Optical Engineering

SPIEDigitalLibrary.org/oe

Image classification with multiple feature channels

Fuxiang Lu
Xiaokang Yang
Weiyao Lin
Rui Zhang
Songyu Yu

Optical Engineering 50(5), 057210 (May 2011)

Image classification with multiple feature channels

Fuxiang Lu
Shanghai Jiao Tong University
Institute of Image Communication and
Information Processing
Shanghai, 200240, China
and
Lanzhou University
School of Information Science and Engineering
Lanzhou, 730000, China
E-mail: lufux@sjtu.edu.cn

Xiaokang Yang
Weiyao Lin
Rui Zhang
Songyu Yu
Jiao Tong University
Institute of Image Communication and
Information Processing
Shanghai, 200240, China

Abstract. In this paper, we propose a framework for image classification. An image is represented by multiple feature channels which are computed by the bag-of-words model and organized in a spatial pyramid. The main difference among feature channels resides in what type of base descriptor in the bag-of-words model is extracted. The overall features achieve different levels of the trade-off between discriminative power and invariance. Support vector machines with kernels based on histogram intersection distance and χ^2 distance are used to obtain *a posteriori* probabilities of the image in each feature channel. Then, four data fusion strategies are proposed to combine intermediate results from multiple feature channels. Experimental results show that almost all the proposed strategies can significantly improve the classification accuracy as compared with the single cue methods and, especially, prod-max performs best in all experiments. The framework appears to be general and capable of handling diverse classification problems due to the multiple-feature-channel–based representation. Also, it is demonstrated that the proposed method achieves higher, or comparable, classification accuracies with less computational cost as compared with other multiple cue methods on challenging benchmark datasets. [DOI: 10.1117/1.3582852]

Subject terms: image classification; object categorization; scene categorization; feature channel; spatial pyramid; visual descriptor.

Paper 100655PRR received Aug. 13, 2010; revised manuscript received Mar. 13, 2011; accepted for publication Apr. 5, 2011; published online May 19, 2011.

1 Introduction

Image classification, sometimes called object categorization or scene categorization, has drawn considerable attention within the computer vision community in the last decade.[1–7] For a given test image, the learned classifier has to decide which class it belongs to. This problem is still a challenging problem, especially in the presence of high intraclass variation, clutter, occlusion, and illumination changes.

Various methods have been proposed for image classification. Fergus et al.[8] proposed a probability model to represent an object class in terms of a constellation of learned parts and achieved high performance on the Caltech-6 dataset. Agarwal et al.[9] learned a vocabulary of parts and represented images using parts from the vocabulary, together with spatial relations observed among the parts. Recently, Felzenszwalb et al.[10] developed an object detection system based on mixtures of a multiscale deformable part model. Although these part-based methods can offer an intellectually satisfying way to represent real-world objects by combining appearance descriptors with their spatial relations, the learning and inference process for these methods is still extremely complex, which greatly limits their applications. Other researchers used a bag-of-words representation together with the support vector machine (SVM) to recognize images.[11–13] The bag-of-words model retains only the frequencies of the individual visual words and discards all information about their spatial layout. Generally, they have lower computational complexity as compared with other methods. In order to further improve the performance, Lazebnik et al.[5] organized the bag-of-words histogram of the image in a spatial pyramid (SP) and achieved highly competitive results on the Scene-15 (Ref. 5) and Caltech-101 (Ref. 14) datasets.

However, the selection of base local descriptors in the bag-of-words model is still an open problem. To date there exist many descriptors in the literature. Scale invariant feature transform[15] (SIFT) is widely used in object recognition.[5] CENsus TRansform hISTgram[16] (CENTRIST) has been shown as a suitable descriptor for place and scene recognition. Wu and Rehg[17] studied the effects of using a different type of base descriptors for image classification. They found that although an optimal descriptor could be hand-crafted for a given task, it might no longer be optimal for another one. Therefore we think that instead of using a single type of descriptor it is better to combine a set of diverse and complementary descriptors in order to discriminate each class best from all other classes. In this paper, we extend spatial pyramid representation based on a single type of descriptor[5] to the case of multiple types of descriptors (see Fig. 1). Specifically, for a given image we compute the bag-of-words representation with the vocabularies based on different types of descriptors (such as SIFT, CENTRIST, etc.) and then organize them in a SP. From another point of view, if we regard each type of overall feature as a "channel," then the new image representation has multiple feature channels and the difference among them resides in what type of descriptor in the bag-of-words model is extracted. Obviously, the proposed representation has high flexibility such that other feature channels can be easily incorporated.

Varma and Ray[18] combined several types of features by means of multiple kernel learning (MKL). They associate a kernel to each image feature and approximate the optimal feature's kernel as $\mathbf{K}_{\text{opt}} = \sum_k d_k \mathbf{K}_k$ where the weights d_k correspond to the relative importance of the k'th feature

0091-3286/2011/$25.00 © 2011 SPIE

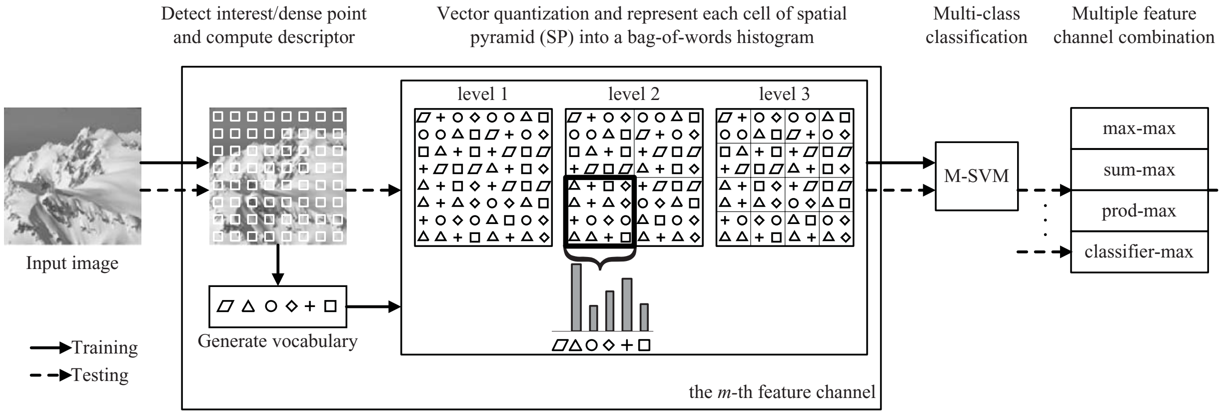


Fig. 1 Our framework for image classification. A bag-of-words histogram of the image is computed and organized in a spatial pyramid. The solid arrows show the training procedure and the dotted ones show the testing procedure.

