Image Classification with Densely Sampled Image Windows and Generalized Adaptive Multiple Kernel Learning

Shengye Yan, XinXing Xu, Dong Xu, Senior Member, IEEE, Stephen Lin, Member, IEEE, and Xuelong Li, Fellow, IEEE

Abstract—We present a framework for image classification that extends beyond the window sampling of fixed spatial pyramids and is supported by a new learning algorithm. Based on the observation that fixed spatial pyramids sample a rather limited subset of the possible image windows, we propose a method that accounts for a comprehensive set of densely sampled windows. This higher level of abstraction removes the need for handling the dense samples and reduced sensitivity to misalignment. In addition to dense window sampling, we introduce generalized adaptive ℓp-norm multiple kernel learning (GA-MKL) to learn a robust classifier based on multiple base kernels constructed from the new image features and multiple sets of prelearned classifiers from other classes. With GA-MKL, multiple levels of image features are effectively fused, and information is shared among different classifiers. Extensive evaluation on benchmark datasets for object recognition and scene categorization demonstrates the proposed method outperforms the state-of-the-art under a wide range of settings.

Index Terms—Adapted classifier, image classification, multiple kernel learning, spatial pyramid.

I. INTRODUCTION

The growing popularity of digital photography has led to a tremendous number of images collected in personal photo albums and online image repositories. To help in managing such large image sets, automatic techniques are highly desirable. One basic task for automatic image database management is to categorize the photos into specified classes, a problem commonly referred to as image categorization. Image categorization is an extremely challenging computer vision task because of the large appearance variation within a class and the cluttered backgrounds in many images. The significance of this application and the challenges that it poses has made image categorization a hot research topic.

Many works have been proposed for image categorization [1]–[11]. One well-established approach among them is spatial pyramid matching (SPM) [4]. It subdivides an image into increasingly finer regions according to a spatial pyramid representation (SPR), then models each of the subregions using a bag of features (BoF) [1], and finally concatenates all the BoFs for classification. In the past several years, a large number of feature extraction techniques have been designed to extend this feature extraction framework [6], [8]–[10], [12].

In spite of its great popularity in image classification, SPR accounts for a rather limited set of spatial regions where features are defined: the regions are with a fixed aspect ratio; their locations must align with a grid; and the number of the regions varies only by multiples of four. Many possible spatial regions, which may carry important discriminative information, are excluded.

In this paper, we seek a general framework for image classification that effectively accounts for a comprehensive set of densely sampled windows with respect to location, size, and aspect ratio, such as sliding windows in object detection [13] and object tracking [14], while allowing existing methods for encoding and pooling to be incorporated. However, two important issues need to be considered. One is that the feature vector would become extremely large, since it is formed by concatenating the features from all the windows. Such lengthy feature vectors would make image classification computationally inefficient. The other issue is that different images are often not aligned with each other in the image classification task. Feature vectors with a strong spatial structure can therefore be detrimental when corresponding features do not coincide in image position.

To address these issues, we propose a simple but effective image feature derived from densely sampled windows that is relatively compact and less sensitive to misalignment. It is obtained via a two-level feature extraction method in which the first level computes window-based features from local descriptors such
as scale-invariant feature transform (SIFT) [15]. For extracting the second level features, the encoding and pooling procedure is applied over multiple groups of the window-based features partitioned by different window sizes. Feature pooling over the image on one hand yields a feature vector with the same number of elements as the codebook and on the other hand leaves out the exact positional information of the window-based features. Moreover, special encoding and pooling over different sizes of windows in the second level leads to a concise feature representation that accounts for window size information, which is important for feature extraction because windows with different sizes are likely to contain different image content.

For classification, motivated by the recent success of MKL methods for various vision applications such as object categorization [16], [17], video concept detection [18] and action recognition [19], a new learning method called generalized adaptive $\ell_p$-norm multiple kernel learning (GA-MKL) is proposed. GA-MKL allows for different features such as our different sets of second level features and the standard first level feature to be effectively fused for classification. Moreover, GA-MKL takes advantage of prelearned classifiers of other classes, based on the intuition that some classes may share common information that can benefit each other. For example, classes like Swan, Duck, and Goose may share the same context of “Water” and similar components like beaks. This kind of shared common information of “same context” and “similar components” may not be learned correctly by the traditional classification method because of the limited number of positive training images within one class. Adapting prelearned classifiers from other classes provides an effective way to leverage the training images for different categories. GA-MKL takes advantage of this by learning an adapted classifier using multiple sets of base kernels from one class and multiple sets of prelearned SVM classifiers from other classes.

Through extensive experiments conducted on three widely-used benchmarks—Caltech256 [20], Caltech101 [21], and 15Scenes [4]—we demonstrate the effectiveness of our feature extraction framework based on the second level features and prelearned classifiers from other classes incorporated through GA-MKL. These results show that our work outperforms the state-of-the-art over a broad range of test cases.

