ITE Trans. on MTA Vol. 1, No. 3, pp. 237-243 (2013) 

Paper

# A Method for Improving SVM-based Image Classification Performance Based on a Target Object Detection Scheme

Soh Yoshida (student member)†, Hiroshi Okada †, Takahiro Ogawa (member)†, Miki Haseyama (member)†

**Abstract** This paper presents a new method to improve performance of SVM-based classification, which contains a target object detection scheme. The proposed method tries to detect target objects from training images and improve the performance of the image classification by calculating the hyperplane from the detection results. Specifically, the proposed method calculates a Support Vector Machine (SVM) hyperplane, and detects rectangular areas surrounding the target objects based on the distances between their feature vectors and the separating hyperplane in the feature space. Then modification of feature vectors becomes feasible by removing features that exist only in background areas. Furthermore, a new hyperplane is calculated by using the modified feature vectors. Since the removed features are not part of the target object, they are not relevant to the learning process. Therefore, their removal can improve the performance of the image classification. Experimental results obtained by applying the proposed methods to several existing SVM-based classification method show its effectiveness.

**Key words**: Image classification, object detection, support vector machine.

## 1. Introduction

Image classification, which assigns an image to its belonging class, is one of the fundamental problems in the field of computer vision. In order to realize image classification, a successful image representation and accurate classifiers are necessary.

In image representation, local features are typically extracted from images. Since SIFT[1] and HOG[2] are capable to capture distinctive details of the images, they have become popular features for image representation. However, since they are sensitive to noise, such features are rarely fed into classifiers directly[3]. Instead, a common strategy is to integrate the local features into a global image representation. Specifically, the Bag of Features (BoF) based approaches have been mainly applied because of their simplicity and effectiveness[3], and have performed to yield good performance on many challenging classification tasks[4)~9)]. In addition, further two improvements have been widely adopted. One is coding methods, which avoid information loss in the feature quantization for training BoF[5)8)9)]. The other is Spatial Pyramid Matching, which considers information about the spatial layout of local features[4)6)7)]. Although these improvements reduce the influence of noisy local features, there still exists a problem that classifiers are sensitive to them. Therefore, we need to reduce noisy features when learning a classifier for each class.

These noisy features tend to be extracted from areas which do not include target objects such as background areas. Thus, removal of areas other than the target objects in feature extraction step is effective for learning a classifier. However, since images are captured in various environments, object conditions, such as size, location and pose are generally unknown, and then, we have to estimate areas including target objects from only the provided training images.

In this paper, we propose a new method to improve the performance of SVM classification by removing non-relevant features to the target object. In the proposed method, we adopt the following two novel procedures: (i) detection of target objects and (ii) calculation of a hyperplane from modified feature vectors. First, in (i), areas including the target objects are detected by monitoring distances between their feature vectors and the separating hyperplane. This hyperplane is calculated by Support Vector Machine (SVM)[10] using entire training images including the target objects. Since the monitored distance provides "the probability of membership in a particular class corresponding to the target object"[11], the detection of the target objects becomes feasible. Therefore, in (ii), we can modify feature vectors by removing features that exist only in background areas. Then a new hyperplane can be also obtained as an updated classifier by using these feature vectors. Since the classifier can be obtained from the target objects, the other background areas do not affect the learning process. Then

Received April 2, 2013; Revised April 19, 2013; Accepted April 22, 2013
† Graduate School of Information Science and Technology, Hokkaido University (Kita-14, Nishi-9, Sapporo-shi, 060-0814 Japan)

the problems of the conventional methods, specifically sensitivity to noisy features are solved, and the performance of image classification can be improved.

This paper is organized as follows. Section 2 provides a brief review of SVM. Section 3 presents the proposed method, which improves the SVM-based image classification performance based on a target object detection scheme. Section 4 provides experimental results that verify the performance of the proposed method. Finally, Section 5 presents concluding remarks.

## 2. Support Vector Machine

This section briefly shows Support Vector Machine (SVM)[10] for two-class classification. Given a training dataset consisting of $N$ vectors $\mathbf{x}_i \in \mathbb{R}^d (i = 1, 2, \cdots, N)$ and their corresponding class labels $y_i \in \{-1, 1\}$, the feature vectors $\mathbf{x}_i$ are mapped into a high-dimensional feature space to obtain $\Phi(\mathbf{x}_i) \in \mathbb{R}^{d'} (d' \gg d)$. Then the separation of the two different classes becomes feasible through an optimal hyperplane defined by a weight factor $\mathbf{w} \in \mathbb{R}^{d'}$ and a bias $b \in \mathbb{R}$. Specifically, the optimal hyperplane is defined as:

$$f^{\mathrm{svm}}(\mathbf{x}) = \mathbf{w}^T \Phi(\mathbf{x}) + b, \tag{1}$$

where $\mathbf{x}$ is an input vector whose class label is unknown. The class label of $\mathbf{x}$ is determined according to $y = \mathrm{sign}[f^{\mathrm{svm}}(\mathbf{x})]$. If $y$ is positive, $\mathbf{x}$ belongs to the positive class whose label is 1. Otherwise, it belongs to the negative class of $-1$.

