Weakly Supervised PatchNets: Describing and Aggregating Local Patches for Scene Recognition

Zhe Wang, Limin Wang, Yali Wang, Bowen Zhang, and Yu Qiao, *Senior Member, IEEE*

***Abstract*— Traditional feature encoding scheme (e.g., Fisher vector) with local descriptors (e.g., SIFT) and recent convolutional neural networks (CNNs) are two classes of successful methods for image recognition. In this paper, we propose a hybrid representation, which leverages the discriminative capacity of CNNs and the simplicity of descriptor encoding schema for image recognition, with a focus on scene recognition. To this end, we make three main contributions from the following aspects. First, we propose a patch-level and end-to-end architecture to model the appearance of local patches, called *PatchNet*. PatchNet is essentially a customized network trained in a weakly supervised manner, which uses the image-level supervision to guide the patch-level feature extraction. Second, we present a hybrid visual representation, called *VSAD*, by utilizing the robust feature representations of PatchNet to describe local patches and exploiting the semantic probabilities of PatchNet to aggregate these local patches into a global representation. Third, based on the proposed VSAD representation, we propose a new state-of-the-art scene recognition approach, which achieves an excellent performance on two standard benchmarks: MIT Indoor67 (86.2%) and SUN397 (73.0%).**

***Index Terms*— Image representation, scene recognition, PatchNet, VSAD, semantic codebook.**

I. Introduction

IMAGE recognition is an important and fundamental problem in computer vision research [1]–[7]. Successful recognition methods have to extract effective visual representations to deal with large intra-class variations caused by scale changes, different viewpoints, background clutter, and so on. Over the past decades, many efforts have been devoted to extracting good representations from images, and these representations may be roughly categorized into two types, namely *hand-crafted* representations and *deeply-learned* representations.

In the conventional image recognition approaches, hand-crafted representation is very popular due to its simplicity and low computational cost. Normally, traditional image recognition pipeline is composed of feature extraction, feature encoding (aggregating), and classifier training. In feature extraction module, local features, such as SIFT [8], HOG [9], and SURF [10], are extracted from densely-sampled image patches. These local features are carefully designed to be invariant to local transformation yet able to capture discriminative information. Then, these local features are aggregated with a encoding module, like Bag of Visual Words (BoVW) [11], [12], Sparse coding [13], Vector of Locally Aggregated Descriptor (VLAD) [14], and Fisher vector (FV) [15], [16]. Among these encoding methods, Fisher Vector and VLAD can achieve good recognition performance with a shallow classifier (e.g., linear SVM).

Recently, Convolutional Neural Networks (CNNs) [17] have made remarkable progress on image recognition since the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012 [18]. These deep CNN models directly learn discriminative visual representations from raw images in an end-to-end manner. Owing to the available large scale labeled datasets (e.g., ImageNet [19], Places [3]) and powerful computing resources (e.g., GPUs and parallel computing cluster), several successful deep architectures have been developed to advance the state of the art of image recognition, including AlexNet [1], VGGNet [20], GoogLeNet [21], and ResNet [2]. Compared with conventional hand-crafted representations, CNNs are equipped with rich modeling power and capable of learning more abstractive and robust visual representations. However, the training of CNNs requires large number of well-labeled samples and long training time even with GPUs. In addition, CNNs are often treated as black boxes for image recognition, and it is still hard to well understand these deeply-learned representations.

In this paper we aim to present a hybrid visual representation for image recognition, which shares the merits of hand-crafted representation (e.g., simplicity and interpretability) and deeply-learned representation (e.g., robustness

Manuscript received August 29, 2016; revised December 24, 2016; accepted January 30, 2017. Date of publication February 8, 2017; date of current version March 9, 2017. This work was supported in part by the National Key Research and Development Program of China under Grant 2016YFC1400704, in part by the National Natural Science Foundation of China under Grant U1613211, Grant 61633021, and Grant 61502470, in part by the Shenzhen Research Program under Grant JCYJ20160229193541167, and in part by the External Cooperation Program of BIC, Chinese Academy of Sciences, under Grant 172644KYSB20160033. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Dacheng Tao. *(Corresponding author: Limin Wang.)*

Z. Wang was with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China. He is now with the Computational Vision Group, University of California at Irvine, Irvine, CA USA (e-mail: buptwangzhe2012@gmail.com).

L. Wang is with the Computer Vision Laboratory, ETH Zürich, Zürich 8092, Switzerland (e-mail: 07wanglimin@gmail.com).

Y. Wang is with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China (e-mail: yl.wang@siat.ac.cn).

B. Zhang is with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China, and also with Tongji University, Shanghai 200092, China (e-mail: 1023zhangbowen@tongji.edu.cn).

Y. Qiao is with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China, and also with The Chinese University of Hong Kong, Hong Kong (e-mail: yu.qiao@siat.ac.cn).

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.2017.2666739

and effectiveness). Specifically, we first propose a patch-level architecture to model the visual appearance of a small region, called as PatchNet, which is trained to maximize the performance of image-level classification. This weakly supervised training scheme not only enables PatchNets to yield effective representations for local patches, but also allows for efficient PatchNet training with the help of global semantic labels. In addition, we construct a semantic codebook and propose a new encoding scheme, called as *vector of semantically aggregated descriptors* (VSAD), by exploiting the prediction score of PatchNet as posterior probability over semantic codewords. This VSAD encoding scheme overcomes the difficulty of dictionary learning in conventional methods like Fisher vector and VLAD, and produce more semantic and discriminative global representations. Moreover, we design a simple yet effective algorithm to select a subset of discriminative and representative codewords. This subset of codewords allows us to further compress the VSAD representation and reduce the computational cost on the large-scale dataset.

To verify the effectiveness of our proposed representations (i.e., PatchNet and VSAD), we focus on the problem of scene recognition. Specifically, we learn two PatchNets on two large-scale datasets, namely ImageNet [19] and Places [3], and the resulted PacthNets denoted as **object-PatchNet** and **scene-PatchNet**, respectively. Due to the different training datasets, object-PatchNet and scene-PatchNet exhibit different but complementary properties, and allows us to develop more effective visual representations for scene recognition. As scene can be viewed as a collection of objects arranged in a certain spatial layout, we exploit the semantic probability of object-PatchNet to aggregate the features of the global pooling layer of scene-PatchNet. We conduct experiments on two standard scene recognition benchmarks (MIT Indoor67 [22] and SUN397 [23]) and the results demonstrate the superior performance of our VSAD representation to the current state-of-the-art approaches. Moreover, we comprehensively study different aspects of PatchNets and VSAD representations, aiming to provide more insights about our proposed new image representations for scene recognition.

The main contributions of this paper are summarized as follows:

- We propose a patch-level CNN to model the appearance of local patches, called as PatchNet. PatchNet is trained in a weakly-supervised manner simply with image-level supervision. Experimental results imply that PatchNet is more effective than classical image-level CNNs to extract semantic and discriminative features from local patches.
- We present a new image representation, called as VSAD, which aggregates the PatchNet features from local patches with semantic probabilities. VSAD differs from previous CNN+FV for image representation on how to extract local features and how to estimate posterior probabilities for features aggregation.
- We exploit VSAD representation for scene recognition and investigate its complementarity to global CNN representations and traditional feature encoding methods. Our method achieves the state-of-the-art performance on the two challenging scene recognition benchmarks, i.e., MIT Indoor67 (86.2%) and SUN397 (73.0%), which outperforms previous methods with a large margin. The code of our method and learned models are made available to facilitate the future research on scene recognition.[1]

The remainder of this paper is organized as follows. In Section II, we review related work to our method. After this, we briefly describe the Fisher vector representation to well motivate our method in Section III. We present the PatchNet architecture and VSAD representation in Section IV and propose a codebook selection method in Section V. Then, we present our experimental results, verify the effectiveness of PatchNet and VSAD, and give a detailed analysis of our method in Section VI. Finally, Section VII concludes this work.

II. Related Work

In this section we review related methods to our approach from the aspects of visual representation and scene recognition.

A. Visual Representation

Image recognition has received extensive research attention in past decades [1]–[4], [13], [16], [24]–[30]. Early works focused on Bag of Visual Word representation [11], [12], where local features were quantinized into a single word and a global histogram was utilized to summarize the visual content. Soft assigned encoding [31] method was introduced to reduce the information loss during quantization. Sparse coding [13] and Locality-constrained linear coding [32] was proposed to exploit sparsity and locality for dictionary learning and feature encoding. High dimensional encoding methods, such as Fisher vector [16], VLAD [14], and Super Vector [24], was presented to reserve high-order information for better recognition. Our VSAD representation is mainly inspired by the encoding method of Fisher vector and VLAD, but differs in aspects of codebook construction and aggregation scheme.

Dictionary learning is another important component in image representation and feature encoding methods. Traditional dictionary (codebook) is mainly based on unsupervised learning algorithms, including k-means [11], [12], Gaussian Mixture Models [16], k-SVD [33]. Recently, to enhance the discriminative power of dictionary, several algorithms were designed for supervised dictionary learning [34]–[36]. Boureau *et al.* [34] proposed a supervised dictionary learning method for sparse coding in image classification. Peng *et al.* [36] designed a end-to-end learning to jointly optimize the dictionary and classifier weights for the encoding method VLAD. Sydorov *et al.* [35] presented a deep kernel framework and learn the parameters of GMM in a supervised way. The supervised GMMs were exploited for Fisher vector encoding. Wang *et al.* [37] proposed a set of good practices to enhance the codebook of VLAD representation. Unlike these dictionary learning method, the learning of our semantic codebook is weakly supervised with image-level labels transferred from the ImageNet dataset. We explicitly exploit object

[1] https://github.com/wangzheallen/vsad

semantics in the codebook construction within our PatchNet framework.

