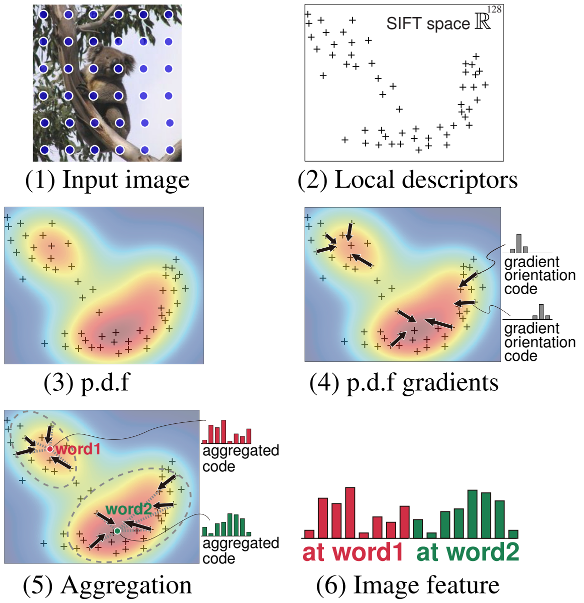
2013 IEEE Conference on Computer Vision and Pattern Recognition

BoF meets HOG: Feature Extraction based on Histograms of Oriented p.d.f Gradients for Image Classification

Takumi Kobayashi
National Institute of Advanced Industrial Science and Technology
1-1-1 Umezono, Tsukuba, Japan
takumi.kobayashi@aist.go.jp

Abstract

Image classification methods have been significantly developed in the last decade. Most methods stem from bag-of-features (BoF) approach and it is recently extended to a vector aggregation model, such as using Fisher kernels. In this paper, we propose a novel feature extraction method for image classification. Following the BoF approach, a plenty of local descriptors are first extracted in an image and the proposed method is built upon the probability density function (p.d.f) formed by those descriptors. Since the p.d.f essentially represents the image, we extract the features from the p.d.f by means of the gradients on the p.d.f. The gradients, especially their orientations, effectively characterize the shape of the p.d.f from the geometrical viewpoint. We construct the features by the histogram of the oriented p.d.f gradients via orientation coding followed by aggregation of the orientation codes. The proposed image features, imposing no specific assumption on the targets, are so general as to be applicable to any kinds of tasks regarding image classifications. In the experiments on object recognition and scene classification using various datasets, the proposed method exhibits superior performances compared to the other existing methods.


SIFT space $\mathbb{R}^{128}$
gradient orientation code
gradient orientation code
word1
word2
aggregated code
aggregated code
at word1
at word2

(1) Input image (2) Local descriptors (3) p.d.f (4) p.d.f gradients (5) Aggregation (6) Image feature

Figure 1. Overview of the proposed method. At dense spatial grid points in an input image (1), a plenty of SIFT local descriptors are extracted (2). The probability density function (p.d.f) in the descriptor space is estimated by applying kernel density estimator to those descriptors (3). The gradients are computed on the p.d.f and then their orientations are coded (4). Those codes are aggregated around respective visual words (5) and the aggregated codes are finally concatenated into the image feature vector (6).

1. Introduction

Image classification has attracted keen attentions in the computer vision community in the last decade. The task includes such as object recognition [1, 14] and scene classification [19, 20], posing a challenge to cope with significant variations of the targets as well as the environmental changes in the image. The image classification is frequently addressed in the framework of bag-of-features (BoF) [9] owing to the advances of the local descriptors such as SIFT [23].

BoF is based on the local descriptors densely extracted in an image which are further coded into *visual words* and produces as the image feature the histogram of the visual words that appear in the image. Thus, BoF mainly consists of the four procedures extracting local descriptors, coding them into words, aggregating (pooling) the words into the histogram, and classifying the histogram feature vector. While the classification is performed such as by simply applying (linear) SVM, much research effort has been made in the past few years to develop effective feature representation related to the first three procedures; for example, [4, 5, 18, 21] for local descriptors, [11, 12, 15, 22, 29, 32] for coding, [19, 34, 33] for aggregation.

1063-6919/13 $26.00 © 2013 IEEE
DOI 10.1109/CVPR.2013.102

In the other direction, the BoF has been recently extended to the methods aggregating vectors [16, 24, 35, 17], not the word codes. It first appeared in Fisher kernel [16] which is derived from the information geometry [2] using Gaussian mixture models (GMM). The method of super vector coding [35] was proposed afterwards to approximate the (nonlinear) classifier function by piece-wise linear models. Although those two methods are defined in different scenarios, they actually result in the same procedure that aggregates the difference vectors between local descriptors and their nearby visual word centers. The image features are obtained by concatenating those difference vectors aggregated around respective words, in contrast to the standard BoF which simply counts the occurrence of the visual words in the histograms. This vector aggregation based method significantly improves the performance on object recognition [16, 24, 35] compared to the BoF approach. It is also employed in the image search task [17].

In this study, we propose a novel method to extract effective features for image classifications; Fig. 1 shows the overview. For describing the images, the proposed method extends the discrete representation in BoF to the continuous *probability density function (p.d.f)*. The p.d.f is estimated by applying kernel density estimator [31] to the densely extracted local descriptors without assuming any specific probabilistic models such as GMM. Since the p.d.f essentially represents the image, we extract features from the p.d.f by means of the gradients on the p.d.f in a manner similar to HOG [10]/SIFT [23] applied to *image pixel function*. The gradients, especially their orientations, can effectively characterize the p.d.f. Through computing the gradients, the mean shift vectors [7] are naturally induced and those vectors are coded in terms of their orientations. Those orientation codes are finally aggregated around respective visual words into the histograms similarly to the above-mentioned methods, Fisher kernel [16, 24] and super vector coding [35] which are also shown to be a special case of the proposed method (Sec.3). The proposed method is defined without any assumption on the target and thus it is applicable to versatile tasks of image classifications, such as object recognition and scene classification.