This paper presents an extension of our previous work [22]. The main extension is a new feature extraction technique proposed for second-level features. In our previous method [22], second level feature extraction was performed similarly to first level feature extraction, with codebook learning, encoding, and pooling performed jointly over all window sizes. Window size information was disregarded, though it includes important discriminative information. Even if two features extracted from two windows are close in terms of the distance, differences in window size may indicate differences in image content. To address this issue, we propose a simple but effective grouping strategy that accounts for window sizes.

II. RELATED WORK

The original SPM work [4] performed image classification using a pyramid of three levels, since it found no improvement with more than three levels. However, other works have reported better classification performance when a fourth level is included [12], [17]. Wu and Rehg [23] added overlapped spatial areas to the nonoverlapped grid in SPR. The winner of VOC 2007 [24] designed a novel spatial pyramid layout which was adopted by many recent state-of-the-art methods [25]–[27]. In [28], areas with irregular shape are used instead of the rectangular spatial areas in SPR. In contrast to these aforementioned methods, our work effectively and efficiently processes a complete set of rectangular windows, instead of a subset of predefined windows.

Jia et al. [29] also presented a method to account for a complete set of rectangular windows. Different from their method which used supervised learning to find some spatial areas, our work proposes an unsupervised technique that directly handles the over-complete set of spatial areas while also providing information complementary to the first layer features.

Three basic steps of BoF—namely local descriptor extraction, coding of the local descriptors, and pooling of the encoded local descriptors—are utilized in SPR [4], our proposed feature extraction method, and in many other methods. Kulkarni and Li [30] proposed affine SIFT to handle pose and viewpoint variance. Boureau et al. [5] proposed a mid-level feature based on a set of neighboring local SIFT features. Yang and Newsam [6] took advantage of spatial pyramid co-occurrence for overhead aerial imagery. Boureau et al. [31] presented a pooling scheme that can effectively handle large codebooks by conducting the pooling multiple times over partitioned subspaces. Feng et al. [10] proposed geometric $\ell_p$-norm pooling that places different importance on different geometric positions. Bosch et al. [32] detected the region of interest before applying BoF feature extraction. Our proposed feature extraction framework can readily be incorporated with these methods to introduce a higher level of feature with densely sampled windows.

Our proposed feature extraction method conducts encoding and pooling multiple times, similar to [31]. However, the multiple rounds of encoding and pooling in [31] are conducted over different subspace partitions, while the multiple poolings in the proposed method are conducted over different sizes of windows.

The work in [33] and [34] proposed to extract new types of higher level feature representations to exploit spatial or spatial-temporal co-occurrences beyond local descriptors. For final classification in both works, their proposed features are pooled to obtain a global histogram over the whole image (i.e., a $1 \times 1$ spatial pyramid) or the whole video (i.e., a $1 \times 1$ spatial-temporal pyramid). In contrast, our method goes beyond spatial pyramids by extracting the final features from windows densely sampled over location, size and aspect ratio.

Another stream of research takes advantage of attribute or object level classifiers to extract high level features directly [35], [36] or use them indirectly for visual word disambiguation [37]. All of these methods involve supervised learning of attribute classifiers using an extra training set collected from Google search or other sources. In contrast, our feature extraction framework does not require an extra training set, and the entire feature extraction process is unsupervised.
Several feature extraction techniques have also been proposed for other related tasks. Duchenne et al. [38] proposed a graph-matching method that matches corresponding object points in different images for object classification. Boiman et al. [39] applied the nearest-neighbor classifier directly on different categories of SIFT features. Gehler and Nowozin [40] combined different kinds of features and showed high performance can be obtained by combining multiple kernels. Bo and Sminchisescu [41] treated image recognition as an image matching problem and solved it by kernel matching.

Recent works [19], [42], [43] demonstrated that it is generally beneficial to utilize prelearned classifiers from other classes for event/action recognition and image retrieval. In contrast to the $\ell_1$-norm constraint used in existing methods like [19], [42], in GA-MKL we utilize the more general $\ell_p$-norm constraint (e.g., $p = 2$ in this paper) which can preserve complementary and orthogonal information [44], [45]. This is particularly important when base kernels contain complementary information as in our two-level feature extraction framework. Furthermore, GA-MKL also learns the weights for multiple sets of prelearned classifiers. Using the prelearned classifiers on other classes also distinguishes GA-MKL from the existing $\ell_p$-MKL technique.