The aim of SVM is finding the optimal hyperplane which minimizes the cost function consisting of two criteria, namely margin maximization and error minimization. The final result of Eq. (1) using a Lagrangian formulation is shown as follows[12]:

$$f^{\mathrm{svm}}(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b, \tag{2}$$

where $K(\cdot, \cdot)$ is a kernel function, which defines an inner product in the feature space $\mathbb{R}^{d'}$. Next, the coefficients $\alpha_i$ are obtained by solving the following convex Quadratic Programming (QP) problem:

$$\begin{aligned} &\text{Maximize} && \textstyle\sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j), \\ &\text{subject to} && \textstyle\sum_{i=1}^{N} \alpha_i y_i = 0, \\ &&& 0 \leqq \alpha_i \leqq C, \ (i = 1, 2, \cdots, N), \end{aligned} \tag{3}$$

where $\alpha_i$ are the Lagrange multipliers and $C$ is a constant which represents a tradeoff between the number of misclassified samples in the training set and separation of the rest samples with maximum margin.

It should be noted that the output values $f^{\mathrm{svm}}(\mathbf{x})$ in Eq. (2) is obtained based on the distance between $\Phi(\mathbf{x})$ and the hyperplane, and it can represent the degree of how the input feature vectors are likely to positive examples[11]. Therefore, in this paper, we focus on the output values of $f^{\mathrm{svm}}(\mathbf{x})$ to realize target object detection by using these values.

## 3. Classification Performance Improvement Using Target Object Detection

This section shows the performance improvement method of image classification. Given a set of images, which contain positive examples including target objects and negative examples not including them, we first calculate a separating hyperplane of SVM as shown in the previous section. In order to improve the image classification performance, we adopt a target object detection scheme. Specifically, the proposed method performs the following two procedures:

**Step 1** Detection of areas where target objects are included based on distances from the hyperplane of SVM

**Step 2** Calculation of a new hyperplane from modified feature vectors which are extracted from the detected areas

First, in **Step 1**, local blocks, which contain the target object, are detected from the positive examples by using the hyperplane of SVM. These blocks are called "positive-blocks", hereafter. Next, in **Step 2**, a new hyperplane is calculated by using modified feature vectors obtained from these positive-blocks. Since the classifiers can be obtained from only the target objects, the other background areas do not affect the learning process. Therefore, by automatically narrowing down the area where the target objects are located in the training images, the classification performance improvement can be realized.

In this section, we first explain the details of the SVM-based detection of the positive-blocks (**Step 1**) in 3.1. Secondly, we show the calculation of a new hyperplane of SVM by using the modified feature vectors of detected positive-blocks (**Step 2**) in 3.2. Finally, multiclass image classification using the proposed method is presented in 3.3.

### 3.1 Detection of Positive-Blocks

In this subsection, we explain the detection of positive-blocks. Let $F_i^{\mathrm{pos}} (i = 1, 2, \cdots, N)$ denote the positive examples, *i.e.*, images including target objects, in the training dataset, where $N$ is the number of the positive examples. First, for each image $F_i^{\mathrm{pos}}$, the proposed method clips local blocks of various sizes, and these clipped blocks are defined as $g_{i,s,m}^{\mathrm{pos}} (s = 1, 2, \cdots, S, m = 1, 2, \cdots, M_{i,s}^{\mathrm{pos}})$. Note that since the notation of the clipped local blocks $g_{i,s,m}^{\mathrm{pos}}$ is complicated, we explain its details below.

First, $s$ represents an index which indicates a group which

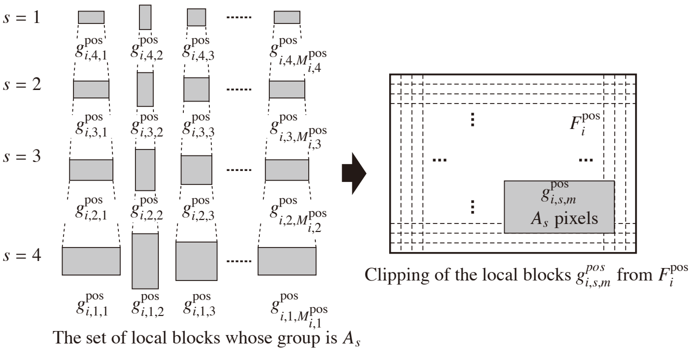



**Fig. 1** An example of clipping local-blocks $g^{\mathrm{pos}}_{i,s,m}$ from the positive example $F^{\mathrm{pos}}_i$.

