Improving Local Descriptors by Embedding Global and Local Spatial Information

Tatsuya Harada, Hideki Nakayama, and Yasuo Kuniyoshi

Graduate School of Information Science and Technology, The University of Tokyo,
7-3-1 Hongo Bunkyo-ku, Tokyo 113-8656, Japan
{harada,nakayama,kuniyosh}@isi.imi.i.u-tokyo.ac.jp
http://www.isi.imi.i.u-tokyo.ac.jp/

Abstract. In this paper, we present a novel problem: "Given local descriptors, how can we incorporate both local and global spatial information into the descriptors, and obtain compact and discriminative features?" To address this problem, we proposed a general framework to improve any local descriptors by embedding both local and global spatial information. In addition, we proposed a simple and powerful combination method for different types of features. We evaluated the proposed method for the most standard scene and object recognition dataset, and confirm the effectiveness of the proposed method from the viewpoint of speed and accuracy.

Key words: Local Auto Correlation, Weight Maps, Probabilistic Linear Discriminant Analysis, Scene and Object Recognition

1 Introduction

Generic image recognition is one of the important problems of computer vision. However, generic image recognition has not yet been put to practical use, though specific detection techniques such as face detection and person detection are at the production level. The difficulty for generic image recognition is that the images should be recognized even if they appear on different scales and cluttered backgrounds when shown from different perspective views. We focus on images with "spatial biases" as the first step in generic image recognition. If a person takes pictures of objects and scenes, the composition of the pictures has some common properties. For example, the objects are arranged in the center of the picture. In this manner, "spatial biases" occur in the images which the person took for the same purpose. Therefore, image features with spatial information are effective in many cases.

Let us briefly review the descriptors and the features for generic image recognition. Self Similarity [1] and Geometric Blur [2] are descriptors containing spatial information. These descriptors represent local spatial structures on interesting points or grid points in the image. The HOG feature [3] is formed by concatenating gradient histograms in each cell, and represents global spatial information. The Gist feature [4] is implemented by dividing the image into grid regions, calculating spatial envelopes in each cell, and finally combining spatial envelopes to form one image feature. Therefore, Gist represents a global spatial structure. The PHOG [5] or PHOW [6] features consist of a weighted concatenation of the HOG or Bags of Words (BoW) features [7] over each image sub-region at each resolution level. They also represent global spatial information. Notice that BoW itself discards spatial information, but PHOW which incorporates spatial information into BoW obtains better results on some object recognition datasets. The above mentioned descriptors and features have been proven experimentally to have good performance for functions such as object recognition, scene recognition and human detection. This fact reveals the importance of designing good features with both local and global information of the image.

In this paper, we present a novel problem: "Given local descriptors, how can we incorporate both local and global spatial information into the descriptors, and obtain compact and discriminative features?" For this problem, we propose a general framework to improve any local descriptors by embedding both local and global spatial information, and improving recognition performance for the task where spatial information is essential.

In addition, by applying the proposed framework to many descriptors, we can obtain multiple features from the image. In this situation, a classifier, which combines multiple features, is also important. Recently, Multiple Kernel Learning (MKL) (for example [8]) has attracted attention as a powerful classifier combining weighted multiple kernel machines. MKL is based on a kernel method, and thus is faced with the problem of learning time for a large amount of training data in exchange for high classification performance. Furthermore, a nearest neighbor approach acquires high classification performance without learning time, and in general, it needs a huge amount of classification time, because an input pattern is compared with all the training patterns or all the training local descriptors. To solve this problem the hashing technique is usually employed. In this paper, we propose a simple classifier "Naive Bayes Probabilistic Linear Discriminant Analysis (PLDA)" which combines many features based on a Naive Bayes scheme. This classifier does not need an optimization process to assign weights to each feature. In addition, it does not need to perform a comparison with all patterns, but with a small number of prototypes. For this reason, the proposed classifier is fast both in learning and classification processes.

Our proposed framework is inspired by many previous studies. The calculation of local spatial information is based on Higher-order Local Auto Correlation (HLAC) features [9]. Improving any local descriptors is inspired by the Covariance [10] and GLC [11] features. The calculation of global spatial information is based on Fisher Weight Maps and Eigen Weight Maps [12]. Our technical contribution is to generalize those techniques, and propose a new framework to incorporate local and spatial information into arbitrary local descriptors. Most descriptors are improved substantially by applying our method (see Section 6). To the best of our knowledge, no one has proposed a general framework to improve any local descriptors by incorporating both local and global spatial information. Furthermore, the Naive Bayes PLDA is a new approach in combining different types of features. This approach is simple and fast, and obtains good results.