for a specified task. $d_k = 0$ means that the k'th feature need not be computed for the final decision function. However, Gehler and Nowozin[6] observe that although the MKL solution is sparse for every class separately, it is not sparse jointly in the multiclass setup and almost every feature is selected at least once. Furthermore, they draw a conclusion that the performance of MKL might have been overestimated in the past. Additionally, inspired by linear programming (LP) Boosting they proposed the LP-β approach which outperformed all other methods considered in Ref. 6. Although LP-β obtains higher classification accuracies than MKL, the common problem is that in most cases both of them are computationally too expensive, which greatly limits their application. Furthermore, the architecture of MKL and LP-β is centralized. Data from multiple data sources are sent to a single location where the data are fused. On the contrary, in the distributed architecture the data from individual data sources in lower level nodes are processed and then only the results are sent to higher level nodes to be fused. Although it is conceptually more complicated, the distributed fusion architecture has the following advantages: less processing load at each fusion node; no need to maintain a large centralized database; lower communication load; higher robustness. The distributed fusion architecture is also a necessity since many fusion systems have to be built with existing fusion systems as components. In this paper we focus on the distributed architecture and propose the following four fusion strategies to combine multiple cues at the decision level: max-max, sum-max, prod-max and classifier-max. All the proposed strategies involve two stages. In the first stage, we employ SVMs with histogram intersection kernels (HIKs) or χ^2 kernels to obtain *a posteriori* probabilities of the image using every feature channel separately. In the second stage, the results of the first stage are combined in different fashions to obtain the final decision. Experimental results show that almost all the proposed strategies can significantly improve the classification accuracies as compared with the single cue methods and especially, prod-max performs best in almost all the experiments. In addition, the proposed method is more computationally efficient than both MKL and LP-β, but get better or comparable classification accuracies.

The rest of this paper is organized as follows. In Sec. 2, we describe the multiple-feature-channel–based image representation. Section 3 describes the procedure for recognizing images with a single feature channel. Section 4 presents the proposed strategies to combine multiple feature channels and compares the computational complexity with other multiple cue methods. Experimental results are shown in Sec. 5 and Sec. 6 concludes the paper.

2 Image Representation

As mentioned in Sec. 1, integrating the spatial layout of the components is very helpful in recognizing images. Also, combining multiple features may greatly improve the performance and the robustness of the algorithm since there is no single type of feature which can be optimal in all situations. Based on these observations, in this paper we propose a multiple-feature-channel–based image representation which extends the SP representation based on a single type of descriptor[5] (i.e., SIFT) to the case of multiple types of descriptors (see Fig. 1).

Let $\mathbf{x}^m$ denote the feature of channel m for an image I, so that $\mathbf{x}^m_{lc}$ is the local feature of the cell c at level l of the SP. Note that the first level of the SP consists of only one cell (i.e., the whole image itself). In each subsequent level each cell is split into four nonoverlapping subcells. The process is repeated up to level L. In this paper, $\mathbf{x}^m_{lc}$ is computed by the bag-of-words model which involves three steps:

- Local descriptors (either at interest points, or, densely or randomly sampled) are extracted.
- A subset of local descriptors extracted from the training images are clustered by K-means to generate a visual vocabulary, and all the descriptors are subsequently quantized into visual words according to nearest neighbor rule.
- Each cell is encoded as a histogram of visual words.

For L levels and V words, $\mathbf{x}^m$ has dimensionality $V\sum_{l=1}^{L} 4^{l-1}$. In order to obtain a generic engine with an acceptable level of complexity, we use pixel patch itself,[12] Self-SIMilarity[19] (SSIM), CENTRIST,[17] Census Transform, SIFT,[15] and oriented gradient (OG) being base descriptors in the bag-of-words model, resulting in six feature channels: SP-PATCH, SP-SSIM, SP-CENTRIST, SP-CT, SP-SIFT, and SP-OG, respectively. Since the only difference among multiple feature channels resides in what type of base descriptor in the bag-of-words model is extracted, the multiple-feature-channel–based representation has high flexibility such that adding more feature channels might further improve the performance with a reasonably additional cost.

3 Classification With a Single Feature Channel

Let $\mathbf{x}_I^m$ and $\mathbf{x}_J^m$ denote the features of channel m found in the images I and J respectively, their similarity can be measured by histogram intersection distance (HID)

$$d_{hi}\left(\mathbf{x}_I^m, \mathbf{x}_J^m\right) = \sum_{i=1}^{D} \min\left[\mathbf{x}_I^m(i), \mathbf{x}_J^m(i)\right] \tag{1}$$

with D the dimension of the feature vectors. HID is well suited to measure the similarity between the histogram-wise features (e.g., SIFT, which is a 3D histogram of gradient location and orientation).