Recently Convolutional Neural Networks (CNNs) [17] have enjoyed great success for image recognition and many effective network architectures have been developed since the ILSVRC 2012 [18], such as AlexNet [1], GoogLeNet [21], VGGNet [20], and ResNet [2]. These powerful CNN architectures have turned out to be effective for capturing visual representations for large-scale image recognition. In addition, several new optimization algorithms have been also proposed to make the training of deep CNNs easier, such as Batch Normalization [38], and Relay Back Propagation [4]. Meanwhile, some deep learning architectures have been specifically designed for scene recognition [39]. Wang *et al.* [39] proposed a multi-resolution CNN architecture to capture different levels of information for scene understanding and introduced a soft target to disambiguate similar scene categories. Our PatchNet is a customized patch-level CNN to model local patches, while those previous CNNs aim to capture the image-level information for recognition.

There are several works trying to combine the encoding methods and deeply-learned representations for image and video recognition [40]–[45]. These works usually were composed of two steps, where CNNs were utilized to extract descriptors from local patches and these descriptors were aggregated by traditional encoding methods. For instance, Gong *et al.* [43] employed VLAD to encode the activation features of fully-connected layers for image recognition. Dixit *et al.* [42] designed a semantic Fisher vector to aggregate features from multiple layers (both convolutional and fully-connected layers) of CNNs for scene recognition. Guo *et al.* [40] developed a locally-supervised training method to optimize CNN weights and proposed a hybrid representation for scene recognition. Arandjelovic *et al.* [44] developed a new generalized VLAD layer to train an end-to-end network for instance-level recognition. Our work is along the same research line of combining conventional and deep image representations. However, our method differs from these works on two important aspects: (1) we design a new PatchNet architecture to learn patch-level descriptors in a weakly supervised manner. (2) we develop a new aggregating scheme to summarize local patches (VSAD), which overcomes the limitation of unsupervised dictionary learning, and makes the final representation more effective for scene recognition.

B. Scene Recognition

Scene recognition is an important task in computer vision research [3], [39], [46]–[50] and has many applications such as event recognition [5], [6] and action recognition [51]–[53]. Early methods made use of hand-crafted global features, such as GIST [46], for scene representation. Global features are usually extracted efficiently to capture the holistic structure and content of the entire image. Meanwhile, several local descriptors (e.g., SIFT [8], HOG [9], and CENTRIST [47]) have been developed for scene recognition within the frameworks of Bag of Visual Words (e.g., Histogram Encoding [12], Fisher vector [15]). These representations leveraged information of local regions for scene recognition and obtained good performance in practice. However, local descriptors only exhibit limited semantics and so several mid-level and high-level representations have been introduced to capture the discriminative parts of scene content (e.g., mid-level patches [54], distinctive parts [55], object bank [50]). These mid-level and high-level representations were usually discovered in an iterative way and trained with a discriminative SVM. Recently, several structural models were proposed to capture the spatial layout among local features, scene parts, and containing objects, including spatial pyramid matching [56], deformable part based model [57], reconfigurable models [58]. These structural models aimed to describe the structural relation among visual components for scene understanding.

Our PatchNet and VSAD representations is along the research line of exploring more semantic parts and objects for scene recognition. Our method has several important differences from previous scene recognition works: (1) we utilize the recent deep learning techniques (PatchNet) to describe local patches for CNN features and aggregate these patches according to their semantic probabilities. (2) we also explore the general object and scene relation to discover a subset of object categories to improve the representation capacity and computational efficiency of our VSAD.

III. Motivating PatchNets

In this section, we first briefly revisit Fisher vector method. Then, we analyze the Fisher vector representation to well motivate our approach.

A. Fisher Vector Revisited

Fisher vector [16] is a powerful encoding method derived from Fisher kernel and has proved to be effective in various tasks such as object recognition [15], scene recognition [55], and action recognition [59], [60]. Like other conventional image representations, Fisher vector aggregates local descriptors into a global high-dimensional representation. Specifically, a Gaussian Mixture Model (GMM) is first learned to describe the distribution of local descriptors. Then, the GMM posterior probabilities are utilized to softly assign each descriptor to different mixture components. After this, the first and second order differences between local descriptors and component center are aggregated in a weighted manner over the whole image. Finally, these difference vectors are concatenated together to yield the high-dimensional Fisher vector ($2KD$), where K is the number of mixture components and D is the descriptor dimension.

B. Analysis

From the above description about Fisher vector, there are two key components in this aggregation-based representation:

- The first key element in Fisher vector encoding method is the local descriptor representation, which determines the feature space to learn GMMs and aggregate local patches.
- The generative GMM is the second key element, as it defines a soft partition over the feature space and determines how to aggregate local descriptors according to this partition.

TABLE I

PATCHNET ARCHITECTURE: WE ADAPT THE SUCCESSFUL INCEPTION V2 [38] STRUCTURE TO THE DESIGN OF PATCHNET, WHICH TAKES A 128×128 IMAGE REGION AS INPUT AND OUTPUTS ITS SEMANTIC PROBABILITY. IN EXPERIMENT, WE ALSO STUDY THE PERFORMANCE OF PatchNet WITH VGGNet16 [20] STRUCTURE

Layer	conv1	conv2	Inception3a	Inception3b	Inception3c	Inception4a	Inception4b
Feature map size	64×64	32×32	16×16	16×16	8×8	8×8	8×8
Stride	2	1	1	1	2	1	1
Channel	64	192	256	320	576	576	576
Layer map	Inception4c	Inception4d	Inception4e	Inception5a	Inception5b	global Avg	prediction
Feature map size	8×8	8×8	4×4	4×4	4×4	1×1	1×1
Stride	1	1	2	1	1	1	1
Channel	608	608	1056	1024	1024	1024	1000

Conventional image representations rely on hand-crafted features, which may not be optimal for classification tasks, while recent methods [42], [43] choose image-level deep features to represent local patches, which are not designed for patch description by its nature. Additionally, dictionary learning (GMM) method heavily relies on the design of patch descriptor and its performance is highly correlated with the choice of descriptor. Meanwhile, dictionary learning is often based on unsupervised learning algorithms and sensitive to the initialization. Moreover, the learned codebook lacks semantic property and it is hard to interpret and visualize these mid-level codewords. These important issues motivate us to focus on two aspects to design effective visual representations: (1) *how to describe local patches with more powerful and robust descriptors;* and (2) *how to aggregate these local descriptors with more semantic codebooks and effective schemes*.

TABLE II

SUMMARY OF NOTATIONS USED IN OUR METHOD

$\mathbf{x}$	local patch sampled from images
$\mathbf{z}$	latent variable to model local patches
$\mathbf{f}$	patch-level descriptor extracted with PatchNet
$\mathbf{p}$	patch-level semantic probability extracted PatchNet
$\mathcal{X}$	a set of local patches
$\mathcal{F}$	a set of patch descriptors
$\mathcal{P}$	a set of semantic probability distributions
$\mathbf{p}_{image}$	image-level semantic probability
$\mathbf{p}_{category}$	semantic probability over images from a category
$\mathbf{p}_{data}$	semantic probability over all images from a dataset

IV. WEAKLY SUPERVISED PATCHNETS

In this section we describe the PatchNet architecture to model the appearance of local patches and aggregate them into global representations. First, we introduce the network structure of PatchNet. Then, we describe how to use learned PatchNet models to describe local patches. Finally, we develop a semantic encoding method (VSAD) to aggregate these local patches, yielding the image-level representation.

A. PatchNet Architectures

The success of aggregation-based encoding methods (e.g., Fisher vector [15]) indicates that the patch descriptor is a kind of rich representation for image recognition. A natural question arises that whether we are able to model the appearance of these local patches with a deep architecture, that is trainable in an end-to-end manner. However, the current large-scale datasets (e.g., ImageNet [19], Places [3]) simply provide the image-level labels without the detailed annotations of local patches. Annotating every patch is time-consuming and sometimes could be ambiguous as some patches may contain part of objects or parts from multiple objects. To handle these issues, we propose a new patch-level architecture to model local patches, which is still trainable with the image-level labels.

Concretely, we aim to learn the patch-level descriptor directly from raw RGB values, by classifying them into predefined semantic categories (e.g., object classes, scene classes). In practice, we apply the image-level label to each randomly selected patch from this image, and utilize this transferred label as supervision signal to train the PatchNet. In this training setting, we do not have the detailed patch-level annotations and exploit the image-level supervision signal to learn patch-level classifier. So, the PatchNet could be viewed as a kind of weakly supervised network. We find that although the image-level supervision may be inaccurate for some local patches and the converged training loss of PatchNet is higher than that of image-level CNN, it is still able to learn effective representation to describe local patches and reasonable semantic probability to aggregate these local patches.

Specifically, our proposed PatchNet is a CNN architecture taking small patches (128×128) as inputs. We adapt two famous image-level structures (i.e., VGGNet [20] and Inception V2 [38]) for the PatchNet design. The Inception based architecture is illustrated in Table I, and its design is inspired by the successful Inception V2 model with batch normalization [38]. The network starts with 2 convolutional and max pooling layers, subsequently has 10 inception layers, and ends with a global average pooling layer and fully connected layer. Different from the original Inception V2 architecture, our final global average pooling layer has a size of 4×4 due to the smaller input size (128×128). The output of PatchNet is to predict the semantic labels specified by different datasets (e.g., 1,000 object classes on the ImageNet dataset, 205 scene classes on the Places dataset). In practice, we train two kinds of PatchNets: **object-PatchNet** and **scene-PatchNet**, and the training details will be explained in subsection VI-A.