2 Outline of the Proposed Method

In this section, we explain the outline of the proposed method. Initially, an input image is partitioned into spatial grids (cell). M is the number of partitioned cells. In each cell, we calculate features representing the cell from local descriptors. We call these features region features. We extract K kinds of features, such as texture, shape and color. The k-th feature of j-th region in the image I_i is denoted by $\boldsymbol{f}_{ij}^{(k)} \in \mathbb{R}^{d^{(k)}}$. For the image I_i, one feature $\boldsymbol{f}_i$ is obtained by concatenating all region features.

$$\boldsymbol{f}_i^{(k)} = (\boldsymbol{f}_{i1}^{(k)T} \cdots \boldsymbol{f}_{iM}^{(k)T})^T, \tag{1}$$

$$\boldsymbol{f}_i = (\boldsymbol{f}_i^{(1)T} \cdots \boldsymbol{f}_i^{(K)T})^T. \tag{2}$$

Now, we consider C classes $\{\omega_l\}_{l=1}^{C}$, and use Bayes decision rule to classify the image feature $\boldsymbol{f}_i$.

$$c = \arg\max_l \{P(\omega_l|\boldsymbol{f}_i)\} \Rightarrow \boldsymbol{f}_i \in \omega_c. \tag{3}$$

Note that $p(\boldsymbol{f}_i)$ is the normalization constant ensuring that the posterior distribution is a valid probability distribution. Moreover, assuming that the prior probability $P(\omega_l)$ is the same value for all the classes, the prior probability can be eliminated.

$$c = \arg\max_l \{p(\boldsymbol{f}_i|\omega_l)\} \Rightarrow \boldsymbol{f}_i \in \omega_c. \tag{4}$$

Here, the problem is how to estimate the probability density $p(\boldsymbol{f}_i|\omega_l)$, and how to handle the high dimensional image feature $\boldsymbol{f}_i$, which is generated by combining many kinds of features. Direct application of the feature $\boldsymbol{f}_i$ is inadvisable because of dimensionality. Therefore, we assume all the k type features $\boldsymbol{f}_i^{(k)}$ are independent conditioning on the class ω_l, and convert the problem into the estimation of each probability density in low dimensional space.

$$\begin{aligned} p(\boldsymbol{f}_i|\omega_l) &= p\big((\boldsymbol{f}_i^{(1)T} \cdots \boldsymbol{f}_i^{(K)T})^T|\omega_l\big), &(5)\\ &= p\big(\boldsymbol{f}_i^{(1)}|\omega_l\big)p\big(\boldsymbol{f}_i^{(2)}|\omega_l\big) \cdots p\big(\boldsymbol{f}_i^{(K)}|\omega_l\big), &(6)\\ &= \prod_{k=1}^{K} p\big(\boldsymbol{f}_i^{(k)}|\omega_l\big). &(7)\end{aligned}$$

The log of Eqn. 7 can be written in the form

$$\ln p(\boldsymbol{f}_i|\omega_l) = \sum_{k=1}^{K} \ln p\big(\boldsymbol{f}_i^{(k)}|\omega_l\big). \tag{8}$$

The problem is simplified to estimating the likelihood $p\big(\boldsymbol{f}_i^{(k)}|\omega_l\big)$ for each k-th feature following the naive Bayes approach. The assumption of conditional independence is a very strict assumption, but the naive Bayes approach has been proved to show higher performance than expected [13]. In fact, in generic object recognition, the Naive Bayes Nearest Neighbor (NBNN) approach [14] achieved satisfactory results, hence we expect our approach to achieve good performance in image recognition.

Although the problem is divided into the estimation of $p\big(\boldsymbol{f}_i^{(k)}|\omega_l\big)$, the feature $\boldsymbol{f}_i^{(k)}$ is still a high dimensional vector, since the k-th feature consists of M region features. Obviously, the k-th feature can be divided into region features with conditional independence assumptions for all sorts of features.

$$p\big(\boldsymbol{f}_i^{(k)}|\omega_l\big) = \prod_{m=1}^{M} p\big(\boldsymbol{f}_{im}^{(k)}|\omega_l\big). \tag{9}$$

In this assumption, we discard spatial information between the cells. Using spatial information enhances classification performance for images with a strong alignment of objects [5]. For this reason, we consider the weighted sum of region features to implicitly present spatial information.

$$\begin{aligned} \boldsymbol{g}_i^{(k)} &= w_1^{(k)}\boldsymbol{f}_{i1}^{(k)} + w_2^{(k)}\boldsymbol{f}_{i2}^{(k)} + \cdots + w_M^{(k)}\boldsymbol{f}_{iM}^{(k)}, & (10)\\ &= \sum_{m=1}^{M} w_m^{(k)}\boldsymbol{f}_{im}^{(k)}, & (11) \end{aligned}$$

where $w_m^{(k)} \in \mathbb{R}$ is a weight for the m-th region of the k-th feature in the image I_i. Let $F_i^{(k)} \in \mathbb{R}^{M\times d^{(k)}}$ denote the $M \times d^{(k)}$ matrix where the M row vectors are the region features.

$$F_i^{(k)T} = (\boldsymbol{f}_{i1}^{(k)} \cdots \boldsymbol{f}_{iM}^{(k)}). \tag{12}$$

Using this matrix, Eqn. 11 can be simplified as follows:

$$\boldsymbol{g}_i^{(k)} = F_i^{(k)T}\boldsymbol{w}^{(k)}, \tag{13}$$

where $\boldsymbol{w}^{(k)} \in \mathbb{R}^M$ is the region weight vector $\boldsymbol{w}^{(k)} = (w_1^{(k)} \ldots w_M^{(k)})^T$.

The region weight is not limited to one weight vector. We can prepare some region weight vectors, and obtain the new feature vectors by concatenating $\{\boldsymbol{g}_{ij}^{(k)} = F_i^{(k)T}\boldsymbol{w}_j^{(k)}\}_{j=1}^{M'}$, which are the weighted region features.

$$\boldsymbol{g}_i^{(k)'} = (\boldsymbol{g}_{i1}^{(k)T} \ldots \boldsymbol{g}_{iM'}^{(k)T})^T \tag{14}$$

Taking the dimensionality reduction into consideration, the number of region weight vectors is generally $M' \ll M$.