2. Proposed method

In the BoF framework [9], an image is represented by a plenty (bag) of local descriptors densely extracted in the image, and then is finally characterized by a histogram of visual words quantizing the underlying probability distribution of the local descriptors. In this work, we explicitly focus on the probability density function (p.d.f) composed of the descriptors; namely, we extract features from the p.d.f which essentially represents the image. An overview of the proposed feature extraction method is shown in Fig. 1.

2.1. Probability density function in BoF

From an input image, N local descriptors, such as SIFT descriptors [23], are extracted at dense spatial positions with various scales; those are denoted by $\boldsymbol{x}_i \in \mathbb{R}^d, i = 1, \cdots, N$. While the bag of those descriptors has so far been used to discretely represent the image, we apply kernel density estimator [31] to obtain the following (continuous) probability density function (p.d.f),

$$p_{\mathtt{f}}(\boldsymbol{x}) = \frac{1}{N}\sum_{i=1}^{N}\mathtt{f}(\|\boldsymbol{x}-\boldsymbol{x}_i\|_2^2), \tag{1}$$

where $\mathtt{f}(z):\mathbb{R}\rightarrow\mathbb{R}$ indicates the (differentiable) profile function for kernel [7]; *e.g.*, $\mathtt{f}(z) = C_{d,h}\exp(-\frac{z}{2h})$ with the bandwidth parameter h, say $h = 0.1$ in this study, and the normalization constant $C_{d,h}$. We begin with this p.d.f (1) for constructing an effective image feature.

2.2. Oriented p.d.f gradients

The gradients, especially their orientations, effectively characterize the "shape" of the p.d.f from the geometrical viewpoint, as is the case with HOG [10]/SIFT [23] applied to extract geometrical feature of an image pixel function. The gradient vector of the p.d.f (1) is simply given by

$$\begin{aligned}\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x}) &= \frac{2}{N}\sum_{i=1}^{N}(\boldsymbol{x}-\boldsymbol{x}_i)\mathtt{f}'(\|\boldsymbol{x}-\boldsymbol{x}_i\|_2^2)\\ &= \frac{1}{N}\sum_{i=1}^{N}(\boldsymbol{x}_i-\boldsymbol{x})\mathtt{g}(\|\boldsymbol{x}-\boldsymbol{x}_i\|_2^2),\end{aligned} \tag{2}$$

where $\mathtt{g}(z) = -2\mathtt{f}'(z)$, the derivative of the profile function $\mathtt{f}(z)$, which is also the profile [7]. Note that it is improper to straightforwardly aggregate the p.d.f gradient vectors themselves since the gradient orientation information is canceled out via summation. Thus, we consider the orientation coding of the p.d.f gradients (2), followed by aggregation into histograms.

The orientation coding is usually applied to image gradients such as in HOG[10] and SIFT[23]. The orientation of the image gradients is coded based on a lot of the bases (bins) that uniformly cover 2-D spatial orientations, forming overcomplete set to describe any oriented gradients. However, it is infeasible to use those overcomplete set of (uniform) bases for coding high-dimensional gradient orientations; *e.g.*, p.d.f by SIFT descriptors is defined in 128-dimensional space. In this study, we employ the complete set of bases given by principal component analysis (PCA).

PCA[1] is applied to the p.d.f gradient vectors normalized in unit L_2 norm $\boldsymbol{\nabla} p_{\mathtt{f}}/\|\boldsymbol{\nabla} p_{\mathtt{f}}\|_2$ which indicate only

[1] Note that, in this case, PCA is applied to the samples *without centering*, which results in the eigen-decomposition of an auto-correlation matrix, not a covariance matrix. And, the samples are all the L_2-normalized gradient vectors drawn from training images.

the orientations on a unit hypersphere. Thereby, we obtain the d orthonormal basis (eigen) vectors, $\boldsymbol{u}_j, j = 1,\cdots,d, \boldsymbol{u}_j^\top \boldsymbol{u}_k = \delta_{jk}$. Along each basis vector, we can consider two orientations, positive and negative ones, which totally provides $2d$ orientation bins by

$$\mathtt{C}(\boldsymbol{v};\{\boldsymbol{u}_j\}_{j=1}^d) = [\max(\boldsymbol{u}_1^\top \boldsymbol{v},0)^2, \max(-\boldsymbol{u}_1^\top \boldsymbol{v},0)^2,\cdots, \max(\boldsymbol{u}_d^\top \boldsymbol{v},0)^2, \max(-\boldsymbol{u}_d^\top \boldsymbol{v},0)^2]^\top \in \mathbb{R}_+^{2d}, \tag{3}$$

where $\boldsymbol{v}$ indicates the d-dimensional vector to be coded. This coding produces rather sparse orientation codes in which at most d components are nonzero, and the code has a unit sum for $\|\boldsymbol{v}\|_2^2 = 1$ since $\|\boldsymbol{v}\|_2^2 = \boldsymbol{1}^\top \mathtt{C}(\boldsymbol{v};\{\boldsymbol{u}\}_{j=1}^d) = 1$. Note again that, without this orientation coding, the gradient vectors, especially oriented in opposite directions, would be canceled out via aggregation. The orientation of the p.d.f gradient vector (2) is coded by $\mathtt{C}\left(\frac{\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x})}{\|\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x})\|_2};\{\boldsymbol{u}_j\}_{j=1}^d\right)$. For simplicity, we omit $\{\boldsymbol{u}_j\}_{j=1}^d$ in $\mathtt{C}$ in the followings.

