Improve the performance of BOF-driven HOG/ SIFT. Experimental results show that the BOF-driven HOG/SIFT outperforms the original ones in pedestrian detection (for HOG), scene matching, and image classification (for SIFT). Our proposed BOF-driven HOG/SIFT can be easily applied as replacements of the original HOG/SIFT in current systems since they are generalized versions of the original ones.

key words: bag of features, connection with HOG/SIFT, pedestrian detection, scene matching, image classification

1. Introduction

Image classification is a fundamental domain in computer vision research. The purpose of image classification is to classify an image into corresponding categories according to its visual information. One typical application of image classification is image retrieval, which aims to find images similar to the queried one.

Bag of features (BOF) model, which borrows idea from document classification, has become prevailing in image classification due to its simplicity and effectiveness. BOF represents an image by an occurrence histogram, which is constructed based on the count of occurrences of local descriptors in this image subject to a vocabulary. Usually, BOF goes through four steps to build an occurrence histogram: descriptor extraction, learning vocabulary, coding of descriptors, pooling of descriptors.

A great number of methods have been proposed to improve the performance of BOF, focusing on different steps of BOF. In the descriptor extraction step, for the sake of obtaining more general and robust descriptors, several techniques, such as deep belief network [2] and mid-level descriptors [3], have been introduced to capture the visual content of local regions in the image more precisely. Studies on the learning of vocabulary also have been presented. Work in [1] learns structured-vocabulary directed by fisher discrimination criterion, enabling the learned vocabulary to carry certain class information. In terms of the step of descriptor coding, Sparse coding [4], Local soft-assignment and Locality-constrained linear coding [5] have made great progresses by minimizing the noise while clamping the descriptor to its relative visual words tightly. The incorporation of spatial information in descriptor pooling step is conducive to recording more discriminative class information, which also helps to raise the accuracy of BOF. Spatial pyramid matching (SPM) technique [6] pools descriptors in a coarse-to-fine grids of the image. Object-centric spatial pooling (OCP) [8] first localizes the pooling region and then pools descriptors in foreground and background regions respectively. Both of these methods have achieved remarkable success in image classification task.

Histogram of oriented gradient (HOG) [1] usually serves as local feature descriptor in the first step of BOF. HOG is capable of retaining discriminative class information while allowing small variations. Due to this property, HOG is widely used in the pedestrian detection task. Experiments in [1] showed that pedestrian detection method using HOG significantly outperforms those former methods based on wavelet and PCA-SIFT [23]. HOG also obtains desirable results in other applications like image classification. To further improve the performance of HOG, plenty of studies have been conducted. Study in [9] uses weak-label structural SVM (WL-SSVM) to learn weights for each entry in the HOG descriptor. Study in [10] adds LBP as a complement to HOG and locates the occlusion region in terms of the SVM output.

Scale invariant feature transform (SIFT) [12] also serves as local feature descriptor in most applications of BOF. SIFT boasts a strong robustness against scale transformation and has establish its dominant position in applications such as image matching and image stitching. Numerous works have been proposed to enhance SIFT’s robustness by magnifying the expressive power of SIFT’s descriptor. Study in [23] combines SIFT with PCA, reduces the dimension of SIFT descriptor while maintaining comparatively agreeable performance. The substitute of the original gradient computation method in SIFT by smooth derivative filters as in [24] gives better performance in image matching and object recognition experiments as well.
In our previous works [13], [14], we improved the performance of HOG/SIFT based on the connection between them and BOF in descriptor construction. The contributions of these papers are two-fold. We a) first revealed that the method used to construct the descriptor of a local region of an image in HOG/SIFT is similar to the one used to build the occurrence histogram of an entire image in BOF. Upon this interpretation, we then b) introduced approaches proposed for BOF into HOG/SIFT to improve their performance. Indeed, the original HOG/SIFT descriptor can be viewed as simplified versions of our BOF-driven HOG/SIFT. As compared to the original HOG in the experiment of pedestrian detection, our BOF-driven HOG reduces the error rate from 19.22% to 7.98% when $FPPW = 10^{-3}$. The BOF-driven SIFT also obtains better performance against the original SIFT in experiments of scene matching and image classification. Because the BOF-driven HOG/SIFT are extensions of the original ones by BOF approaches, they can be easily embedded into current frameworks using the original HOG/SIFT. Moreover, more advances in HOG/SIFT as well as BOF can be applied to further improve the performance of the BOF-driven HOG/SIFT. In this paper, we study this connection between BOF and HOG/SIFT deeper and give more details of our previous works.

The rest of this paper is organized as follows. In Sect. 2, we review recent works which improve the performance of HOG/SIFT in a BOF fashion. In Sect. 3, we give a brief introduction of BOF, HOG/SIFT. In Sect. 4, we show how the method for constructing the descriptor in HOG/SIFT can be viewed as a variant of that used to construct the occurrence histogram in BOF. In Sect. 5, we introduce recent approaches proposed for BOF into the construction of descriptor in HOG/SIFT. Section 6 describes experiments in pedestrian detection to evaluate the performance of the BOF-driven HOG. Section 7 presents the evaluation of BOF-driven SIFT’s performance compared with the original SIFT. Section 8 gives the conclusion of this work.