III. TWO-LEVEL FEATURE EXTRACTION

A. First Level Image Feature

In the proposed feature extraction framework, two sets of features are extracted at two levels. For the first level, BoF with SPR image features are extracted. For the second level, a set of higher level features are extracted based on the first level of feature extraction. We first provide a brief review of BoF with SPR, which is presented in detail in [4]. BoF with SPR consists of five key components conducted sequentially—local descriptor extraction, codebook learning, encoding, spatial pyramid generation and pooling. The first three components are illustrated as rectangles in the upper left of Fig. 1. The fourth and fifth components are shown as rectangles in the upper right of Fig. 1. First, local descriptors such as SIFT are extracted from image patches that describe local image characteristics. A visual word codebook is then generated from these local features by performing clustering to quantize the large number of local descriptors. This visual codebook thereafter is used to represent each local descriptor as an encoded vector by assigning each local descriptor to the codes in the codebook. Next, the encoded local descriptors are partitioned into different groups by different spatial areas in SPR. Finally, pooling is performed over each group of encoded vectors. And the vectors generated from each window after pooling are all concatenated to form the first level image feature. We note that any advanced method for any of these five components [7]–[10], [31], [46] can readily be used in the first level of feature extraction. The implementation details for the first level image feature extraction of this paper will be presented in Section V-A.

To facilitate understanding of the second level of feature extraction, we also describe the region partitioning in SPR. A spatial pyramid [4] subdivides the input image into a sequence of grids with incrementally finer nonoverlapping regions of the same size. As illustrated at the left of Fig. 2, the grid at level $l$ has $2^l$ cells along each dimension, for a total of $D = 2^l \times 2^l$ cells. As one may notice, the regions’ locations, aspect ratios, and sizes are restricted in SPR to a very sparse set that may not capture much of the useful information from different regions. To solve this issue, we propose a new second level feature extraction method based on densely sampled regions.

B. Second Level Image Feature

1) Dense Sampling of Spatial Areas: In the second level feature extraction, we propose to densely sample the spatial areas with respect to location, aspect ratio, and size. This is achieved as follows. Let us denote each spatial area as $Area(x, y, w, h)$, where $(x, y)$ indicates the image position of the upper-left corner of the spatial area, and $(w, h)$ denotes the width and height of the spatial area. The 4-tuples of
Fig. 2. Illustration of spatial sampling. Left: Sparse spatial sampling in [4]. Right: Proposed dense spatial sampling.

Fig. 3. Partitions of the space of window sizes.

Area\((x, y, w, h)\) are enumerated to obtain a comprehensive set of spatial areas with respect to \(x, y, w,\) and \(h\).

In the right part of Fig. 2, we illustrate the dense sampling procedure. For each fixed size of \((\hat{w}, \hat{h})\), all possible locations of \((\hat{x}, \hat{y})\) are sampled. As shown by the red arrows, the location shifts from left to right (X-direction), and from top to bottom (Y-direction). As shown by the black horizontal and vertical axes, all sizes of different widths and heights of the spatial areas are sampled. In the figure, the size of each sampled spatial area is shown beside the top-left corner of each image. Through dense sampling, windows that tightly bound an object or other potentially meaningful image patch can be captured. Three examples are shown by yellow rectangles in Fig. 2 for the bear’s head and leg, and also a log on the ground.

2) Grouping Densely Sampled Spatial Areas: The densely sampled spatial areas have very different sizes, with width and height varying from 30 to 300 pixels. To effectively deal with the size variation, we partition the space of window sizes into a small number of groups. Let us denote the minimal window width, the minimal window height, the maximal window width and the maximal window height as \(w_{\text{min}}, h_{\text{min}}, w_{\text{max}},\) and \(h_{\text{max}}\). Suppose we want to divide the size space into \(S_w \times S_h\) groups, where \(S_w\) is the number of intervals to partition the overall feasible width range \([w_{\text{min}}, w_{\text{max}}]\) and \(S_h\) is the number of intervals to partition the overall feasible height range \([h_{\text{min}}, h_{\text{max}}]\). Then the size space is divided into sub-spaces by dividing the feasible width range and feasible height range uniformly without the overlap by the specified interval numbers of \(S_w\) and \(S_h\). An example of partitioning the size space with \(S_w = S_h = 3\) is illustrated in Fig. 3. Each cell in the gray area represents a range of window sizes.

3) Second Level Clustering, Encoding, and Pooling: We now have a set of spatial areas and their corresponding group assignments according to window size. Pooling of encoded local descriptors from the first level feature extraction is then performed over each of these spatial areas to produce a feature vector which we refer to as a window-based feature. In other words, the first level codebook and the encoded local descriptors are reused to obtain the window-based feature for each densely sampled window by pooling over each sampled window (upper-left part of Fig. 1).

Then we use the window-based features as the input features together with the group information in the second level of window-based feature codebook learning, encoding, and pooling. From each group of window-based features, feature clustering is conducted to obtain a group-dependent codebook. Each group-dependent codebook is then utilized to encode only the window-based features within its group. Finally, pooling over the whole image is performed multiple times for each group of encoded window-based features. This procedure is illustrated in the lower part of Fig. 1. The feature dimension of the resultant second level image features from each group is the same as the size of the corresponding group-dependent codebook.