In addition, if $\boldsymbol{g}_i^{(k)'}$ is still a high dimensional vector, we use principle component analysis (PCA) for dimensionality reduction. Let $\boldsymbol{h}_i^{(k)}$ be the transformed vector by using PCA for $\boldsymbol{g}_i^{(k)'}$. Therefore, the final classification rule of Eqn. 8 becomes:

$$c = \arg\max_l \sum_{k=1}^{K} \ln p\big(\boldsymbol{h}_i^{(k)} | \omega_l\big) \Rightarrow \boldsymbol{h}_i^{(k)} \in \omega_c. \tag{15}$$

We can reduce the problem to the estimation of the region weights and the proper probability distributions. In Sections 4 and 5, we explain the implementation of these methods. In Section 5, we use PLDA to estimate the probability distribution. Therefore, we call the proposed method for the combined multiple features the "Naive Bayes PLDA" method. More importantly, up to this point we have not mentioned the selection of the region features. In the next section, we explain how to build the region features including local spatial information from any local descriptors.

3 Local Spatial Information

In this section, we consider the generation of the region features including local spatial information. To this purpose, we calculate the local auto correlation of arbitrary local descriptors. Let $\boldsymbol{\phi}(\boldsymbol{r}_i)$ be the local descriptor at the reference point $\boldsymbol{r}_i$ and $\boldsymbol{a}_j$ be the displacement vector. Then the first-order auto correlation matrix is obtained by:

$$\varPhi(\boldsymbol{a}_j) = \frac{1}{N_J} \sum_{i \in J} \boldsymbol{\phi}(\boldsymbol{r}_i) \boldsymbol{\phi}(\boldsymbol{r}_i + \boldsymbol{a}_j)^T. \tag{16}$$

where J is a region of the image and N_J is the number of local descriptors in the region J. Noticing that 0-th local auto-correlation in the region is the mean of the local descriptors, we have:

$$\bar{\boldsymbol{\phi}} = \frac{1}{N_J} \sum_{i \in J} \boldsymbol{\phi}(\boldsymbol{r}_i). \tag{17}$$

The local auto-correlation of any local descriptors is considered to be a type of Higher-order Local Auto-Correlation Feature [9]. By using the elements of the mean and the local auto-correlation matrix, we obtain the region feature:

$$\boldsymbol{f} = (\bar{\boldsymbol{\phi}}^T \; \eta^T(\varPhi(\boldsymbol{0})) \; \xi^T(\varPhi(\boldsymbol{a}_1)) \; \cdots \; \xi^T(\varPhi(\boldsymbol{a}_{n_a})))^T, \tag{18}$$

where $\eta(\cdot)$ returns a column vector consisting of the elements of the upper triangular portion of the input matrix, $\xi(\cdot)$ returns a column vector consisting of all the elements of the input matrix, and n_a is the number of displacement vectors. Selection of the displacement vectors is limited in the local area according to [9], and the number of displacement vectors is five. Figure 1 shows the five kinds of displacement vector in this paper. Although we can calculate the higher order auto-correlations, we usually use up to the first or second order, because the feature dimension exponentially increases with the increase in the order.

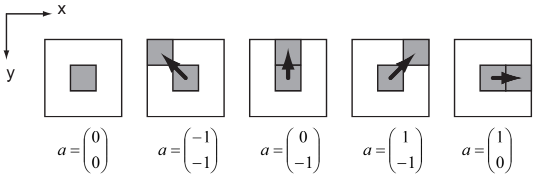

Fig. 1. Displacement vectors of local auto-correlation

Let the dimension of the local descriptor be d, the dimension of the region feature becomes $d + d(d+1)/2 + (n_a - 1)d^2$. In this way, the dimension of the region feature increases as the square of d. If we use the high dimensional local descriptor, it is hard to calculate the region features. Therefore, according to the dimension of descriptors, we can select the calculation of the region feature as follows:

Mean $\boldsymbol{f} = \bar{\boldsymbol{\phi}}$. This is the first statistic of the local descriptor. The dimension is the same as the descriptor d.

Mean + Local Auto-Correlation at $\boldsymbol{a} = 0$ This is the special case of Eqn. 18, $\boldsymbol{f} = (\bar{\boldsymbol{\phi}}^T \ \eta^T(\Phi(\mathbf{0})))^T$. We call this feature the GLC (Generalized Local Correlation) feature [11]. The dimension is $d + d(d+1)/2$.

Mean + all Local Auto-Correlation This is same as Eqn. 18.

The above region features, except the Mean plus all Local Auto-Correlations, do not include spatial information. However, since they calculate the statistics of the local descriptors in the region, we believe these features include meaningful information.

4 Global Spatial Information

In the face recognition, Eigenfaces [15] and Fisherfaces [16] are well known methods for weighting the regions in the image. However, these methods can be applied to images consisting of the scalar values at the pixel such as the brightness, and cannot be applied to the image where each pixel is described by the vector. For this reason, the Fisher Weight Maps (FWM) and Eigen Weight Maps (EWM) are employed as a region weighting method [12]. The original weight maps are defined to weight each pixel in the image. In general, images have different scales and aspect ratios, and the pixel-wise weight maps are not directly utilized in the generic images. Therefore, to absorb the variety of scales and aspect ratios, all images are divided by a regular grid, and weight maps are applied to these regions.

