Multitask Learning of Compact Semantic Codebooks for Context-Aware Scene Modeling

Botao Wang, Hongkai Xiong, *Senior Member, IEEE*, Weiyao Lin, *Senior Member, IEEE*,
Junni Zou, *Member, IEEE*, and Yuan F. Zheng, *Fellow, IEEE*

***Abstract*— In the past few decades, we have witnessed the success of bag-of-features (BoF) models in scene classification, object detection, and image segmentation. Whereas it is also well acknowledged that the limitation of BoF-based methods lies in the low-level feature encoding and coarse feature pooling. This paper proposes a novel scene classification method, which leverages several semantic codebooks learned in a multitask fashion for robust feature encoding, and designs a context-aware image representation for efficient feature pooling. Apart from conventional universal codebook learning approaches, the proposed method encodes each class of local features with a unique semantic codebook, which captures the distinct distribution of different semantic classes more effectively. Instead of learning each semantic codebook separately, we learn a compact global codebook, of which each semantic codebook is a sparse subset, with a two-stage iterative multitask learning algorithm. While minimizing the clustering divergence, the semantic codeword assignment is solved by submodular optimization simultaneously. Built upon the global and semantic codebooks, a context-aware image representation is further developed to encode both global and semantic features in image representation via contextual quantization, semantic response computation, and semantic pooling. Extensive experiments have been conducted to validate the effectiveness of the proposed method on various public benchmarks with several popular local features.**

***Index Terms*— Scene classification, multitask learning, bag-of-features, submodular optimization, clustering.**

I. INTRODUCTION

IN THE past few decades, bag-of-features (BoF) models [1]–[7] have shown much success in many tasks in computer vision, including scene classification, object detection and image segmentation, due to their efficiency in implementation and invariance to rotation, occlusion and scaling. In general, the pipeline of BoF-based scene classification works as follows. First, local features, e.g., SIFT [8], HOG [9] and LBP [10], will be extracted from images, which depict the visual appearance of local patches. Subsequently, a codebook consisting of several codewords is learned with clustering techniques, such as k-means and spectral clustering, to quantize the local features into discrete values. Eventually, the image is represented by the distribution of codewords collected at multiple scales and positions in images.

It is well recognized that, however, the major limitations of BoF-based methods are the low-level feature encoding and coarse feature pooling. Specifically, in order to depict the scenes, which usually refer to places or activities (e.g., beach, street and parade), most of conventional BoF-based methods [1]–[6], [11], [12] utilize a universal codebook to encode the local features of different *semantic classes*, which are the basic visual elements (e.g., sky, water and sand) that compose the *scenes*. The universal codebook only captures the global distribution of the local features, and lacks of semantic interpretations. To improve feature encoding in semantic level, the proposed method not only models the distribution of local features through codebook learning, but also explores the distinct structures of different semantic classes, which provides a richer understanding of the scene categories. Recent approaches, e.g., [13], [14], also learn multiple codebooks to characterize different visual concepts. Nevertheless, they learn each codebook separately, which is incapable of capturing the intrinsic relation across classes. Furthermore, the codebooks learned in this manner contain a large number of redundant codewords, which is inefficient in representation.

On the other hand, pooling local features globally often compromises the discriminative capability of the BoF-based models, since the spatial and semantic information of the local features is not fully utilized. To address this problem, various spatial and semantic pooling approaches have been proposed. Pyramid matching approaches pool the local features in the spatial domain to accurately match two sets of local features. Pyramid matching kernel [4] divides the feature space of the local features into grids of different scales, and the features falling into the same grid will be competed accordingly. Typically, various spatial pyramidal pooling methods [2], [3], [15], [16] were designed, which subdivide the image into multiple spatial grids to pool the local features. Hence, the spatial distribution of the local features can be preserved to some extent in image representation. Acknowledgedly, spatial pooling is often blamed to be inflex-

Manuscript received December 3, 2015; revised May 11, 2016 and July 26, 2016; accepted August 30, 2016. Date of publication September 8, 2016; date of current version September 23, 2016. This work was supported in part by NSFC under Grant 61425011, Grant 61622112, Grant U1201255, Grant 61271218, Grant 61529101, and Grant 61472234, and in part by the Shu Guang Project under Grant 13SG13. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Weisi Lin.

B. Wang, H. Xiong, and W. Lin are with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: botaowang@sjtu.edu.cn; xionghongkai@sjtu.edu.cn; wylin@sjtu.edu.cn).

J. Zou is with the Key Laboratory of Special Fiber Optics and Optical Access Networks, Shanghai University, Shanghai 200072, China (e-mail: zoujn@shu.edu.cn).

Y. F. Zheng is with the Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43210 USA (e-mail: zheng@ece.osu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2016.2607424

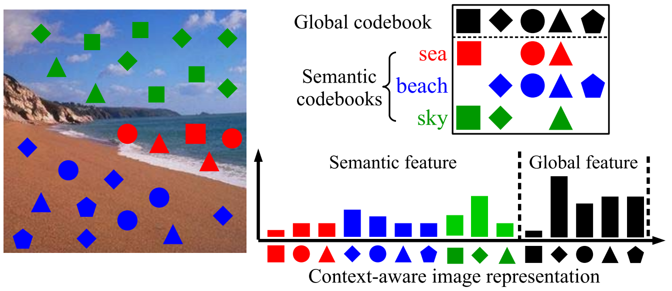


Fig. 1. The proposed method encodes the local features of each semantic class, e.g., *beach*, *sea* and *sky*, with a distinct semantic codebook, which is composed of a sparse subset of the global codebook, as illustrated in the top-right image. The image is represented in a context-aware manner by the distributions of the global and semantic codewords, as illustrated in the bottom-right image.

ible since the local features are pooled over the manually-defined spatial grids, which results in coarse and sub-optimal correspondence. In semantic pooling approaches [1], [7], [17], the pooling region of the local features is not defined spatially but semantically, such as parts [18], objects [7], and visual concepts [1]. However, semantic pooling inevitably brings about the problem of semantic scene parsing [19], [20], which is far more challenging than the task of scene classification itself.

This paper makes two technical contributions to address the problem of low-level feature encoding and inflexible feature pooling in conventional BoF-based scene classification methods, as illustrated in Fig. 1.

The first contribution is to propose a multitask codebook learning approach to learn a compact representation of multiple semantic codebooks, which aims to distinctively encode the local features of various semantic classes. Unlike conventional BoF-based methods that encode all types of local features with a universal codebook, the proposed method leverages a unique semantic codebook to encode the local features of each semantic class, which is more efficient in capturing features of color, shape and texture of the semantic class. In particular, rather than learning each semantic codebook separately, which would result in a huge quantity of redundant codewords, we learn a compact global codebook, of which each semantic codebook is a sparse subset. On the one hand, a common codeword can be shared by many semantic classes, which captures the intrinsic correlation among them. On the other hand, each semantic class may possess some unique codewords, which reflects the distinctiveness of the specific class. Note that this global constraint brings all the individual codebook learning tasks together into a joint framework, and an effective solution is designed to optimize this problem by iteratively minimizing the clustering divergence via convex optimization and optimizing the semantic codeword assignment via submodular minimization.

The second contribution of the paper is to design a context-aware image representation for scene modeling based on the learned global and semantic codebooks, which can be performed via *contextual quantization*, *semantic response computation* and *semantic pooling*. To be concrete, in contextual quantization, each local feature is quantized into a global codeword from the global codebook and multiple semantic codewords from the semantic codebooks, so that the local features can be encoded more discriminatively and robustly with multiple context-specific distributions as opposed to the universal codebook. In semantic pooling, the quantized local features are pooled semantically, and the image is represented by the global and semantic distributions of the codewords. In particular, each local feature casts a vote for the global codeword histogram with unit weight and a vote for each semantic codeword histogram with weight proportional to the corresponding semantic response. In this sense, the local features contribute more to the relevant semantic classes, and less to the irrelevant classes, embedding the relative semantic strength in image representation.

The rest of the paper is organized as follows. Section II reviews the related literatures. Section III presents the overall framework of the proposed method. Section IV reveals the multitask learning of semantic codebooks, while Section V illustrates the context-aware image representation for scene classification. Section VI shows the experimental results. Finally, Section VII concludes the paper.

II. Related Work

The success of BoF models in many tasks of computer vision, such as object detection, scene categorization and face recognition, relies on the delicate design of local features that depict different aspects of visual appearance. Among all the local features, SIFT [8] is presumably the most successful and widely-adopted choice for the past few decades due to its invariance to translation, illumination, and scaling. Since the original SIFT descriptor is not color invariant, many color-based variants have been designed, such as HSV-SIFT [21], Hue-SIFT [22], opponent-SIFT [23] and CSIFT [24]. A comprehensive evaluation of the color descriptors can be found in [23]. Histogram of oriented gradients (HOG) [9] and its variants [18], [25] also characterize the distribution of the gradient orientations, but spread more sophisticated normalization schemes. Felzenszwalb et al. [18] designed a low-dimensional alternative of the HOG descriptor, which pools the original 36-dimensional HOG descriptor along the 18 directed orientation bins, 9 undirected bins and 4 spatial cells, generating a 13-dimensional descriptor that captures the same information. Satpathy et al. [25] suggested an extended HOG to address the problem of high contrast and the confusion of opposite orientations in HOG. In addition to the SIFT- and the HOG-based features, quite a few novel local features have also been put forward in recent years. Margolin et al. [26] developed a local descriptor named Oriented Texture Curves (OTC), which depicts an image patch with curves in multiple orientations while maintaining robustness to illumination changes, geometric distortions and local contrast differences. Dubey et al. [27] derived an interleaved order based local descriptor that considers local neighbors of a pixel as a set of interleaved neighbors and constructs the descriptor over each set separately. Although the local features carry distinct information of the images, in practice, a combination of multiple local features [1], [13] often yields better performance for image representation.

Most of existing BoF-based image classification methods encode local features with a single universal codebook, which is learned with local features all over the images. In [1] and [2], vector quantization (VQ) is used to map a local feature to a single codeword by hard assignment, or multiple codewords by soft assignment based on a universal codebook. Instead of vector quantization, Yang et al. [6] represented the local features with the coefficients of sparse coding, which approximate the features more accurately than k-means quantization. Wang et al. [5] reconstructed the feature with its closest neighbors instead of the entire feature space. Zhou et al. [28] presented the super-vector coding of local features, which is a piece-wise linear approximation to the features and thus achieves a lower approximate error than VQ. Ji et al. [11] compressed a high-dimensional universal codebook into a compact one by learning a compression function with a supervised sparse coding model. Lu et al. [12] refined the BoF based representation by the universal codebook by supervised sparse coding. An alternative strategy to encode the local features emerges, by separately learning multiple codebooks. For instance, [13] learned multiple codebooks from the web images, each of which is trained individually for a visual concept. Wu et al. [14] learned a unique codebook separately for each object category based on semantics-preserving distance metric. Nevertheless, learning multiple codebooks separately will generate a large quantity of redundant codewords, which is inefficient in representation and storage.

After feature encoding, all the encoded local features will be pooled to form an image-level feature vector. However, a global histogram only reflects the holistic distribution of codewords, and the information about the spatial layout and semantic attributes is lost, which could be important cues for scene classification. To take advantage of spatial information into feature matching, a pyramid matching kernel is proposed in [4]. Local features are mapped to multi-resolution histograms based on their distribution in the feature space. Spatial pooling is not only limited to the feature space but also applied in the image space. The most prevailing spatial pooling algorithm is the spatial pyramid matching (SPM) from Lazebnik et al. [2]. SPM repeatedly divides an image into grids of different scales and computes a codeword histogram in each spatial grid. By concatenating the histograms with proper weights, the spatial correspondence of local features can be preserved in the image descriptor. Rather than using absolute spatial arrangement, Yang et al. [3] proposed the spatial pyramid co-occurence algorithm that captures both relative and absolute layout of image. In addition to partitioning the images, Harada et al. [29] tried to find the optimal weights of the spatial histograms that offer more discriminative power.

In addition to the spatial pooling schemes, a few approaches pool the local features based on the mid-level visual cues, such as the semantic category, the objectness, and the foreground/background segmentation, to alleviate the ambiguity of local features. Russakovsky et al. [7] pooled the local features separately in the object region and the background region. However, their method requires accurate object localization and is incapable of handling multiple objects. Su and Jurie [1] considered the semantic contexts of the local features during pooling. A codeword is only collected within a certain context for a specific scene category, which has the most distinct distribution. Obviously, its performance is highly dependent on the accuracy of semantic segmentation, since the contexts are hard-assigned. In view that representing each codeword in a single semantic context can not characterize the visual appearance of the image properly, Li et al. [13] collected images from the Internet to learn visual concepts via multiple instance learning. It remains in deficit by learning each visual concept separately and fails to discover the relations and connections among the concepts.