**Table 1** Four kinds of groups $A_s(s = 1, 2, 3, 4)$, whose areas are $(0.8)^s A_0$ (pixels), used in our method, where $A_0 = 65536$.

| $A_1$ | $A_2$ | $A_3$ | $A_4$ |
|---|---|---|---|
| $0.8A_0$ | $(0.8)^2A_0$ | $(0.8)^3A_0$ | $(0.8)^4A_0$ |

**Table 2** Aspect ratios (vertical:horizontal) used for clipping local blocks of each group $A_s$.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 4 : 1 | 3 : 1 | 2 : 1 | 3 : 2 | 1 : 1 | 2 : 3 | 1 : 2 | 1 : 3 | 1 : 4 |

has the same area, *i.e.*, "the same number of pixels within clipped local blocks". In our method, local blocks of various sizes and aspect ratios are clipped from each positive example $F^{\mathrm{pos}}_i$. Then we group local images, whose areas are the same, together and represent this group as $A_s$ by using the index $s$ as shown in **Fig. 1**. In the proposed method, we set $S = 4$, *i.e.*, the four groups $A_s(s = 1, 2, 3, 4)$ including different areas are used as shown in **Table 1**. For each group $A_s$, local blocks with several aspect ratios are obtained from the positive example $F^{\mathrm{pos}}_i$. Specifically, we use local windows of nine kinds of aspect ratios shown in **Table 2** and clip the local blocks $g^{\mathrm{pos}}_{i,s,m}(m = 1, 2, \cdots, M^{\mathrm{pos}}_{i,s})$ by using these local windows. Note that $M^{\mathrm{pos}}_{i,s}$ represents the number of the clipped local blocks, whose group is $A_s(s = 1, 2, \cdots, S)$, in the positive example $F^{\mathrm{pos}}_i$. As described above, $M^{\mathrm{pos}}_{i,s}$ local images have various aspect ratios, but the areas, *i.e.*, the numbers of pixels, within the blocks are the same. In this way, we obtain $M^{\mathrm{pos}}_{i,s}$ local blocks for each group $A_s$ from $F^{\mathrm{pos}}_i$. This means $\sum_{s=1}^{S} M^{\mathrm{pos}}_{i,s}$ local blocks are obtained for each positive example $F^{\mathrm{pos}}_i$ as shown in an example of Fig. 1.

In the proposed method, we calculate feature vectors $\mathbf{x}^{\mathrm{pos}}_{i,s,m}$ from $g^{\mathrm{pos}}_{i,s,m}$. Then, for each positive example $F^{\mathrm{pos}}_i$, $S$ kinds of positive blocks $b^{\mathrm{pos}}_{i,s}(s = 1, 2, \cdots, S)$ are calculated. Specifically, for each group $A_s(s = 1, 2, \cdots, S)$, its positive-block is obtained as follows:

$$b^{\mathrm{pos}}_{i,s} = g^{\mathrm{pos}}_{i,s,m^{\mathrm{opt}}_{i,s}}, \tag{4}$$

where

$$m^{\mathrm{opt}}_{i,s} = \arg \max_{m=1,2,\cdots,M^{\mathrm{pos}}_{i,s}} f^{\mathrm{svm}}(\mathbf{x}^{\mathrm{pos}}_{i,s,m}). \tag{5}$$

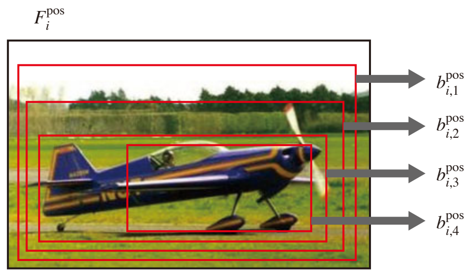


**Fig. 2** An example of positive-blocks $b^{\mathrm{pos}}_{i,s}$ detected from the positive example $F^{\mathrm{pos}}_i$.

In the above equation, $f^{\mathrm{svm}}(\mathbf{x}^{\mathrm{pos}}_{i,s,m})$ is the output value of SVM as shown in Eq. (2) of the previous section. Note that the hyperplane of the SVM is calculated from the positive examples $F^{\mathrm{pos}}_i(i = 1, 2, \cdots, N)$ and other negative examples. From Eq. (4), the proposed method selects the positive-block $b^{\mathrm{pos}}_{i,s}$ for each group $A_s(s = 1, 2, \cdots, S)$. Therefore, $S$ kinds of the positive-blocks are obtained for each positive example $F^{\mathrm{pos}}_i$ as shown in **Fig. 2**.