1) Discussion: Our PatchNet is a customized network for patch modeling, which differs from the traditional CNN architectures on two important aspects: (1) our network is a patch-level structure and its input is a smaller image region (128×128) rather than a image (224×224), compared with those image-level CNNs [1], [20], [21]; (2) our network is trained in a weakly supervised manner, where we directly treat the image-level labels as patch-level supervision information. Although this strategy is not accurate, we empirically demonstrate that it still enables our PacthNet to learn more effective representations for aggregation-based encoding methods in our experiments.

B. Describing Patches

After the introduction of PatchNet architecture, we are ready to present how to describe local patches with PatchNet. The proposed PatchNet is essentially a patch-level discriminative model, which aims to map these local patches from raw RGB space to a semantic space determined by the supervision information. PatchNet is composed of a set of standard convolutional and pooling layers, that process features with more abstraction and downsample spatial dimension to a lower resolution, capturing full content of local patches. During this procedure, PatchNet hierarchically extracts multiple-level representations (hidden layers, denoted as $\mathbf{f}$) from raw RGB values of patches, and eventually outputs the probability distribution over semantic categories (output layers, denoted as $\mathbf{p}$).

The final semantic probability $\mathbf{p}$ is the most abstract and semantic representation of a local patch. Compared with the semantic probability, the hidden layer activation features $\mathbf{f}$ are capable of containing more detailed and structural information. Therefore, multiple-level representations $\mathbf{f}$ and semantic probability $\mathbf{p}$ could be exploited in two different manners: *describing* and *aggregating* local patches. In our experiments, we use the activation features of the last hidden layer as the patch-level descriptors. Furthermore, in practice, we could even try the combination of activation features $\mathbf{f}$ and semantic probability $\mathbf{p}$ from different PatchNets (e.g., object-PatchNet, scene-PatchNet). This flexible scheme decouples the correlation between local descriptor design and dictionary learning, and allows us to make best use of different PatchNets for different purposes according to their own properties.

C. Aggregating Patches

After having introduced the architecture of PatchNet to describe the patches with multiple-level representations $\mathbf{f}$ in the previous subsection, we present how to aggregate these patches with semantic probability $\mathbf{p}$ of PatchNet in this subsection. As analyzed in Section III, aggregation-based encoding methods (e.g., Fisher vector) often rely on generative models (e.g., GMMs) to calculate the posterior distribution of a local patch, indicating the probability of belonging to a codeword. In general, the generative model often introduces latent variables $\mathbf{z}$ to capture the underline factors and the complex distribution of local patches $\mathbf{x}$ can be obtained by marginalization over latent variables $\mathbf{z}$ as follows:

$$p(\mathbf{x}) = \sum_{\mathbf{z}} p(\mathbf{x}|\mathbf{z})p(\mathbf{z}). \tag{1}$$

However, from the view of aggregation process, only the posterior probability $p(\mathbf{z}|\mathbf{x})$ are needed to assign a local patch $\mathbf{x}$ to these learned codewords in a soft manner. Thus, it will not be necessary to use generative model $p(\mathbf{x})$ for estimating $p(\mathbf{z}|\mathbf{x})$, and we can directly calculate $p(\mathbf{z}|\mathbf{x})$ with our proposed PatchNet. Directly modeling posterior probability with PatchNet exhibits two advantages over traditional generative models:

- The estimation of $p(\mathbf{x})$ is a non-trivial task and the learning of generative models (e.g., GMMs) is sensitive to the initialization and may converge to a local minimum. Directly modeling $p(\mathbf{z}|\mathbf{x})$ with PatchNets can avoid this difficulty by training on large-scale supervised datasets.
- Prediction scores of PatchNet correspond to semantic categories, which is more informative and semantic than that of the original generative model (e.g., GMMs). Utilizing this semantic posterior probability enables the final representation to be interpretable.

1) Semantic Codebook: We first describe the semantic codebook construction based on the semantic probability extracted with PatchNet. In particular, given a set of local patches $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$, we first compute their semantic probabilities with PatchNet, denoted as $\mathcal{P} = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_N\}$. We also use PatchNet to extract patch-level descriptors $\mathcal{F} = \{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_N\}$. Finally, we generate semantic mean (center) for each codeword as follows:

$$\mu_k = \frac{1}{N_k} \sum_{i=1}^{N} \mathbf{p}_i^k \mathbf{f}_i, \tag{2}$$

where $\mathbf{p}_i^k$ is the k^{th} dimension of $\mathbf{p}_i$, and N_k is calculated as follows:

$$N_k = \sum_{i=1}^{N} \mathbf{p}_i^k, \quad \pi_k = \frac{N_k}{N}. \tag{3}$$

We can interpret N_k as the prior distribution over the semantic categories and μ_k as the category template in this feature space $\mathbf{f}$. Meanwhile, we can calculate the semantic covariance for each codeword by the following formula:

$$\Sigma_k = \frac{1}{N_k} \sum_{i=1}^{N} \mathbf{p}_i^k \ (\mathbf{f}_i - \mu_k)(\mathbf{f}_i - \mu_k)^\top. \tag{4}$$

The semantic mean and covariance in Equation (2) and (4) constitute our semantic codebook, and will be exploited to semantically aggregate local descriptors in the next paragraph.

2) VSAD: After the description of PatchNet and semantic codebook, we are able to develop our hybrid visual representations, namely *vector of semantically aggregating descriptor* (VSAD). Similar to Fisher vector [15], given a set of local patches with descriptors $\{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_T\}$, we aggregate both

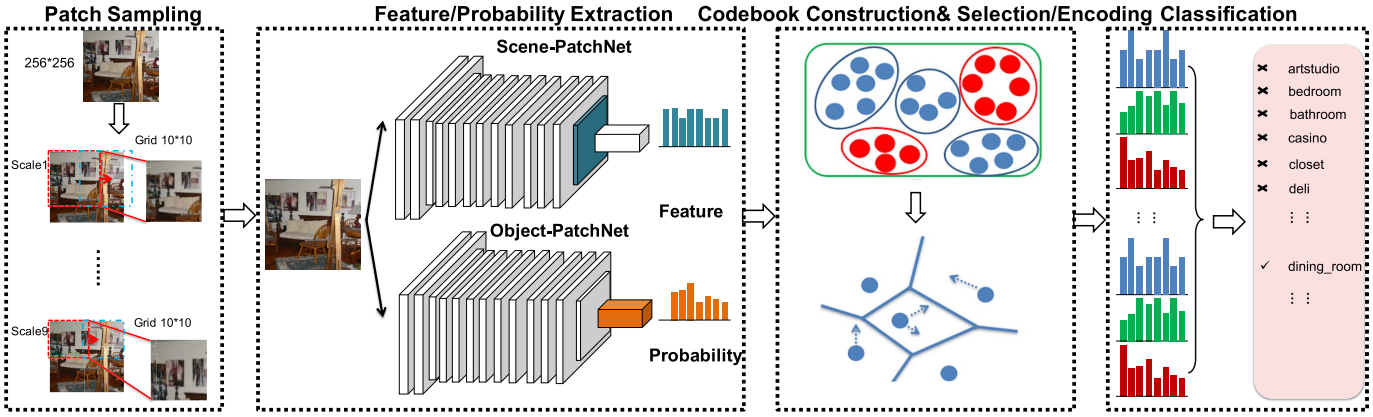


Fig. 1. **Pipeline of our method**. We first densely sample local patches in a multi-scale manner. Then, we utilize two kinds of PatchNets to describe each patch (Scene-PatchNet feature) and aggregate these patches (Object-PatchNet probability). Based on our learned semantic codebook, these local patches are aggregated into a global representation with VSAD encoding scheme. Finally, these global representations are exploited for scene recognition with a linear SVM.

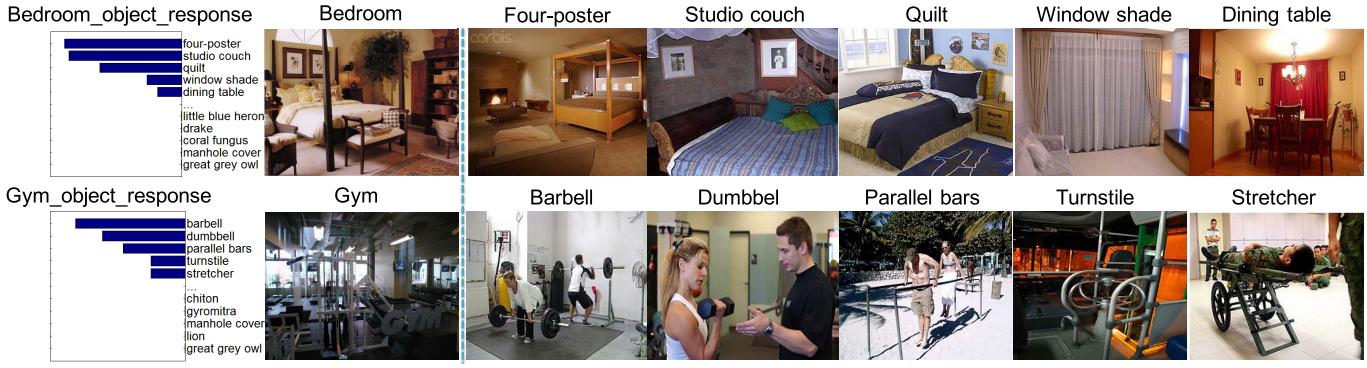


Fig. 2. **Illustration of scene-object relationship**. The first row is the *Bedroom* scene with its top 5 most likely object classes. Specifically, we feed all the training image patches of the *Bedroom* scene into our PatchNet. For each object category, we sum over the conditional probability over all training patches as the response for this object. The results are shown in the 1st column. We then show five object classes (top 5 objects) for the *Bedroom* scene (the second to the sixth column). The second row is an illustration for the *Gym* scene, which is a similar case to *Bedroom*.