III. System Overview

The general framework of the proposed method is illustrated in Fig. 2, where the procedure of *multitask learning of semantic codebooks* and *context-aware image representation* is illustrated by the blue and red paths, respectively.

In multitask learning of semantic codebooks, local features are densely extracted from the images, which are denoted by $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^N$, where $\boldsymbol{x}_i \in \mathbb{R}^D$, N and D is the number and dimension of local features, respectively. Each local feature $\boldsymbol{x}_i$ is associated with a semantic label $s_i \in \{1, 2, \cdots, M\}$, where M is the number of semantic classes. Hence, all local features $\mathcal{X}$ can be grouped into M disjoint semantic bags $\mathcal{X}_s = \{\boldsymbol{x}_i^s\}_{i=1}^{N_s}, s = 1, \cdots, M$, where N_s is the number of local features of the s-th semantic class. Based on the bags of semantic features, M semantic codebooks will be learned by the proposed multitask codebook learning algorithm by simultaneously minimizing the clustering divergence and the sparsity of the semantic codewords. It is worth mentioning that the semantic codebooks are compactly represented by a global codebook $B = \{\boldsymbol{b}_k\}_{k=1}^K$, where $\boldsymbol{b}_k \in \mathbb{R}^D$ is the k-th codeword, and K is the total number of codewords. Each semantic codebook is a subset of the global one, and the codeword indices of the s-th semantic codebook are denoted by $\pi_s \subseteq \{1, \cdots, K\}$. A two-step iterative solution is developed to solve the multitask codebook learning problem effectively. In addition to the semantic codebooks, M semantic classifiers will be learned based on the local features and the superpixels of the images. Detail about the multitask learning of semantic codebooks will be described in Section IV.

After the global and semantic codebooks and the semantic classifiers are learned, a context-aware image representation is devised, which consists of three steps: *contextual quantization*, *semantic response computation* and *semantic pooling*. Following the red path in Fig. 2, in the testing case, local features will be extracted from the image in the first place. Then, contextual quantization assigns each local feature with a global codeword label and a semantic codeword label for each semantic class based on the learned global and semantic codebooks. In the meantime, semantic response of the local feature will be computed by the semantic classifiers. In the next step, the codeword labels attained from contextual quantization will be pooled globally and semantically to form the final image representation. To be specific, the global codewords are pooled with unit weights, regularizing the image representation as

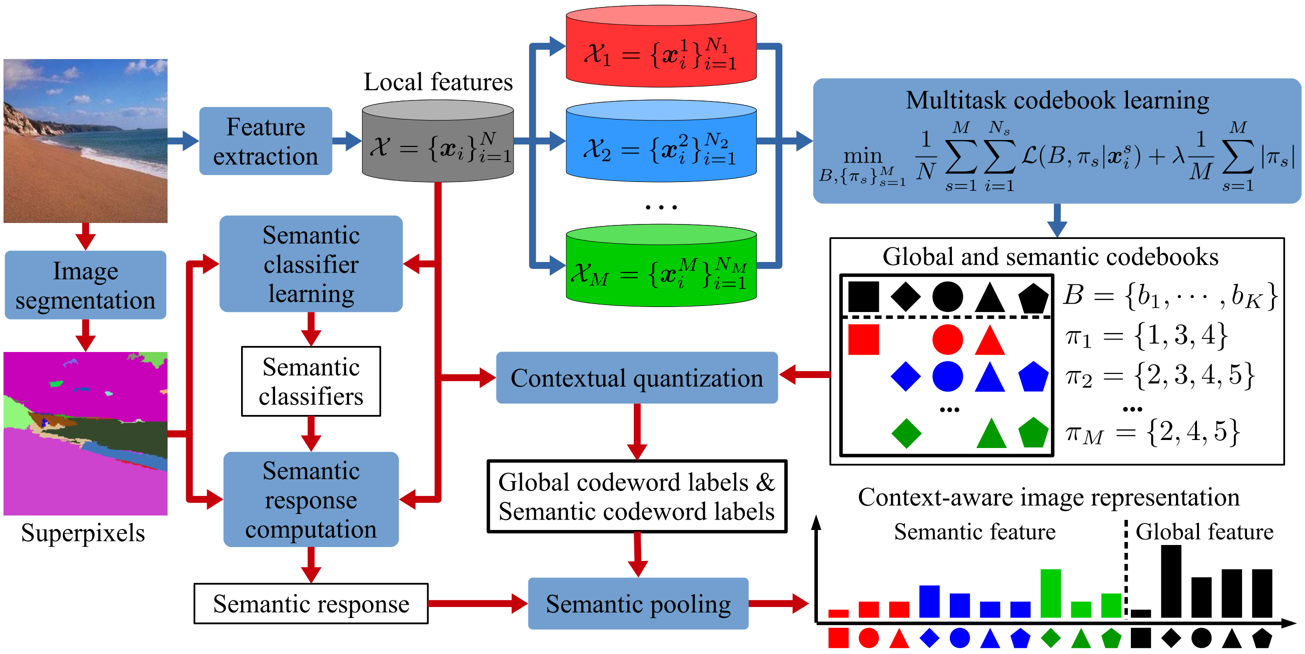

Local features
Feature extraction
$\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^{N}$
$\mathcal{X}_1 = \{\boldsymbol{x}_i^1\}_{i=1}^{N_1}$
$\mathcal{X}_2 = \{\boldsymbol{x}_i^2\}_{i=1}^{N_2}$
$\cdots$
$\mathcal{X}_M = \{\boldsymbol{x}_i^M\}_{i=1}^{N_M}$
Multitask codebook learning
$\min_{B,\{\pi_s\}_{s=1}^M} \frac{1}{N}\sum_{s=1}^{M}\sum_{i=1}^{N_s}\mathcal{L}(B,\pi_s|\boldsymbol{x}_i^s)+\lambda\frac{1}{M}\sum_{s=1}^{M}|\pi_s|$
Global and semantic codebooks
$B=\{b_1,\cdots,b_K\}$
$\pi_1=\{1,3,4\}$
$\pi_2=\{2,3,4,5\}$
...
$\pi_M=\{2,4,5\}$
Image segmentation
Superpixels
Semantic classifier learning
Semantic classifiers
Semantic response computation
Semantic response
Contextual quantization
Global codeword labels & Semantic codeword labels
Semantic pooling
Context-aware image representation
Semantic feature
Global feature

Fig. 2. The general framework of the proposed approach. The *multitask semantic codebook learning* is illustrated by the blue path. The *context-aware image representation* is illustrated by the red path.

a universal codebook. Furthermore, the semantic codeword labels are weighted by the corresponding semantic responses, introducing semantic distribution into the image representation. Detail about the context-aware image representation will be described in Section V.

IV. Multitask Learning of Semantic Codebooks

This section presents the proposed algorithm of multitask learning of semantic codebooks, which is preceded by the review of the single-task codebook learning.

A. A Retrospect of Single-Task Codebook Learning

Most of existing BoF-based image classification approaches learn a universal codebook [2], [5], [6] or multiple codebooks separately [13] in a single-task learning fashion.

To learn a single universal codebook from the images, such as [2], [5], and [6], a codebook $B = \{\boldsymbol{b}_k\}_{k=1}^{K}$ can be attained from the local features $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^{N}$ extracted from the images by solving

$$\min_{B} \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}(B|\boldsymbol{x}_i), \tag{1}$$

where $\mathcal{L}(B|\boldsymbol{x})$ measures the minimum distance of local feature $\boldsymbol{x}$ to the codebook B, which is commonly defined by the L2 norm, i.e.,

$$\mathcal{L}(B|\boldsymbol{x}) = \min_{1\le k\le K} |\boldsymbol{x} - \boldsymbol{b}_k|^2. \tag{2}$$

The solution of Eq. (1) can be obtained by the k-means algorithm.

To learn multiple codebooks separately, such as [13], Eq. (1) is independently solved for each class of local features:

$$\min_{B_s} \frac{1}{N_s}\sum_{i=1}^{N_s}\mathcal{L}(B_s|\boldsymbol{x}_i^s), \quad s = 1,\cdots,M, \tag{3}$$

where $B_s = \{\boldsymbol{b}_k^s\}_{k=1}^{K_s}$ is the codebook and $\mathcal{X}_s = \{\boldsymbol{x}_i^s\}_{i=1}^{N_s}$ is the set of local features of the s-th semantic class, respectively. However, the codebooks learned in this manner contain a large number of codewords, which are very redundant and inefficient in representation, because a codeword may have a couple of duplicates in different codebooks.

B. Multitask Semantic Codebook Learning

To address the redundancy of learning multiple semantic codebooks in single-task fashion, we learn a compact representation of multiple semantic codebooks, and develop a multitask codebook learning algorithm to learn the semantic codebooks efficiently. It is worth mentioning that the key difference of single-task learning and multitask learning is the relation of the learning tasks. In single-task learning, each learning task is independent from the others, and therefore can be solved one by one. However, in multitask learning, all learning tasks are regularized by a global constraint, which is able to improve the generalization capability of each individual learning task. As a result, all learning tasks are solved simultaneously in a joint framework in multitask learning.

To be concrete, multiple semantic codebooks $\{B_s\}_{s=1}^{M}$ are compactly represented by a global codebook $B = \{\boldsymbol{b}_k\}_{k=1}^{K}$, and each semantic codebook is composed of a subset of the global codewords, i.e., $B_s \subseteq B$ for $s = 1,\cdots,M$. With this representation, the proposed semantic codebooks are more discriminative than a universal codebook, because they implicitly characterize the distributions of different semantic classes. Also, the semantic codebooks can be represented more efficiently than separately trained ones. On one hand, many semantic classes can share a common codeword, which effectively captures the intrinsic correlation across classes. On the other hand, each semantic class may preserve some unique codewords reflecting their distinctiveness.

To learn the global codebook B and the semantic codebooks $\{B_s\}_{s=1}^{M}$, a multitask codebook learning algorithm is formulated by solving

$$\min_{B,\{B_s\}_{s=1}^{M}} \frac{1}{N}\sum_{s=1}^{M}\sum_{i=1}^{N_s}\mathcal{L}(B_s|\boldsymbol{x}_i^s)+\lambda\frac{1}{M}\sum_{s=1}^{M}|B_s|$$
$$\text{s.t.}\quad B_s \in B,\quad s=1,2,\cdots,M.X \tag{4}$$

where $|B_s|$ is the number of elements in set B_s, and λ is a parameter that controls the sparsity of the semantic codewords. The larger the λ is, the sparser the semantic codebooks will be. Note that the objective function of Eq. (4) is simply the average clustering loss of the semantic classes. The constraint of Eq. (4) combines the M individual codebook learning tasks together into a unified formulation.

To simplify the notations in Eq. (4), the codeword assignment of the s-th semantic codebook is denoted by $\pi_s \subseteq \{1,2,\cdots,K\}$, so that $B_s = \{\boldsymbol{b}_{\pi_s(1)},\cdots,\boldsymbol{b}_{\pi_s(|\pi_s|)}\}$. Hence, Eq. (4) can be converted to an unconstrained optimization problem:

$$\min_{B,\{\pi_s\}_{s=1}^{M}} \frac{1}{N}\sum_{s=1}^{M}\sum_{i=1}^{N_s}\mathcal{L}(B,\pi_s|\boldsymbol{x}_i^s)+\lambda\frac{1}{M}\sum_{s=1}^{M}|\pi_s|, \tag{5}$$

The first term in Eq. (5) measures the clustering divergence of the local features by the semantic codebooks. The minimization of this term aims at pursuing a close approximation of the local features by the corresponding semantic codebooks. Note that the clustering loss of local feature $\boldsymbol{x}_i^s$ is only measured by the s-th semantic codebook. In this way, the class-specific feature distributions can be efficiently captured by the semantic codebooks.

The second term in Eq. (5) measures the sparsity of the semantic codewords, which is the average number of codewords in the semantic codebooks. Obviously, without this term, each semantic class will take all the codewords in the global codebook to minimize the clustering divergence, which degenerates to a universal codebook. In this case, each semantic codebook is identical to the global one, thus losing the distinctiveness of the semantic classes. On the contrary, with the sparsity of semantic codebooks incorporated in Eq. (5), semantic classes may share a common codeword if the clustering loss is trivial, and a semantic class may preserve some unique codewords for the class-specific feature encoding.

Optimizing Eq. (5) is non-trivial, because it simultaneously optimizes continuous variables $B = \{\boldsymbol{b}_k\}_{k=1}^{K}$ and discrete variables $\{\pi_s\}_{s=1}^{M}$. To solve Eq. (5), we decompose the original problem into two subproblems and develop a two-step iterative scheme to solve them effectively. Specifically, the first subproblem optimizes the global codebooks with the assignment of semantic codewords being fixed. Hence, this subproblem aims at approximating the distribution of local features by the codewords. The second subproblem optimizes the assignment of semantic codewords with the global codebook being fixed. In other words, this subproblem explores the class-specific feature distribution of the semantic classes and the intra-class correlations through codeword sharing. As follows, we elaborate the details about the two-step iterative solution of Eq. (5).