As shown in the previous section, the output value $f^{\mathrm{svm}}(\mathbf{x}^{\mathrm{pos}}_{i,s,m})$ is based on the distance between the vector $\Phi(\mathbf{x}^{\mathrm{pos}}_{i,s,m})$ and the hyperplane in the high-dimensional feature space. Furthermore, $f^{\mathrm{svm}}(\mathbf{x}^{\mathrm{pos}}_{i,s,m})$ represents the degree of how the input feature vectors $\mathbf{x}^{\mathrm{pos}}_{i,s,m}$ are likely to the positive examples corresponding to the target object. Therefore, by selecting the local blocks whose outputs of SVM become maximum, the detection of the positive-blocks including the target objects becomes feasible.

### 3.2 Calculation of New Hyperplane from Modified Feature Vectors

This subsection shows the update of the hyperplane of SVM by using the positive-blocks $b^{\mathrm{pos}}_{i,s}(i = 1, 2, \cdots, N, s = 1, 2, \cdots, S)$ detected in the previous subsection. As shown in Fig. 2, $S$ positive-blocks $b^{\mathrm{pos}}_{i,s}(s = 1, 2, \cdots, S)$ are detected for each positive example $F^{\mathrm{pos}}_i$. They have different areas, *i.e.*, different number of pixels, but tend to include the target object. Before updating the hyperplane of SVM, the proposed method adopts one procedure which selects one optimal positive-block from $b^{\mathrm{pos}}_{i,1}, b^{\mathrm{pos}}_{i,2}, \cdots, b^{\mathrm{pos}}_{i,S}$ for each positive example $F^{\mathrm{pos}}_i$. This procedure aims to select the smallest positive-block which circumscribes the target object. If the positive-blocks are larger than the target object, they also include other background areas. On the other hand, if positive-blocks are smaller than the target object, they may lose important information of the target object. Therefore, in the proposed method, we try to select the smallest positive-block circumscribing the target object. Specifically, as shown in **Fig. 3**, the smallest $s$ satisfying

$$f^{\mathrm{svm}}(\mathbf{x}^{\mathrm{pos}}_{i,s}) - f^{\mathrm{svm}}(\mathbf{x}^{\mathrm{pos}}_{i,s+1}) > 0. \tag{6}$$

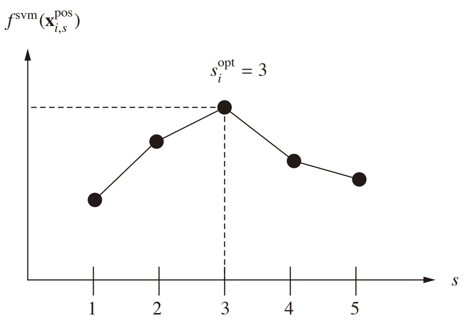



**Fig. 3** An example of "the relationship between $s$ and $f^{\mathrm{svm}}(\mathbf{x}_{i,s}^{\mathrm{pos}})$" and "the detection of $s_i^{\mathrm{opt}}$".


**Fig. 4** Example images of Caltech 101 and Caltech 256 datasets.

is selected as $s_i^{\mathrm{opt}}$ to provide the optimal positive-block $b_{i,s_i^{\mathrm{opt}}}^{\mathrm{pos}}$, where $\mathbf{x}_{i,s}^{\mathrm{pos}}$ is a feature vector of $b_{i,s}^{\mathrm{pos}}$. Generally, for $s(s < s_i^{\mathrm{opt}})$, their positive-blocks tend to include the background, and the output of SVM tends to become smaller. On the other hand, for $s(s > s_i^{\mathrm{opt}})$, their positive-blocks cannot include the whole area of the target object, and the output of SVM tends to also become smaller as shown in Fig. 3. Therefore, the proposed method selects $s_i^{\mathrm{opt}}$ by using Eq. (6).

In this way, we can obtain the optimal positive-block $b_{i,s_i^{\mathrm{opt}}}^{\mathrm{pos}}$ for each positive example $F_i^{\mathrm{pos}}$. Therefore, the proposed method updates the hyperplane of SVM from these positive-blocks $b_{i,s_i^{\mathrm{opt}}}^{\mathrm{pos}}(i = 1, 2, \cdots, N)$ circumscribing the target objects. Note that feature vectors obtained from $b_{i,s_i^{\mathrm{opt}}}^{\mathrm{pos}}$ correspond to the modified feature vectors in this paper. Then improvement of the classification performance can be expected by using the updated hyperplane.

### 3.3 Multiclass Image Classification

This subsection presents the multiclass classification based on the proposed method. Traditionaly, the SVM-based multiclassification has been proposed[4], and we also follow this scheme.

By using the procedures shown in the previous subsections, the proposed method can update the classifier from the positive-blocks $b_{i,s_i^{\mathrm{opt}}}^{\mathrm{pos}}$. In the multiclass classification problem including the two class problem, we generally prepare training samples for each class. Given $K$ classes, we denote each class as $k(k = 1, 2, \cdots, K)$ and define its training examples as $F_i^k(i = 1, 2, \cdots, N^k, N^k$ is the number of training examples belonging to class $k)$.