first order and second order information of local patches with respect to semantic codebook as follows:

$$\mathcal{S}_k = \frac{1}{\sqrt{\pi_k}} \sum_{t=1}^{T} \mathbf{p}_t^k \left(\frac{\mathbf{f}_t - \mu_k}{\sigma_k} \right), \tag{5}$$

$$\mathcal{G}_k = \frac{1}{\sqrt{\pi_k}} \sum_{t=1}^{T} \mathbf{p}_t^k \left[\frac{(\mathbf{f}_t - \mu_k)^2}{\sigma_k^2} - 1 \right], \tag{6}$$

where $\{\pi, \mu, \sigma\}$ is semantic codebook defined above, $\mathbf{p}$ is the semantic probability calculated from PatchNet, $\mathcal{S}$ and $\mathcal{G}$ are first and second order VSAD, respectively. Finally, we concatenate these sub-vectors from different codewords to form our VSAD representation: $[\mathcal{S}_1, \mathcal{G}_1, \mathcal{S}_2, \mathcal{G}_2, \cdots, \mathcal{S}_K, \mathcal{G}_K]$.

V. Codeword Selection for Scene Recognition

In section we take scene recognition as a specific task for image recognition and utilize object-PatchNet for semantic codebook construction and VSAD extraction. Based on this setting, we propose an effective method to discover a set of discriminative object classes to compress VSAD representation. It should be noted that our selection method is general and could be applied to other relevant tasks and PatchNets.

Since our semantic codebook is constructed based on the semantic probability $\mathbf{p}$ of object-PatchNet, the size of our codebook is equal to the number of object categories from our PatchNet (i.e., 1000 objects in ImageNet). However, this fact may reduce the effectiveness of our VSAD representation due to the following reasons:

- Only a few object categories in ImageNet are closely related with scene category. In this case, many object categories in our semantic codebook are redundant. We here use the *Bedroom* and *Gym* scene classes (from MIT Indoor67 [22]) as an illustration for scene-object relationship. As shown in Figure 2, we can see that the *Bedroom* scene class most likely contains the object classes *Four-poster*, *Studio couch*, *Quilt*, *Window shade*, *Dining table*. The *Gym* scene class is a similar case. Furthermore, we feed all the training patches of MIT Indoor 67 into our object-PatchNet. For each object category, we sum over the conditional probability of all the training patches as the response for this object. The result in Figure 3 indicates that around 750 categories of 1000 are not activated. Hence, the redundance using 1,000 object categories is actually large.

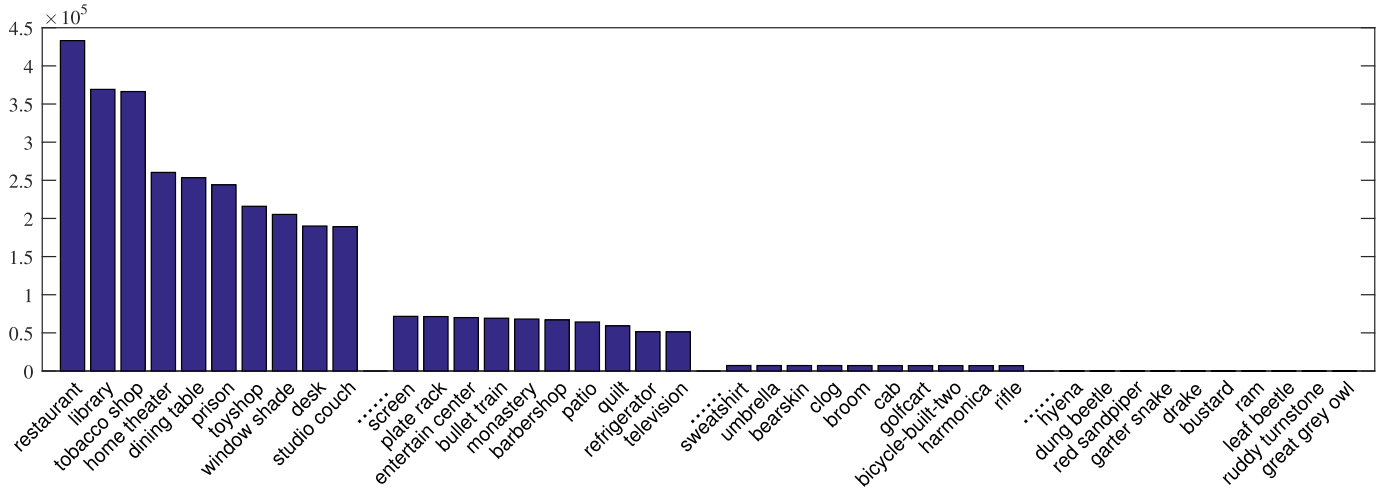


Fig. 3. **Illustration of the object responses in the object-PatchNet**. Specifically, we feed all the training patches (MIT Indoor 67) into our object-PatchNet, and obtain the corresponding probability distribution for each patch. For each object category, we use the sum of probabilities over all the training patches as the response of this object category. Then we sort the responses of all the object categories in a descent order. For visual clarity, we here show four typical groups with high (from *restaurant* to *studio couch*), moderate (from *screen* to *television*), minor (from *sweatshirt* to *rifle*), and low responses (from *hyena* to *great grey owl*). We can see that the groups with the minor and low response (the response rank of these objects: around 250 to 1000) make very limited contribution to the whole scene dataset. Hence, we should design our selection strategy to discard them to reduce the redundance of our semantic codebook.

- From the computational perspective, the large size of codebook will prohibit the application of VSAD on large-scale datasets due to the huge consumption of storage and memory. Therefore, it is also necessary to select a subset of codewords (object categories) to compress the VSAD representation and improve the computing efficiency.

Hence, we propose a codeword selection strategy as follows to enhance the efficiency of our semantic codebook and improve the computation efficiency of our VSAD representation. Specifically, we take advantage of the scene-object relationship to select K classes of 1000 ImageNet objects for our semantic codebook generation. **First**, the probability vector $\mathbf{p}_{patch}$ of the object classes for each training patch is obtained from the output of our PatchNet. We then compute the response of the object classes for each training image $\mathbf{p}_{image}$, each scene category $\mathbf{p}_{category}$ and the whole training data $\mathbf{p}_{data}$

$$\mathbf{p}_{image} = \sum\nolimits_{patch \in image} \mathbf{p}_{patch} \tag{7}$$

$$\mathbf{p}_{category} = \sum\nolimits_{image \in category} \mathbf{p}_{image} \tag{8}$$

$$\mathbf{p}_{data} = \sum\nolimits_{catrgory \in data} \mathbf{p}_{category} \tag{9}$$

Second, we rank p_{data} in the descending order and select $2K$ object classes (with top $2K$ highest responses). We denote the resulting object set as $\mathcal{O}_{data} = \{o_j\}_{j=1}^{2K}$. **Third**, for each scene category, we rank $p_{category}$ in the descending order and select T object classes (with top T highest responses). Then we collect the object classes for all the scene categories together, and delete the duplicate object classes. We denote the object set as $\mathcal{O}_{category} = \{o_i\}_{i=1}^{M}$, where M is the number of object classes in $\mathcal{O}_{category}$. **Finally**, the intersection of $\mathcal{O}_{category}$ and $\mathcal{O}_{data}$ is used as the selected object class set, i.e., $\mathcal{O} \leftarrow \mathcal{O}_{category} \cap \mathcal{O}_{data}$. To constrain the number of object classes as the predefined K, we can gradually increase T (when selecting $\mathcal{O}_{category}$), starting from one. Additionally, to speed up the selection procedure, we choose $2K$ as the size of $\mathcal{O}_{data}$. Note that, our selected object set O is the intersection of $\mathcal{O}_{category}$ and $\mathcal{O}_{data}$. In this case, the selected object classes not only contain the general characteristics of the entire scene dataset, but also the specific characteristics of each scene category. Consequentially, this selection strategy enhances the discriminative power of our semantic codebook and VSAD representations, yet is still able to reduce the computational cost.

VI. Experiments

In this section we evaluate our method on two standard scene recognition benchmarks to demonstrate its effectiveness. First, we introduce the evaluation datasets and the implementation details of our method. Then, we perform exploration experiments to determine the important parameters of the VSAD representation. Afterwards, we comprehensively study the performance of our proposed PatchNets and VSAD representations. In addition, we also compare our method with other state-of-the-art approaches. Finally, we visualize the semantic codebook and the scene categories with the most performance improvement.