C. Optimizing Global Codebook

The first subproblem minimizes the clustering divergence with respect to the global codebook B with the assignment of the semantic codebooks, i.e., $\{\pi_s\}_{s=1}^{M}$, being fixed. Once removing $\{\pi_s\}_{s=1}^{M}$ from Eq. (5), the objective function of the first subproblem is

$$\min_{B}\; E_D = \frac{1}{N}\sum_{s=1}^{M}\sum_{i=1}^{N_s}\mathcal{L}(B|\pi_s,\boldsymbol{x}_i^s). \tag{6}$$

Let A_i^s be the codeword label of x_i^s in the global codebook, i.e.,

$$A_i^s = \pi_s(a_i^s), \tag{7}$$

where

$$a_i^s = \operatorname*{argmin}_{1\le j\le|\pi_s|} |\boldsymbol{x}_i^s - \boldsymbol{b}_{\pi_s(j)}|^2 \tag{8}$$

is the codeword label of $\boldsymbol{x}_i^s$ in the s-th semantic codebook. Then, Eq. (6) can be written as

$$\min_{B}\; E_D = \frac{1}{N}\sum_{s=1}^{M}\sum_{i=1}^{N_s}|\boldsymbol{x}_i^s - \boldsymbol{b}_{A_i^s}|^2. \tag{9}$$

The optimal global codebook B can be solved analytically by

$$\frac{\partial E_D}{\partial \boldsymbol{b}_k} = 0 \Rightarrow \sum_{s=1}^{M}\sum_{i=1}^{N_s}(\boldsymbol{b}_k - \boldsymbol{x}_i^s)\mathbf{1}(A_i^s = k) = 0$$
$$\Rightarrow \boldsymbol{b}_k = \frac{\sum_{s=1}^{M}\sum_{i=1}^{N_s}\boldsymbol{x}_i^s\mathbf{1}(A_i^s = k)}{\sum_{s=1}^{M}\sum_{i=1}^{N_s}\mathbf{1}(A_i^s = k)}, \tag{10}$$

where $\mathbf{1}$(condition) is a boolean function, which equals to 1 if the condition is true, otherwise 0.

Consequently, the optimal global codebook can be obtained by updating B and A_i^s according to Eq. (7), Eq. (8) and Eq. (10) iteratively.

D. Optimizing Semantic Codeword Assignment

The second subproblem optimizes the codeword assignment of semantic codebooks, i.e., $\{\pi_s\}_{s=1}^{M}$, with the global codebook B being fixed. Similarly, once removing B from Eq. (5), the objective function of the second subproblem is

$$\min_{\{\pi_s\}_{s=1}^{M}} \frac{1}{N}\sum_{s=1}^{M}\sum_{i=1}^{N_s}\mathcal{L}(\pi_s|B,\boldsymbol{x}_i^s)+\lambda\frac{1}{M}\sum_{s=1}^{M}|\pi_s|. \tag{11}$$

Since Eq. (11) is irrelevant of B, the M tasks can be decoupled and solved independently, i.e., for $s = 1, \cdots, M$

$$\min_{\pi_s} \frac{1}{N} \sum_{i=1}^{N_s} \mathcal{L}(\pi_s | B, \boldsymbol{x}_i^s) + \frac{\lambda}{M} |\pi_s|, \tag{12}$$

Moreover, we define

$$f(\pi_s) = \frac{N\lambda}{M} |\pi_s|, \tag{13}$$

and

$$g(\pi_s) = -\sum_{i=1}^{N_s} \mathcal{L}(\pi_s | B, \boldsymbol{x}_i^s). \tag{14}$$

Hence, Eq. (12) is equivalent to

$$\begin{aligned} \min_{\pi_s} \ \phi(\pi_s) &= f(\pi_s) - g(\pi_s) \\ &= \frac{N\lambda}{M} |\pi_s| - \Big(-\sum_{i=1}^{N_s} \mathcal{L}(\pi_s | B, \boldsymbol{x}_i^s) \Big). \end{aligned} \tag{15}$$

As follows, we will prove that $f(\pi_s)$ and $g(\pi_s)$ are both submodular functions so that Eq. (15) can be solved efficiently by submodular-supermodular procedure [30]. For convenience, the task label s is omitted in the proof without loss of generality.

Theorem 1: $f(\pi)$ in Eq. (13) *is a submodular function.*

Proof: $\forall \ \pi \subseteq \pi' \subseteq \{1, 2, \cdots, K\}$ and $q \notin \pi'$:

$$\begin{aligned} &\Big(f(\pi \cup \{q\}) - f(\pi) \Big) - \Big(f(\pi' \cup \{q\}) - f(\pi') \Big) \\ &\quad = -|\pi \cup \{q\}| + |\pi| + |\pi' \cup \{q\}| - |\pi'| \\ &\quad = -|\pi| - 1 + |\pi| + |\pi'| + 1 - |\pi'| \\ &\quad = 0 \geq 0 \end{aligned} \tag{16}$$

Thus $f(\pi)$ is a submodular (modular) function by definition. □

Theorem 2: $g(\pi)$ in Eq. (14) *is a submodular function.*

Proof: $\forall \ \pi \subseteq \pi' \subseteq \{1, 2, \cdots, K\}$ and $q \notin \pi'$

$$\begin{aligned} &g(\pi \cup \{q\}) - g(\pi) \\ &\quad = \sum_{i=1}^{N} - \Big(\min\Big(\mathcal{L}(\pi | B, \boldsymbol{x}_i), |\boldsymbol{x}_i - \boldsymbol{b}_q|^2 \Big) - \mathcal{L}(\pi | B, \boldsymbol{x}_i) \Big) \\ &\quad = \sum_{i=1}^{N} - \min\Big(0, |\boldsymbol{x}_i - \boldsymbol{b}_q|^2 - \mathcal{L}(\pi | B, \boldsymbol{x}_i) \Big) \\ &\quad = \sum_{i=1}^{N} \max\Big(0, \mathcal{L}(\pi | B, \boldsymbol{x}_i) - |\boldsymbol{x}_i - \boldsymbol{b}_q|^2 \Big). \end{aligned} \tag{17}$$

Likewise,

$$\begin{aligned} &g(\pi' \cup \{q\}) - g(\pi') \\ &\quad = \sum_{i=1}^{N} \max\Big(0, \mathcal{L}(\pi' | B, \boldsymbol{x}_i) - |\boldsymbol{x}_i - \boldsymbol{b}_q|^2 \Big). \end{aligned} \tag{18}$$

Since $\pi \subseteq \pi'$, we have $\mathcal{L}(\pi | B, \boldsymbol{x}_i) \geq \mathcal{L}(\pi' | B, \boldsymbol{x}_i)$, and

$$g(\pi \cup \{q\}) - g(\pi) \geq g(\pi' \cup q) - g(\pi'). \tag{19}$$

Thus $g(\pi)$ is a submodular function by definition. □

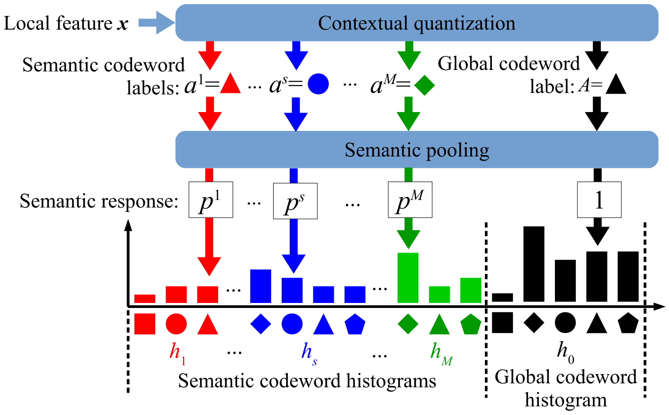

Local feature $\boldsymbol{x}$
Contextual quantization
Semantic codeword labels: a^1=▲ ... a^s=● ... a^M=◆
Global codeword label: A=▲
Semantic pooling
Semantic response: p^1 ... p^s ... p^M
1
h_1 ... h_s ... h_M
Semantic codeword histograms
h_0
Global codeword histogram

Fig. 3. The proposed context-aware image representation is computed via *contextual quantization* and *semantic pooling*. It contains both global and contextual distribution of codewords for representation.

As proved above, Eq. (15) aims at minimizing the difference of two submodular functions, namely, $f(\pi)$ and $g(\pi)$, which can be solved by the submodular-supermodular procedure. We briefly present the algorithm as follows.

The submodular-supermodular procedure minimizes Eq. (15) by finding a sequence $\pi_0, \pi_1, \cdots, \pi_n \subseteq \{1, \cdots, K\}$ satisfying $\phi(\pi_0) \geq \phi(\pi_1) \geq \cdots \geq \phi(\pi_n)$. At iteration n, a modular function $h_n(\pi)$ is constructed, which satisfies:

1) $h_n(\pi_n) = g(\pi_n)$.
2) $h_n(\pi) \leq g(\pi)$, for all $\pi \subseteq \{1, \cdots, K\}$.
3) $\phi_n(\pi) = f(\pi) - h_n(\pi)$ is submodular and can be minimized efficiently.

where h_n is named the *modular approximation* of $g(\pi_n)$. In particular, the mathematical formulation of the modular approximation based on g and π_n can be found in [30, Sec. 3.2].

Subsequently, π_{n+1} can be obtained by

$$\pi_{n+1} = \operatorname*{argmin}_{\pi \subseteq \{1, \cdots, K\}} \phi_n(\pi), \tag{20}$$

which can be computed in time polynomial in K since ϕ_n is submodular. Then we have

$$\phi(\pi_{n+1}) \leq \phi_n(\pi_{n+1}) \leq \phi_n(\pi_n) = \phi(\pi_n). \tag{21}$$

Since ϕ_n is the tight upper bound of ϕ, the optimal solution to Eq. (15) can be acquired accordingly. In practice, Eq. (15) is solved with the SFO toolbox [31].

E. Implementation Details

The optimal solution of Eq. (5) can be obtained by iteratively optimizing Eq. (6) and Eq. (11), as described in Section IV-C and Section IV-D. In particular, the global codebook B is initialized by learning a universal codebook with all local features as Eq. (1). Although it is a complicated continuous- discrete- optimization problem, the proposed two-step iterative solution effectively reduces the value of the objective function in Eq. (5) in each iteration, and generally converges in 5 $\sim$ 6 iterations in experiments.

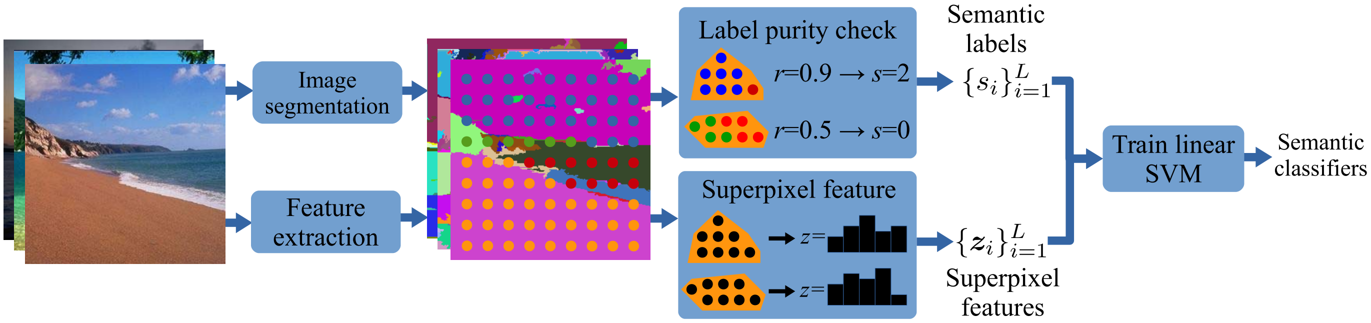


Fig. 4. Illustration of learning semantic classifiers. The input images are segmented into superpixels. The semantic label of a superpixel is determined by the label purity of the local features. The visual feature of each superpixel is encoded by the global codeword histogram. Finally, semantic classifiers can be obtained by learning linear SVM classifiers with the semantic labels and visual features of the superpixels.

V. Context-Aware Image Representation

Leveraging the global codebook and semantic codebooks learned in Section IV, a mid-level context-aware image representation is evolved for scene classification, which incorporates both global and semantic information in scene modeling. In general, the underlying image representation can be fulfilled in three steps: *contextual quantization*, *semantic response computation* and *semantic pooling*, which is illustrated in Fig. 3. Specifically, each local feature is quantized into a global codeword label and several semantic codeword labels during contextual quantization. Then, the semantic responses of the local features will be computed by the semantic classifiers, which serve as the voting weights for feature pooling. Finally, the codeword labels are pooled semantically to generate the final image representation.