PCA produces the eigenvalues e_j as well as the eigenvectors $\boldsymbol{u}_j$ employed for the orientation bases. The eigenvalue represents the power of the code on the corresponding basis, $e_j = \mathrm{E}_{\boldsymbol{x}}\left[\left(\boldsymbol{u}_j^\top \frac{\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x})}{\|\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x})\|_2}\right)^2\right]$, and thus it is utilized to normalize the orientation codes as in tf-idf [28] or PCA whitening:

$$\hat{\mathtt{C}}\left(\frac{\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x})}{\|\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x})\|_2}\right) = \boldsymbol{E}^{-1}\mathtt{C}\left(\frac{\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x})}{\|\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x})\|_2}\right), \tag{4}$$

where $\boldsymbol{E} = \mathrm{diag}(e_1, e_1, \cdots, e_d, e_d) \in \mathbb{R}^{2d\times 2d}$. Through this weighting, the orientation codes are equally dealt with by enhancing the orientations that rarely occur while suppressing the common ones that are frequently found on the whole. The rare orientations would be more discriminative than the common ones [28], and thus the weighting (4) improves the discriminative power.

2.3. Aggregation of p.d.f gradient orientation codes

The orientation codes (4) are finally aggregated around the visual words which are basis points (cluster centers) in the local descriptor space $\mathbb{R}^d$. We define the aggregation in the following continuous form as is the case with the p.d.f;

$$\int \mathtt{W}(\boldsymbol{x},\boldsymbol{\mu})\|\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x})\|_2 \hat{\mathtt{C}}\left(\frac{\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x})}{\|\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x})\|_2}\right) d\boldsymbol{x}, \tag{5}$$

where $\mathtt{W}(\boldsymbol{x},\boldsymbol{y})$ is the weighting function indicating how the local descriptor $\boldsymbol{x}$ contributes to the word $\boldsymbol{\mu}$ as defined in the later, and the magnitude and the orientation of the p.d.f gradient vector $\boldsymbol{\nabla} p_{\mathtt{f}}$ are processed respectively. To reduce the continuous form into a tractable discrete one, it should be noted that the local descriptors $\boldsymbol{x}_i, i = 1,\cdots,N$ are assumed to be randomly sampled according to the p.d.f $p_{\mathtt{f}}(\boldsymbol{x})$. As to the sampling, given arbitrary function $\mathtt{h}(\boldsymbol{x})$, we have the following relationship [3]:

$$\int \mathtt{h}(\boldsymbol{x}) p_{\mathtt{f}}(\boldsymbol{x}) d\boldsymbol{x} \approx \frac{1}{N}\sum_i^N \mathtt{h}(\boldsymbol{x}_i). \tag{6}$$

By using the above relationship, (5) is reduced into

$$\int \mathtt{W}(\boldsymbol{x},\boldsymbol{\mu})\|\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x})\|_2 \hat{\mathtt{C}}\left(\frac{\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x})}{\|\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x})\|_2}\right) d\boldsymbol{x}$$
$$\approx \frac{1}{N}\sum_{i=1}^N \mathtt{W}(\boldsymbol{x}_i,\boldsymbol{\mu})\frac{\|\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x}_i)\|_2}{p_{\mathtt{f}}(\boldsymbol{x}_i)} \hat{\mathtt{C}}\left(\frac{\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x}_i)}{\|\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x}_i)\|_2}\right). \tag{7}$$

This is a summation weighted by the inverse of the probability $p_{\mathtt{f}}(\boldsymbol{x}_i)$. The formulation (7) favorably suppresses the effect of the frequent local descriptors of high probability which are common across the categories and less discriminative for classification [28]. Here, we can induce the *normalized* gradient in (7) as

$$\frac{\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x})}{p_{\mathtt{f}}(\boldsymbol{x})} \approx \frac{\boldsymbol{\nabla} p_{\mathtt{f}}(\boldsymbol{x})}{p_{\mathtt{g}}(\boldsymbol{x})} = \frac{\sum_{i=1}^N \boldsymbol{x}_i \mathtt{g}(\|\boldsymbol{x}-\boldsymbol{x}_i\|_2^2)}{\sum_{i=1}^N \mathtt{g}(\|\boldsymbol{x}-\boldsymbol{x}_i\|_2^2)} - \boldsymbol{x} \triangleq \hat{\boldsymbol{\nabla}} p_{\mathtt{f}}(\boldsymbol{x}), \tag{8}$$

where the profile $\mathtt{g}$ is approximately applied to the normalization since $p_{\mathtt{f}}(\boldsymbol{x}) \approx p_{\mathtt{g}}(\boldsymbol{x})$. Note that the normalized gradient $\hat{\boldsymbol{\nabla}} p_{\mathtt{f}}(\boldsymbol{x})$ is identical to the mean shift vector [7] which has been usually used for clustering and its favorable properties are discussed in [7].

By introducing the normalized gradient (8) into (7), we finally obtain the aggregation form to construct features as the histogram of the oriented p.d.f gradients. Let $\boldsymbol{\mu}_k, k = 1,\cdots,M$ be the k-th visual word center, and the aggregation around $\boldsymbol{\mu}_k$ is given by

$$\boldsymbol{d}_k = \frac{1}{N}\sum_{i=1}^N \mathtt{W}(\boldsymbol{x}_i,\boldsymbol{\mu}_k)\|\hat{\boldsymbol{\nabla}} p_{\mathtt{f}}(\boldsymbol{x}_i)\|_2 \hat{\mathtt{C}}\left(\frac{\hat{\boldsymbol{\nabla}} p_{\mathtt{f}}(\boldsymbol{x}_i)}{\|\hat{\boldsymbol{\nabla}} p_{\mathtt{f}}(\boldsymbol{x}_i)\|_2}\right). \tag{9}$$

These features around the respective visual words are concatenated into the final feature vector;

$$\boldsymbol{d} = [\boldsymbol{d}_1^\top,\cdots,\boldsymbol{d}_M^\top]^\top \in \mathbb{R}_+^{2dM}. \tag{10}$$

2.4. Practical techniques

In this section, we present the techniques which are somewhat apart from the essence in the proposed method but are practically effective.