Another popular similarity measure is χ^2 distance

$$d_{\chi^2}\left(\mathbf{x}_I^m, \mathbf{x}_J^m\right) = \frac{1}{2}\sum_{i=1}^{D} \frac{\left[\mathbf{x}_I^m(i) - \mathbf{x}_J^m(i)\right]^2}{\mathbf{x}_I^m(i) + \mathbf{x}_J^m(i)}. \tag{2}$$

Note here that we assume $\frac{[\mathbf{x}_I^m(i)-\mathbf{x}_J^m(i)]^2}{\mathbf{x}_I^m(i)+\mathbf{x}_J^m(i)} = 0$ if both $\mathbf{x}_I^m(i)$ and $\mathbf{x}_I^m(i)$ are zeros.

SVM can then be used for classifying the images. In a two-class case, the decision function of SVM for a test image with feature vector $\mathbf{x}^m$ has the following form:[20]

$$g(\mathbf{x}^m) = \sum_{n=1}^{N} \alpha_n^m y_n \mathbf{K}\left(\mathbf{x}^m, \mathbf{x}_n^m\right) + b^m, \tag{3}$$

where $\mathbf{K}(\mathbf{x}^m, \mathbf{x}_n^m)$ is the value of a kernel function for the n'th training image and the test image, y_n the class label of $\mathbf{x}_n^m$ ($+1$ or -1), α_n^m the learned weight of the n'th training image, and b^m the learned threshold parameter. To incorporate HID into the SVM framework, we simply set $\mathbf{K}_{hi}(\mathbf{x}_I^m, \mathbf{x}_J^m) = d_{hi}(\mathbf{x}_I^m, \mathbf{x}_J^m)$. For the case of χ^2 distance, the kernel function is defined as follows:

$$\mathbf{K}_{\chi 2}\left(\mathbf{x}_I^m, \mathbf{x}_J^m\right) = \exp\left(-\frac{1}{\mu} d_{\chi^2}(\mathbf{x}_I^m, \mathbf{x}_J^m)\right), \tag{4}$$

where μ is set to the average χ^2 distance between all the training images.

We employ the LIBSVM package[21] to train the two-class classifier. One advantage of LIBSVM is that it is able to provide probabilistic output for the test image.[21] Multiclass classification is done using the one-versus-the-rest scheme. For $K > 2$ classes, we learn K binary SVMs where the k'th decision function $g_k(\mathbf{x}^m)$ is trained using the data from class $\mathcal{C}_k$ as the positive examples and the data from the remaining $K - 1$ classes as the negative examples. During the training phase, the weight of positive class is set to $w_p = (n_p + n_n)/n_p$, and that of negative class is set to $w_n = (n_p + n_n)/n_n$, where n_p and n_n are the numbers of positive and negative samples, respectively. The test image is assigned to class $\mathcal{C}_i$ if $g_i(\mathbf{x}^m) > g_j(\mathbf{x}^m)$ for all $j \neq i$.

4 Multiple Feature Channel Combination

For M feature channels and K classes, we learn $M \times K$ binary SVMs $\{g_{mk}(\mathbf{x}^m)\}$ as described in Sec. 3, where $m = 1, \ldots, M$ and $k = 1, \ldots, K$. Given a test image X with the multiple-feature-channel–based representation $\{\mathbf{x}^m\}$, we define a $M \times K$ matrix $\mathbf{T} = (T_{mk})$ with $T_{mk} = g_{mk}(\mathbf{x}^m)$ denoting the probability that X belongs to class $\mathcal{C}_k$ when only using the feature of channel m. The final decision function is of the following form:

$$y = f(\mathbf{T}) = f_2[f_1(\mathbf{T})] = f_2(\mathbf{y}), \tag{5}$$

where $\mathbf{y} = f_1(\mathbf{T})$ and $y = f_2(\mathbf{y}) = \arg\max_{k \in \{1,\ldots,K\}} \mathbf{y}(k)$. Four formulations of $f_1(\mathbf{T})$ have been tried. The overall strategies are termed max-max, sum-max, prod-max, and classifier-max.

1. max-max. The probability of the test image X belonging to $\mathcal{C}_k$ is decided by the most discrimination feature channel, i.e., $\mathbf{y}(k) = \max_{j \in \{1,\ldots,M\}} T_{jk}$.
2. sum-max. In this case, the probability of the test image belonging to $\mathcal{C}_k$ is given by the arithmetic average of the probabilities obtained with every feature channel, i.e. $\mathbf{y}(k) = 1/M \sum_{m=1}^{M} T_{mk}$.
3. prod-max. The probability of the test image belonging to $\mathcal{C}_k$ is given by the geometric average, i.e. $\mathbf{y}(k) = (\prod_{m=1}^{M} T_{mk})^{1/M}$. For practical implementation, $\mathbf{y}(k) = \prod_{m=1}^{M} T_{mk}$ is used for efficiency.
4. classifier-max. Consider a $M \times K$ probabilistic response matrix $\mathbf{T}^n$, whose element T_{mk}^n is given by the previous learned classifier $g_{mk}(\mathbf{x}^m)$ evaluated at a training image X_n. Then we rearrange the rows of $\mathbf{T}^n$ into an $M \times K$-dimensional row vector $\mathbf{t}_n$, such that $\mathbf{t}_n = (T_{11}^n, \ldots, T_{1K}^n, \ldots, T_{M1}^n, \ldots, T_{MK}^n)$. For a set of training images $\{X_n\}$ where $n = 1, \ldots, N$, we obtain a matrix $\mathbb{T} = (\mathbf{t}_1^T, \ldots, \mathbf{t}_N^T)^T$. Depending on $\mathbb{T}$ together with the corresponding class labels $\{y_n\}$, additional K binary SVMs $\{g_k'(\mathbf{t})\}$ are learned using the one-versus-the-rest scheme. In the testing phase, for a test image X with the multiple-feature-channel–based representation $\{\mathbf{x}^m\}$, the classifiers $\{g_{mk}(\mathbf{x}^m)\}$ first take $\{\mathbf{x}^m\}$ as input and generate a vector $\mathbf{t}$. Then all classifiers $\{g_k'(\mathbf{t})\}$ are evaluated, and the test image X is finally assigned to class $\mathcal{C}_i$ if $g_i'(\mathbf{t}) > g_j'(\mathbf{t})$ for all $j \neq i$.