Now, we have the labeled training samples $\{(\boldsymbol{f}_i^{(k)}, y_i)\}_{i=1}^N$. Let $\tilde{\Sigma}_W$ be the within-class covariance matrix of region features, and $\tilde{\Sigma}_B$ be the between-class covariance matrix. The Fisher criterion is given by $J(\boldsymbol{w}) = \frac{tr\tilde{\Sigma}_B}{tr\tilde{\Sigma}_W}$. The traces of

$\tilde{\Sigma}_W$ and $\tilde{\Sigma}_B$ are given by:

$$tr\tilde{\Sigma}_W = \boldsymbol{w}^T \Sigma_W \boldsymbol{w}, \tag{19}$$

$$tr\tilde{\Sigma}_B = \boldsymbol{w}^T \Sigma_B \boldsymbol{w}, \tag{20}$$

where

$$\Sigma_W = \frac{1}{N} \sum_{l=1}^{C} \sum_{i \in \omega_l} (F_i^{(k)} - M_l)(F_i^{(k)} - M_l)^T, \tag{21}$$

$$\Sigma_B = \frac{1}{N} \sum_{l=1}^{C} n_l (M_l - M)(M_l - M)^T, \tag{22}$$

M_l is the mean of $F_i^{(k)}$ belonging to the class ω_l, and M is the mean of total data set. By maximizing the Fisher criteria under the condition $\boldsymbol{w}^T \Sigma_W \boldsymbol{w} = 1$, we can obtain the weight vector $\boldsymbol{w}$ as the eigen vector of the generalized eigenvalue problem.

$$\Sigma_B \boldsymbol{w} = \lambda \Sigma_W \boldsymbol{w}, \tag{23}$$

where λ is the eigen value corresponding to the eigen vector $\boldsymbol{w}$. We select the M' largest eigen values $\lambda_1, \cdots, \lambda_{M'}$, and the corresponding eigen vectors $\boldsymbol{w}_1, \cdots, \boldsymbol{w}_{M'}$, and calculate the weighted feature vector by using Eqn. 13 and Eqn. 14. These weight vectors are called Fisher Weight Maps (FWM). Because this method uses the matrix consisting of the region features, it is considered to be the generalized Fisher discriminant analysis.

In the case of the presence of clutter or occlusion, there would be difficulty establishing recognition if we use the global spatial information. However, the method automatically weights the discriminative regions, and ignores less discriminative regions. If unobservable regions are less important, this method is expected to work properly in the presence of clutter or occlusion.

5 Classifier

For the estimation of the probability density, Probabilistic Linear Discriminant Analysis is employed. Some variations of PLDA have been proposed [17][18][19]. In this paper, reference [17] is utilized whose solution is similar to that of Linear Discriminant Analysis (LDA). We note that the density estimation can be replaced by [18] and [19].

Suppose that N training samples $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathbb{R}^d$ are given, and all training samples belong to one of the C classes $\omega_1, \ldots, \omega_C$. In addition, assuming that the test sample $\boldsymbol{x}^t$ belonging to one of the C classes is given, we want to decide the class of the test sample. Let $\boldsymbol{u} = A^{-1}(\boldsymbol{x} - \boldsymbol{m})$ ($A \in \mathbb{R}^{d \times d'}$, $\boldsymbol{m} \in \mathbb{R}^d$) be the Affine transformation that transforms the input vector to the latent variable $\boldsymbol{u}$. Let $\boldsymbol{u}^t$ be the latent variable of the test sample, and $\{\boldsymbol{u}_i\}_{i=1}^N$ be the latent variables of the training samples. The probability that $\boldsymbol{u}^t$ belongs to the class ω_j is given by:

$$p(\boldsymbol{u}^t|\omega_j) = \mathcal{N}(\boldsymbol{u}^t|\frac{n_j\Psi}{n_j\Psi + I}\bar{\boldsymbol{u}_j}, I + \frac{\Psi}{n_j\Psi + I}), \tag{24}$$

where n_j is the number of samples belonging to the class ω_j, $\bar{\boldsymbol{u}_j}$ is the mean of the samples $\bar{\boldsymbol{u}_j} = \frac{1}{n_j}\sum_{\boldsymbol{u}_i \in \omega_j} \boldsymbol{u}_i$ belonging to the class ω_j, and $\Psi \in \mathbb{R}^{d' \times d'}$ is the diagonal matrix.