Weights to visual words. We define the weighting function $\mathtt{W}(\boldsymbol{x},\boldsymbol{\mu})$ from the local descriptor $\boldsymbol{x}$ to the visual word $\boldsymbol{\mu}$. We simply apply the following method based on the distance ratio, though the other word coding methods [12, 15, 33, 32] are also applicable. According to the Euclidean distance $\mathtt{D}(\boldsymbol{x},\boldsymbol{y}) = \|\boldsymbol{x}-\boldsymbol{y}\|_2^2$, only K nearest neighbor words around $\boldsymbol{x}$ are picked up, say $K = 10$ in this study, and assume $\mathtt{D}(\boldsymbol{x},\boldsymbol{\mu}_k)\leq\mathtt{D}(\boldsymbol{x},\boldsymbol{\mu}_l)$ for $k<l$. Based on the distance ratio compared to the nearest one $\mathtt{D}(\boldsymbol{x},\boldsymbol{\mu}_1)$, the weighting function is determined by

$$\mathtt{W}(\boldsymbol{x},\boldsymbol{\mu}_k)=\frac{\mathtt{D}(\boldsymbol{x},\boldsymbol{\mu}_1)/\mathtt{D}(\boldsymbol{x},\boldsymbol{\mu}_k)}{\sum_{l=1}^{K}\mathtt{D}(\boldsymbol{x},\boldsymbol{\mu}_1)/\mathtt{D}(\boldsymbol{x},\boldsymbol{\mu}_l)}=\frac{\prod_{\bar{l}\neq k}^{K}\mathtt{D}(\boldsymbol{x},\boldsymbol{\mu}_{\bar{l}})}{\sum_{l=1}^{K}\prod_{\bar{l}\neq l}^{K}\mathtt{D}(\boldsymbol{x},\boldsymbol{\mu}_{\bar{l}})}.$$

The weights to the words other than those K-NN ones are assigned with 0.

Normalization. The final feature vector $\boldsymbol{d}$ in (10) is normalized by SIFT-like normalization [23, 10] with the threshold $\tau=\frac{1}{\sqrt{2dM}}$; $\boldsymbol{d}\leftarrow\frac{\min\left(\frac{\boldsymbol{d}}{\|\boldsymbol{d}\|_2},\tau\right)}{\|\min\left(\frac{\boldsymbol{d}}{\|\boldsymbol{d}\|_2},\tau\right)\|}$.

Spatial pooling. Following the idea of spatial pyramid matching [19], we employ the spatial pooling using spatial bins (partitions) in order to roughly take into account the spatial alignment in the image. Hence, the weighting function $\mathtt{W}$ is modified into

$$\hat{\mathtt{W}}(\boldsymbol{x}_i,\boldsymbol{\mu}_k,\boldsymbol{r}_i,\boldsymbol{s}_l)=\mathtt{W}(\boldsymbol{x}_i,\boldsymbol{\mu}_k)\mathtt{W}_s(\boldsymbol{r}_i,\boldsymbol{s}_l),\qquad(11)$$

where $\boldsymbol{r}_i$ indicates the 2-D position vector at which the local descriptor $\boldsymbol{x}_i$ is extracted, $\boldsymbol{s}_l, l=1,\cdots,S$ be the l-th spatial bin, and the spatial weighting function $\mathtt{W}_s$ is defined by bilinear interpolation as in SIFT spatial pooling [23]. This enables us to characterize the spatial information in the image, improving the classification performances. In this study, we use three levels of spatial partitioning as 1×1, 2×2, and 3×1 in the image, which results in the following feature,

$$\boldsymbol{d}=\left[\boldsymbol{d}_1^{1\times1\top},\boldsymbol{d}_1^{2\times2\top},\cdots,\boldsymbol{d}_4^{2\times2\top},\boldsymbol{d}_1^{3\times1\top},\cdots,\boldsymbol{d}_3^{3\times1\top}\right]^{\top}\in\mathbb{R}_{+}^{16dM}.$$

3. Comparison to the previous methods

The proposed method can be regarded as HOG [10] or SIFT [23] applied to the p.d.f of densely extracted local descriptors in the BoF framework. However, it is not straightforward to apply HOG to the p.d.f since the p.d.f is composed of (discrete) samples of local descriptors and thus defined in a high dimensional space, while HOG [10]/SIFT [23] directly deals with an image, a (continuous) image pixel function well defined in 2-D spatial dimensions. Table 1 shows the comparison of the proposed method with HOG/SIFT.

In the BoF framework, some methods based on vector aggregation have been proposed, showing promising performances on object recognition, such as Fisher kernel [16, 24] and super vector coding [35]. The Fisher kernel [16] is derived from the information geometry [2] with Gaussian mixture model (GMM), and the super coding [35] is motivated by the piece-wise linear approximation of the (nonlinear) classification function. These methods eventually result in aggregating the difference vectors $\boldsymbol{x}_i-\boldsymbol{\mu}_k$ around the visual word $\boldsymbol{\mu}_k$ as shown in Table 2. The aggregation of the difference vectors roughly corresponds to the p.d.f gradient vector $\boldsymbol{\nabla}p_{\mathtt{f}}(\boldsymbol{\mu}_k)$ in (2), or mean shift vector $\hat{\boldsymbol{\nabla}}p_{\mathtt{f}}(\boldsymbol{\mu}_k)$ in (8), at the word $\boldsymbol{\mu}_k$; thus, we can say that the Fisher kernel [16] and super coding [35] are viewed as the special case of the proposed method. The gradient vectors only at those sparse word points would be less discriminative for characterizing the distribution of local descriptors; especially, the information of the distribution around the word is canceled out by simply aggregating those difference vectors as shown in Fig. 2a. The Fisher kernel [16, 24] additionally employs the variances of the difference vector in order to compensate it, but the integration of those different kinds of statistical quantities is itself difficult, though those are simply concatenated in [16, 24]. In contrast, the proposed method can effectively exploit the distribution (p.d.f) of the local descriptors by means of the histograms of the p.d.f gradient orientation which is densely computed at every sample point, as shown in Fig. 2b. Our formulation is compared to those methods including the standard BoF in Table 2.