A. Contextual Quantization

Conventional methods utilizing a universal codebook would quantize each local feature to a single codeword, which is coarse in representation and weak for discrimination. On the contrary, the proposed method quantizes a local feature to a global codeword and M semantic codewords. With this context-dependent quantization scheme, local features can be represented in a more robust and discriminative way by the global and semantic codebooks.

The *global codeword label* of a local feature $\boldsymbol{x}$ is the codeword index of its closest neighbor in the global codebook B, i.e.,

$$A = \underset{1\leq k\leq K}{\operatorname{argmin}} |\boldsymbol{x} - \boldsymbol{b}_k|^2. \tag{22}$$

Likewise, the s-th *semantic codeword label* of the local feature is the codeword index of its closest neighbor in the s-th semantic codebook, i.e.,

$$a^s = \underset{1\leq k\leq |\pi_s|}{\operatorname{argmin}} |\boldsymbol{x} - \boldsymbol{b}_{\pi_s(k)}|^2, \quad s = 1, \cdots, M. \tag{23}$$

In this way, a local feature is quantized to $M + 1$ codeword labels, namely, $(A, a^1, \cdots, a^M)$, which satisfy $A \in \{1, \cdots, K\}$ and $a^s \in \{1, 2, \cdots, |\pi_s|\}$ for $s = 1, \cdots, M$.

B. Semantic Response Computation

Aside from the codeword labels, the semantic responses of the local features will be computed, which serve as the voting weights for feature pooling. To compute the semantic responses of the local features, a set of semantic classifiers will be learned during training, which is illustrated in Fig. 4.

Since a local feature alone does not contain sufficient information for semantic discrimination, we perform scene classification over the superpixels, which are large enough to carry semantic meaning. To be specific, the images are segmented into overlapping superpixels at multiple scales by graph-based image segmentation [32]. Each superpixel $l = 1, \cdots, L$ is represented by a *semantic label* $c_l \in \{0, 1, \cdots, M\}$ and a *visual feature* $z_l \in \mathbb{R}^K$, where L is the number of superpixels. In particular, the semantic label of a superpixel is determined by the *label purity* of the local features inside the superpixel, which is defined as

$$r_l = \max_{j=1,\cdots,M} \frac{1}{N_l} \sum_{i=1}^{N_l} \mathbf{1}(s_i = j), \tag{24}$$

where N_l and $\{s_i\}_{i=1}^{N_l}$ is the number and semantic labels of the local features in the superpixel, respectively. In other words, the label purity depicts the percentage of the dominant semantic class in the superpixel. Obviously, the superpixel can be associated with the label of the dominant class if its label purity is sufficiently large. Otherwise, the semantic label of the superpixel is zero, which means the *unknown* class. In practice, the threshold of label purity is 0.8, i.e.,

$$c_l = \begin{cases} \underset{j=1,\cdots,M}{\operatorname{argmax}} \dfrac{1}{N_l} \displaystyle\sum_{i=1}^{N_l} \mathbf{1}(s_i = j), & \text{if } r_l \geq 0.8 \\ 0, \ \text{if } r_l < 0.8 \end{cases} \tag{25}$$

Furthermore, the visual appearance of a superpixel is represented by the histogram of the global codeword labels of the local features, which is denoted by $z \in \mathbb{R}^K$. Finally, the semantic classifiers are learned by training linear SVMs over $\{(z_i, c_i)\}_{i=1}^{L}$.

To predict the semantic response of the local features, we also segmented an image into overlapping superpixels by graph-based image segmentation, and represent each superpixel by the global codeword histogram. Subsequently, the

TABLE I

SCENE CLASSIFICATION ACCURACY ON THE MSRC DATASET

		1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*	*16*	*17*	*18*	*19*	*20*	avg.
SIFT	SPM	.417	**1.00**	.583	**1.00**	.333	**1.00**	**.833**	**1.00**	.750	.846	.583	.500	.833	.250	.500	.250	.750	.417	.500	.375	.641
	LLC	.417	**1.00**	.667	**1.00**	.667	**1.00**	**.833**	**1.00**	**.917**	.923	.750	.714	.750	.500	.600	.167	.833	.500	.333	.500	.709
	ScSPM	**.583**	**1.00**	.750	**1.00**	.667	**1.00**	**.833**	**1.00**	.833	.923	.667	.714	.833	.500	.500	.250	.833	.583	.333	.500	.721
	CEIR	.167	**1.00**	**.917**	**1.00**	**.833**	**1.00**	**.833**	**1.00**	.750	.923	**1.00**	.143	**1.00**	.000	**1.00**	**1.00**	.833	.417	.833	.500	.755
	FV	.500	**1.00**	.667	**1.00**	.750	**1.00**	.750	**1.00**	.833	.846	**1.00**	.714	.750	.583	.500	.250	.750	.667	.333	.375	.722
	VLAD	.167	**1.00**	.417	**1.00**	.583	**1.00**	**.833**	**1.00**	.750	.923	.917	.429	.750	.417	.500	.250	.583	.583	.167	.375	.637
	Ours	**.583**	**1.00**	.750	**1.00**	.667	**1.00**	**.833**	**1.00**	**.917**	**1.00**	**1.00**	**1.00**	**1.00**	**.917**	**1.00**	**1.00**	**.917**	**.833**	**.917**	**.875**	**.911**
HOG	SPM	.083	**1.00**	.583	.917	.583	.917	**.833**	**1.00**	.750	.769	.750	.714	.667	.333	.600	.167	**.917**	.500	.167	.625	.646
	LLC	.250	**1.00**	.583	.917	**.917**	**1.00**	**.833**	**1.00**	.750	.923	.917	.643	.750	.500	.600	.250	.833	.500	.333	.500	.705
	ScSPM	**.333**	**1.00**	.667	.917	.667	**1.00**	**.833**	.917	.750	.846	.833	.571	.833	.583	.600	.250	.750	.500	.167	.750	.688
	CEIR	**.333**	.917	**1.00**	**1.00**	**.917**	**1.00**	**.833**	**1.00**	**.917**	.923	**1.00**	.857	**1.00**	.000	**1.00**	**1.00**	.833	.667	.833	.000	.814
	FV	.167	**1.00**	.333	**1.00**	.583	.917	**.833**	**1.00**	.833	.769	.917	.571	.833	.250	.500	.333	.667	.417	.167	.500	.633
	VLAD	.167	.750	.417	.833	.417	.917	**.833**	**1.00**	.833	.923	.917	.500	.750	.250	.200	.333	.750	.417	.083	.625	.599
	Ours	.250	.917	.917	**1.00**	.833	**1.00**	**.833**	**1.00**	**.917**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**.917**	**.750**	**1.00**	**.875**	**.911**
LBP	SPM	.167	**1.00**	.750	.917	.417	.833	**.833**	**1.00**	.750	.769	.500	.214	.583	.500	.300	.167	.833	.500	.250	.625	.595
	LLC	.083	**1.00**	.667	**1.00**	.417	.917	**.833**	**1.00**	.833	.769	.667	.214	.833	.667	.500	.167	**1.00**	**.833**	.250	.625	.662
	ScSPM	.083	**1.00**	.750	**1.00**	.500	.917	**.833**	**1.00**	.833	.846	.667	.286	.833	.833	.400	.250	.917	.750	.250	.625	.679
	CEIR	**.417**	**1.00**	**1.00**	**1.00**	**.917**	**1.00**	**.833**	**1.00**	.750	.923	**1.00**	.786	**1.00**	.000	**1.00**	.917	.917	.333	.750	**.875**	.819
	FV	.250	.917	.333	**1.00**	.417	.917	**.833**	**1.00**	.833	.769	.667	.357	.583	.667	.400	.333	.667	.583	.083	.375	.603
	VLAD	.333	.833	.250	**1.00**	.333	.917	**.833**	.917	.750	.923	.667	.286	.667	.500	.500	.417	.833	.333	.167	.750	.608
	Ours	.250	**1.00**	**1.00**	**1.00**	.833	**1.00**	**.833**	**1.00**	**.917**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	.917	.667	**1.00**	.750	**.911**
OTC	SPM	.417	**1.00**	.500	.917	.583	**1.00**	**.833**	**1.00**	.750	.846	.833	.500	.750	.500	.400	.333	.833	.333	.250	.750	.666
	LLC	.333	**1.00**	.667	**1.00**	**.917**	**1.00**	**.833**	**1.00**	.750	**.923**	.750	.500	.750	.500	.600	.417	**.917**	.583	.250	.750	.722
	ScSPM	.250	**1.00**	.667	**1.00**	**.917**	**1.00**	**.833**	**1.00**	.750	**.923**	.833	.286	.750	.500	.600	.333	.833	.417	.417	.750	.700
	CEIR	.167	**1.00**	.750	**1.00**	.833	**1.00**	**.833**	**1.00**	**1.00**	**.923**	**1.00**	.143	**1.00**	.500	**1.00**	**1.00**	**.917**	.417	.667	.750	.789
	FV	.167	**1.00**	.500	**1.00**	.583	**1.00**	**.833**	**1.00**	.750	.846	.917	.286	.917	.333	.300	.417	.583	**.667**	.167	.500	.641
	VLAD	.250	.917	.583	**1.00**	.667	.917	.750	**1.00**	.750	.846	.583	.071	.833	.333	.300	.500	.750	.333	.333	.750	.620
	Ours	**.500**	.917	**.833**	**1.00**	.833	**1.00**	**.833**	**1.00**	.917	**.923**	**1.00**	**.929**	**1.00**	**.917**	**1.00**	**1.00**	**.917**	**.667**	**.750**	**.875**	**.890**
ALL	SPM	**.500**	**1.00**	.750	**1.00**	.500	**1.00**	**.833**	**1.00**	.750	.846	.833	.714	.833	.583	.600	.333	.833	.667	.583	.625	.743
	LLC	**.500**	**1.00**	.750	**1.00**	**.833**	**1.00**	**.833**	**1.00**	.750	.923	.833	.643	.750	.583	.600	.250	**.917**	.667	.583	.625	.755
	ScSPM	**.500**	**1.00**	.833	**1.00**	**.833**	**1.00**	**.833**	**1.00**	.833	.923	.833	.643	.833	.583	.600	.250	**.917**	.667	.583	.625	.768
	CEIR	.250	**1.00**	**1.00**	**1.00**	.750	**1.00**	**.833**	**1.00**	**.917**	.923	**1.00**	.286	**1.00**	.000	**1.00**	.833	**.917**	.417	.667	.625	.768
	FV	.333	**1.00**	.667	**1.00**	.667	**1.00**	**.833**	**1.00**	.833	.846	.917	.571	.917	.583	.500	.417	.750	.583	.167	.625	.713
	VLAD	.333	**1.00**	.417	.917	.667	.917	**.833**	**1.00**	.750	.923	**1.00**	.357	.917	.333	.500	.417	.667	.500	.167	**.875**	.671
	Ours	.333	**1.00**	**1.00**	**1.00**	**.833**	**1.00**	**.833**	**1.00**	**.917**	**1.00**	**1.00**	**1.00**	**1.00**	**.917**	**1.00**	**1.00**	**.917**	**.750**	**.917**	**.875**	**.916**
CNN		.583	1.00	.917	1.00	.667	1.00	.833	1.00	.833	.923	1.00	.929	1.00	.917	.900	.750	1.00	.750	1.00	.625	.886

classification score of a superpixel can be computed by the semantic classifiers. Since the superpixels are overlapping, a local feature may be contained in multiple superpixels. Therefore, the semantic response of class s of a local feature is the average classification score of the relevant superpixels, which is denoted by p'_s. After normalization over classes, the semantic response of a local feature is

$$p^s = \frac{\exp(p'_s)}{\sum_{j=1}^{M} \exp(p'_j)}. \tag{26}$$

C. Semantic Pooling

Eventually, the codeword labels obtained in Section V-A and the semantic responses obtained in Section V-B will be pooled semantically to generate the context-aware image representation.

Specifically, a global codeword histogram $h_0 \in \mathbb{R}^K$ and M semantic codeword histograms $\{h_s \in \mathbb{R}^{|\pi_s|}\}_{s=1}^M$ will be calculated, each of which depicts the codeword distribution of the corresponding codebook. Each local feature casts a vote for the A-th bin in h_0 and the a^s-th bin in h_s for $s = 1, \cdots, M$, where A and $\{a^s\}_{s=1}^M$ are the global codeword label and the semantic codeword labels, as defined in Eq. (22) and Eq. (23).

Each local feature casts a vote for the global codeword histogram with *unit* weight. Therefore, the global codeword histogram depicts the distribution of global codewords independent of the semantic classes, as a universal codebook does. To incorporate semantic information into image representation, the voting weight of each local feature to the s-th semantic codeword histogram is the semantic response p^s. Intuitively, a local feature is supposed to make more contribution to the relevant semantic classes which have large semantic responses, and less to the irrelevant semantic classes which have small semantic responses. As a result, the semantic codeword histograms depict not only the semantic codeword distribution, but also the relative strength of the semantic classes, which is informative in modeling a scene.