2. Related Studies

Several studies, although not explicitly and deliberately, have presented schemes to improve the performance of HOG/SIFT in a BOF fashion. Study in [15] learns a vocabulary of filters and then assign pixels in a block to filters in which the pixel gets maximal response. A descriptor, in which each entry represents its respective filter and the value of the entry indicates how many pixels are assigned to this filter, is built upon assignment results. Since those filters are learned on data with class information, the generated descriptor has the power of discrimination and is more suitable for classification task. Study in [16] pools gradients on a spatial pyramid of the whole image in order to retain more visual information. Gradient location orientation histogram (GLOH) in [25] extends the SIFT descriptor by computing the descriptor on a manually defined log-polar grid, which raises the accuracy of SIFT in image matching. Differing from these studies, our works established a tight connection between BOF and HOG/SIFT, which allows approaches of BOF to be introduced into HOG/SIFT to improve their performance. Therefore, we can anticipate that more useful approaches of BOF can be applied into our BOF-driven HOG/SIFT to achieve better performance.

3. BOF, HOG/SIFT

3.1 BOF

Generally, BOF follows 4 steps to build the vocabulary-based occurrence histogram of an image: descriptor extraction, learning vocabulary, coding of descriptors, pooling of descriptors. In the following, we go through these 4 steps in short to show how an occurrence histogram is generated. The (a) in Fig. 1 schematizes this procedure.

Let $p_i \in R^r$ denote a descriptor extracted from some local region, $B \in R^{r \times n}$ a vocabulary with $r$ visual words, $b_j \in R^n$ the $j$th visual word in this vocabulary. Let $u_{ij}$ denote the coefficient of $p_i$ to $b_j$.

3.1.1 Descriptor Extraction

Local feature descriptors, such as HOG/SIFT, are used to encapsulate distinct information of local regions in an image. Usually, those descriptors are randomly or densely extracted. The dense extraction has been verified to have better performance in BOF [17] than random extraction.

3.1.2 Learning Vocabulary

Vocabulary comprises distinctive descriptors (visual words) which are discriminative enough to represent class-wise feature information. For example, a wheel will be a good feature to discriminate car from pedestrian, then descriptors of the wheel should be incorporated into the vocabulary. Usually, the vocabulary is generated on extracted descriptors

![Fig. 1](image-url) (a): Four steps for the construction of occurrence histogram of an image in BOF model. Symbols (cross, multiple sign and circle) represent descriptors belonging to different visual words. (b): The method used to construct the descriptor of an local region (red square) of an image in HOG/SIFT. Arrows represent gradient orientations belonging to different clusters.
from Sect. 3.1.1 by certain methods such as k-means. Generally, the larger the size of the vocabulary is, the more data information it can carry. Nonetheless, a large size vocabulary comes with more computational cost.

3.1.3 Coding of Descriptors

Given descriptors extracted from an image, we would like to know what visual words in the vocabulary those descriptors are close to and how close they are. The procedure seeking reasonable assignments of those descriptors to their corresponding visual words in a vocabulary is called the coding of descriptors. One naive method is to assign a descriptor to just one nearest visual word measured by the euclidian distance between them. Apart from this simple method, numerous methods such as sparse coding are proposed to achieve more robust, reasonable assignments.

3.1.4 Pooling of Descriptors

During the coding step, coefficients are obtained for each descriptor subject to their corresponding visual words. To summarize those coefficients of all descriptors for each visual word, methods like max-pooling, sum-pooling and average pooling are proposed. Max-pooling selects the largest coefficient for each visual word. Sum-pooling adds up all coefficients of each visual word. Average pooling also adds up all coefficients of each visual word but divides the sum by the number of coefficients.

3.2 HOG and SIFT

HOG, as can be inferred from its name, groups gradient orientations to their corresponding clusters to build a descriptor. Since the construction method of descriptor in SIFT is similar to that of HOG, we only introduce the method in HOG in detail and what the SIFT differs from that of HOG briefly. The (b) in Fig. 1 schematizes this method.

Firstly, gradients are extracted from each pixel in a local region (usually of size 16 × 16 in both HOG and SIFT, we call this local region block afterwards) of an image (such as the red square in (b) of Fig. 1). In order to tolerate significant positional shift of gradients in the block, the block is usually evenly divided into cells (usually, there are 2 × 2 cells in HOG and 4 × 4 cells in SIFT).

Secondly, the orientation value of the gradient of each pixel in a cell is assigned to its corresponding clusters (usually, the total number of clusters is 9 in HOG and 8 in SIFT), which are determined by the distances between this orientation value and those cluster centers, which are also orientation values. For the purpose of avoiding abrupt changes in the cell descriptor resulted by boundary effect and smooth shift of gradient from one cell to another, trilinear interpolation is applied to assign this orientation value to its nearest two clusters (rather than only one) in the same cell descriptor, and the same clusters in its neighboring cell descriptors. Also, a gaussian weight, which is calculated based on the relative spatial position of the orientation value with respect to the center of the block, is multiplied during assignment to give less emphasis on orientation values that are far away from the block center. Along with this, the magnitude value related to this orientation value is also multiplied as a weight.