Table 1. Comparison with HOG/SIFT

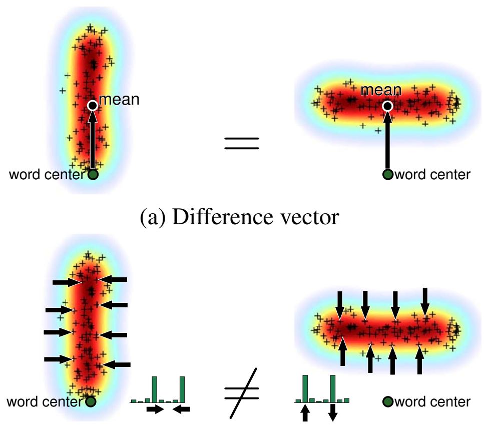
	Ours	HOG/SIFT
target	p.d.f $p_{\mathtt{f}}(\boldsymbol{x})$	image pixel function $I(\boldsymbol{r})$
variable	local descriptor $\boldsymbol{x}\in\mathbb{R}^d$	2-D position vector $\boldsymbol{r}\in\mathbb{R}^2$
orientation code by	PCA bases	predefined orientation bins
aggregation at	visual words	predefined spatial bins

Table 2. Comparison with BoF-based image features. For Fisher kernel [24], GMM produces the mixture weight η_k and the diagonal covariance $\boldsymbol{\Sigma}_k=\mathrm{diag}(\boldsymbol{\sigma}_k)^2$ as well as the mean (word) $\boldsymbol{\mu}_k$, and consequently the posterior $\alpha_{ik}=\frac{\eta_k\mathcal{N}(\boldsymbol{x}_i;\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)}{\sum_{l=1}^{M}\eta_l\mathcal{N}(\boldsymbol{x}_i;\boldsymbol{\mu}_l,\boldsymbol{\Sigma}_l)}$. Note that $()^{\bullet2}$ indicates the operation of the component-wise square. Super vector coding [35] has a (scalar) parameter s.

Method	Feature vector around word $\boldsymbol{\mu}_k$	Dim.
BoF	$\sum_i\mathtt{W}(\boldsymbol{x}_i,\boldsymbol{\mu}_k)$	1
Fisher kernel [24]	$\frac{1}{\sqrt{\eta_k}}\sum_i\alpha_{ik}\begin{bmatrix}\boldsymbol{\Sigma}_k^{-1/2}(\boldsymbol{x}_i-\boldsymbol{\mu}_k)\\\frac{1}{\sqrt{2}}\left\{\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_i-\boldsymbol{\mu}_k)^{\bullet2}-\mathbf{1}\right\}\end{bmatrix}$	$2d$
Super coding [35]	$\frac{1}{\sqrt{\sum_i\mathtt{W}(\boldsymbol{x}_i,\boldsymbol{\mu}_k)}}\sum_i\mathtt{W}(\boldsymbol{x}_i,\boldsymbol{\mu}_k)\begin{bmatrix}\boldsymbol{x}_i-\boldsymbol{\mu}_k\\s\end{bmatrix}$	$d+1$
Ours	$\sum_i\mathtt{W}(\boldsymbol{x}_i,\boldsymbol{\mu}_k)\|\hat{\boldsymbol{\nabla}}p_{\mathtt{f}}(\boldsymbol{x}_i)\|_2\hat{\mathtt{C}}\left(\frac{\hat{\boldsymbol{\nabla}}p_{\mathtt{f}}(\boldsymbol{x}_i)}{\|\hat{\boldsymbol{\nabla}}p_{\mathtt{f}}(\boldsymbol{x}_i)\|_2}\right)$	$2d$

4. Experimental Results

We apply the proposed method to image classification tasks in the following setting. The SIFT local descriptors [23] are extracted at dense spatial grid points in 4 pixel step with three scales of $\{16,24,32\}$ pixels. Visual words are obtained by applying k-means clustering to one million local descriptors which are randomly sampled from the training images. For orientation coding (Sec.2.2), the bases $\boldsymbol{u}_j$ and the weights e_j are obtained by applying PCA to all the gradient vectors in the training images. The resultant



(a) Difference vector

(b) Orientation code

Figure 2. Aggregation around the word center. '+' indicates the local descriptor. While the aggregated difference vectors, denoted by the large arrows, improperly coincide for different distributions (a), the aggregation (histogram) of the orientation codes discriminatively describes the distribution around the word (b).

feature vectors are finally classified by linear SVM [30].

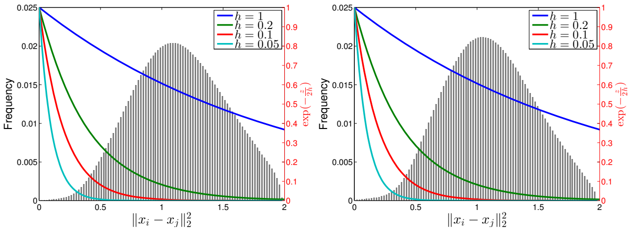
First, we analyze the performance of the proposed method to validate the effectiveness of the procedures described in Sec.2. Then, the proposed method is compared to the other recently developed methods on various datasets for scene classification and object recognition.

4.1. Performance Analysis

We use PASCAL-VOC2007 dataset [1] to analyze the performances of the proposed method from various aspects. The dataset contains objects of 20 categories and it poses a challenging task of object recognition due to significant variations in terms of appearances and poses even with occlusions. There are 5,011 training images (`train`+`val` sets) and 4,952 test images (`test` set). The performance is evaluated by the standard PASCAL protocol which computes average precision (AP) based on the precision/recall curve; we report the mean of AP (mAP) across the 20 categories in the results other than Table 4.

The following four issues are conceivable in the proposed method. Here, we use 128 visual words.