Finally, by concatenating the global codeword histogram and semantic codeword histograms collected in spatial grids followed by L2 normalization, the final image representation is achieved. It is worth mentioning that the global codeword histogram and the semantic codeword histograms are normalized separately, i.e.,

$$|h_0|_2 = 1 \quad \text{and} \quad |(h_1, \cdots, h_M)|_2 = 1, \tag{27}$$

so that the relative semantic strength can be kept.

In summary, the algorithm of the proposed context-aware image representation is described in Algorithm 1.

Algorithm 1: Context-Aware Image Representation

```
Input: Input image, global and semantic codebooks,
       semantic classifiers
Output: Image feature

Extract local features from the image;
# Contextual quantization
for each local feature do
    Compute the global codeword label $A$ by Eq. (22);
    Compute the semantic codeword labels $a^s$ by Eq.
    (23);
end
# Computation of semantic responses
Segment the image into superpixels by [32];
for each superpixel do
    Compute the global codeword histogram within the
    superpixel;
    Compute the classification scores by the semantic
    classifiers;
end
# Semantic pooling
for each local feature do
    Compute the semantic responses $p_s, s = 1, \cdots, M$;
    Cast a unit vote for the $A$-th bin of the global
    codeword histogram;
    Cast a vote for the $a^s$-th bin of the semantic
    codeword histogram with weight $p_s$;
end
Concatenate and normalize the global and semantic
codeword histograms to generate the image feature.
```

TABLE II

SCENE CLASSIFICATION ACCURACY ON THE SIFT FLOW DATASET

		coast	*forest*	*high.*	*city*	*moun.*	*coun.*	*street*	*build.*	avg.
SIFT	SPM	.799	.954	.846	.919	.933	.842	.889	.965	.893
	LLC	.896	.962	.904	.951	.920	.842	.940	.979	.922
	ScSPM	.868	.962	.885	**.959**	.927	.829	.923	.979	.914
	CEIR	.944	.924	.933	.927	.927	.829	.897	.979	.918
	FV	.917	.962	.885	.943	.947	.884	.897	.965	.926
	VLAD	.854	.954	.856	.951	.933	.829	.906	.930	.900
	Ours	**.986**	**1.00**	**.971**	.951	**.987**	**.896**	**.949**	**1.00**	**.967**
HOG	SPM	.868	.939	.817	.878	.900	.707	.872	.894	.857
	LLC	.819	.931	.846	.911	.887	.793	.889	.951	.876
	ScSPM	.847	.939	.827	.862	.867	.726	.863	.951	.858
	CEIR	.965	.901	.923	.870	.913	.506	.872	.909	.847
	FV	.813	.962	.769	.927	.907	.756	.838	.944	.864
	VLAD	.806	.939	.779	.902	.887	.689	.863	.859	.837
	Ours	**.993**	**.985**	**.962**	**.959**	**.987**	**.884**	**.949**	**1.00**	**.964**
LBP	SPM	.771	.908	.817	.837	.867	.774	.880	.894	.842
	LLC	.819	.916	.817	.854	.840	.738	.906	.937	.850
	ScSPM	.806	.924	.827	.886	.860	.762	.855	.930	.854
	CEIR	.986	.863	.923	.805	.900	.640	.889	.909	.895
	FV	.792	.931	.769	.886	.893	.738	.812	.930	.844
	VLAD	.771	.886	.798	.894	.840	.726	.821	.859	.821
	Ours	**.993**	**.970**	**.952**	**.919**	**.980**	**.902**	**.940**	**1.00**	**.957**
OTC	SPM	.819	.947	.846	.902	.880	.793	.872	.937	.873
	LLC	.861	.939	.856	.919	.913	.854	.897	.965	.901
	ScSPM	.868	.947	.865	.911	.913	.842	.897	.951	.900
	CEIR	.944	.893	.885	.837	.947	.829	.821	.951	.890
	FV	.833	.954	.817	.935	.887	.793	.795	.944	.870
	VLAD	.722	.924	.789	.902	.820	.646	.803	.810	.796
	Ours	**.993**	**1.00**	**.942**	**.927**	**.980**	**.860**	**.906**	**1.00**	**.951**
ALL	SPM	.854	.962	.894	.935	.933	.842	.897	.951	.907
	LLC	.882	.954	.914	.935	.933	.872	.923	.979	.923
	ScSPM	.903	.962	.904	.943	.947	.860	.915	.979	.926
	CEIR	.972	.931	.952	.935	.947	.860	.932	.965	.935
	FV	.875	.977	.846	.927	.933	.872	.880	.972	.912
	VLAD	.847	.970	.865	.943	.907	.805	.906	.965	.899
	Ours	**.986**	**.985**	**.962**	**.968**	**.987**	**.915**	**.957**	**.993**	**.968**
CNN		.938	.954	.933	.919	.960	.884	.940	.986	.939

VI. EXPERIMENTAL RESULTS

The proposed algorithm is validated on the MSRC dataset [33] and the SIFT flow dataset [34], where pixel-wise semantic annotations are available.

The MSRC dataset contains 591 images from 20 scene categories, which are composed of 21 semantic classes: *building*, *grass*, *tree*, *cow*, *sheep*, *sky*, *aeroplane*, *water*, *face*, *car*, *bicycle*, *flower*, *sign*, *bird*, *book*, *chair*, *road*, *cat*, *dog*, *body* and *boat*. Besides, the SIFT flow dataset consists of 2688 images from 8 outdoor scene categories, namely, *coast*, *forest*, *highway*, *city*, *mountain*, *country*, *street* and *building*, which contains 33 semantic classes, including *awning*, *balcony*, *bird*, *boat*, *bridge*, *building*, *bus*, *car*, *cow*, *crosswalk*, *desert*, *door*, *fence*, *field*, *grass*, *moon*, *mountain*, *person*, *plant*, *pole*, *river*, *road*, *rock*, *sand*, *sea*, *sidewalk*, *sign*, *sky*, *staircase*, *streetlight*, *sun*, *tree* and *window*.

The proposed method is evaluated comprehensively with four popular local features, namely,

1) 128-dimensional SIFT descriptor [8];
2) 31-dimensional HOG descriptor [9];
3) 58-dimensional local binary pattern (LBP) [10];
4) 185-dimensional Oriented Texture Curves (OTC) [26].

To be concrete, the local features are extracted from the images with the *VLFeat* [35] package as follows. The image is evenly divided into overlapping blocks at a stride of 8 pixels. In each block, a local feature is computed in 2×2 spatial cells at three scales by varying the cell sizes to 4, 6 and 8 pixels.

For each dataset, 60% of the images are used for training and the rest for testing in each scene category. By default, for the small-sized MSRC dataset, a global codebook of 500 codewords is learned, while for the mid-sized SIFT flow dataset, a global codebook of 1000 codewords is learned. The sparsity parameter λ is set to 0.003 via cross-validation. Finally, a linear SVM classifier is trained upon the proposed context-aware image features over the 3-level spatial pyramid.

A. Comparison With BoF-Based Scene Classification Methods

In the first experiment, we compare the performance of the proposed method with six conventional BoF-based scene classification methods, i.e.,

- Spatial pyramid matching (SPM) [2];
- Locality-constrained linear coding (LLC) [5];
- Linear SPM using sparse coding (ScSPM) [6];
- Context embedded image representation (CEIR) [1];
- Fisher vector (FV) [36];
- Vector of locally aggregated descriptors (VLAD) [37].

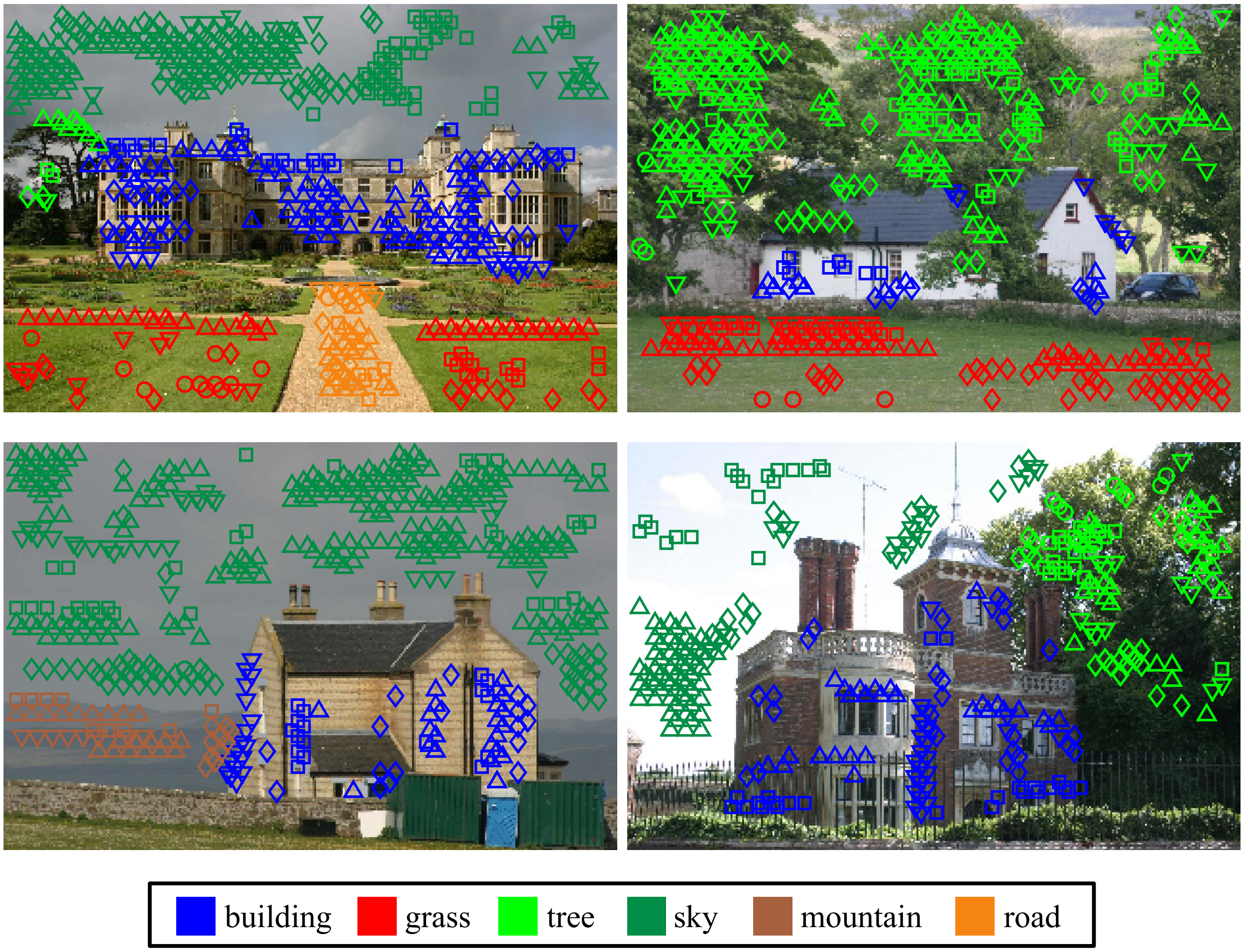


Fig. 5. The spatial distribution of the top-5 most frequent codewords in each semantic class. Each semantic class is represented by a unique color, and each codeword is represented by a unique symbol.

In addition to these BoF-based approaches, we also compare the performance of the proposed method with the state-of-the-art convolutional neural network (CNN). Specifically, each image is represented by a 4096-dimensional features vector by the CNN used in [38], and then predicted by a linear SVM classifier.

The class-specific scene classification accuracy of the MSRC dataset and the SIFT flow dataset is displayed in Table I and Table II, respectively. In addition, we also test the performance when all local features are concatenated, which is represented by ALL in Table I and Table II.

Table I shows that, among the 20 scene classes in the MSRC dataset, the proposed method achieves the highest scene classification accuracy in 18 classes with SIFT, in 16 classes with HOG, in 15 classes with LBP, in 17 classes with OTC, and in 19 classes with ALL. On the other hand, the second best methods achieve the highest scene classification accuracy only in 11, 12, 12, 12 and 11 out of 20 classes for SIFT, HOG, LBP, OTC and ALL, respectively. In terms of average accuracy, the proposed method achieves the highest average accuracy with all local features, and outperforms the second best ones by 21%, 12%, 11%, 13% and 19% for SIFT, HOG, LBP, OTC and ALL, respectively.

Table II shows that, on the SIFT flow dataset, the proposed method achieves the highest classification accuracy in all scene classes for HOG, LBP, OTC and ALL. Meanwhile, for SIFT, the proposed method performs best in 7 out of 8 scene categories, and ScSPM in 1 out of 8 scene category. On average, the proposed method outperforms the second best methods with SIFT, HOG, LBP, OTC and ALL by 4%, 10%, 7%, 6% and 4%, respectively.