Thirdly, assignment results of all orientation values to their corresponding clusters are added up for each cluster to build the cell descriptor. All cell descriptors are then concatenated to form the block descriptor. As a result, the number of entries in the block descriptor is 2 × 2 × 9 = 36 for HOG and 4 × 4 × 8 = 128 for SIFT. Normalization, as well as thresholding of large entry values, are applied afterwards for the block descriptor to achieve invariance to affine changes in illumination.

4. Interpret HOG/SIFT as BOF in Descriptor Construction

As can be observed from Sect. 3.1 and Sect. 3.2, methods for histogram construction in BOF and descriptor construction in HOG/SIFT have a tight connection. In the following, we address how this connection become explicit following the typical streamline of constructing the occurrence histogram in BOF as illustrated in Sect. 3.1.

Let \( p, b \), and \( r \) represent a gradient. Let \( (p_i, p_m) \) and \( (b_j, b_m) \) denote the orientation value and magnitude value of \( p_i \) and \( b_j \), respectively. If we scale the range of orientation dimension and magnitude dimension into \([0, \pi]\) and \([0, 1]\), then \( p_i, b_j \in [0, \pi] \) and \( p_m, b_m \in [0, 1] \).

4.1 Descriptor Extraction

The orientation value \( p_i \) of a gradient extracted from a pixel in a cell can be viewed as the descriptor. In this case, \( n = 1 \). Then, the procedure of calculating the orientation value for all pixels in the cell is dense descriptor extraction.

4.2 Learning Vocabulary

We can treat those cluster centers in HOG/SIFT as manually defined visual words of a vocabulary \( B \in \mathbb{R}^{\times 1} \), which differs from typical BOF using data-driven vocabulary generated by k-means or structured vocabulary learning method as in [11]. Then, \( r = 9 \) in HOG and \( r = 8 \) in SIFT. These visual words are evenly distributed along the orientation dimension of the gradient space. Therefore, a visual word \( b_j \) in \( B \) can be simply calculated as follows:

\[
b_j = \frac{2j + 1}{2r} \pi, \quad j \in [0, r - 1].
\]  

Since these visual words are evenly distributed, the distance between any two adjacent visual words is the same. If we denote this distance as \( \text{dis}_0 \), then \( \text{dis}_0 = \frac{1}{2r} \pi \). For \( r = 9 \), \( b_j \in \left( \frac{1}{18} \pi, \frac{1}{18} \pi, \frac{5}{18} \pi, \frac{7}{18} \pi, \frac{9}{18} \pi, \frac{11}{18} \pi, \frac{13}{18} \pi, \frac{15}{18} \pi, \frac{17}{18} \pi \right), \text{dis}_0 = \frac{1}{9} \pi \).
4.3 Coding of Descriptors

The assignment of orientation values to their corresponding clusters can be viewed as the coding of orientation values by the local soft-assignment scheme [7]. Also, the assignment result of orientation value \( p_i^\theta \) to the \( j \)th cluster, namely, visual word \( b_j^\theta \), obtained by the linear interpolation distributing \( p_i^\theta \) into neighboring clusters can be assumed as the coding coefficient \( u_{ij} \). The number of nearest visual words is two in both HOG and SIFT. Benefitting from the even distribution of visual words in the descriptor space, the coding coefficient \( u_{ij} \) can be obtained through simple calculation as follows,

\[
u_{ij} = \begin{cases} 
1 - \frac{|p_i^\theta - b_j^\theta|}{\text{dis}_{ij}} & b_j^\theta \in \mathbf{b}_V^\theta, j \in [0, r - 1] \\
0 & b_j^\theta \notin \mathbf{b}_V^\theta 
\end{cases}
\]

where \( \mathbf{b}_V^\theta \) represents a collection of locally nearest visual words of \( p_i^\theta \) and \( |p_i^\theta - b_j^\theta| \) is the euclidian distance between \( p_i^\theta \) and \( b_j^\theta \).

4.4 Pooling of Descriptors

The summation of orientation value’s assignment results to relative clusters can be viewed as the sum-pooling of coding coefficients and weights to their relative visual words. Here, the descriptor is \( p_i^\theta \), the coding coefficient is \( u_{ij} \), and the weights are the magnitude value \( w_i^\theta \), the gaussian weight \( w_i^\theta \), and the spatial weight \( w_j^\theta \), which is calculated by the bilinear interpolation distributing \( p_i^\theta \) into the same clusters of neighboring cell descriptors\(^\dagger\). Therefore, the final weight \( w_j \) of visual word \( b_j^\theta \) is calculated by

\[
w_j = \sum_i u_{ij} \times p_i^\theta \times w_i^\theta \times w_j^\theta.
\]

5. Improvement of HOG/SIFT by Approaches of BOF

Since the construction method of block descriptor in HOG/SIFT can be treated as a variant of that in BOF, we can apply approaches of BOF into HOG/SIFT to improve the performance of them. In this paper, those approaches applied are 1) the preservation of locality, 2) the data-driven vocabulary, and 3) the preservation of spatial information.