First, for each class $k$, we calculate an initial hyperplane $H^k$ of SVM by using $F_i^k(i = 1, 2, \cdots, N^k)$ as positive examples and $F_i^{k'}(k' = \{1, 2, \cdots, K | k' \neq k\}, i = 1, 2, \cdots, N^{k'})$ as negative examples. Next, by using the obtained hyperplane $H^k$, the calculation of the optimal positive-blocks $b_{i,s_i^{\mathrm{opt}}}^{k}$ is performed, and the updated hyperplane $\hat{H}^k$ is obtained for each class $k$. Note that the updated hyperplane is calculated by using $b_{i,s_i^{\mathrm{opt}}}^{k}$ as positive examples and "$F_i^{k'}$ and $b_{i,s_i^{\mathrm{opt}}}^{k'}$" $(k' = \{1, 2, \cdots, K | k' \neq k\}, i = 1, 2, \cdots, N^{k'})$ as negative examples. Then, for all classes $k(k = 1, 2, \cdots, K)$, the proposed method can provide the classifiers which can consider their target objects.

Finally, the proposed method performs classification of a new sample by using the obtained hyperplane $\hat{H}^k(k = 1, 2, \cdots, K)$. Given a feature vector $\mathbf{x}$ of a target sample whose belonging class is unknown, the proposed method determines its class $\hat{k}$ as follows:

$$\hat{k} = \arg \max_{k=1,2,\cdots,K} f_k^{\mathrm{svm}}(\mathbf{x}), \tag{7}$$

where $f_k^{\mathrm{svm}}(\cdot)$ is the function of class $k$, *i.e.*, it represents the hyperplane $\hat{H}^k$. Then, by using Eq. (7), image classification becomes feasible.

## 4. Experimental Results

In this section, we verify the performance of the proposed method by using the general object datasets including Caltech 101[13] and Caltech 256[14].

### 4.1 Datasets Used for Verification

In this subsection, we explain the details of the datasets Caltech 101 and Caltech 256, in which each image contains a certain object and a cluttered background. Their details are shown as follows.

**(i) Caltech 101**

This dataset (collected by Fei-Fei *et al.*[13]) contains 9,144 images in 101 object categories including animals, vehicles, flowers, buildings, *etc*. The number of images per category varies from 31 to 800. The significance of this dataset is its large inter-class variability.