Table I and Table II also indicate that although CNN is more discriminative than other BoF methods, it is still outperformed by the proposed method, because we explicitly incorporate mid-level semantic information into image representation, which is more characteristic of the scene classes.

In addition, the proposed method slightly outperforms CNN in both the MSRC and the SIFT flow dataset. It should also be noted that CNN achieves better performance than other BoF-based methods in all types of local features, which demonstrates the discriminative capability of deep neural networks in visual classification.

Furthermore, we visualize the spatial distribution of the top-5 most frequent codewords for each semantic class in Fig. 5, where each semantic class is represented by a unique color, and each codeword is represented by a unique symbol.

In conclusion, extensive experiments upon multiple datasets with various local features demonstrate the effectiveness of the proposed method in comparison with other BoF-based scene classification approaches.

B. Evaluation of Multitask Codebook Learning

In this experiment, we evaluate the performance of the proposed *multitask codebook learning* algorithm in comparison with

1) *Universal codebook learning*: A *universal* codebook of K codewords is learned with all local features, which serves as the global codebook and the semantic codebooks as well.
2) *Separate codebook learning*: Each semantic codebook contains approximately K/M^1 codewords, and is learned *separately* with the corresponding local features. The global codebook is composed of K semantic codewords.

In particular, the semantic codeword distribution of the SIFT descriptors on the SIFT flow dataset obtained by the universal codebook learning, the separate codebook learning and the proposed multitask codebook learning is illustrated in Fig. 13. Each column represents a global codeword, and each row represents a semantic class. The codewords that a semantic class selects as its semantic codewords will be highlighted with a distinct color, while the unselected codewords will be in black. It is clearly demonstrated that each semantic codebook learned by the universal codebook learning is composed of all the global codewords, and the semantic codebooks learned by the separate codebook learning are mutually disjoint. Meanwhile, the semantic codebooks learned by the proposed multitask codebook learning algorithm are represented compactly via sharing some codewords across classes.

First, we test the three codebook learning methods with the same number of codewords. The average scene classification accuracy of different codebook learning methods on the MSRC dataset ($K = 500$) and the SIFT flow dataset ($K = 1000$) is displayed in Fig. 6.

Fig. 6 indicates that training the semantic codebooks separately obtains the worst result, because there are not sufficient number of codewords in each semantic codebook, which is not capable of characterizing the complicated distribution of local features. The performance of the universal codebook learning is much better than the separate codebook learning, but still inferior than the proposed method, because the distinct feature distribution of the semantic classes is not considered. The proposed multitask codebook learning method achieves the best results, which demonstrates the effectiveness of sharing common codewords across semantic classes.

In the next experiment, we fix the number of codewords for the proposed method, i.e., 500 for the MSRC dataset and 1000 for he SIFT flow dataset, and increase the number of codewords for universal codebook learning and separate codebook learning to test how many codewords they require to achieve similar performance as the proposed method. The average scene classification accuracy on the MSRC dataset and the SIFT flow dataset is illustrated in Fig. 7 and Fig. 8, respectively.

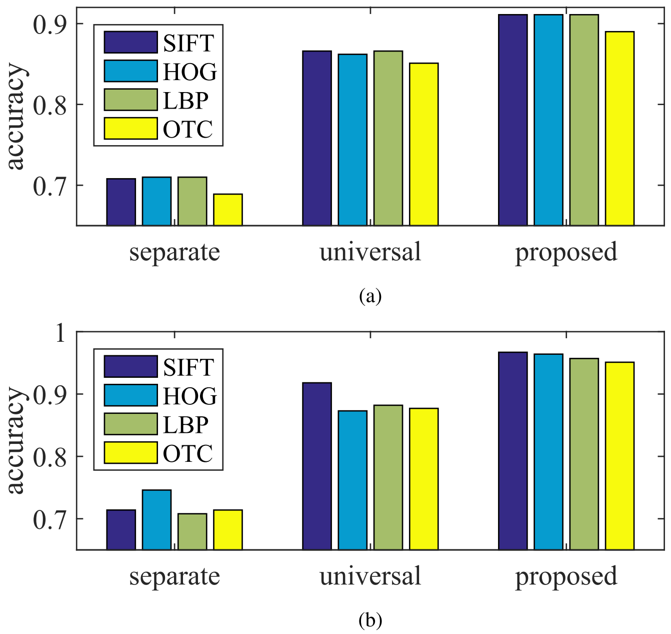


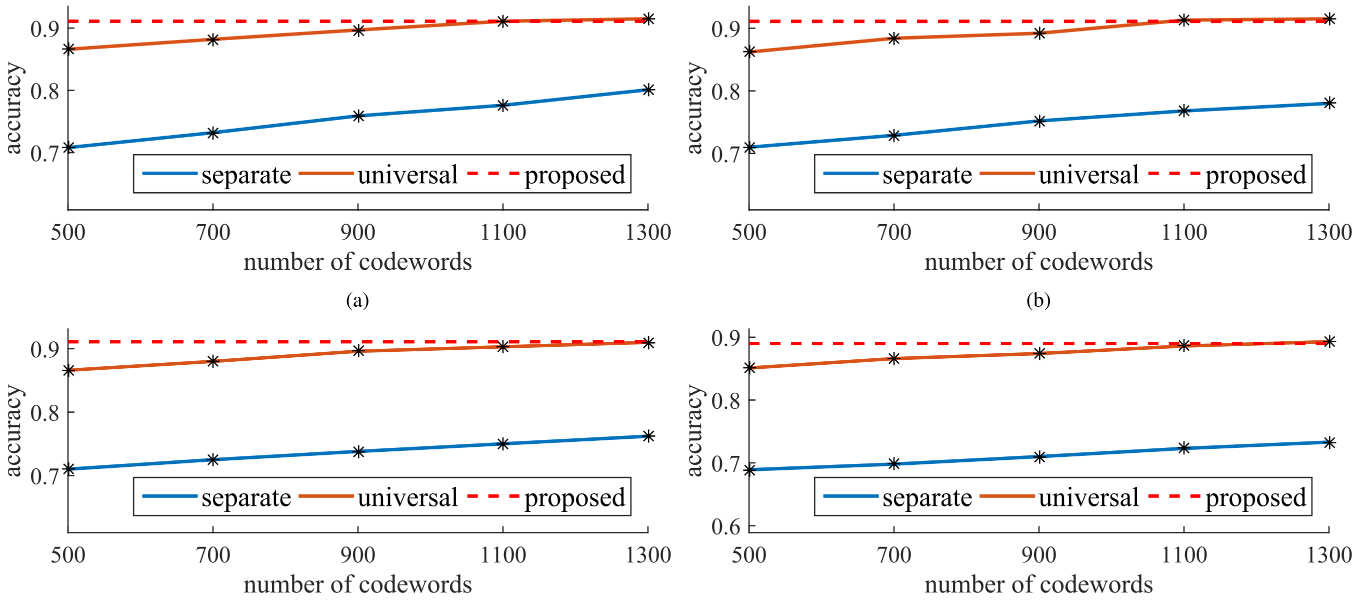
Fig. 6. Average scene classification accuracy of different codebook learning methods. (a) MSRC dataset ($K = 500$). (b) SIFT flow dataset ($K = 1000$).

Fig. 7 and Fig. 8 demonstrate that the performance of both the universal codebook leaning and the separate codebook learning is improved as the number of codewords increases. Specifically, the universal codebook learning requires approximately double the number of codewords as the proposed method to obtain similar performance, which proves that the proposed method effectively leverages the structure of the semantic features to distinguish scene classes. Whereas, the performance of the separate codebook learning is significantly lower than the universal codebook learning and the multitask codebook learning, because a large number of redundant codewords are learned by each semantic classes individually.

In addition, we evaluate the number of *common* codewords of the SIFT descriptors between separate codebook learning, universal codebook learning and the proposed multitask codebook learning. In practice, it hardly happens that two codewords in different codebooks are exactly identical. Instead, we consider that two codewords are common if and only if their distance is smaller than half the minimum distance of any two codewords in the multitask codebook. Thus, the number of common codewords in (1) multitask codebook learning and separate codebook learning, (2) multitask codebook learning and separate codebook learning is illustrated in Fig. 9.

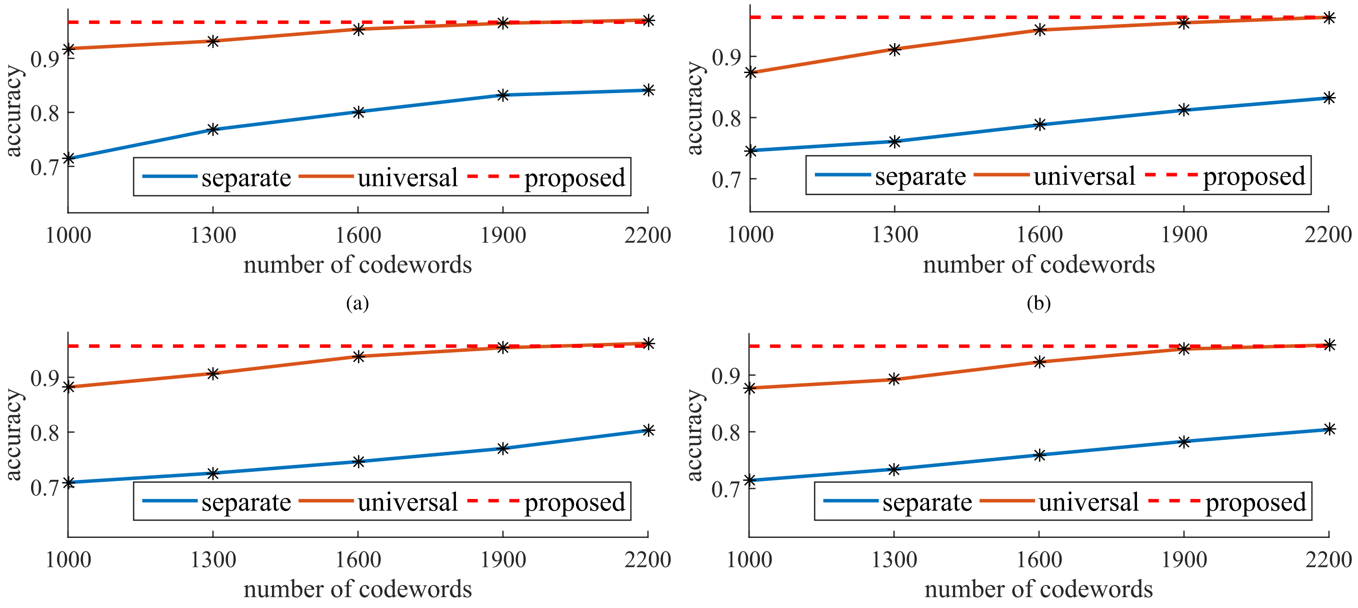
From Fig. 9, we can draw two conclusions. First, the proposed method shares a very small number of codewords with separate codebook learning and universal codebook learning due to different codebook learning mechanisms. Second, universal codebook learning has a larger number of common codewords with the proposed method than separate codebook

[1] A random set of $K - \lfloor K/M \rfloor \cdot M$ semantic codebooks contain $\lfloor K/M \rfloor$ codewords, and the rest contain $\lfloor K/M \rfloor + 1$ codewords.



(a) (b) (c) (d)

Fig. 7. Average scene classification accuracy of the universal codebook learning and the separate codebook learning with different number of codewords on the MSRC dataset. The performance of the proposed multitask codebook learning with 500 codewords is illustrated with the dash line. (a) SIFT. (b) HOG. (c) LBP. (d) OTC.



(a) (b) (c) (d)

Fig. 8. Average scene classification accuracy of universal codebook learning and separate codebook learning with different number of codewords on the SIFT flow dataset. The performance of the proposed multitask codebook learning with 1000 codewords is illustrated with the dash line. (a) SIFT. (b) HOG. (c) LBP. (d) OTC.

learning, because separate codebook learning is highly redundant.

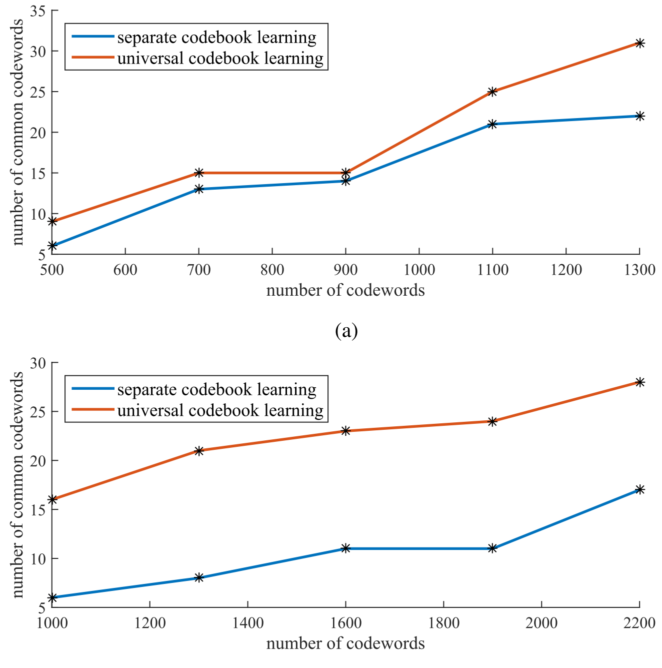
C. Evaluation of Feature Pooling

In this experiment, we evaluate effectiveness of the proposed *context-aware image representation* in comparison with the following feature pooling schemes:

1) *Hard assignment + unit voting*: Each local feature only casts a unit vote for the most likely semantic class.
2) *Soft assignment + unit voting*: Each local feature casts a unit vote for each semantic histogram.
3) *Soft assignment + weighted voting*: Each local feature casts a weighted vote for each semantic histogram, which is the proposed scheme.