The second level differs from the first level because clustering, encoding, and pooling are carried out over the window-based features instead of local SIFT descriptors. The window-based codewords essentially represent a set of window clusters that each shares similar content in terms of the first level feature. We will later show in the experiments that this higher level abstraction of window-based features provides useful complementary information to first level image features.

In our previous work [22], second level image feature extraction involves codebook learning, encoding, and pooling only once by taking all the window-based features as a whole, which yields only a single second level image feature. By contrast, in the current method the second level codebook learning, encoding, and pooling are conducted multiple times, each over one group of window-based features from windows with the similar size. As a result, multiple second level image features are obtained for an image. Since these features additionally encode information with respect to window size, they are expected to benefit the final classification.
The proposed two-level feature extraction framework can be applied to any kind of local descriptor individually, such as SIFT [15], Spatial HOG [47], and LBP [48]. Using different local features would allow us to obtain multiple corresponding sets of first level and second level image features and capture more characteristic properties of the local patches. In our experiments, the above-mentioned three descriptors are all used to extract the corresponding two levels of image features for better classification performance.

IV. GENERALIZED ADAPTIVE \( \ell_p \)-NORM MULTIPLE KERNEL LEARNING (GA-MKL)

GA-MKL aims to fuse different types of visual features such as the multiple sets of features extracted in Section III as well as take advantage of existing SVM classifiers trained for different types of visual features and different classes. In GA-MKL, we follow the recent \( \ell_p \)-norm multiple kernel learning [44] for learning the kernel coefficients, which can preserve complementary and orthogonal information [44], [45] in contrast to \( \ell_1 \)-norm MKL.

In the following, let us denote the \( \ell_p \)-norm of the \( M \)-dimensional vector \( d \) as \(||d||_p = (\sum_{m=1}^{M} d_m^p)^{1/p} \), and specially denote the \( \ell_2 \)-norm of \( d \) as \(||d||_2 \) for ease of presentation. We also denote the transpose of a vector or matrix with the superscript \( \prime \), the element-wise product between two vectors \( \alpha \) and \( y \) as \( \alpha \odot y = [\alpha_1 y_1, ... , \alpha_l y_l]' \). We denote an \( l \)-D vector with all elements of 1 as \( 1 \in \mathbb{R}^l \), and \( M \)-dimensional vector with all elements of 1 to be \( 1_M \). The inequality \( d = [d_1, ... , d_M]' \geq 0 \) is used to indicate that \( d_m \geq 0 \) for all \( m = 1, ..., M \). Under the one-versus-rest classification framework for multiclass classification, let us denote the training set as \( \{(x_i, y_i)\}_{i=1}^{S} \) where \( x_i \) is the \( i \)-th training image with \( y_i \in \{ +1, -1 \} \) being the corresponding label.

Suppose that in total we have \( H \) classes and \( S \) sets of prelearned classifiers \( \{f_1^s(x), ... , f_H^s(x)\}_{s=1}^{S} \), which can be learned with different image representations or visual features. Following [42], we assume that the decision function for the new classifier is a linear combination of all the prelearned classifiers with a perturbation function modeled by using the original visual features. Specifically, we define the decision function as

\[
f(x) = \sum_{i=1}^{S} \sum_{m=1}^{M} d_m \psi_m(x) + b \tag{1}
\]

where \( f_s(x) = [f_1^s(x), ... , f_H^s(x)]' \) is the \( s \)-th decision value vector for the input image \( x \) from the prelearned classifiers, \( \beta_s = [\beta_1^s, ... , \beta_H^s]' \) is the corresponding weight vector to be optimized, and \( \Delta f(x) \) is the perturbation function based on multiple kernels. For a total of \( M \) base kernels, we have \( \Delta f(x) = \sum_{m=1}^{M} d_m \psi_m(x) + b \), where \( \psi_m(\cdot) \) is the mapping function induced from the \( m \)-th base kernel, \( d = [d_1, ... , d_M]' \) is the vector of base kernel coefficients. In our work, \( d, M, \psi_m, b \) are the variables to be learnt.

We formulate the learning problem within an MKL framework. The adapted classifier \( f(x) \) in GA-MKL is learned by minimizing the following objective function:

\[
\begin{align*}
\min_{d_m, \beta_s, \mu_s} & \quad \min_{\alpha, \mu} \frac{1}{2} \sum_{s=1}^{S} \frac{\|\beta_s\|^2}{\mu_s} + \frac{\lambda}{2} \sum_{s=1}^{S} \mu_s^2 \\
& + \frac{1}{2} \sum_{m=1}^{M} ||\psi_m||^2 + C \sum_{i=1}^{l} \xi_i \tag{2}
\end{align*}
\]

s.t. \( y_i \left( \sum_{s=1}^{S} \beta_s f_s(x_i) + \sum_{m=1}^{M} \psi_m(x_i) + b \right) \geq 1 - \xi_i, \xi_i \geq 0, i = 1, ... , l \)