4.1 *Computational Complexity*

MKL (Ref. 18) and LP-β (Ref. 6) are representative methods on feature combination. Table 1 summarizes the complexity of our proposed method, MKL and LP-β.

Let N be the number of training images, M the number of kernels, and K the number of categories. The time for training a binary SVM is dominated by the time for solving the underlying quadratic programming (QP), and so the computational complexity is about $O(N^\gamma)$, where N is the number of training samples and γ is a constant.[22] γ varies depending on the method used to solve the underlying QP. In the worst case, $\gamma = 3$. Empirically, $1 < \gamma < 2$.[22] In the proposed method, multiclass classification is done using the one-versus-the-rest scheme, so the complexity of our proposed method is $O(\lambda K M N^\gamma)$, where $\lambda = 1 + 1/M$ if classifier-max is used and $\lambda = 1$ otherwise.

In each iteration, the MKL algorithm in Ref. 18 proceeds in two stages. The first stage would simply consist of optimizing a standard SVM with a kernel $\mathbf{K} = \sum_k d_k \mathbf{K}_k$ while for the second stage, the gradient of object function

Table 1 The complexity of various methods. K, M and N are the number of categories, kernels, and training images, respectively. $\lambda = 1 + 1/M$ if classifier-max is used and 1 otherwise. I is the number of iterations for MKL, while N_S is the average number of support vectors at each iteration. γ is a constant. Usually, $1 < \gamma < 2$.

Methods	Ours	MKL (Ref. 18)	LP-β (Ref. 6)
Complexity	$O(\lambda KMN^{\gamma})$	$O\left[KI\left(N^{\gamma} + MN_S^3\right)\right]$	$O[KMN^{\gamma} + (M + N + 1)^2 N]$

according to d_k is needed. The two stages are repeated until convergence or a maximum number of iterations is reached. The gradient computation has a complexity of the order of MN_S^3, where N_S $(0 \le N_S \le N)$ is the average number of support vectors at each iteration.[23] Therefore, the MKL algorithm has a complexity of $O[KI(N^{\gamma} + MN_S^3)]$, where I is the number of iteration. If $I \gg M$ (this is usually the case), MKL has a much higher complexity than the proposed method, especially for the case of $N_S \approx N$.

The LP-β algorithm in Ref. 6 learns all parameters of the model in two separate steps. The parameters for all classes and features are learned in the first step. The mixing coefficients d_k are then learned by the multiclass extension of LPboost, which is a linear program. The complexity of solving a linear program using an interior-point method is of order D^2N, where D is the number of variables.[24] Therefore, LP-β has a complexity of $O[KMN^{\gamma} + (M + N + 1)^2 N]$. It is obvious that LP-$\beta$ has a much higher complexity than our proposed method also.

5 Experiments and Results

5.1 *Datasets and Setup*

We apply the proposed method to three scene datasets [Scene-8 (Ref. 25), Scene-13 (Ref. 4), and Scene-15 (Ref. 5)] and two object datasets [Caltech-6 (Ref. 8) and Caltech-101 (Ref. 14)]. All experiments are performed in gray scale. We repeat the experiments on each dataset over 5 random train/test splits. Performance for all datasets is measured as the average classification accuracy per class.

To evaluate classification performance, we stick to the methodologies defined by the designers of the corresponding datasets. Specifically, for Scene-8, Scene-13, Scene-15, and Caltech-6 datasets 100 images per category are randomly selected for training and the remainder for testing in each run. For the Caltech-101 datasets, 30 images per category is used for training from all 102 classes (i.e., including the background) and up to 50 images per category for testing for efficiency.

5.2 *Implementation Detail*

For categorization of the Caltech-101 dataset, we populate the training set of positive images by synthetically generating additional training images since the number of training images per category is too limited to estimate *a posteriori*. Specifically, we generate 4 virtual images for each original training image by zooming it by a factor of 1.1, 1.2, or by rotating it 5°, –5°, respectively, resulting in a total of 150 images per category for training.

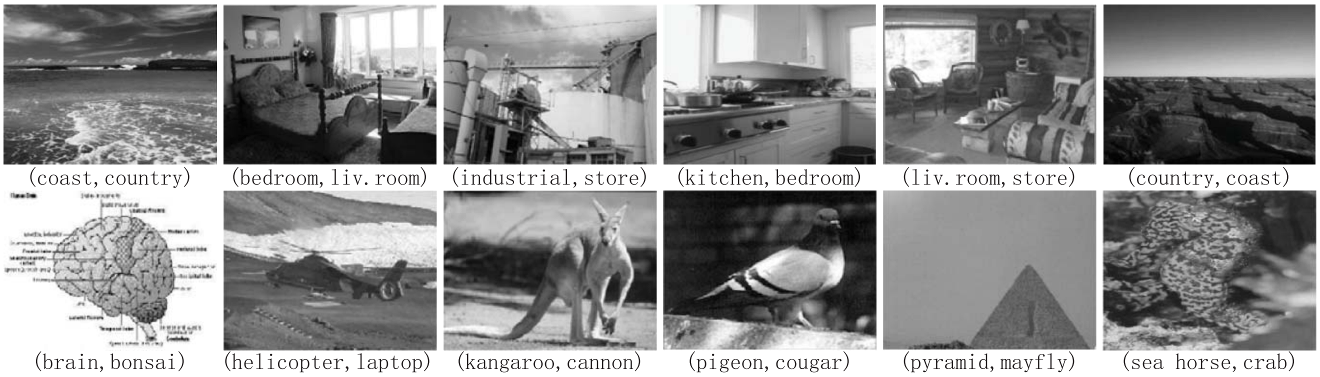
For SP-PATCH, SP-SSIM, SP-CENTRIST, and SP-SIFT, local descriptors are computed at points on a regular grid with a spacing of 8 pixels for all the datasets except the Caltech-101 dataset, where the spacing is set to 4 pixels. A smaller step size means that more local descriptors are extracted. At

Table 2 Evaluation of different feature channels using HIK and χ^2 kernel.