Bandwidth parameter in the profile function. We use the profile function $\mathtt{f}(z) = \exp(-\frac{z}{2h})$ with $h = 0.1$ in the kernel density estimation (1). The determination of the bandwidth parameter h still remains as an open problem [7, 31], and there are some attempts to determine it adaptively from the data, such as in [8, 27, 6]. Those methods, however, are intended mainly for the lower dimensional data rather than the high-dimensional descriptors that we use. Due to the curse of dimensionality [26], such adaptive bandwidth selection becomes less effective in the higher-dimensional space, since the data samples are *sparsely* distributed around each sample point. Actually,



(a) PASCAL-VOC2007

(b) Scene-15

Figure 3. Pairwise distances with the profile $\mathtt{f}(z) = \exp(-\frac{z}{2h})$

Table 3. Performance analysis on PASCAL-VOC2007

(a) Bandwidth in the profile $\mathtt{f}$

$h = 1$	0.2	0.1	0.05
59.41	59.54	59.82	58.55

(b) Orientation coding

PCA bases	random bases
59.82	58.05

(c) Component weighting

inverse eigenvalues	none
59.82	59.13

(d) p.d.f gradient

normalized	none
59.82	57.97

Fig. 3a shows the distribution of pairwise squared Euclidean distances in PASCAL-VOC2007 dataset. Note that the SIFT descriptors are non-negative in unit L_2 norm, which makes the squared distance range in $[0, 2]$. The distribution has the peak at $\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^2 = 1$ and this can be also seen in Scene-15 dataset of scene images (Fig. 3b) which are different contents from object images in PASCAL-VOC2007 dataset. We apply four bandwidths $h \in \{1, 0.2, 0.1, 0.05\}$ to the profile $\mathtt{f}(z) = \exp(-\frac{z}{2h})$ which are superimposed over the distribution of the distances in Fig. 3. The profile of $h = 0.1$ appropriately picks up the neighboring samples, while those of the other bandwidths cover too small or too large portion of neighbors. And, the favorable performance is obtained at $h = 0.1$ as shown in Table 3a; the larger bandwidth $h = 1$ produces better performance than the smaller one $h = 0.05$, showing that it is favorable to pick up somewhat large amount of neighbors for constructing discriminative p.d.f gradients. In this study, we employ $h = 0.1$.

Orientation coding of p.d.f gradients. Next, we focus on the way of coding p.d.f gradient orientations. As described in Sec.2.2, those orientations are coded by using the PCA basis vectors. For the alternative to the PCA bases, we can also employ the random orthonormal bases to code them in a similar way. The performance comparison is shown in Table 3b, demonstrating that the PCA bases substantially improve the performance. In such a case of complete set of orientation bases, which is smaller than overcomplete one, the data-driven bases provided by PCA effectively code the orientations.

Component weighting. The third issue is related to the weighting of the orientation codes as described in the last paragraph of Sec.2.2. The effectiveness of the weighting by the inverse of the PCA eigenvalues is shown in Table 3c with comparison to the case without weighting. The per-

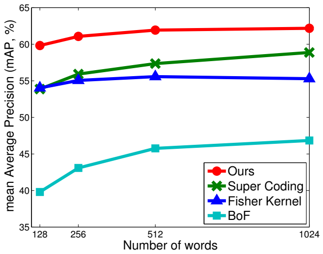


Figure 4. Performances on various numbers of words

Table 4. Comparison with VOC2007 winner

category	Ours 256 w.	Ours 1024 w.	winner
aeroplane	77.1	75.0	**77.5**
bicycle	66.6	**68.3**	63.6
bird	54.7	**58.2**	56.1
boat	69.7	69.5	**71.9**
bottle	29.1	**33.3**	33.1
bus	67.8	**68.9**	60.6
car	79.1	**80.0**	78.0
cat	63.5	**65.8**	58.8
chair	55.8	**55.9**	53.5
cow	49.8	**50.9**	42.6
diningtable	**61.4**	60.6	54.9
dog	47.0	**50.4**	45.8
horse	77.6	**77.6**	77.5
motorbike	69.8	**70.6**	64.0
person	85.2	**86.2**	85.9
pottedplant	27.8	31.6	**36.3**
sheep	**50.7**	49.6	44.7
sofa	55.2	**56.9**	50.6
train	**79.2**	78.9	79.2
tvmonitor	54.4	**55.5**	53.2
mAP	**61.1**	**62.2**	59.4

formance is improved by the weighting which suppresses the orientations commonly occurring across the categories while enhancing the less-frequent but discriminative ones.

p.d.f gradients. The *normalized* p.d.f gradient $\hat{\nabla} p_{\mathfrak{f}}$ that corresponds to the mean-shift vector [7] is naturally induced via the aggregation considering sampling procedure as described in Sec.2.3. In case that we simply use the (original) p.d.f gradient $\nabla p_{\mathfrak{f}}$ without normalization in (9), the performance is deteriorated as shown in Table 3d. The method employing $\nabla p_{\mathfrak{f}}$ amounts to the aggregation (5) weighted by the probability $p_{\mathfrak{f}}$ which would highly enhance the samples (local descriptors) frequently found in the image. Such samples, however, are regarded as common ones across the categories being less discriminative for the classification. On the other hand, the normalized gradient $\hat{\nabla} p_{\mathfrak{f}}$ leads to the uniform aggregation (5) over $\boldsymbol{x}$; even the less-frequent samples which would be discriminative are fairly treated. In addition, the mean-shift vector $\hat{\nabla} p_{\mathfrak{f}}$ is stable in that it always points to the direction where the p.d.f is increased [7].

Number of visual words. We show the performances on various numbers of words $M \in \{128, 256, 512, 1024\}$ in Fig. 4 with comparison to the methods of Fisher kernel [24], super vector coding [35] and the standard BoF. The proposed method produces stably high performances even on the small amount of words, exhibiting significant improvement over the BoF and the other two methods. The proposed feature is twice the dimensionality of super vector coding [35] when the same number of words are used, but our method produces superior performances to [35] under the same dimensionality; *e.g.*, the proposed method of 256 words is superior to super vector coding of 512 words. These results demonstrates that the p.d.f gradient orientations more effectively characterizes the distribution of the local descriptors, compared to the difference vectors.