5.1 Approach 1: The Preservation of Locality

5.1.1 Locality

The preservation of locality in the descriptor space generally assumes that in operations like pooling and coding, a descriptor only interacts with its neighboring descriptors (or visual words) or these operations should be implemented within a local region of this descriptor. Usually, even descriptors nearby a descriptor will be rather heterogeneous. The preservation of locality helps to discriminate a descriptor from those descriptors that diverge greatly. In terms of the view of manifold theory [5], [19], similar descriptors approximately dwell in a low-dimensional manifold in an ambient descriptor space, which suggests that only within locally small region, the euclidian distance is applicable to approximate the geodesic distance. Outside this region, the euclidian distance between two descriptors may largely differs from their actual geodesic distance.

Recent studies have revealed the merit of locality preservation in coding and pooling steps of BOF model. Local linear constraint [5] maps each descriptor into a space expanded by its locally neighboring descriptors. Local soft-assignment [7] suggested that in soft-assignment, a descriptor should only have contributions to its locally nearest visual words, which differs from traditional soft-assignment distributing the contribution of a descriptor to all visual words in terms of their distances\(^\dagger\). For pooling step, Study in [20] splits the descriptor space into several regions by mixture modeling and perform coding and pooling separately in each region. This method is simplified and streamlined by work [21], which merely pools descriptor in Voronoi regions of clusters obtained using k-means. The success of these works in tasks such as image classification has certified the substance of locality preservation in BOF.

5.1.2 Implementation Into HOG/SIFT

The application of the locality preservation into HOG/SIFT can be simply achieved by treating the entire gradient as descriptor. Then, the vocabulary can span the entire gradient space and the coding and pooling of descriptors can be performed more locally as depicted in Fig. 2. In the original HOG and SIFT, the coding and pooling are biased merely along the orientation dimension of the gradient space.

The merit of locality preservation can be stated ex-
This operation indeed amounts to dividing the gradient merely the orientation value since now the vocabulary is the entire gradient as the descriptor rather than large magnitude values tend to be in the edge, while gradients of small magnitude values often come from non-edge regions. This problem can be solved by the locality preservation using the entire gradient as the descriptor rather than merely the orientation value since now the vocabulary can also span the magnitude dimension, therefore, descriptors become closer to their actual visual words.

We manually redefine the entire gradient space and evenly distributed, as in the original HOG/SIFT. This operation indeed amounts to dividing the gradient space evenly into r regions\(^1\) and the descriptor in the center of each region is a visual word of the vocabulary as shown in Fig. 3. Let \(S_\theta\) and \(S_m\) denote the number of regions in the orientation dimension and magnitude dimension respectively, \(b_{jk} = (b^\theta_{jk}, b^m_{jk}) (j \in [0, S_\theta - 1], k \in [0, S_m - 1])\) a visual word in the vocabulary. Then, \(b^\theta_{jk}\) and \(b^m_{jk}\) can be calculated as follows,

\[
b^\theta_{jk} = \frac{2j + 1}{2S_\theta}, \quad b^m_{jk} = \frac{2k + 1}{2S_m}.
\]

Also the coefficient \(u_{ijk}\) can be calculated as follows,

\[
u_{ijk} = \left\{ \begin{array}{ll}
(1 - \left|\frac{\theta^\theta_i - \theta^\theta_j}{\text{dis}_\theta}\right|) \times (1 - \left|\frac{\theta^m_i - \theta^m_j}{\text{dis}_m}\right|) & b_{jk} \in b_p^N \\
0 & b_{jk} \notin b_p^N,
\end{array} \right.
\]

where \(\text{dis}_\theta\) and \(\text{dis}_m\) are lengths of the region in the orientation dimension and magnitude dimension respectively. Since the entire gradient space is evenly distributed, the \(\text{dis}_\theta\) and \(\text{dis}_m\) are identical for all regions. We limit the number of nearest visual words to 4 (see Fig. 3 for an example).

5.2 Approach 2: The Data-Driven Vocabulary

5.2.1 Vocabulary

A vocabulary is expected to represent the property of the training data. A naive approach is to treat the entire training data as a vocabulary, which is, however, infeasible in many real applications and some times superfluous. To remedy these deficiencies, numerous vocabulary learning methods, ranging from simple clustering technique such as k-means to sophisticated approaches incorporating sparse coding or fisher discriminative learning [11], [22], are proposed to learn a refined, compact, and discriminative data-driven vocabulary.

5.2.2 Implementation into HOG/SIFT

The manually defined vocabulary \(B\) in the original HOG/SIFT fails to capture the distribution information of the training data. To incorporate this information, we use k-means to cluster descriptors extracted from the training data to generate a data-driven vocabulary, as what conventional BOF does.

The clustering is executed for \(p^\theta_i\) and \(p^m_i\) independently\(^1\) to obtain \(S_\theta\) cluster centers in the orientation dimension and \(S_m\) cluster centers in the magnitude dimension. These cluster centers are then combined together to build the vocabulary \(B\). In this case, \(r = S_\theta \times S_m\) and the coefficient \(u_{ijk}\) is calculated as follows,

\[
u_{ijk} = \frac{\exp\left(-\frac{\theta^\theta_i - \theta^\theta_j}{\text{dis}_\theta} - \frac{\theta^m_i - \theta^m_j}{\text{dis}_m}\right)}{\sum_{s} \exp\left(-\frac{\theta^\theta_i - \theta^\theta_s}{\text{dis}_\theta} - \frac{\theta^m_i - \theta^m_s}{\text{dis}_m}\right)}
\]

where \(\beta\) is a parameter adjusting the softness of the exponential function.