Here, we show the calculation of the parameters $\boldsymbol{m}$, A, and Ψ in Eqn. 24. At first, the between-class covariance matrix $S_b \in \mathbb{R}^{d \times d}$ and the within-class covariance matrix $S_w \in \mathbb{R}^{d \times d}$ are given by:

$$S_w = \frac{1}{N}\sum_l \sum_{i \in \omega_l} (\boldsymbol{x}_i - \boldsymbol{m}_l)(\boldsymbol{x}_i - \boldsymbol{m}_l)^T, \tag{25}$$

$$S_b = \frac{1}{N}\sum_l n_l(\boldsymbol{m}_l - \boldsymbol{m})(\boldsymbol{m}_l - \boldsymbol{m})^T, \tag{26}$$

where n_l is the number of samples in class ω_l, N is the number of the total training samples $N = \sum_l n_l$, $\boldsymbol{m}_l = \frac{1}{n_l}\sum_{i \in \omega_l} \boldsymbol{x}_i$ is the mean of the samples in class ω_l, and $\boldsymbol{m} = \frac{1}{N}\sum_i \boldsymbol{x}_i$ is the mean of the total training samples. Next, we calculate the transformation $\boldsymbol{y} = W^T\boldsymbol{x}$, $\boldsymbol{y} \in \mathbb{R}^{d'}$, $W \in \mathbb{R}^{d \times d'}$ that maximizes the between-class covariance matrix to the within-class covariance matrix. This process is the same as LDA, and the optimal projection matrix is given by the solution of generalized eigenvalue problem:

$$S_b W = S_w W \Lambda, \tag{27}$$

where Λ is the diagonal matrix with eigenvalues. The dimension d' of the discriminant space is given by $d' \leq \min(C-1, d)$.

Now we diagonalize the between-class covariance matrix and the within-class covariance matrix ($\Lambda_b = W^T S_b W$, $\Lambda_w = W^T S_w W$). From these diagonalizations, the parameters $\boldsymbol{m}$, A, and Ψ are given by:

$$\boldsymbol{m} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i, \tag{28}$$

$$A = W^{-T}(\frac{n}{n-1}\Lambda_w)^{1/2}, \tag{29}$$

$$\Psi = \max\big(0, \frac{n-1}{n}\frac{\Lambda_b}{\Lambda_w} - \frac{1}{n}\big), \tag{30}$$

where $n = N/C$.

To solve the PLDA, we calculate only the d dimensional generalized eigenvalue problem in Eqn. 27. Calculation complexity depends on the dimension. It is true that S_b and S_w depend on the number of samples, but it is easy to modify the calculation of correlation matrices into incremental manner. Moreover, because Eqn. 24 is the uni-modal Gaussian distribution, the classification rule in Eqn. 15 is simplified.

6 Experiment

6.1 Setup

We selected the following descriptors and features.

HLAC We used at most second order HLAC features [9]. As a preprocessing step, we extracted edges by using the Canny operator, and obtains the binary images. The HLAC features were extracted for the binary images. The dimension of at most second order HLAC features was 25.

Color HLAC The Color HLAC feature was one of the HLAC features [9] except that the RGB values were utilized as the local descriptor. We used at most first order Color HLAC features. The dimension of at most first order Color HLAC features was 45.

HOG The HOG (Histograms of Oriented Gradients) [3] was implemented by dividing the image window into small spatial cells. In each cell, the histogram of oriented gradients was calculated. All histograms were concatenated to represent the image. We considered the histogram in each cell as the descriptor. We used an unsigned gradient HOG descriptor, and used a 20° bin size. The dimension of the local descriptor was 9 in our setting.

SIFT We used the densely sampled SIFT descriptor [20] for each 5 pixels. We did not use the orientation normalization, and calculates the gray SIFT with $r = 8$. The dimension of the SIFT was 128. 128 dimensions were too high to calculate the local auto-correlation. Because the SIFT descriptor consists of 4×4 cells, we considered the histogram of each cell as the local descriptor. The dimension of the histogram in the cell was 8. Therefore, since we did not use the SIFT directly, we denote the SIFT- with the ”-” mark.

Self Similarity (SS) The Self Similarity [1] calculates the correlation between the reference point and the surrounding points around the reference point. We used the angle bin = 8, and the radial interval = 3. The dimension of this descriptor was 24.

PHOG The PHOG (Pyramid Histogram of Oriented Gradients) [5] is a global feature, and therefore we did not calculate the GLC and the local auto-correlation, but used the weight maps to reduce the dimensionality.

Gist The Gist [4] is also a global feature. For this reason, we did not calculate the GLC and the local auto-correlation, but used the weight maps to reduce the dimensionality. We calculated both the gray Gist and RGB Gist for the 6 directions and 6 scales. The dimension of the gray Gist and the RGB Gist were 36 and 108 respectively.

We denote the descriptor, which is improved by embedding Global and Local Spatial (GLS) information, as (descriptor) + GLS. In the same manner, we denote the descriptor with GLC and FWM as (descriptor) + GLC + FWM.

We tested the experiments on the standard workstation (XeonW5590 (3.33 GHz) ×2 = 8 core, 48GB ram) using Matlab. We did not use special acceleration techniques, such as MEX, for the implementation of both learning and classification.

6.2 Scene classification

We experimented with a commonly used scene classification benchmark dataset by Lazebnik et al., [21] (LSP15). LSP15 consists of gray images of OT8 [4] plus seven additional classes. OT8 consists of 2,688 color images of eight classes. Each class has 260 to 410 sample images. In all, it has 4,492 gray images. LSP15 has the largest number of target classes among scene datasets currently in use. We randomly chose 100 training images for each class in LSP15. We used the remaining samples as test data, and calculated the mean of the classification rate for each class. This score was averaged over many trials replacing the training and test samples randomly. This is the methodology used in previous studies.