Table 5. Performance comparison on Scene-15 dataset

Method	Acc. (%)
Lazebnik *et al.* [19]	81.40 ±0.50
Yand and Newsam [34]	82.51 ±0.43
Dixit *et al.* [11]	85.4
Huang *et al.* [15]	82.55 ±0.41
Liu *et al.* [22]	83.76 ±0.59
Boureau *et al.* [5]	84.3 ±0.5
Fisher kernel (256 words) [24]	82.94 ±0.78
super coding (1024 words) [35]	84.79 ±0.76
ours (256 words)	**85.63** ±0.67

Finally, Table 4 shows the performance comparison of the proposed method with the winner (`INRIA_Genetic`) in PASCAL-VOC2007 [1]. The proposed method is competitive to the winner, and thus we can say that the method effectively works for object recognition.

Since the performance is sufficiently improved by 256 words, in the following experiments, we apply the proposed method with 256 words and similarly Fisher kernel with 256 words and super vector coding with 1024 words.

4.2. Comparison on various datasets

We then apply the proposed method to the datasets of Scene-15 [19] and MIT-Scene [25] for scene classification, UIUC-sports [20] for event classification and Caltech-256 for object classification, with performance comparison to the other methods; for the methods other than Fisher kernel [24] and super vector coding [35], we show the performances that are reported in the referenced papers.

Scene-15 [19]. The dataset contains totally 4,485 images of 15 scene categories in indoor/outdoor scenes, such as *store*, *bedroom* and *kitchen* for indoor, and *coast*, *city* and *forest* for outdoor. Each category includes 200∼400 images of about 300×300 pixels. We follow the standard experimental setup [19]; 100 images per class are randomly selected for training and the rest images are used for test. The averaged classification accuracies over 10 trials are shown in Table 5. The proposed method exhibits the favorable performance compared to the others, though the improvement is not so significant. This is because the dataset is somewhat *easy* due to the strong spatial alignment in the images and the small number of categories, which saturates the performances.

MIT-Scene [25]. This dataset contains 15,620 images from 67 indoor scene categories and all images have a minimum resolution of 200 pixels in the smallest axis. In contrast to Scene-15, this classification task is very challenging due to the large within-class variability and small between-class variability in a large number of categories. Following the experimental protocol in [25], we use 80 images per category for training and 20 images for test. We report the classification accuracy on the training/test split

Table 6. Performance comparison on MIT-Scene dataset

Method	Acc. (%)
Quattoni and Torralba [25]	26.0
Li *et al.* [21]	37.6
Bo *et al.* [4]	41.8
Fisher kernel (256 words) [24]	53.27
super coding (1024 words) [35]	56.17
ours (256 words)	**58.91**

Table 7. Performance comparison on UIUC-sports dataset

Method	Acc. (%)
Li *et al.* [21]	77.88
Li and Fei-Fei [20]	73.40
Dixit *et al.* [11]	84.4
Liu *et al.* [22]	84.56 ±1.5
Gao *et al.* [12]	85.31 ±0.51
Bo *et al.* [4]	85.7 ±1.3
Fisher kernel (256 words) [24]	88.61 ±1.16
super coding (1024 words) [35]	90.83 ±1.06
ours (256 words)	**90.42** ±1.03

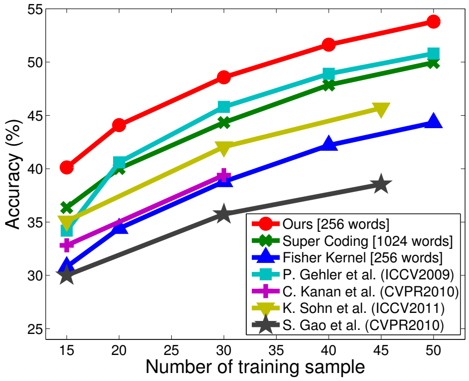


Figure 5. Performance results on Caltech-256 dataset

Table 8. Performance (%) of our method on Caltech-256 dataset

#sample	15	20	30	40	50
Ours	40.1±0.23	44.1±0.46	48.6±0.26	51.6±0.09	53.8±0.26

given by the authors' web page[2]. The performance results are shown in Table 6. We can see that the proposed method significantly outperforms the others including Fisher kernel [24] and super vector coding [35]; the proposed method improves the performance by 17% over the recently developed method [4].

UIUC-sports [20]. This dataset is collected by [20] for image-based event classification. It contains 1,792 images of eight sport categories; *badminton*, *bocce*, *croquet*, *polo*, *rowing*, *rock climbing*, *sailing* and *snowboarding*. Each category contains 137∼250 images. This is a challenging dataset since variations of poses and sizes are quite large across diverse event categories with the cluttered backgrounds. According to the experimental setup used in [20], we randomly select 70 training and 60 test images from each category. We report in Table 7 the averaged classification accuracies over three random training/test splits. The vector aggregation based methods effectively work compared to the other methods; especially, the proposed method significantly outperforms others, though it is comparable to the super vector coding [35]. However, it should be noted that the dimensionality of the proposed feature with 256 words is half of that in super vector coding with 1024 words, which speeds up the classification.

Caltech-256 [14]. This is a challenging dataset for object recognition task. It contains 256 object categories and 30,607 images besides a background (clutter) category in which none of the images belonging to those 256 categories. Each category contains at least 80 images. The intra-class variances regarding such as object locations, sizes and poses in the images are quite large, which makes Caltech-256 a challenging benchmark dataset for object recognition. According to the standard experimental setting, we randomly picked up for training 15, 20, 30, 40, and 50 images per category and 30 images for test. We report the averaged classification accuracy over three trials, and the results are shown in Fig. 5 and Table 8. The proposed method significantly outperforms the other methods [13, 18, 29, 12]; the performances are improved by $3 \sim 5\%$ over the method [13] which uses multiple types of features while the proposed method is based on single type of SIFT local descriptors.