A. Evaluation Datasets and Implementation Details

Scene recognition is a challenging task in image recognition, due to the fact that scene images of the same class exhibit large intra-class variations, while images from different categories contain small inter-class differences. Here, we choose this challenging problem of scene recognition as the evaluation task to demonstrate the effectiveness of our proposed PatchNet architecture and VSAD representation. Additionally, scene image can be viewed as a collection of objects arranged in the certain layout, where the small patches may contain rich

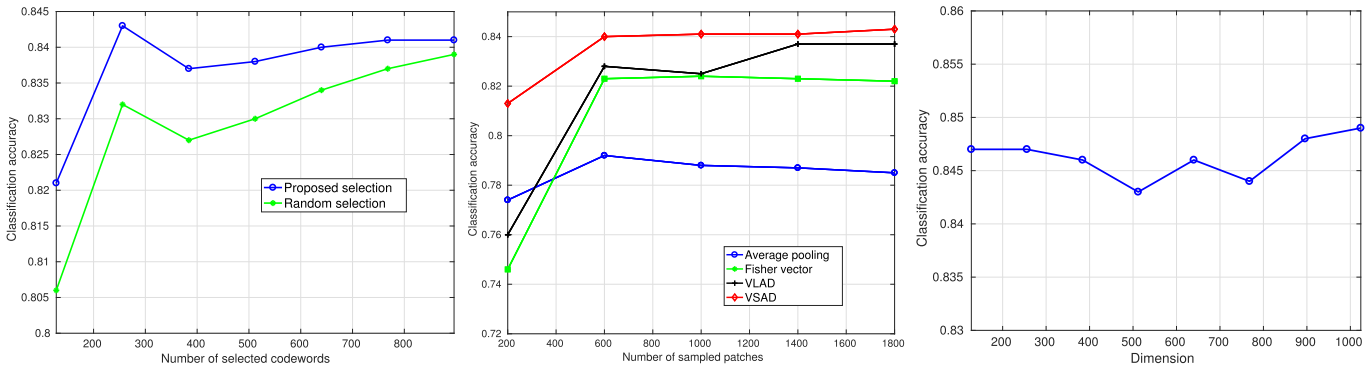


Fig. 4. **Exploration study on the MIT Indoor67 dataset**. Left: performance comparison of different codebook selection methods; Middle: performance comparison of different numbers of sampled patches; Right: performance comparison of different descriptor dimension reduced by PCA.

object information and can be effectively described by our PatchNet. Thus scene recognition is more suitable to evaluate the performance of VSAD representation.

1) Evaluation Datasets: In our experiment, we choose two standard scene recognition benchmarks, namely MIT Indoor67 [22] and SUN397 [23]. The MIT Indoor67 dataset contains 67 indoor-scene classes and has 15,620 images in total. Each scene category contains at least 100 images, where 80 images are for training and 20 images for testing. The SUN397 dataset is a larger scene recognition dataset, including 397 scene categories and 108,754 images, where each category also has at least 100 images. We follow the standard evaluation from the original paper [23], where each category has 50 images for training and 50 images for testing. Finally, the average classification accuracy over 10 splits is reported.

2) Implementation Details of PatchNet Training: In our experiment, to fully explore the modeling power of PatchNet, we train two types of PatchNets, namely **scene-PatchNet** and **object-PatchNet** with the MPI extension [61] of Caffe toolbox [62]. The scene-PatchNet is trained on the large-scale Places dataset [3], and the object-PatchNet is learned from the large-scale ImageNet dataset [19]. The Places dataset contains around 2,500,000 images and 205 scene categories and the ImageNet dataset has around 1,300,000 images and 1,000 object categories. We train both PatchNets from scratch on these two-large scale datasets. Specifically, we use the stochastic gradient decent (SGD) algorithm to optimize the model parameters, where momentum is set as 0.9 and batch size is set as 256. The learning rate is initialized as 0.01 and decreased to its $\frac{1}{10}$ every K iterations. The whole learning process stops at $3.5K$ iterations. K is set as 200,000 for the ImageNet dataset and 350,000 for the Places dataset. To reduce the effect of over-fitting, we adopt the common data augmentation techniques. We first resize each image into size of 256×256. Then we randomly crop a patch of size $s \times s$ from each image, where $s \in \{64, 80, 96, 112, 128, 144, 160, 176, 192\}$. Meanwhile, these cropped patches are horizontally flipped randomly. Finally, these cropped image regions are resized as 128×128 and fed into PatchNet for training. The object-PatchNet achieves the recognition performance of **85.3**% (top-5 accuracy) on the ImageNet dataset and the scene-PatchNet obtains the performance of **82.7**% (top-5 accuracy) on the Places dataset.

3) Implementation Details of Patch Sampling and Classifier: An important implementation detail in the VSAD representation is how to densely sample patches from the input image. To deal with the large intra-class variations existed in scene images, we design a *multi-scale dense sampling* strategy to select image patches. Specifically, like training procedure, we first resize each image to size of 256×256. Then, we sample patches of size $s \times s$ from the whole image in the grid of 10×10. Sizes s of these sampled patches range from $\{64, 80, 96, 112, 128, 144, 160, 176, 192\}$. These sampled image patches also go under horizontal flipping for further data augmentation. Totally, we have 9 different scales and each scale we sample 200 patches (10×10 grid and 2 horizontal flips). Normalization and recognition classifier are other important factors for all encoding methods (i.e., average pooling, VLAD, Fisher vector, and VSAD). In our experiment, the image-level representation is signed-square-rooted and L2-normalized for all encoding methods. For classification, we use a linear SVM (C=1) trained in the one-vs-all setting. The final predicted class is determined by the maximum score of different binary SVM classifiers.

B. Exploration Study

In this subsection we conduct exploration experiments to determine the parameters of important components in our VSAD representation. First, we study the performance of our proposed codeword selection algorithm and determine how many codewords are required to construct efficient VSAD representation. Then, we study the effectiveness of proposed multi-scale sampling strategy and determine how many scales are needed for patch extraction. Afterwards, we conduct experiments to explore the dimension reduction of PatchNet descriptors. Finally, we study the influence of different network structures and compare Inception V2 with VGGNet16. In these exploration experiments, we choose scene-PatchNet to describe each patch (i.e., extracting descriptors $\mathbf{f}$), and object-PatchNet to aggregate patches (i.e., utilizing semantic probability $\mathbf{p}$). We perform this exploration experiment on the dataset of MIT Indoor67.

1) Exploration on Codeword Selection: We begin our experiments with the exploration of codeword selection. We propose a selection strategy to choose the number of object categories (the codewords of semantic codebook) in Section V. We report the performance of VSAD representation with different codebook sizes in the left of Figure 4. To speed up this exploration experiment, we use PCA to pre-process the patch descriptor $\mathbf{f}$ by reducing its dimension from 1,024 to 100. In our study, we compare the performance of our selection method with the random selection. As expected, our selection method outperforms the random selection, in particular when the number of selected codewords are small. Additionally, when selecting 256 codewords, we can already achieve a relatively high performance. Therefore, to keep a balance between recognition performance and computing efficiency, we fix the number of selected codewords as 256 in the remaining experiments.

2) Exploration on Multi-Scale Sampling Strategy: After the exploration of codeword selection, we investigate the performance of our proposed multiï¼Žscale dense sampling strategy for patch extraction. In this exploration study, we choose four types of encoding methods: (1) average pooling over patch descriptors $\mathbf{f}$, (2) Fisher vector, (3) VLAD, and (4) our proposed VSAD. We sample image patches from 1 scale to 9 scales, resulting in the number of patches from 200 to 1800. The experimental results are summarized in the middle of Figure 4. We notice that the performance of traditional encoding methods (i.e., Fisher vector, VLAD) is more sensitive to the number of sampled patches, while the performance of our proposed VSAD increases gradually as more patches are sampled. We analyze that the traditional encoding methods heavily rely on unsupervised dictionary learning (i.e., GMMs, k-means), whose training is unstable when the number of sampled patches is small. Moreover, we observe that our VSAD representation is still able to obtain high performance when only 200 patches are sampled, which again demonstrates the effectiveness of semantic codebook and VSAD representations. For real application, we may simply sample 200 patches for fast processing, but to fully reveal the representation capacity of VSAD, we crop image patches from 9 scales in the remaining experiments.

3) Exploration on Dimension Reduction: The dimension of scene-PatchNet descriptor $\mathbf{f}$ is relatively high (1,024) and it may be possible to reduce its dimension for VSAD representation. So we perform experiments to study the effect of dimension reduction on scene-PatchNet descriptor. The numerical results are reported in the right of Figure 4 and the performance difference is relatively small for different dimensions (the maximum performance difference is around 0.5%). We also see that PCA dimension reduction can not bring the performance improvement for VSAD representation, which is different from traditional encoding methods (e.g., Fisher vector, VLAD). This result could be explained by two possible reasons: (1) PatchNet descriptors are more discriminative and compact than hand-crafted features and dimension reduction may cause more information loss; (2) Our VSAD representation is based on the semantic codebook, which does not rely on any unsupervised learning methods (e.g., GMMs, k-means). Therefore de-correlating different dimensions of descriptors can not bring any advantage for semantic dictionary learning. Overall, in the case of fully exploiting the representation power of VSAD, we could keep the dimension of PatchNet descriptor as 1,024, and in the case of high computational efficiency, we could choose the dimension as 100 for fast processing speed and low dimensional representation.