5.3 Approach 3: The Preservation of Spatial Information

5.3.1 Spatial Information

In traditional BOF model, the spatial information of descriptors is lost during pooling step since the aggregation and assignment of those descriptors are carried out merely in the descriptor space. However, the preservation of spatial information of descriptors may contribute to a more expressive occurrence histogram and achieve better performance. Several studies have manifested the benefit of integrating spatial information. Study in [6] proposed a technique called spatial pyramid matching (SPM) which pools descriptors in a spatial pyramid of the image. Experiment result shows that SPM successfully mines the spatial information and significantly raises the accuracy of BOF in image classification task. Studies in [8] and [18] also exhibit the advantage of the preservation of spatial information of descriptors.

In HOG/SIFT, the spatial information is also utilized by dividing the block into cells, calculating the cell descriptor for each cell and then concatenating these cell descriptors to form the block descriptor. As well as this, the gaussian weight \(w^\theta_i\), which is calculated based on the distance

\[\text{distance} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}\]

\(^1\)The GLOH in [25] also computes the descriptor on a hand-crafted log-polar grid of the gradient space, this is different from our method since ours is motivated by the BOF approach and splits the gradient space by a vocabulary, which can be hand-crafted or data-driven.

\(^1\)This is done to avoid the bias brought by the unequal importance of magnitude and orientation in discrimination.
between the pixel where $p_i^b$ is extracted and the center of the block, along with the spatial weight $w_s$, which is calculated based on the distance between this pixel and the centers of its neighboring cells, are used to further record the location information of $p_i^b$.

5.3.2 Implementation into HOG/SIFT

To integrate more spatial information into the descriptor of HOG/SIFT, we follow the idea of SPM to compute the block descriptor on a spatial pyramid of the block. A spatial pyramid comprises $T$ layers, which are copies of the block. Each layer is divided into cells and the number of cells in a layer increases progressively with respect to the index of the layer. When the ratio of numbers of cells between adjacent layers is set to 4, the $r$th $(r \in [1, T])$ layer has $4^{T-r}$ cells. Figure 4 gives an example of a spatial pyramid of $T = 3$. Cell descriptors are computed on each cell and concatenated into layer descriptors. Layer descriptors are then concatenated together to construct a block descriptor. In HOG/SIFT, the block descriptor is also computed on a simplified spatial pyramid, which has only one layer divided into $2 \times 2$ cells ($4 \times 4$ cells in SIFT). The multi-level HOG introduced in [16] also computes the HOG descriptor on a spatial pyramid of the whole image, which differs from ours since we use the spatial pyramid for the block. Moreover, the spatial weight $w_s$ is removed in multi-level HOG.

5.4 Computational Cost

Compared with the original HOG/SIFT, our BOF-driven HOG/SIFT have larger computational cost since they have more visual words regarding the parameter $S_m$ and larger computing area caused by the spatial pyramid regarding the parameter “$T$”. Generally, in the steps of coding and pooling of descriptors, the complexity of the BOF-driven HOG (SIFT) is $S_m \times \frac{4^T-1}{12} (S_m \times \frac{4^T-1}{48})$ times that of the original HOG (SIFT) and is less than $S_m \times \frac{4^{T-1}-1}{12} (S_m \times \frac{4^{T-1}-1}{48})$ times in total complexity for all steps. However, as shown in the experiment Sects. 6 and 7, the BOF-driven HOG/SIFT can achieve agreeable performance even $S_m$ and $T$ have small values. Moreover, since BOF-driven HOG/SIFT are generalizations of the original ones. They can leverage acceleration techniques proposed for the original ones. Thus, our BOF-driven HOG/SIFT are acceptable when higher accuracy in requested.

6. Experiment and result on HOG

We do experiments to evaluate how the application of three approaches of BOF introduced in Sect. 5 improve the the performance of HOG. The evaluation is implemented on pedestrian detection task.

We term the original HOG (SIFT) “original”. We term the approach 1 using entire gradient as descriptor “GRAD#”, where # is the value of “$S_m$”. We term the approach 2 using data-driven vocabulary “DV”. We term the approach 3 using spatial pyramid in pooling “SP#”, where # is the value of “$T$”. The BOF-driven HOG combing the original HOG with approach 1, 2 and 3 with $S_m = 3$, $T = 2$ is then denoted as GRAD3 $\text{DV}_{SP2}$.

We conduct three experiments to evaluate the performance of these BOF-driven variants of HOG. Experiment 1 compares the performance of GRAD# with the original HOG and examine how its performance is affected by different values of $S_m$. Experiment 2 compares the performance of GRAD3 $\text{DV}$ with that of GRAD3. Experiment 3 compares the performance of GRAD3 $\text{DV}_{SP#}$ with GRAD3 $\text{DV}$ and examines the impact of different $T$ to its performance.