Summary. These experimental results demonstrate that the proposed method produces favorable performances compared to the other existing methods on various tasks of object recognition and scene/event classification using various datasets. It should be noted again that all the results are produced by the proposed method with the identical parameter setting described in this paper, without carefully tuning them for respective datasets. Thus, the parameter setting, especially the bandwidth $h = 0.1$, is shown to be robust, while the performances might be further improved by tuning the parameter setting carefully in each dataset.

We obtain greater performance improvements over the methods of super vector coding [35] and Fisher kernel [24] on PASCAL-VOC2007, MIT-Scene and Caltech-256 datasets. This is because the performances on the remaining datasets of Scene-15 and UIUC-sports are saturated in the proposed method and super coding [35]. Those two datasets contain smaller number of categories, rendering rather easier classification tasks than the other three datasets; actually, the performances produced by the methods on Scene-15 and UIUC-sports are much higher (about 90%) than those on the other three datasets. The proposed method improves the performances more significantly on the more challenging (difficult) datasets due to its high discriminative power.

[2] http://web.mit.edu/torralba/www/indoor.html

5. Conclusion

We have proposed a novel method to extract effective features for image classification. In the framework of BoF which extracts a plenty of local descriptors from an image, the proposed method is built upon the probability density function (p.d.f) obtained by applying kernel density estimator to those local descriptors. The method exploits the oriented p.d.f gradients to effectively characterize the p.d.f, which are subsequently coded and aggregated into the orientation histograms. The proposed method produces generic image features without imposing any assumption on the task, and thus it is applicable to any kinds of image classification tasks. In the experiments on object recognition and scene/event classification using various datasets, the proposed method exhibited the superior performances, compared even to the recently developed methods. Our future work includes to apply the proposed method to an image search as in [17].

References

[1] The PASCAL Visual Object Classes Challenge 2007 (VOC2007). http://www.pascal-network.org/challenges/VOC/voc2007/index.html.

[2] S. Amari. Natural gradien tw orks encien tly in learning. *Neural Computation*, 10:251–276, 1998.

[3] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2007.

[4] L. Bo, X. Ren, and D. Fox. Hierarchical matching pursuit for image classification: Architecture and fast algorithms. In *NIPS*, pages 2115–2123, 2011.

[5] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning mid-level features for recognition. In *CVPR*, pages 2559–2566, 2010.

[6] D. Comaniciu. An algorithm for data-driven bandwidth selection. *Pattern Analysis and Machine Intelligence*, 25(2):281–288, 2003.

[7] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

[8] D. Comaniciu, V. Ramesh, and P. Meer. The variable bandwidth mean shift and data-driven scale selection. In *ICCV*, pages 438–445, 2001.

[9] G. Csurka, C. Bray, C. Dance, and L. Fan. Visual categorization with bags of keypoints. In *ECCV Workshop on Statistical Learning in Computer Vision*, pages 1–22, 2004.

[10] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, pages 886–893, 2005.

[11] M. Dixit, N. Rasiwasia, and N. Vasconcelos. Adapted gaussian models for image classification. In *CVPR*, pages 937–943, 2011.

[12] S. Gao, I.-H. Tsang, L.-T. Chia, and P. Zhao. Local features are not lonely - laplacian sparse coding for image classification. In *CVPR*, pages 3555–3561, 2010.

[13] P. Gehler and S. Nowozin. On feature combination for multiclass object classification. In *ICCV*, pages 221–228, 2009.

[14] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical Report 7694, Caltech, 2007.

[15] Y. Huang, K. Huang, Y. Yu, and T. Tan. Salient coding for image classification. In *CVPR*, pages 1753–1760, 2011.

[16] T. S. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. In *NIPS*, pages 487–493, 1999.

[17] H. Jégou, M. Douse, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *CVPR*, pages 3304–3311, 2010.

[18] C. Kanan and G. Cottrell. Robust classification of objects, faces, and flowers using natural image statistics. In *CVPR*, pages 2472–2479, 2010.

[19] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, pages 2169–2178, 2006.

[20] L.-J. Li and L. Fei-Fei. What, where and who? classifying events by scene and object recognition. In *ICCV*, pages 1–8, 2007.

[21] L.-J. Li, H. Su, Y. Lim, and L. Fei-Fei. Objects as attributes for scene classification. In *International Workshop on Parts and Attributes*, 2010.

[22] L. Liu, L. Wang, and X. Liu. In dense of soft-assignment coding. In *ICCV*, pages 2486–2493, 2011.

[23] D. Lowe. Distinctive image features from scale invariant features. *International Journal of Computer Vision*, 60:91–110, 2004.

[24] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, pages 1–8, 2007.

[25] A. Quattoni and A. Torralba. Recognizing indoor scenes. In *CVPR*, pages 413–420, 2009.

[26] D. W. Scott and M. P. Wand. Feasibility of multivariate density estimates. *Biometrika*, 78(1):197–205, 1991.

[27] M. Singh and N. Ahuja. Regression based bandwidth selection for segmentation using parzen windows. In *ICCV*, pages 2–9, 2003.

[28] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV*, pages 1470–1477, 2003.

[29] K. Sohn, D. Y. Jung, H. Lee, and A. O. Hero III. Efficient learning of sparse, distributed, convolutional feature representations for object recognition. In *ICCV*, pages 2643–2650, 2011.

[30] V. Vapnik. *Statistical Learning Theory*. Wiley, 1998.

[31] M. Wand and M. Jones. *Kernel Smoothing*. Chapman and Hall, 1995.

[32] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *CVPR*, pages 3360–3367, 2010.

[33] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *CVPR*, pages 1794–1801, 2009.

[34] Y. Yang and S. Newsam. Spatial pyramid co-occurrence for image classification. In *ICCV*, pages 1465–1472, 2011.

[35] X. Zhou, K. Yu, T. Zhang, and T. S. Huang. Image classification using super-vector coding of local image descriptors. In *ECCV*, pages 141–154, 2010.