TABLE III

COMPARISON OF DIFFERENT STRUCTURES FOR THE PatchNet DESIGN ON THE DATASET OF MIT INDOOR67

Descriptor $\mathbf{f}$	MIT Indoor67
scene-PatchNet (VGGNet16)+ average pooling	81.1
scene-PatchNet (Inception V2) + average pooling	78.5
scene-PatchNet (VGGNet16)+ VLAD	83.7
scene-PatchNet (Inception V2) + VLAD	83.9
scene-PatchNet (VGGNet16)+ Fisher vector	81.2
scene-PatchNet (Inception V2) + Fisher vector	83.6
scene-PatchNet (VGGNet16)+ VSAD	83.9
scene-PatchNet (Inception V2) + VSAD	84.9

4) Exploration on Network Architectures: We explore different network architectures to verify the effectiveness of PatchNet and VSAD representation on the MIT Indoor67 dataset. Specifically, we compare two network structures: VGGNet16 and Inception V2. The implementation details of VGGNet16 PatchNet are the same with those of Inception V2 PatchNet, as described in Section VI-A. We also train two kinds of PatchNets for VGGNet16 structure, namely object-PatchNet on the ImageNet dataset and scene-PatchNet on the Places dataset, where the top5 classification accuracy is 80.1% and 82.9%, respectively. As the last hidden layer (fc7) of VGGNet16 has a much higher dimension (4096), we decreases its dimension to 100 as the patch descriptor $\mathbf{f}$ for computational efficiency. For patch aggregating, we use the semantic probability from object-PatchNet, where we select the most 256 discriminative object classes. The experimental results are summarized in Table III and two conclusions can be drawn from this comparison. First, for both structures of VGGNet16 and Inception V2, our VSAD representation outperforms other three encoding methods. Second, the recognition accuracy of Inception V2 PatchNet is slightly better than that of VGGNet16 PatchNet, for all aggregation based encoding methods, including VLAD, Fisher vector, and VSAD. So, in the following experiment, we choose the Inception V2 as our PatchNet structure.

C. Evaluation on PatchNet Architectures

After exploring the important parameters of our method, we focus on verifying the effectiveness of PatchNet on patch modeling in this subsection. Our PatchNet is a patch-level architecture, whose hidden layer activation features $\mathbf{f}$ could be exploited to describe patch appearance and prediction probability $\mathbf{p}$ to aggregate these patches. In this subsection we compare two network architectures: image-level CNNs (ImageCNNs) and patch-level CNNs (PatchNets), and demonstrate the superior performance of PatchNet on describing and aggregating local patches on the dataset of MIT Indoor67.

TABLE IV

COMPARISON OF PATCHNET AND IMAGECNN FOR PATCH MODELING ON THE DATASET OF MIT INDOOR67

Descriptor $\mathbf{f}$	object-PatchNet $\mathbf{p}$	object-ImageCNN $\mathbf{p}$
scene-PatchNet (1,024D)	84.9	84.7
scene-PatchNet (100D)	84.3	84.0
scene-ImageCNN (1,024D)	83.8	83.4
scene-ImageCNN (100D)	83.6	83.1
objcet-PatchNet (1,024D)	79.6	79.4
object-PatchNet (100D)	79.5	79.3
object-ImageCNN (1,024D)	79.3	79.2
object-ImageCNN (100D)	79.1	78.7

For fair comparison, we also choose the Inception V2 architecture [38] as our ImageCNN structure, and following the similar training procedure to PatchNet, we learn the network weights on the datasets of ImageNet [19] and Places [3]. The resulted CNNs are denoted as **object-ImageCNN** and **scene-ImageCNN**. The main difference between PatchNet and ImageCNN is their receptive filed, where PatchNet operates on the local patches (128×128), while ImageCNN takes the whole image (224×224) as input. In this exploration experiment, we investigate four kinds of descriptors, including $\mathbf{f}$ extracted from scene-PatchNet, scene-ImageCNN, object-PatchNet, and object-ImageCNN. Meanwhile, we compare the descriptor $\mathbf{f}$ without dimension reduction (i.e., 1,024) and with dimension reduction to 100. For aggregating semantic probability $\mathbf{p}$, we choose two types of probabilities from object-PatchNet and object-ImageCNN respectively.

The experiment results are summarized in Table IV and several conclusions can be drawn as follows: (1) From the comparison between object network descriptors and scene network descriptors, we see that scene network descriptor is more suitable for recognizing the categories from MIT Indoor67, no matter which architecture and aggregating probability is chosen; (2) From the comparison between descriptors from image-level and patch-level architectures, we conclude that PatchNet is better than ImageCNN. This superior performance of descriptors from PatchNet indicates the effectiveness of training PatchNet for local patch description; (3) From the comparison between aggregating probabilities from PatchNet and ImageCNN, our proposed PatchNet architecture again outperforms the traditional image-level CNN, which implies the semantic probability from the PatchNet is more suitable for VSAD representation. Overall, we empirically demonstrate that our proposed PatchNet architecture is more effective for describing and aggregating local patches.

D. Evaluation on Aggregating Patches

In this subsection we focus on studying the effectiveness of PatchNet on aggregating local patches. We perform experiments with different types of descriptors and compare VSAD with other aggregation based encoding methods, including average pooling, Fisher vector (FV), and VLAD, on both datasets of MIT Indoor67 and SUN397.

1) Performance With SIFT Descriptors: We first verify the effectiveness of our VSAD representation by using the hand-crafted features (i.e., SIFT [8]). For each image, we extract the SIFT descriptors from image patches (in grid of 64×64, a stride of 16 pixels). These SIFT descriptors are square-rooted and then de-correlated by PCA processing, where the dimension is reduced from 128 to 80. We compare our VSAD with traditional encoding methods of VLAD [14] and Fisher vector [16]. For traditional encoding methods, we directly learn the codebooks with unsupervised learning methods (i.e., GMMs, k-means) based on SIFT descriptors, where the codebook size is set as 256. For our VSAD, we first resize the extracted patches of training images to 128×128. Then we feed them to the learned object-PatchNet and obtain their corresponding semantic probabilities $\mathbf{p}$. Based on the SIFT descriptors $\mathbf{f}$ and the semantic probabilities $\mathbf{p}$ of these training patches, we construct our semantic codebook and VSAD representations by Equation (2) and (6).

TABLE V

PERFORMANCE COMPARISON WITH SIFT DESCRIPTORS ON THE DATASETS OF MIT INDOOR67 AND SUN397

Method	MIT indoor67	SUN397
SIFT+VLAD	32.6	19.2
SIFT+FV	42.8	24.4
Dense-Multiscale-SIFT+VLAD+aug. [63]	53.3	-
Dense-Multiscale-SIFT+Fisher vector [63]	58.3	-
Dense-Multiscale-SIFT+Fisher vector [23]	-	38.0
SIFT+ VSAD	**60.8**	**40.3**

TABLE VI

PERFORMANCE COMPARISON WITH SCENE-PATCHNET DESCRIPTOR ON THE DATASETS OF MIT INDOOR67 AND SUN397

Method	MIT indoor67	SUN397
scene-PatchNet+average pooling	78.5	63.5
scene-PatchNet+Fisher vector	83.6	69.0
scene-PatchNet+VLAD	83.9	70.1
scene-PatchNet+VSAD	**84.9**	**71.7**

The experimental results are reported in Table V. We see that our VSAD significantly outperforms the traditional VLAD and Fisher vector methods on both datasets of MIT Indoor67 and SUN397. Meanwhile, we also list the performance of VLAD and Fisher vector with multi-scale sampled SIFT descriptors from previous works [23], [63]. Our VSAD from single-scale sampled patches is still better than the performance of traditional methods with multi-scale sampled patches, which demonstrates the advantages of semantic codebook and VSAD representations.

2) Performance With Scene-PatchNet Descriptors: After evaluating VSAD representation with SIFT descriptors, we are ready to demonstrate the effectiveness of our complete framework, i.e. *describing* and *aggregating* local patches with PatchNet. According to previous study, we choose the multi-scale dense sampling method (9 scales) to extract patches. For each patch, we extract the scene-PatchNet descriptor $\mathbf{f}$ and use the semantic probabilities $\mathbf{p}$ obtained from object-PatchNet to aggregate these descriptors.

We make comparison among the performance of VSAD, Average Pooling, Fisher vector, and VLAD. For fair comparison, we fix the dimension of PatchNet descriptor as 1,024 for all encoding methods, but de-correlate different dimensions to make GMM training easier. The numerical results are summarized in Table VI and our VSAD encoding method achieves the

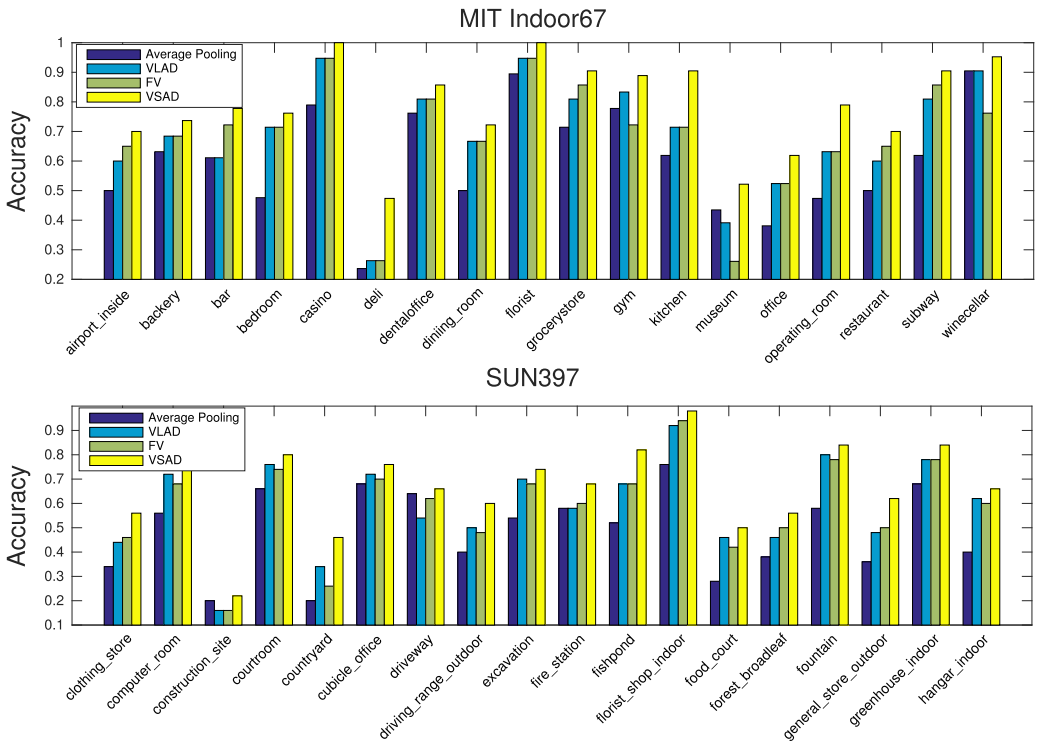


Fig. 5. Several categories with significant improvement on MIT Indoor67 and SUN397. These results show the strong ability of VSAD encoding for scene recognition.