**(ii) Caltech 256**

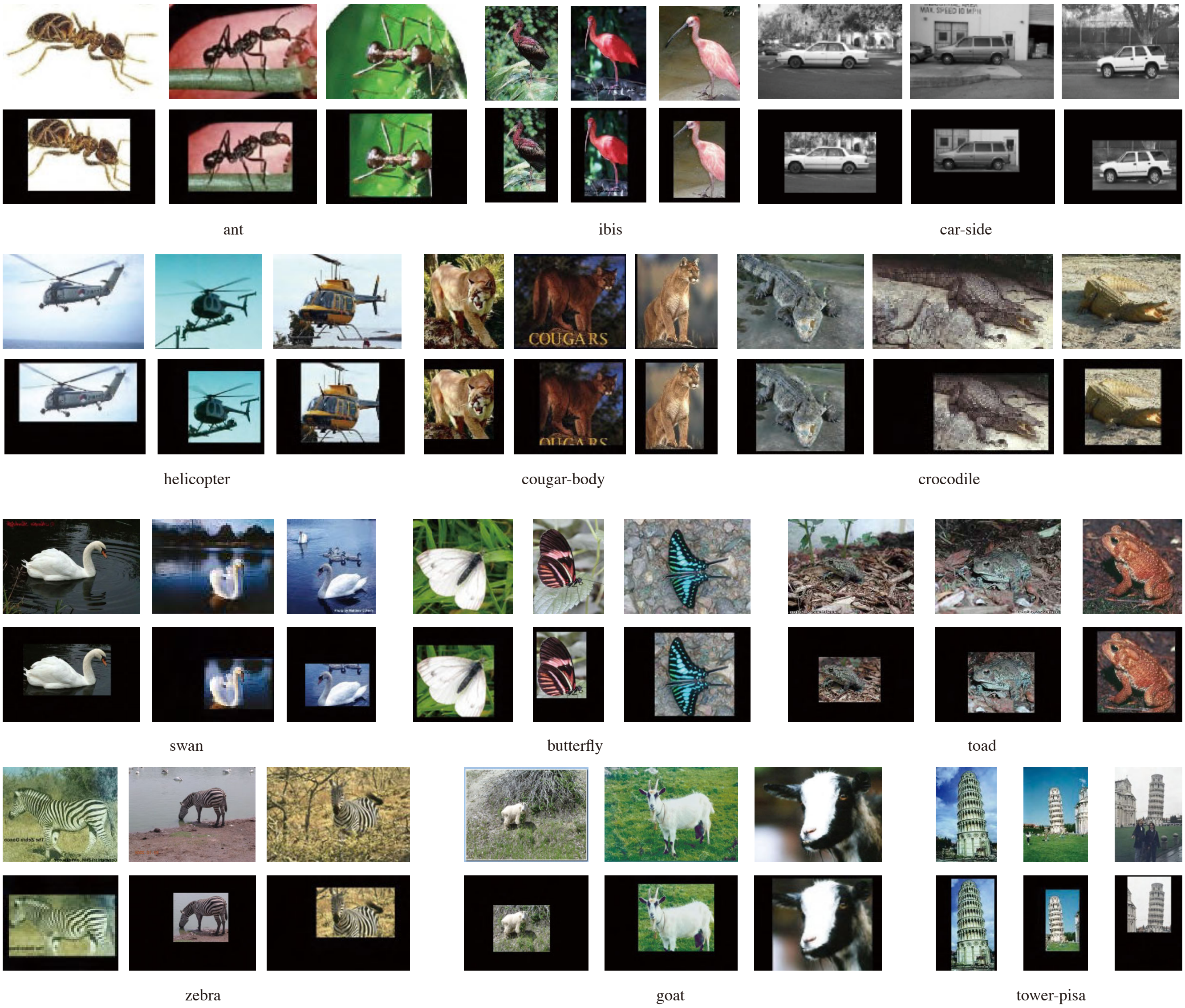



**Fig. 5** An examples of target object detection from Caltech 101 and Caltech 256. Categories "ant", "ibis", "car-side", "helicopter", "cougar-body", "crocodile" are included in Caltech 101. All categories shown in this figure are included in Caltech 256. This means Caltech 256 includes Caltech 101.

This dataset (collected by Griffin *et al.*[14]) contains 30, 607 images from 256 object categories, and each category contains at least 80 images. Caltech 256 is an extension of Caltech 101. The significance of this dataset is its large inter-class variability, as well as a larger intra-class variability than Caltech 101. Furthermore, there is no alignment among the object categories.

**Fig. 4** gives some example images. Compared with Caltech 101, Caltech 256 presents much greater variation in object size, location and pose.

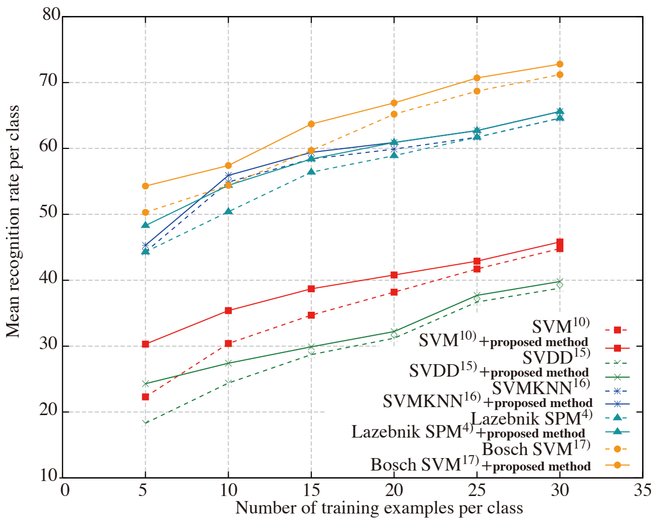
### 4.2 Image Classification Results

In this subsection, we first verify the efficiency of the target object detection realized by the proposed method. **Fig. 5** shows detection results of the optimal positive-blocks. As shown in this figure, it can be seen that the proposed method can detect blocks circumscribing the target objects.

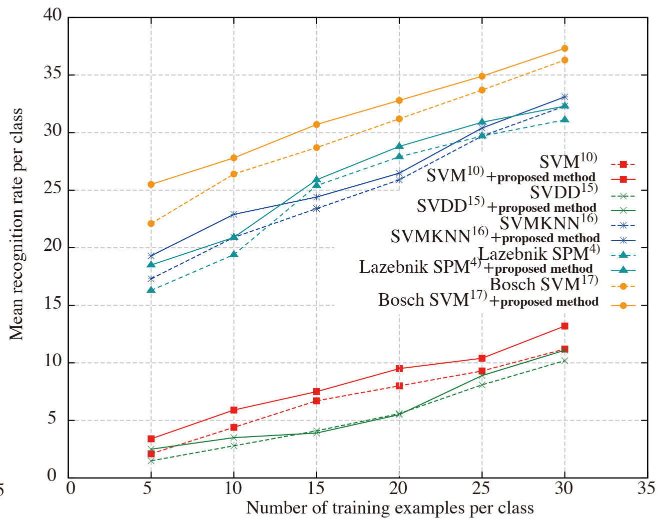
**Table 3** Classification accuracy (%) comparison on Caltech 101 and Caltech 256.