Initially, we tested the performance of the framework of embedding global and spatial information into any descriptors. We compared (descriptor) + GLS with the baseline features and (descriptor) + GLC + FWM. The baseline features were obtained by concatenating all mean descriptors in each grid. FWM was not applied to the baseline features. The dimensions of PCA were selected to get the best performance for all features. The dimensions of weight maps were also selected to get the best performance for (descriptor) + GLC + FWM and (descriptor) + GLS. We used the simple LDA as the classifier in order to compare the performances of the features themselves.

Table 1 shows these results on LSP15 with the single features. The bold number means the best score in each feature. We can see that GLS improves the classification performance significantly for all descriptors. We changed the classifier to PLDA. SIFT- (2 and 8 scales) + GLS + PLDA obtained (80.1 [%]), which is comparable to SIFT + hard quantization + intersection kernel (80.1 [%]), but inferior to SIFT + sparse codes + intersection kernel (84.3 [%]) [22][23].

We evaluated the combination of multiple features on the LSP15. We used four features (HOG + GLS, SIFT-(2 and 8 scales) + GLS, SS + GLS, gray Gist), and combined them with the Naive Bayes PLDA. In LSP15, our method obtained the comparable score (86.6 [%]) to the state-of-the-art methods (Xiao et al. [24] (88.1 [%]), Zhou et al. [25] (85.2 [%]), Nakayama et al. [11] (84.1 [%]), Bosch et al. [26] (83.7 [%]), Lazebnik et al. [21] (81.4 [%])).

The Naive Bayes PLDA calculates the classifiers of each feature independently, and requires no optimization process to weight each classifier. The learning cost of PLDA is almost same as LDA. On classification, the Naive Bayes PLDA only sums the log likelihoods of each classifier. The calculation times of LDA on both learning and classification are shown in Table 1. Learning finished within 1 minutes for all features, and classification finished within 0.1 second for all features. Therefore, GLS + (Naive Bayes) PLDA approach is fast, and obtains good performance for the scene classification.

6.3 Object recognition

Caltech-101 [27] is the de-facto standard object recognition dataset. This dataset consists of images from 101 object categories and one background class, and contains from 31 to 800 images per category. This dataset has large intra-class

Table 1. Classification results on LSP15 with single features

Feature	Grid	LDA dim	Maps dim	PCA dim	Classification rate [%]	Learn [sec]	Classify [sec]
HOG (baseline)	2x2	14	4	36	54.8 ± 1.4	0.04	0.02
	4x4	14	16	100	61.5 ± 1.8	0.11	0.02
	6x6	14	36	100	**62.3 ± 0.6**	0.28	0.02
	8x8	14	64	100	61.2 ± 1.1	1.35	0.02
HOG +GLC +FWM	2x2	14	4	216	67.6 ± 1.0	0.20	0.02
	4x4	14	8	300	72.9 ± 1.3	0.64	0.02
	6x6	14	8	300	74.2 ± 0.5	0.64	0.02
	8x8	14	8	300	**74.5 ± 0.7**	0.71	0.02
HOG +GLS (proposed)	2x2	14	4	300	72.6 ± 1.0	24.29	0.04
	4x4	14	5	500	75.9 ± 0.9	35.99	0.04
	6x6	14	5	500	**77.3 ± 0.8**	36.18	0.04
	8x8	14	5	500	77.1 ± 0.7	36.35	0.05
SIFT- (baseline)	2x2	14	4	32	56.6 ± 1.2	0.04	0.02
	4x4	14	16	60	**63.1 ± 0.9**	0.10	0.02
	6x6	14	36	80	62.5 ± 0.7	0.23	0.02
	8x8	14	64	120	61.1 ± 1.4	1.02	0.03
SIFT- +GLC +FWM	2x2	14	4	176	56.9 ± 1.6	0.15	0.02
	4x4	14	8	350	66.9 ± 0.9	0.55	0.03
	6x6	14	8	350	69.0 ± 1.3	0.60	0.03
	8x8	14	8	350	**70.4 ± 0.8**	0.70	0.03
SIFT- +GLS (proposed)	2x2	14	4	600	66.2 ± 1.3	14.39	0.03
	4x4	14	4	600	73.1 ± 1.0	14.50	0.04
	6x6	14	4	600	74.3 ± 0.9	14.65	0.04
	8x8	14	4	600	**75.3 ± 0.7**	15.00	0.04
SS (baseline)	2x2	14	4	96	64.3 ± 1.3	0.07	0.02
	4x4	14	16	100	**65.9 ± 0.7**	0.47	0.02
	6x6	14	36	100	63.3 ± 0.9	4.93	0.03
	8x8	14	64	100	60.9 ± 1.4	40.51	0.04
SS +GLC +FWM	2x2	14	2	400	**80.0 ± 0.6**	2.04	0.03
	4x4	14	2	400	79.4 ± 0.7	2.24	0.03
	6x6	14	2	400	78.7 ± 0.9	2.40	0.03
	8x8	14	2	400	78.3 ± 0.9	2.72	0.03
SS+GLS (proposed)	2x2	14	2	400	**80.8 ± 0.7**	36.03	0.04
	4x4	14	2	400	80.4 ± 0.7	36.43	0.04

variety. The spatial information is essential for the image recognition, because the images in the same category are well centered. To evaluate classification performance, we followed the most standard methodology. 15 images are randomly selected from all 102 categories for training, and another random 15 for testing.

