performing the linear SVM approximation calculations for many of the detection windows (Fig. 5).

4 Conclusion

This paper has two contributions to faster classification processing in object detection: 1. Faster raster scanning by introducing early judgment to linear SVM classifiers and 2. Higher accuracy in object detection by binary coded features (B-HOG) that represent feature co-occurrence. The former increases the speed of raster scanning by a factor of about 17 while maintaining accuracy by adaptively making early judgments on the classification results in the linear SVM approximation calculations according to the features of the detection window. The latter increases accuracy by about 6.1% by applying bit operations on the binary codes of B-HOG features to represent co-occurrence for binary coding at high speed and with good efficiency, thus increasing classification accuracy while reducing memory size to about 1/3. Future work includes an implementation of faster feature extraction to realize fast object detection.

References

[1] C. Cortes and V. Vladimir. Support-Vector Networks. In *Machine Learning*, volume 20, pages 273–297, 1995.

[2] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *Computer Vision and Pattern Recognition*, volume 1, pages 886–893, 2005.

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

[4] P. Dollár, S. Belongie, and P. Perona. The Fastest Pedestrian Detector in the West. In *BMVC*, volume 2, 2010.

[5] P. Dollár, Z. Tu, P. Perona, and D. Ramanan. Integral Channel Features. In *BMVC*, volume 2, 2009.

[6] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester. Object Detection with Discriminatively Trained Part Based Models. In *PAMI*, volume 32, pages 1627–1645, 2010.

[7] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. *ICML*, pages 148–156, 1996.

[8] S. Hare, A. Saffari, and P. H. S. Torr. Efficient online structured output learning for keypoint-based object tracking, 2012.

[9] V. Prisacariu and I. Reid. fastHOG-a real-time GPU implementation of HOG. In *Department of Engineering Science*, volume 2310, 2009.

[10] P. Viola and M. Jones. Rapid Object Detection using a Boosted Cascade of Simple Features. In *CVPR*, volume 1, pages 511–518, 2001.

[11] X. Wang, T. X. Han, and S.yan. An HOG-LBP human detector with partial occlusion handling. In *ICCV*, pages 32–39, 2009.

[12] C. Wojek, S. Walk, S. Roth, and B. Schiele. Monocular 3D scene understanding with explicit occlusion reasoning. In *CVPR*, pages 1993–2000, 2011.

[13] Y. Yamauchi, C. Matsushima, T. Yamashita, and H. Fujiyoshi. Relational HOG Feature with Wild-Card for Object Detection. In *Workshop on Visual Surveillance(in conjunction with ICCV2011)*, 2011.

[14] Q. Zhu, S. Avidan, M. Yeh, and K. Cheng. Fast Human Detection Using a Cascade of Histograms of Oriented Gradients. In *CVPR*, pages 1491–1498, 2006.