Also, a global codebook of 500 codewords is learned for the MSRC dataset, and a global codebook of 1000 codewords is learned for the SIFT flow dataset by the proposed multitask codebook learning algorithm. The average scene classification accuracy of different feature pooling schemes is shown in Fig. 10.

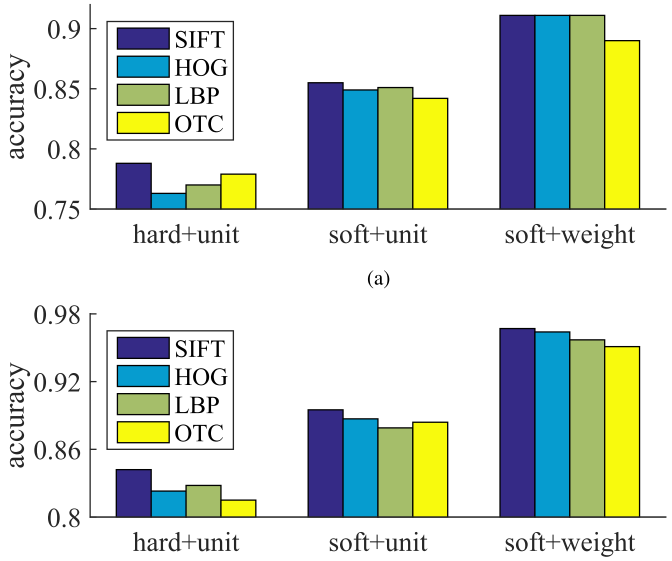
It could be observed from Fig. 10 that only voting to the most likely semantic class obtains the worst result, because the semantic response is not completely reliable, so that hard



(a)

(b)

Fig. 9. The number of common codewords in multitask codebook learning and separate codebook learning, multitask codebook learning and separate codebook learning. (a) MSRC dataset. (b) SIFT flow dataset.

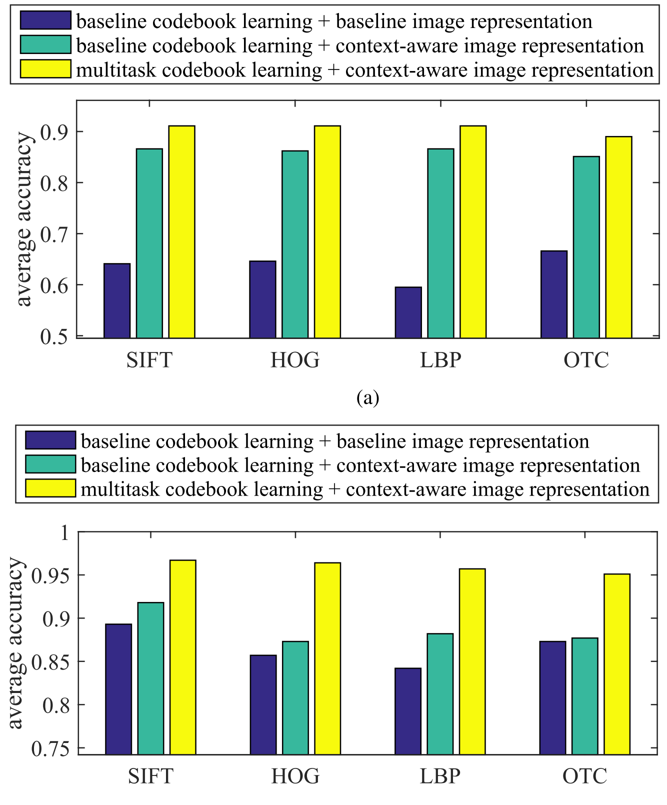


(a)

(b)

Fig. 10. Average scene classification accuracy of different pooling schemes. (a) MSRC dataset. (b) SIFT flow dataset.

voting will introduce large bias to the image representation. On the other hand, uniform voting to all semantic classes works better than hard voting, because the contextual quantization to multiple semantic codewords preserves more information in feature encoding. The proposed method achieves the best performance, because the relative strength of semantic classes is preserved in the image representation, which is important for scene classification.



(a)

(b)

Fig. 11. Average scene classification accuracy of (1) baseline codebook learning + baseline image representation, (2) baseline codebook learning + context-aware image representation, and (3) multitask codebook learning + context-aware image representation. (a) MSRC dataset. (b) SIFT flow dataset.

D. Impact of Multitask Codebook Learning and Context-Aware Image Representation

In this experiment, we analyze the gain of multitask codebook learning and context-aware image representation. Specifically, the baseline codebook learning scheme is universal codebook learning, and the baseline image representation is a global codeword histogram collected from the 3-level spatial pyramid. The average scene classification accuracy of nolistsep

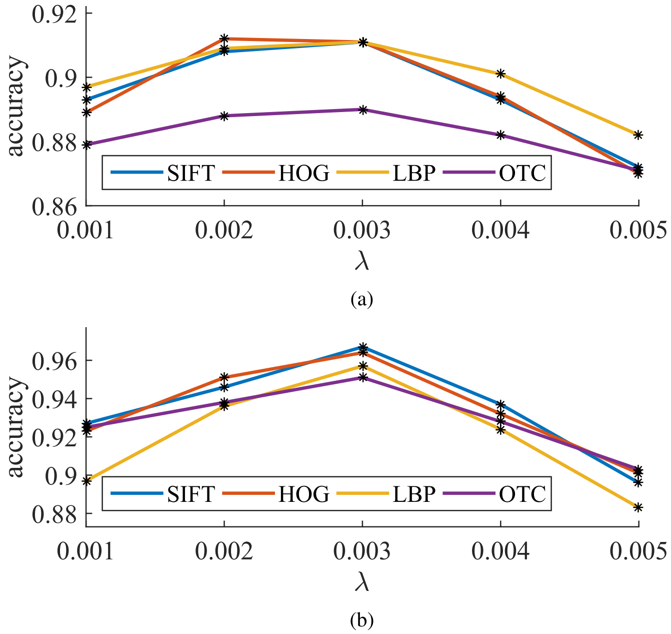
- baseline codebook learning + baseline image representation;
- baseline codebook learning + *context-aware image representation*;
- *multitask codebook learning* + *context-aware image representation*.

is illustrated in Fig. 11. It should be noted that learning the semantic codebooks without using it for context-aware image representation is meaningless, so that this situation is not evaluated.

On average, the proposed context-aware image representation alone outperforms the baseline system by 35% for MSRC dataset and 2% for the SIFT flow dataset. After incorporating the multitask codebook learning, the proposed method outperforms the baseline system by 42% for the MSRC dataset and 11% for the SIFT flow dataset.

E. Sparsity of Semantic Codewords

In this experiment, we evaluate the influence of the sparsity of the semantic codewords, which is controlled by λ in Eq. (5),



(a)

(b)

Fig. 12. Average scene classification accuracy for $\lambda = 0.001, 0.002, 0.003, 0.004, 0.005$. (a) MSRC dataset. (b) SIFT flow dataset.

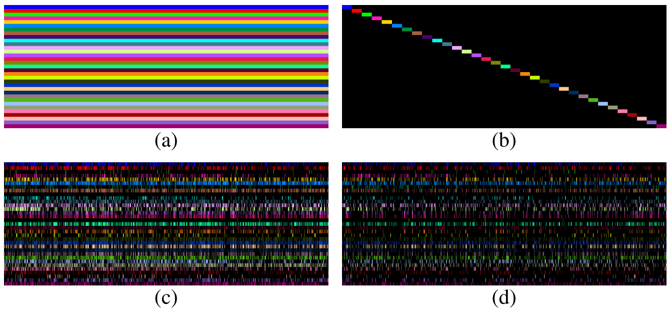
to the performance of scene classification. The average scene classification accuracy for $\lambda = 0.001, 0.002, 0.003, 0.004$ and 0.005 is tested, and the result is shown in Fig. 12.

In general, for both datasets, the average scene classification accuracy is improved as λ increases when $\lambda < 0.003$ and drops quickly when $\lambda > 0.003$. Two conclusions can be drawn from Fig. 12. First, the scene classification accuracy will be low if the semantic codewords are too dense, because the distinct feature distribution of semantic classes is not properly modeled. Second, the scene classification accuracy will also be low if the semantic codewords are too sparse, because a small number of codewords are not capable of capturing the complicated distribution of local features accurately.

Moreover, the semantic codeword distribution of the SIFT descriptors for $\lambda = 0.001$ and 0.005 on the SIFT flow dataset is visualized in Fig. 13 (c) and Fig. 13 (d). It is clearly demonstrated in Fig. 13 that the distribution of semantic codewords becomes sparser with λ getting larger.

F. Computational Complexity

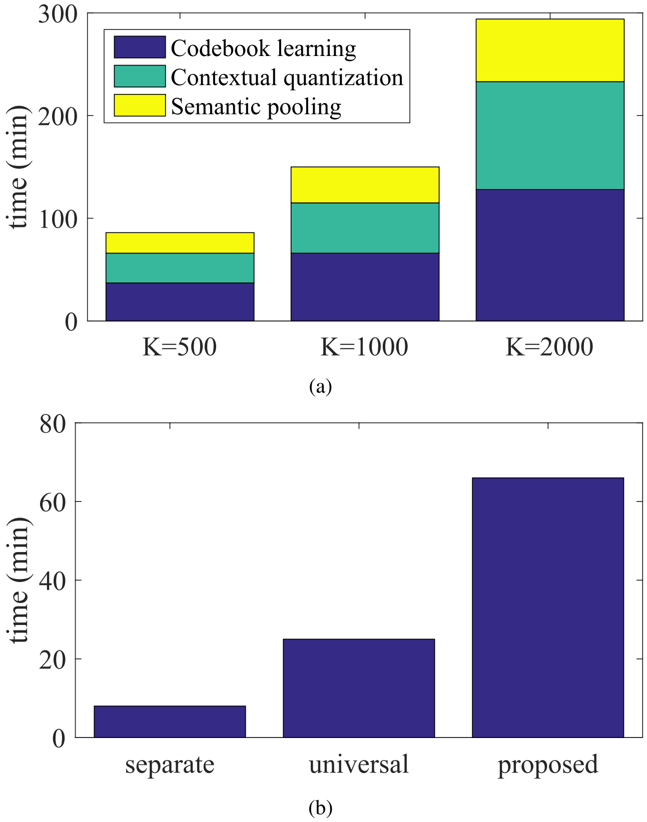
Finally, we analyze the computational complexity of the proposed method. In general, the computational complexity of the proposed method is dependent on two factors: (1) the dimension and number of the local descriptors, and (2) the number of global codewords. We evaluate the time consumption of the proposed method on the SIFT flow dataset, where 1613 image are used for training. Approximately four million 128-dimensional SIFT descriptors are then computed from those images, from which one million local features are randomly sampled to train the semantic codebooks. The computation of the proposed method is mainly occupied by three parts: (1) the learning of the semantic codebooks, (2) the contextual quantization, and (3) semantic pooling.



(a) (b)

(c) (d)

Fig. 13. Semantic codeword distribution of the universal codebook learning, the separate codebook learning and the multitask codebook learning on the SIFT flow dataset ($K = 1000$, $\lambda = 0.001, 0.005$). Each column represents a global codeword, and each row represents a semantic class. Codewords selected by the semantic classes are marked in distinct colors, and the unselected ones are marked in black. (a) Universal codebook learning. (b) Separate codebook learning. (c) Multitask codebook learning ($\lambda = 0.001$). (d) Multitask codebook learning ($\lambda = 0.005$).



(a)

(b)

Fig. 14. Computational complexity analysis. (a) time consumption of codebook learning, contextual quantization and semantic pooling in the proposed algorithm ($K = 500, 1000, 2000$); (b) time consumption of separate codebook learning, universal codebook learning and the proposed multitask codebook learning.

The composition of runtime of the proposed method is displayed in Fig. 14(a) for $K = 500, 1000$ and 2000. Note that the contextual quantization and semantic pooling are applied to all four million local features. Fig. 14(a) demonstrates that the time for codebook learning, contextual quantization and semantic pooling is almost proportional to the number of codewords.