We tested the performance of the framework of embedding global and spatial information into any descriptors in the same manner as the scene classification experiment. Table 2 shows these results on Caltech-101 with the single features. The bold number means the best score in each feature. We can see also that (descriptor) + GLS improves the classification performance significantly for all descriptors even in the object recognition dataset. We also checked (descriptor) + GLS + PLDA. SIFT- (2 and 8 scales) + GLS + PLDA obtained (55.4 [%]), which is comparable to SIFT + spatial pyramid + hard quantization + kernel SVM (56.4 [%]), but inferior to SIFT + spatial pyramid + sparse codes + max pooling + kernel SVM (67.0 [%]) [22][23].

We evaluated the performance of the combination of multiple features. We use eight features (HLAC (1/1 and 1/2 resolutions), Color HLAC, HOG + GLS, SIFT- (2 and 8 scales)+ GLS, SS + GLS, PHOG + FWM, gray Gist + FWM,

Table 2. Classification results on Caltech101 with single features

Feature	Grid	LDA dim	Maps dim	PCA dim	Classification rate [%]	Learn [sec]	Classify [sec]
HOG (baseline)	2x2	36	4	36	22.1 ± 1.5	0.1	0.1
	4x4	101	16	120	35.3 ± 1.4	0.2	0.4
	6x6	101	36	120	36.9 ± 0.8	0.3	0.4
	8x8	101	64	120	**37.0 ± 1.0**	1.4	0.4
HOG +GLC +FWM	2x2	101	4	216	30.8 ± 0.9	0.3	0.4
	4x4	101	8	300	40.1 ± 1.6	0.8	0.4
	6x6	101	8	300	41.0 ± 0.6	0.8	0.4
	8x8	101	8	300	**41.2 ± 1.5**	0.9	0.4
HOG +GLS (proposed)	2x2	101	4	300	41.7 ± 1.3	25.1	0.4
	4x4	101	8	300	44.3 ± 1.2	33.9	0.4
	6x6	101	8	300	44.3 ± 1.0	34.2	0.4
	8x8	101	8	300	**45.2 ± 0.6**	34.4	0.4
SIFT- (baseline)	2x2	32	4	32	28.1 ± 1.3	0.1	0.1
	4x4	101	16	120	42.9 ± 1.2	0.2	0.4
	6x6	101	36	120	**44.8 ± 1.7**	0.3	0.4
	8x8	101	64	120	44.4 ± 0.8	1.1	0.4
SIFT- +GLC +FWM	2x2	101	4	176	31.7 ± 1.3	0.2	0.4
	4x4	101	8	350	44.8 ± 0.7	0.9	0.4
	6x6	101	8	350	44.6 ± 0.6	0.9	0.4
	8x8	101	8	350	**47.2 ± 1.4**	1.0	0.4
SIFT- +GLS (proposed)	2x2	101	4	600	46.3 ± 1.0	15.1	0.5
	4x4	101	4	600	50.2 ± 1.2	15.4	0.5
	6x6	101	4	600	52.6 ± 1.1	15.6	0.5
	8x8	101	4	600	**53.4 ± 1.0**	16.2	0.5
SS (baseline)	2x2	96	4	96	40.3 ± 1.4	0.1	0.4
	4x4	101	16	120	**43.8 ± 1.4**	0.5	0.4
	6x6	101	36	120	42.6 ± 1.5	5.3	0.5
	8x8	101	64	120	42.3 ± 1.3	42.4	0.4
SS +GLC +FWM	2x2	101	4	350	50.3 ± 1.0	16.5	0.4
	4x4	101	6	350	50.7 ± 0.9	34.7	0.4
	6x6	101	6	350	**51.7 ± 1.4**	35.0	0.4
	8x8	101	6	350	51.4 ± 1.4	35.5	0.4
SS+GLS (proposed)	2x2	101	4	350	**52.6 ± 0.8**	64.7	0.5
	4x4	101	6	350	52.5 ± 1.3	100.4	0.5

RGB Gist + FWM). Our classification rate achieves 66.4 ± 1.0 [%]. This performance is lower than the state-of-the-art methods (Lin et al. [28] (75.8 ± 1.1 [%]), Boiman et al. [14] (72.8 ±0.39 [%]), Bosch et al. [6] (70.4 ± 0.7 [%])), but has comparable results with Frome et al. [29] (63.2 [%]) , Zhang et al. [30] (59.1 ± 0.56 [%]), and Lazebnik et al. [21] (56.4 [%]). It should be noted that our classification method is very simple and does not use the optimization of weight for each feature, and the learning and classification times are about 150 [sec] and 10 [msec/frame]. Our combination method shows a good trade-off between the computational costs and the classification performance for Caltech-101.

Finally, we evaluated the effect of the dimension of the weight maps and the PCA dimension on the Caltech-101. Figure 2 shows the results of the SIFT- + GLC + FWM, and SIFT- + GLS with 8 grid cells using PLDA. Since there are no spaces to show all results, we picked out only two typical results. We can see that the peak classification performances are achieved around 10 dimensions with the SIFT- + GLC + FWM and SIFT- + GLS. These results show the effectiveness of dimensionality reduction of the weight maps.