6.1 Data Set

We use the INRIA pedestrian data set from [1] for experiment. Both positive and negative training data set contain 1000 images of size $64 \times 128$, which are generated by cropping positive and negative training images in INRIA. The positive testing data set contains 476 positive images with their reflections of size $64 \times 128$ as well. We randomly selected 45300 patches of size $64 \times 128$ from negative testing images in INRIA to form the negative testing data set. Several examples from these data sets are displayed in Fig. 5.

6.2 Setting the Parameters

All parameters are set to the same for the original HOG and BOF-driven HOG unless otherwise noted$^1$. The $\beta = 10$ in experiments using it. We use L2-norm rule $H = H/\sqrt{\|H\|_2} + \epsilon$ for normalization, where $\epsilon$ is a regulating parameter. For the original HOG, GRAD# and GRAD3 $\text{DV}$, the $H$ is the block descriptor, for GRAD3 $\text{DV}_{SP#}$, the $H$ is the layer descriptor. To prevent the alias of improper thresholds, we don’t use the thresholding of large entry values for all HOGs. For the purpose of capturing the discriminant information of the pedestrian, the k-means is performed

$^1$The cell size in GRAD3 $\text{DV}_{SP#}$ varies in different layers of the spatial pyramid.
merely on positive training set. We use linear SVM classifier trained by Libsvm [26] for detection.

Experimental results are presented in the detection error tradeoff (DET) curve, where the horizontal axis represents the logarithmic value of the false positives per window (FPPW) and the vertical axis represents the logarithmic value of its corresponding error rate \( \frac{\text{false negative}}{\text{true negative} + \text{false positive}} \). In general, A good descriptor should have low error rate for any FPPW. However, the error rate usually increases along with the reduction of FPPW due to the increment of false negatives.

### 6.3 Result of Experiment 1

As can be observed from the experimental result in Fig. 6, the error rate of GRAD# becomes lower than that of HOG when the value of \( S_m \) steps up from 2 to 3 and 4. This improvement is consistent for almost all FPPW, which indicates the merit of the locality preservation. The GRAD4 has lower error rate compared with that of GRAD3, however, this improvement is not as remarkable as when \( S_m \) increases from 2 to 3.

### 6.4 Result of Experiment 2

From the experimental result in Fig. 7, we can see that the curve of GRAD3 DV obtains better performance for all FPPW compared with GRAD3, which uses manually-defined vocabulary. This result verifies the capability of data-driven vocabulary in capturing more class information, which consists with many other experimental results of BOF. Whereas, unlike other experiments, in which different classes have explicitly different visual words, the visual words of pedestrian might not stay far away from those of the background in the descriptor space, which impairs the efficacy the data-driven vocabulary and results in the trivial reduction of error rate in this experiment.

### 6.5 Result of Experiment 3

From the experimental result in Fig. 8, we can see that performance of GRAD3 DV SP1 is worse than that of GRAD3 DV. This can be attributed to the reason that the former one pools gradients in a spatial pyramid with one undivided layer while the latter one in a spatial pyramid with one layer divided into 4 cells. This indeed serves as a proof of the importance of spatial information preservation. The error rate of GRAD3 DV SP# declines notably when the value of \( T \) increases to 2 and 3, which complies with our anticipation. The GRAD3 DV SP3 gives worse performance compared with that of GRAD3 DV SP2, this is probably resulted by the too finely divided layers in the spatial pyramid, which introduces in superfluous spatial information and deteriorates the performance.

### 7. Experiment and Result on SIFT

We conduct experiments to evaluate how the application of approach 1 and approach 3 introduced in Sect. 5 improve the performance of SIFT\(^\dagger\). The evaluation is implemented on scene matching and image classification. Please refer to Sect. 6 for the meaning of the notations.

#### 7.1 Parameters of SIFT Descriptors

All parameters are set to the same for the original SIFT and BOF-driven SIFT unless otherwise noted. In GRAD2 SP3,

\(^\dagger\)The approach 2 is not utilized since it is difficult to learn a representative data-driven vocabulary on gradients for a small block when the training images in the experiment have large sizes and highly diversified appearances even in the same category.
The thresholding of large entry values is applied for each layer descriptor and the values of threshold are 0.16, 0.2 and 0.02 for the first, second and third layer descriptor respectively.

7.2 Scene Matching

7.2.1 Data Set and Evaluation Method

The data set used in this experiment is the Oxford eight scenes [25] as shown in Fig. 9. This data set contains eight scene categories for evaluation on rotation and scale changes (boat, bark), viewpoint changes (graffiti, wall), image blur (bike, tree), jpeg compression (ubc), and illumination changes (leuven). Each category comprises one reference image (image 1) and five transformed images (image 2 to 6). The transformation increments progressively with respect to the index of the image. Namely, e.g. image 3 has lower illumination compared to image 2 in leuven. The matching is performed between the first image and the remaining five. The result is presented with “recall versus 1-Precision (1 minus precision)” curve as in [25]. The recall and 1-Precision are calculated as follows respectively,

\[
\text{recall} = \frac{\text{#correct matches}}{\text{#correspondences}}, \quad (7)
\]

\[
1 - \text{Precision} = \frac{\text{#false matches}}{\text{#correct matches} + \text{#false matches}}, \quad (8)
\]

where the #correspondences represents the number of corresponding regions in two images. A favorable descriptor should have a low 1-Precision and high recall. Two descriptors in two images are deemed to match if their euclidian distance is below a given threshold. We vary this threshold to obtain the “recall versus 1-Precision” curve.