| Methods | Datasets | Conventional method | Proposed method |
|---|---|---|---|
| SVM[10] | Caltech 101 | 44.8 ± 0.7 | **46.2 ± 0.9** |
| | Caltech 256 | 11.6 ± 0.5 | **13.2 ± 0.9** |
| SVDD[15] | Caltech 101 | 38.8 ± 0.6 | **39.9 ± 0.8** |
| | Caltech 256 | 10.1 ± 0.7 | **11.1 ± 0.5** |
| SVM-KNN[16] | Caltech 101 | 64.6 ± 0.3 | **65.3 ± 0.5** |
| | Caltech 256 | 32.3 ± 0.5 | **33.1 ± 0.8** |
| Lazebnik SPM[4] | Caltech 101 | 64.2 ± 0.3 | **65.6 ± 0.7** |
| | Caltech 256 | 31.1 ± 0.8 | **32.5 ± 0.6** |
| Bosch SVM[17] | Caltech 101 | 71.2 ± 0.3 | **72.8 ± 0.5** |
| | Caltech 256 | 36.3 ± 0.5 | **37.6 ± 0.7** |

Next, we show the image classification results obtained by using the two datasets, Caltech 101 and Caltech 256. We followed the experimental setup of Grauman and Darrell[18] and J. Zhang et al.[19], namely, we trained on 30 images per




(a) Caltech 101.




(b) Caltech 256.

**Fig. 6** Relationship between the number of training examples per class and the mean recognition rate per class.

class and tested on the rest. For efficiency, we limit the number of test images to 50 for Caltech 101 and 25 for Caltech 256 per class. **Table 3** provides the resultant classification performance on these two datasets. Note that we repeated this verification three times and then calculated the average classification accuracy and the corresponding standard deviation.

As shown in Table 3, we applied the proposed method to several conventional classification methods based on SVM. In order to confirm the improvement by our method, we show the classification results with those of individual classification methods. Note that in this experiment, SVM in[10)] and Support Vector Data Description(SVDD) in[15)] adopted SIFT features for constructing their classifiers. The settings of SIFT features were based on those in[17)]. In addition, experimental settings of SVM-KNN[16)], Lazebnik SPM[4)] and Bosch SVM[17)] were followed in each reference.

When detecting positive-blocks by the proposed method, we used different image representation with those used in the conventional methods. Specifically, we used two representations[17)], Pyramid Histogram Of visual Words (PHOW) and Pyramid Histograms Of Gradients (PHOG) which are calculated based on SIFT and HOG, respectively. Note that in this calculation, we used the same settings shown in[17)]. In this experiment, HOG is discretized into 40 orientation bins. Furthermore, SPM kernel[4)] with three levels of $1 \times 1$, $2 \times 2$ and $4 \times 4$ is adopted. After detecting the positive-blocks, we calculated the classifiers by the conventional methods based on the same conditions shown in the above.

From the results shown in Table 3, we can see that the use of the proposed method always outputs better performance than the individual one. In addition, for both datasets, we randomly select 5, 10, 15, 20, 25, 30 images for training, respectively, and test on the rest. In this experiment, we verify the robustness of the image classification performance based on the proposed and conventional methods for the number of training images. This verification scheme was adopted among several reports[4)14)16)], and thus, we also adopted it in this experiment. **Fig. 6** shows our results and the results obtained by others as the number of training images is varied. It can be also seen that the performance improvement is realized by introducing our method into those SVM-based conventional classification methods.

From the above experimental results on the two datasets, we can verify the effectiveness of the proposed method. Then performance improvements of SVM-based classification methods can be achieved by the proposed method.

## 5. Conclusions

This paper has presented a method to improve performance of SVM-based classification, which contains a target object detection scheme. The proposed method is composed of two procedures, detection of target objects and calculation of a hyperplane from modified feature vectors. By using these procedures, we can obtain blocks circumscribing the target objects and calculate the new hyperplane of SVM by the modified feature vectors which extracted from these blocks. Then improvements of the classification performance becomes feasible by using the modified hyperplane. Experimental results using the two datasets, Caltech 101 and Caltech 256, have shown the effectiveness of the proposed method.

## Acknowledgment

This work was partly supported by Grant-in-Aid for Scientific Research on Innovative Areas 24120002 from the MEXT.