Datasets	SP-PATCH	SP-SSIM	SP-CENTRIST	SP-CT	SP-OG	SP-SIFT
HIK						
Scene-8	82.9 ± 0.4	87.1 ± 0.7	84.4 ± 0.6	83.8 ± 1.1	82.6 ± 0.5	$\mathbf{87.5} \pm 0.2$
Scene-13	79.4 ± 0.4	81.5 ± 0.6	82.4 ± 0.3	82.8 ± 0.5	75.5 ± 0.4	$\mathbf{84.6} \pm 0.5$
Scene-15	76.1 ± 0.5	77.7 ± 0.2	79.9 ± 0.5	80.0 ± 0.4	69.1 ± 0.9	$\mathbf{80.1} \pm 0.2$
Caltech-6	$\mathbf{99.7} \pm 0.1$	99.3 ± 0.1	99.4 ± 0.4	99.4 ± 0.3	98.8 ± 0.1	$\mathbf{99.7} \pm 0.1$
Caltech-101	56.8 ± 0.3	$\mathbf{66.6} \pm 1.5$	56.3 ± 1.0	56.9 ± 1.0	57.7 ± 0.8	65.3 ± 1.3
χ^2 kernel						
Scene-8	83.5 ± 0.7	$\mathbf{87.6} \pm 0.6$	84.5 ± 0.6	82.8 ± 0.9	83.2 ± 0.4	$\mathbf{87.6} \pm 0.2$
Scene-13	79.6 ± 0.7	81.6 ± 0.7	82.9 ± 0.3	81.7 ± 0.5	76.5 ± 0.5	$\mathbf{84.5} \pm 0.6$
Scene-15	76.7 ± 0.8	77.2 ± 0.4	80.1 ± 0.5	79.8 ± 0.8	70.1 ± 0.2	$\mathbf{81.0} \pm 0.6$
Caltech-6	99.7 ± 0.1	99.5 ± 0.1	99.4 ± 0.4	99.3 ± 0.3	98.8 ± 0.4	$\mathbf{99.8} \pm 0.1$
Caltech-101	58.1 ± 0.6	$\mathbf{67.6} \pm 1.0$	57.3 ± 1.0	55.9 ± 0.5	59.3 ± 0.6	64.7 ± 1.2

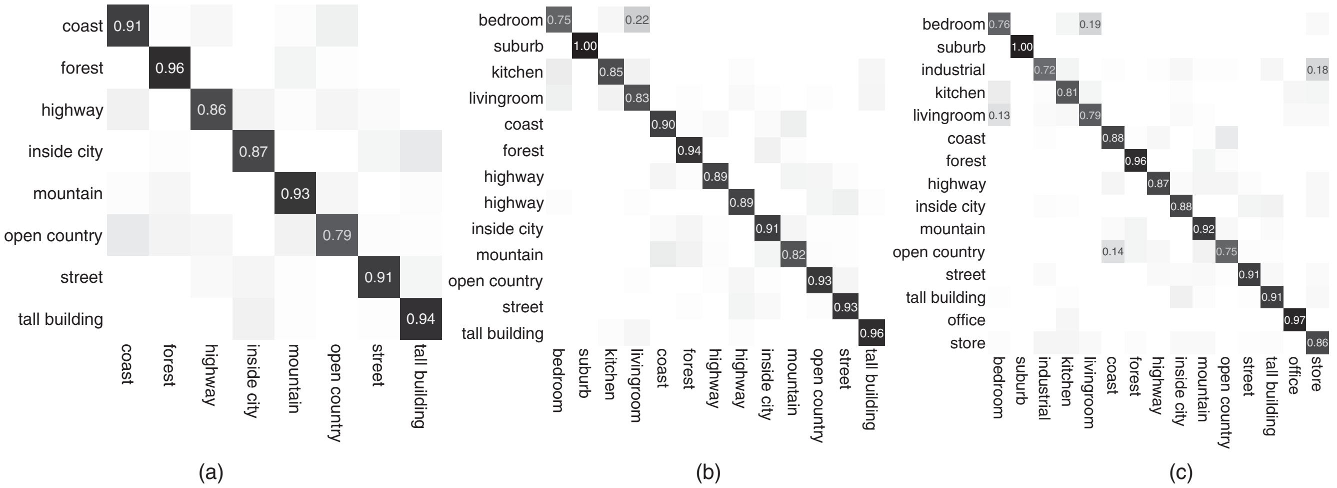
Table 3 Evaluation of different information combination methods using HIK and χ^2 kernel. Note that the first column lists the best results with a single feature channel.

Datasets	Single	max-max	sum-max	prod-max	classifier-max
HIK					
Scene-8	87.5 ± 0.2	88.3 ± 0.3	89.0 ± 0.2	$\mathbf{89.4} \pm 0.3$	88.8 ± 0.2
Scene-13	84.6 ± 0.5	86.4 ± 0.2	88.7 ± 0.2	$\mathbf{89.0} \pm 0.2$	88.3 ± 0.3
Scene-15	80.1 ± 0.2	83.5 ± 0.6	86.2 ± 0.5	$\mathbf{86.6} \pm 0.1$	85.8 ± 0.4
Caltech-6	99.7 ± 0.1	99.8 ± 0.1	99.8 ± 0.1	99.8 ± 0.1	99.8 ± 0.1
Caltech-101	66.6 ± 1.5	70.8 ± 0.9	73.8 ± 0.3	$\mathbf{75.0} \pm 0.8$	73.9 ± 0.2
χ^2 kernel					
Scene-8	87.6 ± 0.2	88.0 ± 0.3	89.1 ± 0.2	$\mathbf{89.2} \pm 0.4$	88.8 ± 0.4
Scene-13	84.5 ± 0.6	86.4 ± 0.6	88.1 ± 0.4	$\mathbf{88.7} \pm 0.4$	88.0 ± 0.4
Scene-15	81.0 ± 0.6	83.5 ± 0.1	86.0 ± 0.2	$\mathbf{86.5} \pm 0.1$	85.6 ± 0.5
Caltech-6	99.8 ± 0.1	99.8 ± 0.1	99.8 ± 0.1	99.8 ± 0.1	99.8 ± 0.1
Caltech-101	67.6 ± 1.0	70.7 ± 0.8	73.9 ± 1.0	$\mathbf{74.5} \pm 1.2$	73.8 ± 1.1