We use the same SIFT key-point detector embedded in opencv [27] for all descriptors since the comparison is designed for the descriptor, also, all descriptors’ covered regions are identical. Consequently, the value of #correspondences is the same for all descriptors. Note that in some matching pairs (e.g., matching between image 1 and 3 in bark), the #correspondences is zero due to the severe distortion of the transformed image.

7.2.2 Results

In total there are 40 comparison results, due to the limitation of space, we only display the comparison result of the first matching pair of each category except wall\(^1\) since the comparison results are consistent for most matchings in each category (please refer to the appendix for other results). The results are shown in Fig. 10.

From those results, we can observe that in overall comparison, GRAD2 and GRAD2_SP3 outperform original SIFT descriptor in all categories. Also, the changing of curves of the former two descriptors is more consistent and natural, which indicates their robustness against the diversification and variation of transformation. Among these categories, GRAD2 and GRAD2_SP3 gain the most significant improvement against original SIFT descriptor in boat and graffiti, which are devised for the evaluation of affine transformation in structured scenes. This manifests these two descriptors’ capability of alleviating the changing of descriptor in affine transformation since the locality preservation forces the coding and pooling to perform within a small region, where those features greatly diverge after affine transformation are screened outside. For bark and wall, which are also for evaluation of affine transformation, the improvement is not as remarkable as in boat and graffiti. This may be due to the unstructured and repeated textures in these two scenes, which undermine the efficacy of the locality preservation. The GRAD2 and GRAD2_SP3 also obtain a notable increment of recall compared with original SIFT descriptor in ubc for evaluation in jpeg compression. In general, GRAD2_SP2 gives higher recall than GRAD2 due to the addition of spatial information preservation. However, It has worse performance in bark, boat and trees. This might be resulted by the irregular clusters and blobs in those scenes. The application of spatial information preservation conversely brings in undesirable information and corrupts the performance.

7.3 Image Classification

7.3.1 Data Set and Evaluation Method

The experiment of image classification is implemented on the fifteen scenes data set (see Fig. 11 for examples.) from [6] which contains fifteen scene categories ranging from kitchen to mountain. Each category has 200 to 400 images of sizes around 250 × 300. This data set has large intra-class variation and small cross-class variation, rendering it a challenging data set for image classification. We transform all images in this data set into gray scale image for experiment.

We repeat the experiment for 5 times and calculate the average of all per-class recognition rates for each descriptor along with the standard deviation. In each time’s experiment, 50 images from each category are selected randomly as training images and the remaining ones are used for testing. We use k-means to cluster SIFT descriptors, which are extracted densely on a grid with spacing of 8 pixels from

\(^1\)For wall, the result of the fifth matching pair (matching between image 1 and 6) is given as a replacement because in the result of the first matching pair, the 1-Precision is zero for all descriptors.
YANG and KURITA: IMPROVEMENTS OF LOCAL DESCRIPTOR IN HOG/SIFT BY BOF APPROACH

the training images, to build a vocabulary with 200 visual words. Classification is carried out on a multi-class SVM trained by libsvm.

7.3.2 Results

The results of image classification of three descriptors are given in Table 1. The GRAD2_SP3 gives the highest average recognition rate and GRAD2 follows it as the second best. This result further confirms the efficacy of the application of approaches from BOF to enhance the expressive power of SIFT’s descriptor.

8. Conclusion

In this paper, we summarize and elaborate our previous works [13], [14] in the learning of the connection between BOF and HOG/SIFT. We interpret the method of block descriptor construction in HOG/SIFT as a variant of that in BOF used to construct the occurrence histogram. Upon this interpretation, we apply approaches of BOF, such as the locality preservation, data-driven vocabulary and the spatial information preservation into HOG/SIFT to improve their performance. Experimental results on pedestrian detection (for HOG), scene matching and image classification (for SIFT) have shown us the efficacy of BOF-driven HOG/SIFT, which also confirms the close relationship between BOF and HOG/SIFT in descriptor construction. We expect that more improvements can be made to HOG/SIFT and even BOF by leveraging each other’s advancements based on this connection.

Although HOG/SIFT are similar to BOF in descriptor construction, their specialities still restrain their abilities to benefit from more advancements of BOF. To break this limitation, we would like to deepen this connection and make it more general in our future work. Also, we would like to conduct more experiment to examine the influence of some parameters, such as the number of visual words, in the future work.

Acknowledgements

We would like to express our great gratitude to the reviewers of this paper, who have pointed out some insufficiencies in the experiment as well as problems in the logical structure of this paper. It is their discreet reading and incisive comments that make this paper’s publication available. This work was partly supported by JSPS KAKENHI Grand Number 23500211.

References

[1] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” Proc. 2005 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, San Diego, CA, USA, pp.886–893, June 2005.
[2] G. Hinton and R.R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” Science, vol.313. no.5786, pp.504–507, July 2006.
[3] Y.L. Boureau, F. Bach, Y. LeCun, and J. Ponce, “Learning mid-level
features for recognition,” Proc. 2010 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, San Francisco, CA, USA, pp.2559–2566, June 2010.