\( d \geq 0, ||d||_p^2 \leq 1, \mu \geq 0 \)

where \( C > 0 \) is the regularization parameter. \( \psi_m = d_m \psi_m \), \( J(\Delta f) \) is the standard MKL structural risk functional before considering the prelearned classifiers, and \( p \geq 1 \) is the norm parameter for the base kernel coefficients introduced in \( \ell_p \)-MKL [44]. Besides the structural risk term \( J(\Delta f) \) in standard MKL, the coefficients \( \beta_s \) for the prelearned classifiers are also penalized by \( \|\beta_s\|^2 \). Considering that prelearned classifiers from different visual features have different classification capacity, we further introduce an intermediate vector \( \mu = [\mu_1, ... , \mu_S]' \) to assign different weights to different sets of prelearned classifiers related to different types of features (e.g., dense SIFT feature, spatial HOG and LBP features). The regularization term \( \frac{1}{2} \sum_{s=1}^{S} \mu_s^2 \) with the regularization parameter \( \lambda > 0 \) is added in the objective function to avoid a trivial solution for \( \mu \). In this way, we not only fuse different types of visual features but also utilize the prelearned classifiers of all the classes. As GA-MKL additionally considers multiple groups of prelearned classifiers, the first term and the second term in the objective function in (2) are intrinsically different from A-MKL in [42].

The optimization problem in (2) is jointly convex with respect to \( v_m, b, \xi, \beta, d, \mu \). First, as in the derivation of [45] and [44], the term \( J(\Delta f) \) is jointly convex with respect to \( v_m, b, d \). Second, following the derivation for the first term in \( J(\Delta f) \), the first term in (2) is jointly convex with respect to \( \beta, \mu \). Finally, the second term in (2) is obviously convex with respect to \( \mu \). Thus, we arrive at the conclusion that the optimization problem in (2) is jointly convex with respect to \( v_m, b, \xi, \beta, d, \mu \). The global optimum can be obtained by using the block-wise coordinate descent algorithm [44], [45], [49] by alternatively optimizing these variables with the following two steps.

In the first step, we optimize \( v_m, b, \xi, \beta, \mu \) with fixed \( d, \mu \). By introducing the nonnegative Lagrangian multipliers \( \alpha, i = 1, ... , l \), the dual of (2) can be derived as follows:

\[
\max_{\alpha} \quad \alpha' \frac{1}{2} \alpha \odot y' \left( \sum_{m=1}^{M} d_m K_m + \sum_{s=1}^{S} \mu_s F_s \right) \alpha \odot y \tag{3}
\]

s.t. \( \alpha'y = 0, 0 \leq \alpha \leq C \)

where \( \alpha = [\alpha_1, ... , \alpha_l]' \), \( y = [y_1, ... , y_l]' \), \( K_m(x_i, x_j) = \psi_m(x_i)' \psi_m(x_j) \), and \( F_s(x_i, x_j) = f_s(x_i)' f_s(x_j) \). It can be seen that (3) shares the same form as the dual problem of SVM
by replacing the kernel as $K = \sum_{m=1}^{M} d_m K_m + \sum_{s=1}^{S} \mu_s F_s$. Therefore, it can be solved via existing SVM solvers.

After obtaining the optimum $v$ from (3), we can recover the square of primal variables $v_m, \beta_s$ accordingly:

$$\|v_m\|^2 = d_m^2 (\alpha \circ \gamma) K_m (\alpha \circ \gamma), \quad m = 1, \ldots, M$$  \hspace{1cm} (4)

$$\|\beta_s\|^2 = \mu_s^2 (\alpha \circ \gamma) F_s (\alpha \circ \gamma), \quad s = 1, \ldots, S.$$  \hspace{1cm} (5)

In the second step, we optimize $d, \mu$ with fixed $v_m, b, \xi, \beta_s$. With fixed $v_m, b, \xi, \beta_s$, the problem in (2) reduces to the following constrained convex minimization problem:

$$\min_{d_m, \mu_s} \frac{1}{2} \sum_{s=1}^{S} \|\beta_s\|^2 + \lambda \sum_{s=1}^{S} \mu_s^2 + \frac{1}{2} \sum_{m=1}^{M} \|y_m\|^2 \frac{1}{d_m}$$

s.t. $d \geq 0, \|d\|_p^2 \leq 1, \mu \geq 0.$

Similar to the derivations in [44], the closed-form solutions can be obtained as follows:

$$d_m = \frac{\|y_m\|^2 \frac{2}{p} \frac{1}{\|y_m\|^2}}{\left( \sum_{i=1}^{M} \|y_i\|^2 \right)^{\frac{1}{p}}}, \quad m = 1, \ldots, M$$ \hspace{1cm} (7)

$$\mu_s = \sqrt{\frac{\|\beta_s\|^2}{2 \lambda}}, \quad s = 1, \ldots, S.$$ \hspace{1cm} (8)

where $\|y_m\|^2$ and $\|\beta_s\|^2$ can be calculated by using (4) and (5), respectively.