TABLE VII

PERFORMANCE COMPARISON WITH CONCATENATED DESCRIPTOR (HYBRID-PATCHNET) FROM OBJECT-PATCHNET AND SCENE-PATCHNET ON THE DATASETS OF MIT INDOOR67 AND SUN397

Method	MIT indoor67	SUN397
hybrid-PatchNet+average pooling	80.6	65.7
hybrid-PatchNet+Fisher vector	82.6	68.4
hybrid-PatchNet+VLAD	84.9	70.9
hybrid-PatchNet+VSAD	**86.1**	**72.0**

best accuracy on both datasets of MIT Indoor67 and SUN397. Some more detailed results are depicted in Figure 5, where we show the classification accuracy on a number of scene categories from the MIT Indoor67 and SUN397. VSAD achieves a clear performance improvement over other encoding methods.

3) Performance With Hybrid-PatchNet Descriptors: Finally, to further boost the performance of VSAD representation and make comparison more fair, we extract two descriptors for each patch, namely descriptor from scene-PatchNet and object-PatchNet. We denote this fused descriptor as hybrid-PatchNet descriptor. For computational efficiency, we first decrease the dimension of each descriptor to 100 for feature encoding. Then, we concatenate the image-level representation from two descriptors as the final representation. As shown in Table VII, our VSAD encoding still outperforms other encoding methods, including average pooling, VLAD, Fisher vector, with this new hybrid-PatchNet descriptor, which further demonstrates the effectiveness of PatchNet for describing and aggregating local patches.

E. Comparison With the State of the Art

After the exploration of different components of our proposed framework, we are ready to present our final scene recognition method in this subsection and compare its performance with these sate-of-the-art methods. In our final recognition method, we choose the VSAD representations by using scene-PatchNet to describe each patch (**f**) and object-PatchNet to aggregate these local pathces (**p**). Furthermore, we combine our VSAD representation, with Fisher vector and deep features of Place205-VGGNet-16 [75] to study the complementarity between them, and achieve the new state of the art on these two challenging scene recognition benchmarks.

The results are summarized in Table VIII and Table IX, which show that our VSAD representation outperforms the previous state-of-the-art method (LS-DHM [40]). Furthermore, we explore the complementary properties of our VSAD from the following three perspectives. (1) The semantic codebook of our VSAD is generated by our discriminative PatchNet, while the traditional codebook of Fisher vector (or VLAD) is generated in a generative and unsupervised manner. Hence, we combine our VSAD with Fisher vector to integrate both discriminative and generative power. As shown in Table VIII and Table IX, the performance of this combination further improves the accuracy. (2) Our VSAD is based on local patches and is complementary to those global representations of image-level CNN. Hence, we combine our VSAD and the deep global feature (in the FC6 layer) of Place205-VGGNet-16 [75] to take advantage of both patch-level and image-level features. The results in Table VIII and Table IX show that this combination

Fig. 6. **Analysis of semantic codebook**. The codeword (the 1st column) appears in its related scene categories (the 2nd-5th column), which illustrates that our codebook contains important semantic information.

TABLE VIII

COMPARISON WITH RELATED WORKS ON MIT INDOOR67. NOTE THAT THE CODEBOOK OF FV AND OUR VSAD ARE ENCODED BY DEEP FEATURE FROM OUR SCENE-PatchNet

Method	**Publication**	**Accuracy**(%)
Patches+Gist+SP+DPM [64]	ECCV2012	49.4
BFO+HOG [65]	CVPR2013	58.9
FV+BoP [55]	CVPR2013	63.1
FV+PC [66]	NIPS2013	68.9
FV(SPM+OPM) [67]	CVPR2014	63.5
Zhang *et al.* [68]	TIP2014	39.9
DSFL [69]	ECCV2014	52.2
LCCD+SIFT [70]	arXiv2015	66.0
OverFeat+SVM [71]	CVPRW2014	69.0
AlexNet fc+VLAD[43]	ECCV2014	68.9
DSFL+DeCaf [69]	ECCV2014	76.2
DeCaf [72]	ICML2014	59.5
DAG+VGG19 [73]	ICCV2015	77.5
C-HLSTM [74]	arXiv2015	75.7
VGG19 conv5+FV [75]	arXiv2015	78.3
Places205-VGGNet-16 [76]	arXiv2015	81.2
VGG19 conv5+FV [77]	CVPR2015	81.0
Semantic FV [42]	CVPR2015	72.9
LS-DHM [40]	TIP2017	83.8
Our VSAD	-	84.9
Our VSAD+FV	-	84.4
Our VSAD+Places205-VGGNet-16	-	85.3
Our VSAD+FV+ Places205-VGGNet-16	-	**86.2**

TABLE IX

COMPARISON WITH RELATED WORKS ON SUN397. NOTE THAT THE CODEBOOK OF FV AND OUR VSAD ARE ENCODED BY DEEP FEATURE FROM OUR PatchNet. OUR VSAD IN COMBINATION WITH PLACES205-VGGNet-16 OUTPERFORM STATE-OF-THE-ART AND SURPASS HUMAN PERFORMANCE

Method	**Publication**	**Accuracy**(%)
Xiao *et al.* [23]	CVPR2010	38.0
FV(SIFT+LCS) [16]	IJCV2013	47.2
FV(SPM+OPM) [67]	CVPR2014	45.9
LCCD+SIFT [70]	arXiv2015	49.7
DeCaf [72]	ICML2014	43.8
AlexNet fc+VLAD [43]	ECCV2014	52.0
Places-CNN [3]	NIPS2014	54.3
Semantic FV [42]	CVPR2015	54.4
VGG19 conv5+FV [75]	arXiv2015	59.8
Places205-VGGNet-16 [76]	arXiv2015	66.9
LS-DHM [40]	TIP2017	67.6
Human performance [23]	CVPR2010	**68.5**
Our VSAD	-	71.7
Our VSAD+FV	-	72.2
Our VSAD+Places205-VGGNet-16	-	**72.5**
Our VSAD+FV+ Places205-VGGNet-16	-	**73.0**

surpasses the human performance on SUN 397 dataset. (3) Finally, we combine our VSAD, Fisher vector, and deep global feature of Place205-VGGNet-16 to put the state-of-the-art performance forward with a large margin. To our best knowledge, the result of this combination in Table VIII and Table IX is one of the best performance on both MIT Indoor67 and SUN397, which surpasses human performance (68.5%) on SUN 397 by 4 percents.

F. Visualization of Semantic Codebook

Finally, we show the importance of object-based semantic codebook in Figure 6. Here we use four objects from ImageNet (*desk*, *file*, *slot*, *washer*) as an illustration of the codewords in our semantic codebook. For each codeword, we find five scene categories from either MIT Indoor67 or SUN 397 (the 2nd to 5th column of Figure 6), based on their semantic conditional probability (more than 0.9) with respect to this codeword. As shown in Figure 6, the object (codeword)

appears in its related scene categories, which makes our codebook contains important semantic cues to improve the performance of scene recognition.

VII. Conclusions

In this paper we have designed a patch-level architecture to model local patches, called as *PatchNet*, which is trainable in an end-to-end manner with a weakly supervised setting. To fully unleash the potential of PatchNet, we proposed a hybrid visual representation, named as *VSAD*, by exploiting PatchNet to both describe and aggregate these local patches, whose superior performance was verified on two challenging scene benchmarks: MIT indoor67 and SUN397. The excellent performance demonstrates the effectiveness of PatchNet for patch description and aggregation.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1106–1114.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, 2016, pp. 770–778.

[3] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning deep features for scene recognition using places database," in *Proc. NIPS*, 2014, pp. 487–495.

[4] L. Shen, Z. Lin, and Q. Huang, "Relay backpropagation for effective learning of deep convolutional neural networks," in *Proc. ECCV*, 2016, pp. 467–482.

[5] Y. Xiong, K. Zhu, D. Lin, and X. Tang, "Recognize complex events from static images by fusing deep channels," in *Proc. CVPR*, 2015, pp. 1600–1609.

[6] L. Wang, Z. Wang, W. Du, and Y. Qiao, "Object-scene convolutional neural networks for event recognition in images," in *Proc. CVPRW*, 2015, pp. 30–35.

[7] L. Wang, Z. Wang, Y. Qiao, and L. Van Gool, "Transferring object-scene convolutional neural networks for event recognition in still images," *CoRR*, vol. abs/1609.00162, 2016.

[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. CVPR*, 2005, pp. 886–893.

[10] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Understand.*, vol. 110, no. 3, pp. 346–359, 2008.

[11] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proc. ECCVW*, 2004, pp. 1–16.

[12] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. ICCV*, 2003, pp. 1470–1477.

[13] J. Yang, K. Yu, Y. Gong, and T. S. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Proc. CVPR*, 2009, pp. 1794–1801.

[14] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid, "Aggregating local image descriptors into compact codes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1704–1716, Sep. 2012.

[15] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *Proc. ECCV*, 2010, pp. 143–156.