## References

1) D.G.Lowe: "Distinctive image features from scale-invariant keypoints", Int. J. Comput. Vis., **60**, 2, pp. 91-110, (2004)
2) N. Dalal and B. Triggs: "Histograms of oriented gradients for human detection", in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., **1**, pp. 886-893, (2005)
3) Li. F.-F. and P. Perona: "A Bayesian hierarchical model for learning natural scene categories", in Proc. IEEE Conf. Comput. Vis. Pattern Recognit, **2**, pp. 524-531, (2005)
4) S. Lazebnik, C. Schmid and J. Ponce: "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories", in Proc. IEEE Conf. Comput. Vis. Pattern Recognit, **2**, pp. 2169-2178, (2006)
5) J. van Gemert, J. Geusebroek, C. Veenman and A. Smeulders: "Kernel codebooks for scene categorization", in Proc. Eur. Conf. Comput. Vis., pp. 696-709, (2008)
6) J. Yang, K. Yu, Y. Gong and T. Huang: "Linear spatial pyramid matching using sparse coding for image classification", in Proc. IEEE Conf. Comput. Vis. Pattern Recognit, pp. 1794-1801, (2009)
7) S. Gao, I. Tsang, L. Chia and P. Zhao: "Local features are not lonely Laplacian sparse coding for image classification", in Proc. IEEE Conf. Comput. Vis. Pattern Recognit, pp. 3555-3561, (2010)
8) J. Wang, J. Yang, K. Yu, F. Lv and T. Huang: "Locality-constrained linear coding for image classification", in Proc. IEEE Conf. Comput. Vis. Pattern Recognit, pp. 3360-3367, (2010)
9) L. Liu, L. Wang and X. Liu: "In defense of soft-assignment coding", in Proc. Int. Conf. Comput. Vis., pp. 2486-2493, (2011)
10) Christopher J. C. Burges: "A Tutorial on Support Vector Machines for Pattern Recognition", Data Min. Knowl. Discov., **2**, pp. 121-167, (1998)
11) J. C. Platt: "Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods", In Advances In Large Margin Classifiers, pp. 61-74, (1999)
12) V. N. Vapnik, "Statistical Learning Theory", New York: Wiley, (1998)
13) Li. F.-F., R. Fergus and P. Perona: "Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories", Comput. Vis. Image Understand., **106**, pp.59-70, (2007)
14) G. Griffin, A. Holub and P. Perona: "Caltech-256 object category dataset", Dept. Comput. Sci., California Inst. Technology, Tech. Rep.7694, (2007)
15) David M. J. Tax and Robert P. W. Duin: "Support Vector Data Description", Mach. Learn., **54**, pp. 45-66, (2004)
16) H. Zhang, A. C. Berg, M. Maire and J. Malik: "SVM-KNN: Discriminative nearest neighbor classification for visual category recognition", in Proc. IEEE Conf. Comput. Vis. Pattern Recognit, **2**, pp. 2126-2136, (2006)
17) A. Bosch, A, Zisserman and X. Munoz: "Representing shape with a spatial pyramid kernel", Proc. ACM Conf. on Image and Video Retrieval, pp. 401-408, (2007)
18) K. Grauman and T. Darrell: "The pyramid match kernel: Discriminative classication with sets of image features", in Proc. Int. Conf. Comput. Vis., **2**, pp, 1458-1465, (2005)
19) J. Zhang, M. Marszalek, S. Lazebnik and C. Schmid: "Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study", Int. J. Comput. Vision, **73**, pp. 213-238, (2007)

**Soh Yoshida** received his B.S. degree in Electronics and Information Engineering from Hokkaido University, Sapporo, Japan in 2012. He is currently pursuing an M.S. degree at the Graduate School of Information Science and Technology, Hokkaido University. His research interests include image and video processing. He is a member of IEICE.

**Hiroshi Okada** received his B.S. and M.S. degrees in Electronics and Information Engineering from Hokkaido University, Sapporo, Japan in 2009 and 2011, respectively. He is currently working in Hokkaido Electric Power Company. His research interests include digital image processing.

**Takahiro Ogawa** received his B.S., M.S. and Ph.D. degrees in Electronics and Information Engineering from Hokkaido University, Japan in 2003, 2005 and 2007, respectively. He is currently an assistant professor in the Graduate School of Information Science and Technology, Hokkaido University. His research interests are digital image processing and its applications. He is a member of the IEEE, EURASIP,IEICE, and Institute of Image Information and Television Engineers (ITE).

**Miki Haseyama** received her B.S., M.S. and Ph.D. degrees in Electronics from Hokkaido University, Japan in 1986, 1988 and 1993, respectively. She joined the Graduate School of Information Science and Technology, Hokkaido University as an associate professor in 1994. She was a visiting associate professor of Washington University, USA from 2005 to 2006. She is currently a professor in the Graduate School of Information Science and Technology, Hokkaido University. Her research interests include image and video processing and its development into semantic analysis. She is a member of the IEEE, IEICE, Institute of Image Information and Television Engineers (ITE) and Acoustical Society of Japan (ASJ).