In Algorithm 1, the entire optimization procedure is summarized. After obtaining the optimal $d, \mu$, and $\alpha$ using Algorithm 1, the final classifier for a test image can be expressed as

$$f(x) = \sum_{i=1}^{l} \alpha_i y_i \left( \sum_{s=1}^{S} \mu_s f_s(x) \right) + b.$$ 

V. EXPERIMENTS

In this section, we evaluate the proposed two-level feature extraction framework and GA-MKL on three benchmark databases: Caltech256 [20], 15Scenes [4], and Caltech101 [21].

A. Experimental Setup

Each image is preprocessed by downsizing to a maximal width or height of 300, and converting color images to grayscale.

1) Local Descriptor Extraction: Three types of local descriptors—dense SIFT [15], spatial HOG [47], and LBP [48]—are used in our experiments. SIFT is extracted from densely located patches at every 4 pixels in the image, with a patch size of $16 \times 16$ pixels. For spatial HOG, the HOG descriptors are extracted from densely located patches at every 8 pixels in the image, with a patch size of $8 \times 8$ pixels. Then the spatial HOG descriptor is formed by stacking together $2 \times 2$ neighboring local HOG descriptors. For LBP, the uniform LBP as described in [48] is adopted.

2) Window Sampling: More densely sampled windows at the second level would bring higher precision in locating regions of interest. However, denser sampling comes with the expense of greater computational cost. In practice, we do not exhaustively sample all the spatial areas. Instead, our implementation uses a step size of 30 pixels for $x, y, w$, and $h$. Under this setting, the number of windows is about 3000 for a $300 \times 300$ image.

3) Window Grouping: The densely sampled windows at the second level are grouped by window size. With finer partitioning, the feature dimension becomes higher and the generalization ability may be reduced. Moreover, greater memory would be needed. With coarser partitioning, there is weaker consideration of size variation. In our experiments, we empirically use $3 \times 3$ groups (i.e., $S_w = S_h = 3$).

4) Codebook Learning: K-means is employed for feature extraction at both levels.

The codebook sizes for all second level feature extractions and the first level feature extraction using SIFT feature are set to 4096. All other codebook sizes are set to 1024.

5) Encoding: Localized soft assignment [9] is used for encoding at both levels.

6) Pooling: The first level feature extraction using the LBP feature is pooled by average pooling. In all other cases, max pooling is used. A three level spatial pyramid of $1 \times 1, 2 \times 2$ and $4 \times 4$ is used.

7) Feature Normalization: The first level image features using the LBP local descriptors are normalized with the $\ell_1$-norm equal to 1 as in previous works [40], [48]. Each of the other types of image features is normalized with the $\ell_2$-norm equal to 1.

8) Designation of Different Features: The first level image feature is referred to as a SPR feature. The first level feature together with the second level feature without considering grouping information is referred to as the beyond spatial pyramid representation (BSPR) feature [22]. The first level feature together with the second level feature with grouping information is referred to as the grouped beyond spatial pyramid representation (BSPR-G) feature.

9) Kernel Learning: $\ell_p$-MKL and GA-MKL are implemented using libsvm [50]. Linear kernels with $C$ set to 10 are used throughout the experiments. In $\ell_p$-MKL and GA-MKL, we fix $p$ to 2. In GA-MKL, we empirically set $\lambda$ to
10. The parameters $C$, $p$, and $\lambda$ are fixed empirically based on the Scene15 data set, and then the fixed values are used for all three datasets. For GA-MKL, different kinds of local descriptors and different groups present a natural way to obtain multiple prelearned classifiers. For the base kernels in the first level, there are three sets in total, with each set learned by using each type of local descriptor. For the base kernels in the second level, there are nine sets in total, with each set learned from each group of window-based features by using all three types of local descriptors together. Correspondingly, for each class three prelearned classifiers from the first level and nine prelearned classifiers from the second level are constructed.

All experiments on each dataset are repeated five times with different randomly selected training images. The results are reported in terms of the mean and standard deviation from all five runs.

### B. Results on the Caltech256 Dataset

Caltech256 [20] provides challenging images for object recognition. It consists of 30,608 images with 256 object categories. In our experiments on Caltech256, we take 30, 45, and 60 images for training and use the rest for testing.

A performance evaluation of the three techniques described in this paper is presented in Table I. These three techniques include the two-level feature extraction without grouping strategy (BSPR), the two-level feature extraction with grouping strategy (BSPR-G), and GA-MKL. As the baseline, we use SPR with $\ell_p$-norm MKL (first row).