In addition, we further compare the time consumption of separate codebook learning, universal codebook learning and the proposed multitask codebook learning on the SIFT flow dataset. Specifically, 1000 codewords are learned from approximately one million SIFT descriptors as described in Section VI-B by the three codebook learning approaches. The result is shown in Fig. 14(b). It can be observed from Fig. 14(b) that computational complexity of the proposed multitask codebook learning is significantly larger than separate codebook learning and universal codebook learning. However, in practice, since the codebook learning is accomplished off-line and the semantic codeword assignment tasks can be optimized in parallel, the time consumption of the proposed method can be further reduced.

VII. Conclusions

This paper proposes a novel scene classification method that enriches the conventional BoF-based framework with two improvements: *the multitask learning of compact semantic codebooks* and *context-aware image representation*. It encodes the local features of the images with a set of distinct semantic codebooks, which are more characteristic of the visual appearance of different types of semantic categories than a single universal codebook. Furthermore, in order to suppress the dimensionality of the codebooks, a compact representation of multiple semantic codebooks is designed. Specifically, instead of learning each semantic codebook separately in a single-task learning manner, we learn a global codebook, so that each semantic codebook is composed of a sparse subset of the codewords from the global codebook. Since this problem is non-trivial, we decompose the original problem into two subproblems, and optimize them iteratively via convex optimization and submodular optimization techniques. Based on the learned semantic codebooks, a discriminative context-aware image representation, incorporating both global and semantic information, is designed, which can be computed by *contextual quantization* and *semantic pooling*. Experiments on multiple datasets with various local features validated the effectiveness of the proposed method in comparison with other conventional approaches. Along this trajectory, our future effort is to make pixel-level annotation of semantic classes more convenient to train the semantic codebooks by semi-supervised data mining approaches.

References

[1] Y. Su and F. Jurie, "Visual word disambiguation by semantic contexts," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Barcelona, Spain, Nov. 2011, pp. 311–318.

[2] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2. Jun. 2006, pp. 2169–2178.

[3] Y. Yang and S. Newsam, "Spatial pyramid co-occurrence for image classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Barcelona, Spain, Nov. 2011, pp. 1465–1472.

[4] K. Grauman and T. Darrell, "The pyramid match kernel: Discriminative classification with sets of image features," in *Proc. ICCV*, vol. 2. Oct. 2005, pp. 1458–1465.

[5] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 3360–3367.

[6] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 1794–1801.

[7] O. Russakovsky, Y. Lin, K. Yu, and L. Fei-Fei, "Object-centric spatial pooling for image classification," in *Proc. 12th Eur. Conf. Comput. Vis.*, Oct. 2012, pp. 1–15.

[8] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[9] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, vol. 1, no. 1, pp. 886–893.

[10] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.

[11] R. Ji, H. Yao, W. Liu, X. Sun, and Q. Tian, "Task-dependent visual-codebook compression," *IEEE Trans. Image Process. (TIP)*, vol. 21, no. 4, pp. 2282–2293, Apr. 2012.

[12] Z. Lu, P. Han, L. Wang, and J. R. Wen, "Semantic sparse recoding of visual content for image applications," *IEEE Trans. Image Process.*, vol. 24, no. 1, pp. 176–188, Jan. 2015.

[13] Q. Li, J. Wu, and Z. Tu, "Harvesting mid-level visual concepts from large-scale Internet images," in *Proc. CVPR*, Jun. 2013, pp. 851–858.

[14] L. Wu, S. C. H. Hoi, and N. Yu, "Semantics-preserving bag-of-words models and applications," *IEEE Trans. Image Process.*, vol. 19, no. 7, pp. 1908–1920, Jul. 2010. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/20227977

[15] S. Yan, X. Xu, D. Xu, S. Lin, and X. Li, "Beyond spatial pyramids: A new feature extraction framework with dense spatial sampling for image classification," in *Proc. 12th Eur. Conf. Comput. Vis.*, Florence, Italy, Oct. 2012, pp. 473–487.

[16] D. Lin, C. Lu, R. Liao, and J. Jia, "Learning important spatial pooling regions for scene classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 3726–3733.

[17] J. Vogel and B. Schiele, "Semantic modeling of natural scenes for content-based image retrieval," *Int. J. Comput. Vis.*, vol. 72, no. 2, pp. 133–157, Apr. 2007.

[18] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.

[19] J. Yang, B. Price, S. Cohen, and M.-H. Yang, "Context driven scene parsing with attention to rare classes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 3294–3301.

[20] A. Sharma, O. Tuzel, and D. W. Jacobs, "Deep hierarchical parsing for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015.

[21] A. Bosch, A. Zisserman, and X. Muñoz, "Scene classification using a hybrid generative/discriminative approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 4, pp. 712–727, Apr. 2008.

[22] J. van de Weijer, T. Gevers, and A. D. Bagdanov, "Boosting color saliency in image feature detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 1, pp. 150–156, Jan. 2006.

[23] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek, "Evaluating color descriptors for object and scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1582–1596, Sep. 2010.

[24] A. Abdel-Hakim and A. Farag, "Csift: A sift descriptor with color invariant characteristics," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2. New York, NY, USA, Jun. 2006, pp. 1978–1983.

[25] A. Satpathy, X. Jiang, and H.-L. Eng, "Human detection by quadratic classification on subspace of extended histogram of gradients," *IEEE Trans. Image Process.*, vol. 23, no. 1, pp. 287–297, Jan. 2014.

[26] R. Margolin, L. Zelnik-Manor, and A. Tal, "Otc: A novel local descriptor for scene classification," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Zurich, Switzerland, Sep. 2014, pp. 377–391.

[27] S. R. Dubey, S. K. Singh, and R. K. Singh, "Rotation and illumination invariant interleaved intensity order-based local descriptor," *IEEE Trans. Image Process.*, vol. 23, no. 12, pp. 5323–5333, Dec. 2014.

[28] X. Zhou, K. Yu, T. Zhang, and T. S. Huang, "Image classification using super-vector coding of local image descriptors," in *Proc. 11th Eur. Conf. Comput. Vis.*, Oct. 2010, pp. 141–154.

[29] T. Harada, Y. Ushiku, Y. Yamashita, and Y. Kuniyoshi, "Discriminative spatial pyramid," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Portland, OR, USA, Jul. 2013, pp. 1617–1624.

[30] M. Narasimhan and J. A. Bilmes, "A submodular-supermodular procedure with applications to discriminative structure learning," in *Proc. 21st Conf. Uncertainty Artif. Intell. (UAI)*, Edinburgh, Scotland, Jul. 2005, pp. 404–412.

[31] A. Krause, "Sfo: A toolbox for submodular function optimization," *J. Mach. Learn. Res.(JMLR)*, vol. 11, pp. 1141–1144, Mar. 2010.

[32] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *Int. J. Comput. Vis.*, vol. 59, no. 2, pp. 167–181, Sep. 2004.

[33] J. Winn, A. Criminisi, and T. Minka, "Object categorization by learned universal visual dictionary," in *Proc. 10th IEEE ICCV*, Oct. 2005, pp. 1800–1807.

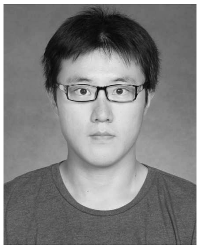
[34] C. Liu, J. Yuen, and A. Torralba, "Nonparametric scene parsing via label transfer," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2368–2382, Dec. 2011.

[35] A. Vedaldi and B. Fulkerson. (2008). *VLFeat: An Open and Portable Library of Computer Vision Algorithms*. [Online]. Available: http://www.vlfeat.org/

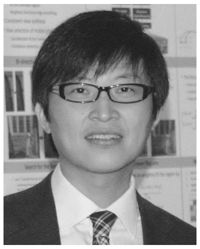
[36] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the Fisher kernel for large-scale image classification," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Heraklion, Greece, Sep. 2010, pp. 143–156.

[37] H. Jégou, M. Douze, C. Schmid, and P. Perez, "Aggregating local descriptors into a compact image representation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, San Francisco, CA, USA, Jun. 2010, pp. 3304–3311.

[38] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Stateline, NV, USA, Dec. 2012, pp. 1097–1105.



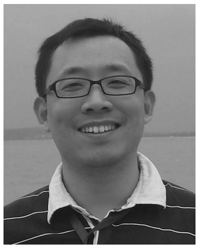
Botao Wang received the B.S. degree in electronic engineering from Shanghai Jiao Tong University, Shanghai, China, in 2010, where he is currently pursuing the Ph.D. degree. His main research interests include object detection, scene classification, and image understanding.



Hongkai Xiong (M'01–SM'10) received the Ph.D. degree in communication and information system from Shanghai Jiao Tong University (SJTU), Shanghai, China, in 2003. Since then, he has been with the Department of Electronic Engineering, SJTU, where he is currently a Distinguished Professor. From 2007 to 2008, he was with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, USA, as a Research Scholar. From 2011 to 2012, he was a Scientist with the Division of Biomedical Informatics, University of California, San Diego, CA, USA.

He has authored over 140 refereed journal/conference papers. His research interests include source coding/network information theory, signal processing, computer vision, and machine learning. He is the recipient of the Best Student Paper Award at the 2014 IEEE Visual Communication and Image Processing, the best paper award at the 2013 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, and the Top 10% Paper Award at the 2011 IEEE International Workshop on Multimedia Signal Processing.

Dr. Xiong received the National Science Fund for Distinguished Young Scholar and Shanghai Youth Science and Technology Talent as well. In 2013, he was a recipient of Shanghai Shu Guang Scholar. From 2012, he is a Member of Innovative Research Groups of the National Natural Science. In 2011, he obtained the First Prize of the Shanghai Technological Innovation Award for Network-oriented Video Processing and Dissemination: Theory and Technology. In 2010 and 2013, he received the SMC-A Excellent Young Faculty Award from Shanghai Jiao Tong University. In 2009, he was a recipient of New Century Excellent Talents in University, Ministry of Education of China. He served as a TPC Member for prestigious conferences such as ACM Multimedia, ICIP, ICME, and ISCAS.



Weiyao Lin (M'10–SM'16) received the B.E. and M.E. degrees from Shanghai Jiao Tong University, China, in 2003 and 2005, and the Ph.D degree from the University of Washington, Seattle, USA, in 2010. He is currently an Associate Professor with the Department of Electronic Engineering, Shanghai Jiao Tong University. His research interests include image/video processing, video surveillance, and computer vision.

He has authored or coauthored 80+ technical papers including 30+ referred journal papers and one book chapter. He served as an Associate Editor for *Journal of Visual Communication and Image Representation*, *Signal Processing: Image Communication*, *Journal of Circuits, Systems, and Signal Processing*, and the IEEE ACCESS.

Junni Zou (M'07) received the M.S. and Ph.D. degrees in communication and information system from Shanghai University, Shanghai, China, in 2004 and 2006, respectively. Since then, she has been with the School of Communication and Information Engineering, Shanghai University, where she is a Full Professor. From 2011 to 2012, she was with the Department of Electrical and Computer Engineering, University of California, San Diego, as a Visiting Professor.

In 2016, she received the National Science Fund for Outstanding Young Scholar. Her research interests include distributed resource allocation, multimedia networking and communications, and network information theory. She has authored over 70 referred journal/conference papers on these topics. She is the recipient of Shanghai Young Rising-Star Scientist in 2011.He acts as a Member of Technical Committee on Signal Processing of Shanghai Institute of Electronics.

Yuan F. Zheng (F'97) received the B.S. degree from the Tsinghua University, Beijing, China, in 1970, and the M.S. and Ph.D. degrees in electrical engineering from The Ohio State University, Columbus, OH, USA, in 1980 and 1984, respectively. From 1984 to 1989, he was with the Department of Electrical and Computer Engineering, Clemson University, Clemson, South Carolina. Since 1989, he has been with The Ohio State University, where he is currently a Professor and was the Chairman of the Department of Electrical and Computer Engineering from 1993 to 2004. From 2004 to 2005, he spent sabbatical year with the Shanghai Jiao Tong University, Shanghai, China and continued to be involved as the Dean with the School of Electronic, Information and Electrical Engineering until 2008.

His research interests include two aspects. One is in wavelet transform for image and video, and object classification and tracking, and the other is in robotics, which includes robotics for life science applications, multiple robots coordination, legged walking robots, and service robots. He has been on the Editorial Board of five international journals. He received the Presidential Young Investigator Award from Ronald Reagan in 1986, and the Research Awards from the College of Engineering, The Ohio State University, in 1993, 1997, and 2007, respectively. He received the Best Conference and Best Student Paper Award in 2000, 2002, and 2006, and received the Fred Diamond for Best Technical Paper Award from the Air Force Research Laboratory, Rome, New York, in 2006. In 2004, he was appointed to the International Robotics Assessment Panel by the NSF, NASA, and NIH to assess the robotics technologies worldwide in 2004 and 2005.