(coast, country) (bedroom, liv. room) (industrial, store) (kitchen, bedroom) (liv. room, store) (country, coast)

(brain, bonsai) (helicopter, laptop) (kangaroo, cannon) (pigeon, cougar) (pyramid, mayfly) (sea horse, crab)

Fig. 2 Image examples of Scene-15 (top) and Caltech-101 (bottom). The images are correctly classified by our proposed method. (`class1`, `class2`) denotes that `class1` image is misclassified as a `class2` image when a single feature channel (SP-SIFT) is used.



(a) (b) (c)

Fig. 3 Confusion matrices demonstrating accuracies on (a) Scene-8, (b) Scene-13, and (c) Scene-15. Only accuracies higher than 0.1 are shown.

Table 4 Average classification accuracies on Scene-8.

Methods	Reference 25	Reference 12	Ours
Accuracy	83.7	87.8	**89.4**

Table 5 Average classification accuracies on Scene-13.

Methods	Reference 4	Reference 13	Reference 26	Reference 12	Ours
Accuracy	65.2	66.5	75.0	85.9	**89.0**

each grid point, the descriptor is computed over a 16×16 pixel patch. The visual vocabulary size V can be obtained through cross validation for selecting the optimal value. In our experiments V is empirically set to 200. This value is selected based on the comparative evaluation of Lazebnik et al..[5] For SP-CT, CTs are computed at every point and we use PCA to reduce the dimension of the bag-of-words representation of each cell to 40. In this case we need not construct the vocabulary because the CT value is a base-10 number in [0, 255] (please refer to Ref. 17). For SP-OG, oriented gradients are first computed at every point on the output of a Canny edge detector and then quantized to 36 bins. Similar to SP-CT, we also need not construct the vocabulary for SP-OG. For all the feature channels, spatial pyramids of depth $L = 3$ are used except for SP-OG where $L = 4$.

5.3 *Main Results*

5.3.1 *Evaluation of different feature channels*

Table 2 shows the results when HIK is used for recognition. As seen, SP-SIFT performs best on scene datasets while SP-SSIM outperforms SP-SIFT on Caltech-101. All the feature channels work very well on Caltech-6 even though SP-SIFT and SP-SSIM obtain the best results. If χ^2 kernel is used for recognition, we find from Table 2 that SP-SIFT performs best on Scene-13 and Scene-15 and obtains the same result on Scene-8 as SP-SSIM. Similar to the case of HIK, SP-SSIM gives the best result on Caltech-101. To summarize, when only a single feature channel is used, SP-SIFT is the best choice for scene categorization, and SP-SSIM is the best one for object categorization.

5.3.2 *Evaluation of different type of kernels*

The power of a kernel classifier such as SVM depends heavily on the type of kernel used. Here we compare two types of kernels: HIK and χ^2 kernel. To the best of our knowledge, there exist no literatures which compare the HIK and χ^2 kernel under the same conditions. Table 2 summarizes the results of the HIK and χ^2 kernel. We find that the performance rank between the HIK and χ^2 kernel depends on the feature channels. For example, HIKs obtain better results than χ^2 kernels on all the datasets for PACT, whereas χ^2 kernels give slightly better, or equivalent, results than HIKs for the other feature channels except for SP-SIFT on Caltech-101. However, the difference in performance between the HIK and χ^2 kernel in most cases is insignificant. Either of the kernels seems to be a good choice for image classification tasks. In the following, we only report the HIK results unless stated otherwise.

5.3.3 *Evaluation of different information combination strategies*

Table 3 shows results of max-max, sum-max, prod-max, and classifier-max. For comparison, we also list the best results using a single feature channel. As expected, using the features together improves the performance significantly. We observe that in most cases prod-max performs best while max-max performs the worst. Sum-max and classifier-max can achieve comparable accuracies. For instance, prod-max obtains the accuracy of 75.0% on Caltech-101, whereas max-max obtains the accuracy of 70.8% which is still higher than the best single feature channel accuracy of 66.6%. The images in Fig. 2 are misclassified when using a single feature channel, but they are correctly classified by our proposed method with prod-max. In short, we suggest that: 1. as much information cues as possible should be incorporated if the appropriate feature is unknown in advance for a specific task, and 2. prod-max should be the preferable information combination choice in the context of image classification. In the following experiments, we adopt prod-max to perform feature combination.

Figure 3 shows the resulting confusion matrices on Scene-8, Scene-13, and Scene-15 from one run of using our proposed method. For Scene-8, we observe that: a. there exists some confusion between coast and open country categories. This can be explained by the fact that some images of both categories share similar image components such as sky, waters, or trees. b. The confusion matrix also shows that inside city images are sometimes misclassified as a tall building. This can be explained by the fact that inside city images often contain patterns such as windows and vertical lines which are characteristic of tall building images. For Scene-13, it is not surprising that more heavy confusion occurs among the indoor classes (kitchen, bedroom, and living room), due to similar image configurations and similar components (such as windows, doors). Analyzing the confusion matrix from Scene-15, we observe that some mistakes are made between the industrial and store categories besides those observed from Scene-13.