From the table, one can see that the classification accuracy with BSPR features (second row) consistently yields better results than the one with SPR features (first row) in all three scenarios using different numbers of training data. With $\ell_p$-norm MKL, the improvements of the BSPR feature over the SPR feature are 2.03%, 2.38%, and 2.73%, respectively. This demonstrates that the proposed second level features provide additional information which is complementary to SPR with only the first level features. Also, it is shown in the table that the results using the BSPR feature and our proposed GA-MKL (third row) are better than those using BSPR and $\ell_p$-MKL (second row) by 1.04%, 1.08%, and 1.26%, which indicates that it is beneficial to learn an adapted classifier that leverages prelearned classifiers from other classes. This finding is consistent with those of previous work [19], [42]. Finally, for the proposed BSPR-G feature using GA-MKL (fourth row), the improvements over the BSPR feature using GA-MKL (third row) are 0.85%, 0.66%, and 0.92%, respectively. This demonstrates that the proposed grouping strategy is more effective than that without distinguishing different sizes of windows. In total, the proposed BSPR-G feature with GA-MKL (fourth row) improves upon the baseline method SPR with $\ell_p$-MKL (first row) by 3.92%, 4.12%, and 4.91%, respectively.

After learning the adapted classifiers, we observe that similar concepts have higher weights than dissimilar ones. Taking for instance the concepts of Swan and Gorilla, the two largest values of $\beta$ are as follows: Swan ($\beta_{duck} = 0.092$, $\beta_{goose} = 0.078$), Gorilla ($\beta_{chimp} = 0.195$, $\beta_{raccoon} = 0.106$). These learned values also reflect the benefit by leveraging prelearned classifiers of other classes.

#### 1) Comparisons With State-of-the-Art

In the lower part of Table I, comparisons with state-of-the-art methods are provided. The listed methods include the most recently reported techniques as well as the most competitive methods from the past. Our method outperforms all the existing methods with various numbers of training samples. To be exact, our method exceeds the existing best results [30] by 1.84%, 2.05%, and 2.47% for 30, 45, and 60 training samples, respectively.

### C. Results on the 15Scenes Dataset

The 15Scenes dataset [4] is composed of 15 classes of scenes and contains 4,485 images in total. Following the common evaluation protocol on this dataset, we randomly select 100 images for training and use the remaining images for testing.
Table II presents performance comparisons. Using $\ell_p$-MKL, classification accuracy with BSPR features (second row) exceeds that of the method with SPR features (first row), which again demonstrates the effectiveness of our proposed two level feature extraction framework. The result using the BSPR feature and GA-MKL (third row) is also better than that from the BSPR feature and $\ell_p$-MKL (second row), which validates GA-MKL in leveraging prelearned classifiers from other classes. When using $\ell_p$-MKL, the result from the proposed BSPR-G feature (fourth row) is observed to be better than that of the BSPR feature (third row). This demonstrates the effectiveness of the proposed grouping strategy. In total, our proposed BSPR-G feature with GA-MKL brings an overall improvement of 2.72% over the baseline using SPR feature and $\ell_p$-MKL.

1) **Performance of Individual Features:** For individual BSPR features, the results are 83.2%, 84.6%, and 70.4% (resp. 75.8%, 69.8%, 69.5%) using SIFT, SHOG, and LBP features at the first (resp. the second) level. Note that the result after combining all three first level features using $\ell_p$-MKL (86.6%) is better than the best result from each individual feature at the first level, which shows the effectiveness of $\ell_p$-MKL. Though the individual results at the second level are not as good as those corresponding to the first level, they are complementary to them, and the combination of two level features using $\ell_p$-MKL leads to a better result (i.e., 88.32% versus 86.6% in Table II).

2) **Comparisons With State-of-the-Art:** In the lower part of Table II, we compare our work with state-of-the-art methods including the latest techniques and other top performers. Our method achieves the second best performance on this dataset.

![Table III](image)