[4] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, and Y. Ma, “Robust face recognition via sparse representation,” IEEE Trans. Pattern Anal. Mach. Intell., vol.31, no.2, pp.210–227, Feb. 2009.

[5] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, “Locality-constrained linear coding for image classification,” Proc. 2010 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, pp.3360–3367, San Francisco, CA, USA, June 2010.

[6] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” Proc. 2006 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, vol.2, pp.2169–2178, New York, NY, USA, June 2006.

[7] L. Liu, L. Wang, and X. Liu, “In defense of soft-assignment coding,” Proc. 2011 IEEE Intl. Conf. on Computer Vision, pp.2486–2493, Barcelona, Spain, Nov. 2011.

[8] O. Russakovsky, Y. Lin, K. Yu, and F. Li, “Object-centric spatial pooling for image classification,” Proc. 12th European conf. on Computer Vision, Florence, Italy, vol.2, pp.1–15, Oct. 2012.

[9] R. Girshick, P. Felzenszwalb, and D. Mcallester, “Object detection with grammar models,” Proc. 25th Annu. Conf. on Neural Information Processing Systems, pp.442–450, Granada, Spain, Dec. 2011.

[10] X. Wang, T. Han, and S. Yan, “An HOG-LBP human detector with partial occlusion handling,” Proc. 2009 IEEE Intl. Conf. on Computer Vision, pp.32–39, Kyoto, Japan, Sept. 2009.

[11] M. Yang, L. Zhang, X. Feng, and D. Zhang, “Fisher discrimination dictionary learning for sparse representation,” Proc. 2011 IEEE Intl. Conf. on Computer Vision, pp.543–550, Barcelona, Spain, Nov. 2011.

[12] D.G. Lowe, “Distinctive Image Features from Scale-Invariant Key-points,” Intl. J. Comput. Vis., vol.60, no.2, pp.91–110, Nov. 2004.

[13] Z. Yang and T. Kurita, “BOG: an extension of hog by interpreting it as bag of features,” Proc. 13th IAPR Intl. Conf. on Machine Vision Applications, Kyoto, Japan, May 2013.

[14] Z. Yang and T. Kurita, “Improvements to the descriptor of SIFT by BOF approaches,” Proc. 2nd IAPR Asian Conf. on Pattern Recognition, Okinawa, Japan, Nov. 2013.

[15] M. Dikmen, D. Hoiem, and T. Huang, “A data driven method for feature transformation,” Proc. 2012 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, pp.3314–3321, Providence, RI, USA, June 2012.

[16] S. Maji, A.C. Berg, and J. Malik, “Classification using intersection kernel support vector machines is efficient,” Proc. 2008 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, pp.1–8, Anchorage, AK, USA, June 2008.

[17] F. Li and P. Perona, “A bayesian hierarchical model for learning natural scene categories,” Proc. 2005 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, pp.524–531, San Diego, CA, USA, June 2005.

[18] Y. Cao, C. Wang, Z. Li, L. Zhang, and L. Zhang, “Spatial-Bag-of-Features,” Proc. 2010 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, pp.3352–3359, San Francisco, CA, USA, June 2010.

[19] K. Yu, T. Zhang, and Y. Gong, “Nonlinear learning using local coordinate coding,” Proc. 22nd Annu. Conf. on Neural Information Processing Systems, Vancouver, British Columbia, Canada, pp.2223–2231, Dec. 2009.

[20] J. Yang, K. Yu, and T. Huang, “Efficient highly over-complete sparse coding using a mixture model,” Proc. 10th European conf. on Computer Vision, Heraklion, Crete, Greece, pp.113–126, Sept. 2010.

[21] Y.L. Boureau, N.L. Roux, and F. Bach, “Ask the locals: multi-way local pooling for image recognition,” Proc. 2011 IEEE Intl. Conf. on Computer Vision, Barcelona, Spain, pp.2651–2658, Nov. 2011.

[22] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, “Supervised dictionary learning,” Proc. 21st Annu. Conf. on Neural Information Processing Systems, Vancouver, British Columbia, Canada, Dec. 2008.

[23] Y. Ke and R. Sukthankar, “PCA-SIFT: A more distinctive representation for local image descriptors,” Proc. 2004 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, pp.506–513, Washington, DC, USA, June 2004.

[24] P. Moreno, A. Bernardino, and J. Santos-Victor, “Improving the SIFT descriptor with smooth derivative filters,” Pattern Recognit. Lett., vol.30, no.1, pp.18–26, Jan. 2009.

[25] K. Mikolajczyk and C. Schmid, “A performance evaluation of local descriptors,” IEEE Trans. Pattern Anal. Mach. Intell., vol.27, no.10, pp.1615–1630, Oct. 2005.

[26] C. Chang and C. Lin, “LIBSVM: A library for support vector machines,” ACM Trans. Intelligent Systems and Technology, vol.2, no.3, pp.27:1–27:27, April 2011.

[27] G. Bradski, “The openvc library,” Dr. Dobbs’ J. Software Tools, 2000.
Appendix: Other Comparison Results of Scene Matching

Fig. A.1 Other comparison results of all categories in scene matching.