5.4 *Comparison With State-of-the-Art Single Cue Methods*

Table 4 compares the proposed method with state-of-the-art single cue methods on Scene-8. Our proposed method achieves the accuracy of 89.4% which is much higher than those reported by Oliva et al.[25] and Bosch et al.[12]

Table 6 Average classification accuracies on Scene-15.

Methods	Reference 5	Reference 27	Reference 12	Reference 17	Reference 16	Ours
Accuracy	81.4	83.3	83.7	83.3	84.1	**86.6**

Table 7 ROC equal error rates on Caltech-6.

Methods	Reference 8	Reference 28	Reference 29	Reference 11	Ours
Airplanes	90.2	98.3	98.6	98.8	**100**
Cars (rear)	90.3	N/A	N/A	98.3	**99.7**
Cars (side)	88.5	N/A	N/A	95.0	**96.7**
Faces	96.4	99.7	96.3	100	**100**
Motorbikes	92.5	99.0	98.9	98.5	**99.7**
Leopards	90.0	N/A	N/A	97.0	**99.3**

Table 5 summarizes the results on Scene-13. Our proposed method achieves an accuracy of 89.0%. This exceeds the highest accuracy previously published, that of 85.9% reported by Bosch et al.[12] The variant of latent dirichlet allocation (LDA) (Ref. 4) obtained an accuracy of 65.2%. Classification accuracy for the bag-of-visterms representation[13] was 66.5%.

On Scene-15, the best single feature channel (SP-SIFT) yields an accuracy of 80.1%. We record an accuracy of 86.6%. Table 6 also lists the results of the state-of-the-art single cue methods. The spatial pyramid match method[5] achieved 81.4%. Wu and Rehg[17] and Liu et al.[27] both reported an 83.3% accuracy. Recently, Wu and Rehg have achieved an accuracy of 84.1%.[16] In Ref. 16, they create the visual vocabularies using the HID instead of Euclidean distance.

Besides the average classification accuracies (see Tables 2 and 3), we also report the receiver operating characteristic (ROC) equal error rates for two-class classification (object versus background) on Caltech-6. Table 7 compares our method with four state-of-the-art methods: Fergus et al.,[8] Deselaers et al.,[28] Zhang et al.,[29] and Zhang et al..[11] Our method performs best on all six object classes. The results obtained by the other methods are also quite high, indicating the relatively low level of difficulty of Caltech-6.

Table 8 shows the results on Caltech-101. Mutch and Lowe[30] obtained an accuracy of 56% based on sparse image representation. The spatial pyramid matching method achieved an accuracy of 64.6%.[5] Zhang et al.[31] yielded a classification accuracy of 66.2% using SVM-KNN, which is a hybrid of SVM and the K nearest neighbor classifier. We obtain the accuracy of 75.0% using multiple cues together.

5.5 *Comparison With State-of-the-Art Multiple Cue Methods*

In this section, we compare our proposed method with MKL and LP-β using our features. The classification accuracies are summarized in Table 9 (we do not report results for Caltech-6 as the performance is already at ceiling). In all experiments, we associate a HIK with the feature from each channel giving a total of 6 kernels in all.* As can be seen, the proposed method consistently performs better than MKL while producing slightly inferior accuracies to LP-β. However, compared with LP-β, our proposed method has two advantages. First, it is more efficient. Table 1 shows that the proposed method has a much lower complexity than both MKL and LP-β. Empirically, our proposed method is an order of magnitude faster than both MKL and LP-β, whereas MKL and LP-β have comparable runtimes. Second, it is easy to implement and is potential for parallel computation due to the distributed fusion architecture as mentioned in Sec. 1.

Table 9 Comparison with state-of-the-art multiple cue methods using the same types of features. See text for detail.

Dataset	Ours	MKL	LP-β
Scene-8	89.4 ± 0.3	89.0 ± 0.2	89.5 ± 0.3
Scene-13	89.0 ± 0.2	88.7 ± 0.3	89.2 ± 0.4
Scene-15	86.6 ± 0.1	86.1 ± 0.4	86.7 ± 0.4
Caltech-101	75.0 ± 0.8	74.2 ± 0.9	75.3 ± 0.7

6 Conclusion

In this paper, we proposed a new framework for image classification. Our proposed framework has certain attractive properties. First, it can be used to combine heterogeneous sources of data effectively and outperforms all other single cue methods. Second, the proposed method is very suitable for practical applications due to its efficiency and easy implementation. Third, it appears to be general and capable of handling diverse recognition problems including scene categorization and object categorization.

Table 8 Average classification accuracies on Caltech-101.

Methods	Reference 32	Reference 33	Reference 11	Reference 30	Reference 31	Reference 5	Ours
Accuracy	43.0	49.0	53.9	56.0	66.2	64.6	**75.0**

---

*Note that an even better performance may be obtained by adding more kernels (features) for all multiple cue methods. For example, LP-β achieved an accuracy of 77.7% on Caltech-101 using 39 kernels (Ref. 6).

Some researchers have shown that most existing algorithms for image classification perform poorly at the indoor recognition task. In the future, we will further test the scalability of the proposed method on the datasets covering a wide range of domains, especially indoor scene categories.

Acknowledgments

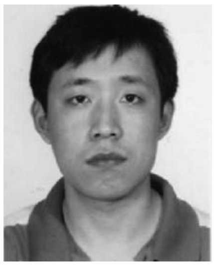
We would like to thank the anonymous reviewers and the editor for their thoughtful and constructive comments that helped improve the quality of the paper. We are very grateful to the following for providing some source code: V. Gulshan, J. X. Wu, S. Lazebnik, P. Gehler, A. Vedaldi, and M. Varma. This work was supported by 973 Program (2010CB731401, 2010CB731406), NSFC (60932006, 60828001, and 61001146), SRFDP (20090073110022), and the 111 Project (B07022).