**TABLE III**

<table>
<thead>
<tr>
<th>Method</th>
<th>30 training images</th>
<th>Classification Accuracy (%) on the Caltech101 Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPR feature ($\ell_p$-MKL)</td>
<td>80.81 ± 0.97</td>
<td></td>
</tr>
<tr>
<td>BSPR feature ($\ell_p$-MKL)</td>
<td>81.88 ± 1.08</td>
<td></td>
</tr>
<tr>
<td>BSPR feature (GA-MKL)</td>
<td>82.51 ± 0.48</td>
<td></td>
</tr>
<tr>
<td>BSPR-G feature (GA-MKL)</td>
<td>83.28 ± 0.78</td>
<td></td>
</tr>
<tr>
<td>NBN [39]</td>
<td>70.4 ± -</td>
<td></td>
</tr>
<tr>
<td>ScSPM [7]</td>
<td>73.20 ± 0.54</td>
<td></td>
</tr>
<tr>
<td>LLC [46]</td>
<td>73.44 ± -</td>
<td></td>
</tr>
<tr>
<td>Localized Soft-assignment [9]</td>
<td>74.21 ± 0.81</td>
<td></td>
</tr>
<tr>
<td>Beyond Spatial Pyramids [29]</td>
<td>75.30 ± 0.70</td>
<td></td>
</tr>
<tr>
<td>Feature Context [28]</td>
<td>77.09 ± 0.74</td>
<td></td>
</tr>
<tr>
<td>Local Pooling [31]</td>
<td>77.30 ± 0.60</td>
<td></td>
</tr>
<tr>
<td>GMK [38]</td>
<td>80.30 ± 1.20</td>
<td></td>
</tr>
<tr>
<td>GLP [10]</td>
<td>82.60 ± -</td>
<td></td>
</tr>
<tr>
<td>ASC [30]</td>
<td>83.28 ± -</td>
<td></td>
</tr>
</tbody>
</table>

E. Computation Time

For feature extraction, the proposed grouping strategy increases the computation cost mainly in the multiple times of dictionary learning when compared with the method [22] without the grouping setting. Taking the SIFT descriptor as an example, here we report the computation time of our feature extraction method at each level. We conduct the experiments on an IBM workstation (3.33GHz CPU with 18GB RAM) with an unoptimized MATLAB implementation. For codebook learning, in the first level, one codebook with the size of 4096 is learned from 2 000 000 randomly selected 128-dimension SIFT descriptors. The codebook learning time is about 1 h. For the second level with and without the grouping strategy, one codebook and nine codebooks are learned, respectively. The computation time to learn each codebook (with the size of 1024) from 2 000 000 randomly selected window-based features is about 4 h. With the learned codebooks, the computation times of the encoding and pooling for the first level (5184 SIFT descriptors with the feature dimension of 128), the second level (3025 windows with the window-based feature dimension of 4096) without the grouping strategy and with the grouping strategy are about 10s, 15s, and 16s on a 300 × 300 image from Caltech256. For GA-MKL, taking Caltech-256 as an example, 256 classifiers are learned in total. The computation time to learn the 256 classifiers is about 12 h.

VI. CONCLUSION

In this paper, two technical contributions are proposed for image classification. The first technique is a novel two level feature extraction framework that extends the popular spatial pyramid representation, which efficiently and concisely accounts for densely sampled windows with different sizes and allows for existing encoding and pooling techniques to be used. The second technique is a novel multiple kernel learning method called GA-MKL, which fuses different kinds of image features and leverages multiple sets of prelearned classifiers from other classes. Extensive experiments are conducted and the experimental results on three benchmark datasets show that the proposed method outperforms the state-of-the-art.
Shengye Yan received the Ph.D. degree from the Institute of Computing Technology of the Chinese Academy of Sciences, Xi’an, China, in 2009. He is currently a Professor with the School of Automation and Control, Nanjing University of Information Science and Technology, Nanjing, China. His current research interests include computer vision, pattern recognition, machine learning, and their applications in imaging.

Xinxing Xu received the B.E. degree from the University of Science and Technology of China, Hefei, China, in 2009. He is currently pursuing the Ph.D. degree from the School of Computer Engineering, Nanyang Technological University, Singapore. His current research interests include multiple kernel learning as well as image and video understanding.

Dong Xu (M’07–SM’13) received the B.E. and Ph.D. degrees from the University of Science and Technology of China, Anhui, China, in 2001 and 2005, respectively. While pursuing the Ph.D. degree, he was with Microsoft Research Asia, Beijing, China, and the Chinese University of Hong Kong, Hong Kong, for over two years. He was a Post-Doctoral Research Scientist with Columbia University, New York, NY, USA, for one year. In 2007, he joined Nanyang Technological University, Singapore, where he is currently an Associate Professor with the School of Computer Engineering. His current research interests include computer vision, statistical learning, and multimedia content analysis.

Dr. Xu was the co-author of a paper that won the Best Student Paper Award in the IEEE International Conference on Computer Vision and Pattern Recognition, in 2010.

Stephen Lin (M’10) received the B.S.E. degree from Princeton University, Princeton, NJ, USA, and the Ph.D. degree from the University of Michigan, Ann Arbor, MI, USA. He is currently a Senior Researcher with the Internet Graphics Group of Microsoft Research, Beijing, China. His current research interests include computer vision, image processing, and computer graphics.

Mr. Lin served as a Program Co-Chair for the International Conference on Computer Vision 2011 and the Pacific-Rim Symposium on Image and Video Technology 2009.

Xuelong Li (M’02–SM’07–F’12) is a Full Professor with the Center for Optical Imagery Analysis and Learning, State Key Laboratory of Transient Optics and Photonics, Xi’an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi’an, China.