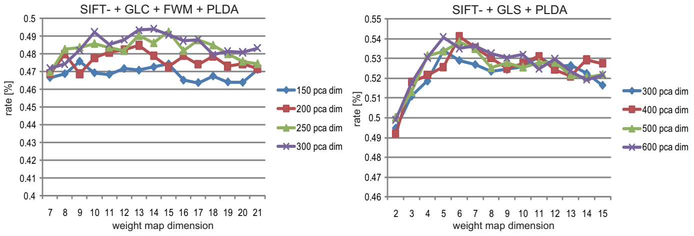

Fig. 2. Effect of the dimension of the weight maps and the PCA dimension on Caltech-101 dataset. The left figure shows the result with the single SIFT- + GLC + FWM. Right figure shows the result with the single SIFT- + GLS (proposed).

7 Conclusions

In this paper, we proposed a general framework to improve any local descriptors by embedding both local and global spatial information. To incorporate local spatial information, we calculated the local auto-correlation of the densely sampled local descriptors, and generated the region features. Then we calculated the weighted sum of the region features by using discriminative weight maps to embed global spatial information. We also proposed a simple classifier "Naive Bayes PLDA" which combined many features based on a Naive Bayes scheme. Experimental results show that our method is very simple and fast, and boosts all descriptors substantially.

There are a lot of things to improve the performance. Here we calculated the spatial correlation of the same descriptor. Our idea can be easily extended to the spatial correlation of different descriptors by which the conditional independence of Naive Bayes PLDA can be relaxed. The performance of GLC is improved by using the information geometory based metric [31]. By following this idea, we will also invent the proper similarity measure for our framework.

References

1. Shechtman, E., Irani, M.: Matching local self-similarities across images and videos. In: CVPR. (2007)
2. Berg, A.C., Berg, T.L., Malik, J.: Shape matching and object recognition using low distortion correspondence. In: CVPR. (2005)
3. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR. (2005)
4. Oliva, A., Torralba, A.: Modeling the shape of the scene: a holistic representation of the spatial envelope. IJCV **42** (2001) 145–175
5. Bosch, A., Zisserman, A., Munoz, X.: Representing shape with a spatial pyramid kernel. In: ACM CIVR. (2007)
6. Bosch, A., Zisserman, A., Munoz, X.: Image classification using random forests and ferns. In: ICCV. (2007)

7. Csurka, G., Dance, C.R., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: ECCV International Workshop on Statistical Learning in Computer Vision. (2004)
8. Varma, M., Varma, M.: Learning the discriminative power-invariance trade-off. In: ICCV. (2007)
9. Otsu, N., Kurita, T.: A new scheme for practical, flexible and intelligent vision systems. In: Proc. IAPR Workshop on Computer Vision. (1988)
10. Tuzel, O., Porikli, F., Meer, P.: Region covariance: A fast descriptor for detection and classification. In: ECCV. (2006)
11. Nakayama, H., Harada, T., Kuniyoshi, Y.: Dense sampling low-level statistics of local features. In: ACM CIVR. (2009)
12. Shinohara, Y., Otsu, N.: Facial expression recognition using fisher weight maps. In: IEEE FG. (2004)
13. Domingos, P., Pazzani, M.J.: On the optimality of the simple bayesian classifier under zero-one loss. Machine Learning **29** (1997) 103–130
14. Boiman, O., Shechtman, E., Irani, M.: In defense of nearest-neighbor based image classification. In: CVPR. (2008)
15. Turk, M., Pentland, A.: Face recognition using eigenfaces. In: CVPR. (1991)
16. Belhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. IEEE Trans. on PAMI **19** (1997) 711–720
17. Ioffe, S.: Probabilistic linear discriminant analysis. In: ECCV. (2006)
18. Yu, S., Yu, K., Tresp, V., Kriegel, H.P., Wu, M.: Supervised probabilistic principal component analysis. In: ACM SIGKDD. (2006)
19. Prince, S.J., Elder, J.H.: Probabilistic linear discriminant analysis for inferences about identity. In: ICCV. (2007)
20. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV **60** (2004) 91–110
21. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: CVPR. (2006)
22. Boureau, Y.L., Bach, F., LeCun, Y., Ponce, J.: Learning mid-level features for recognition. In: CVPR. (2010)
23. Yang, J., Yu, K., Gong, Y., Huang, T.: Linear spatial pyramid matching using sparse coding for image classification. In: CVPR. (2009)
24. Xiao, J., Hays, J., Ehinger, K., Oliva, A., Torralba, A.: Sun database: Large scale scene recognition from abbey to zoo. In: CVPR. (2010)
25. Zhou, X., Cui, N., Li, Z., Liang, F., Huang, T.S.: Hierarchical gaussianization for image classification. In: ICCV. (2009)
26. Bosch, A., Zisserman, A., Munoz, X.: Scene classification using a hybrid generative/discriminative approach. IEEE Trans. on PAMI **30** (2008) 712–727
27. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. In: CVPR Workshop on Generative-Model Based Vision. (2004)
28. Lin, Y.Y., Tsai, J.F., Liu, T.L.: Efficient discriminative local learning for object recognition. In: ICCV. (2009)
29. Frome, A., Singer, Y., Sha, F., Malik, J.: Learning globally-consistent local distance functions for shape-based image retrieval and classification. In: ICCV. (2007)
30. Zhang, H., Berg, A.C., Maire, M., Malik, J.: Svm-knn: Discriminative nearest neighbor classification for visual category recognition. In: CVPR. (2006)
31. Nakayama, H., Harada, T., Kuniyoshi, Y.: Global gaussian approach for scene categorization using information geometry. In: CVPR. (2010)