References

1. J. Gao, N. Sang, C. Gao, Q. Tang, and J. Sang, "Visual codebook construction for class-specific recognition," *Opt. Eng.* **48**(7), 077201 (2009).
2. Y. Jiang, J. Chen, and R. Wang, "Fusing local and global information for scene classification," *Opt. Eng.* **49**(10), 047001 (2010).
3. J. Qin and N. H. C. Yung, "Scene categorization with multiscale category-specific visual words," *Opt. Eng.* **48**(4), 047203 (2009).
4. L. Fei-Fei and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *IEEE Conference on Computer Vision and Pattern Recognition* (2005).
5. S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bag of features: Spatial pyramid matching for recognizing natural scene categories," in *IEEE Conference on Computer Vision and Pattern Recognition* (2006).
6. P. Gehler and S. Nowozin, "On feature combination for multiclass object classification," in *IEEE International Conference on Computer Vision* (2009).
7. W. Lin, M.-T. Sun, R. Poovendran, and Z. Zhang, "Activity recognition using a combination of category components and local models for video surveillance," *IEEE Transactions on Circuits and Systems for Video Technology* **18**(8), 1128–1139 (2008).
8. R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *IEEE Conference on Computer Vision and Pattern Recognition* (2003).
9. S. Agarwal, A. Awan, and D. R. Roth, "Learning to detect objects in images via a sparse, part-based representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **26**(11), 1475–1490 (2004).
10. P. F. Felzenszwalb, R. B. Girshichk, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **32**(9), 1627–1645 (2010).
11. J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, "Local feature and kernels for classification of texture and object categories: a comprehensive study," *International Journal of Computer Vision* **73**(2), 213–238 (2007).
12. A. Bosch, A. Zisserman, and X. Munoz, "Scene classification using a hybrid generative/discriminative approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **30**(4), 712–727 (2008).
13. P. Quelhas, F. Monay, J.-M. Odobez, D. G. Perez, and T. Tyytelaars, "A thousand words in a scene," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **29**(9), 1575–1589 (2007).
14. L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories," in *IEEE Conference on Computer Vision and Pattern Recognition Workshop on Generative-Model Based Vision* (2004).
15. D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision* **60**(2), 91–110 (2004).
16. J. Wu and J. M. Rehg, "Beyond the Euclidean distance: Creating effective visual codebooks using the histogram intersection kernel," in *IEEE International Conference on Computer Vision* (2009).
17. J. Wu and J. M. Rehg, "CENTRIST: A visual descriptor for scene categorization," IEEE Transactions on Pattern Analysis and Machine Intelligence (accepted).
18. M. Varma and D. Ray, "Learning the discriminative power-invariance trade-off," *IEEE International Conference on Computer Vision* (2007).
19. E. Shechtman and M. Irani, "Matching local self-similarities across images and videos," in *IEEE Conference on Computer Vision and Pattern Recognition* (2007).
20. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Min. Knowl. Discov.* **2**(2), 121–167 (1998).
21. C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," *software available at* http://www.csie.ntu.edu.tw/~cjlin/libsvm (2001).
22. C. D. Manning, P. Raghavan and H. Schutze, *An Introduction to Information Retrieval*, Cambridge University Press, Cambridge, England (2009).
23. A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet, "More efficiency in multiple kernel learning," *International Conference on Machine Learning* (2007).
24. S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, England (2009).
25. A. Oliva and A. Torralaba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision* **42**(3), 145–175 (2001).
26. F. Lu, X. Yang, R. Zhang, and S. Yu, "Image classification based on pyramid histogram of topics," *IEEE International Conference on Multimedia Expo* (2009).
27. J. Liu and M. Shah, "Scene modeling using co-clustering," *IEEE International Conference on Computer Vision* (2007).
28. T. Deselaers, D. Keysers, and H. Ney, "Discriminative training for object recognition using image patches," *IEEE Conference on Computer Vision and Pattern Recognition* (2005).
29. W. Zhang, B. Yu, G. J. Zelinsky, and D. Samaras, "Object class recognition using multiple layer boosting with heterogeneous features," *IEEE Conference on Computer Vision and Pattern Recognition* (2005).
30. J. Mutch and D. Lowe, "Multiclass object recognition with sparse, localized features," *IEEE Conference on Computer Vision and Pattern Recognition* (2006).
31. H. Zhang, A. C. Berg, and J. Malik, "SVM-KNN: discriminative nearest neighbor classification for visual category recognition," *IEEE Conference on Computer Vision and Pattern Recognition* (2006).
32. K. Grauman and T. Darrell, "Pyramid match kernels: Discriminative classification with sets of image features," in *IEEE International Conference on Computer Vision* (2005).
33. J. Amores, N. Sebe, and P. Radeva, "Context-based object-class recognition and retrieval by generalized correlograms," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **29**(10), 1818–1833 (2007).

Fuxiang Lu is currently pursuing a PhD degree at the Institute of Image Communication and Information Processing, Shanghai Jiao Tong University, Shanghai, China. His current research interests include image processing, computer vision, and pattern recognition.

Xiaokang Yang is currently a professor with the Institute of Image Communication and Information Processing, Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China. From April 2002 to October 2004, he was a research scientist in the Institute for Infocomm Research, Singapore. His current research interests include scalable video coding, video transmission over networks, video quality assessment, digital television, and pattern recognition.



Weiyao Lin received a PhD degree from the University of Washington, Seattle, Washington, in 2010, in electrical engineering. Since 2010, he has been an assistant professor at the Institute of Image Communication and Information Processing, Department of Electronic Engineering, Shanghai Jiao Tong University. His research interests include video processing, machine learning, computer vision and video coding and compression.

Rui Zhang received a PhD degree from Shanghai Jiao Tong University, Shanghai, China, in 2008. She is currently an associate professor with the Institute of Image Communication and Information Processing, Shanghai Jiao Tong University. Her research interests include image communication, image processing, and DTV.

Songyu Yu is currently a professor and vice director of the Institute of Image Communication and Information Processing, Shanghai Jiao Tong University, Shanghai, China. His current research interests include image processing, digital television, and image communication.