[16] J. Sánchez, F. Perronnin, T. Mensink, and J. J. Verbeek, "Image classification with the Fisher vector: Theory and practice," *Int. J. Comput. Vis.*, vol. 105, no. 3, pp. 222–245, 2013.

[17] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[18] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[19] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and F. Li, "ImageNet: A large-scale hierarchical image database," in *Proc. CVPR*, 2009, pp. 248–255.

[20] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, 2015, pp. 1–14.

[21] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. CVPR*, 2015, pp. 1–9.

[22] A. Quattoni and A. Torralba, "Recognizing indoor scenes," in *Proc. CVPR*, 2009, pp. 413–420.

[23] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba, "SUN database: Large-scale scene recognition from abbey to zoo," in *Proc. CVPR*, 2010, pp. 3485–3492.

[24] X. Zhou, K. Yu, T. Zhang, and T. S. Huang, "Image classification using super-vector coding of local image descriptors," in *Proc. ECCV*, 2010, pp. 141–154.

[25] J. Yu, X. Yang, F. Gao, and D. Tao, "Deep multimodal distance metric learning using click constraints for image ranking," *IEEE Trans. Cybern.*, doi: 10.1109/TCYB.2016.2591583.

[26] J. Yu, Y. Rui, and D. Tao, "Click prediction for Web image reranking using multimodal sparse coding," *IEEE Trans. Image Process.*, vol. 23, no. 5, pp. 2019–2032, May 2014.

[27] Z. Xu, D. Tao, S. Huang, and Y. Zhang, "Friend or foe: Fine-grained categorization with weak supervision," *IEEE Trans. Image Process.*, vol. 26, no. 1, pp. 135–146, Jan. 2017.

[28] Z. Xu, S. Huang, Y. Zhang, and D. Tao, "Webly-supervised fine-grained visual categorization via deep domain adaptation," *IEEE Trans. Pattern Anal. Mach. Intell.*, doi: 10.1109/TPAMI.2016.2637331.

[29] T. Liu, D. Tao, M. Song, and S. J. Maybank, "Algorithm-dependent generalization bounds for multi-task learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 2, pp. 227–241, Feb. 2017.

[30] T. Liu, M. Gong, and D. Tao, "Large cone nonnegative matrix factorization," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published.

[31] J. C. van Gemert, J. Geusebroek, C. J. Veenman, and A. W. M. Smeulders, "Kernel codebooks for scene categorization," in *Proc. ECCV*, 2008, pp. 696–709.

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

[33] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[34] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce, "Learning mid-level features for recognition," in *Proc. CVPR*, 2010, pp. 2559–2566.

[35] V. Sydorov, M. Sakurada, and C. H. Lampert, "Deep Fisher kernels—End to end learning of the Fisher Kernel GMM parameters," in *Proc. CVPR*, 2014, pp. 1402–1409.

[36] X. Peng, L. Wang, Y. Qiao, and Q. Peng, "Boosting VLAD with supervised dictionary learning and high-order statistics," in *Proc. ECCV*, 2014, pp. 660–674.

[37] Z. Wang, Y. Wang, L. Wang, and Y. Qiao, "Codebook enhancement of VLAD representation for visual recognition," in *Proc. ICASSP*, 2016, pp. 1258–1262.

[38] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. ICML*, 2015, pp. 448–456.

[39] L. Wang, S. Guo, W. Huang, Y. Xiong, and Y. Qiao, "Knowledge guided disambiguation for large-scale scene classification with multi-resolution CNNs," *CoRR*, vol. abs/1610.01119, 2016.

[40] S. Guo, W. Huang, L. Wang, and Y. Qiao, "Locally supervised deep hybrid model for scene recognition," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 808–820, Feb. 2017.

[41] G. Xie, X. Zhang, S. Yan, and C.-L. Liu, "Hybrid CNN and dictionary-based models for scene recognition and domain adaptation," *CoRR*, vol. abs/1601.07977, 2016.

[42] M. Dixit, S. Chen, D. Gao, N. Rasiwasia, and N. Vasconcelos, "Scene classification with semantic Fisher vectors," in *Proc. CVPR*, 2015, pp. 2974–2983.

[43] Y. Gong, L. Wang, R. Guo, and S. Lazebnik, "Multi-scale orderless pooling of deep convolutional activation features," in *Proc. ECCV*, 2014, pp. 392–407.

[44] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," in *Proc. CVPR*, 2016, pp. 5297–5307.

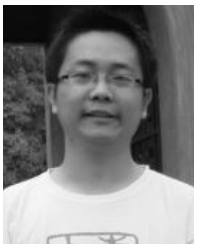
[45] L. Wang, Y. Qiao, and X. Tang, "Action recognition with trajectory-pooled deep-convolutional descriptors," in *Proc. CVPR*, 2015, pp. 4305–4314.

[46] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Comput. Vis.*, vol. 42, no. 3, pp. 145–175, 2001.

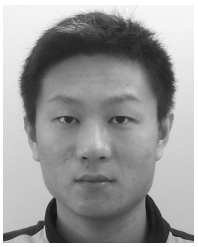
[47] J. Wu and J. M. Rehg, "Centrist: A visual descriptor for scene categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1489–1501, Aug. 2011.

[48] D. Song and D. Tao, "Biologically inspired feature manifold for scene classification," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 174–184, Jan. 2010.
[49] J. Yu, D. Tao, Y. Rui, and J. Cheng, "Pairwise constraints based multiview features fusion for scene classification," *Pattern Recognit.*, vol. 46, no. 2, pp. 483–496, 2013.
[50] L.-J. Li, H. Su, E. P. Xing, and L. Fei-Fei, "Object bank: A high-level image representation for scene classification & semantic feature sparsification," in *Proc. NIPS*, 2010, pp. 1378–1386.
[51] L. Wang, Y. Qiao, and X. Tang, "MoFAP: A multi-level representation for action recognition," *Int. J. Comput. Vis.*, vol. 119, no. 3, pp. 254–271, 2016.
[52] L. Wang, Y. Qiao, X. Tang, and L. Van Gool, "Actionness estimation using hybrid fully convolutional networks," in *Proc. CVPR*, 2016, pp. 2708–2717.
[53] L. Wang, Y. Qiao, and X. Tang, "Latent hierarchical model of temporal structure for complex activity classification," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 810–822, Feb. 2014.
[54] S. Singh, A. Gupta, and A. A. Efros, "Unsupervised discovery of mid-level discriminative patches," in *Proc. ECCV*, 2012, pp. 73–86.
[55] M. Juneja, A. Vedaldi, C. V. Jawahar, and A. Zisserman, "Blocks that shout: Distinctive parts for scene classification," in *Proc. CVPR*, 2013, pp. 923–930.
[56] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. CVPR*, 2006, pp. 2169–2178.
[57] M. Pandey and S. Lazebnik, "Scene recognition and weakly supervised object localization with deformable part-based models," in *Proc. ICCV*, Nov. 2011, pp. 1307–1314.
[58] S. N. Parizi, J. G. Oberlin, and P. F. Felzenszwalb, "Reconfigurable models for scene recognition," in *Proc. CVPR*, 2012, pp. 2775–2782.
[59] X. Wang, L. Wang, and Y. Qiao, "A comparative study of encoding, pooling and normalization methods for action recognition," in *Proc. ACCV*, 2012, pp. 572–585.
[60] X. Peng, L. Wang, X. Wang, and Y. Qiao, "Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice," *Comput. Vis. Image Understand.*, vol. 150, pp. 109–125, Sep. 2016.
[61] L. Wang *et al.*, "Temporal segment networks: Towards good practices for deep action recognition," in *Proc. ECCV*, 2016, pp. 20–36.
[62] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," *CoRR*, vol. abs/1408.5093, 2014.
[63] A. Vedaldi and B. Fulkerson. (2008). *VLFeat: An Open and Portable Library of Computer Vision Algorithms*. [Online]. Available: http://www.vlfeat.org/
[64] T. Kobayashi, "BFO meets HOG: Feature extraction based on histograms of oriented p.d.f. gradients for image classification," in *Proc. CVPR*, 2013, pp. 747–754.
[65] C. Doersch, A. Gupta, and A. Efros, "Mid-level visual element discovery as discriminative mode seeking," in *Proc. NIPS*, 2013, pp. 494–502.
[66] L. Xie, J. Wang, B. Guo, B. Zhang, and Q. Tian, "Orientational pyramid matching for recognizing indoor scenes," in *Proc. CVPR*, 2014, pp. 3734–3741.
[67] L. Zhang, X. Zhen, and L. Shao, "Learning object-to-class kernels for scene classification," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3241–3253, Aug. 2014.
[68] Z. Zuo, G. Wang, B. Shuai, L. Zhao, Q. Yang, and X. Jiang, "Learning discriminative and shareable features for scene classification," in *Proc. ECCV*, 2014, pp. 552–568.
[69] S. Guo, W. Huang, and Y. Qiao, "Local color contrastive descriptor for image classification," *CoRR*, vol. abs/1610.01119, 2015.
[70] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: An astounding baseline for recognition," in *Proc. CVPRW*, 2014, pp. 512–519.
[71] J. Donahue *et al.*, "DeCAF: A deep convolutional activation feature for generic visual recognition," in *Proc. ICML*, 2014, pp. 647–655.
[72] S. Yang and D. Ramanan, "Multi-scale recognition with DAG-CNNs," in *Proc. ICCV*, 2015, pp. 1215–1223.
[73] Z. Zuo *et al.*, "Learning contextual dependence with convolutional hierarchical recurrent neural networks," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 2983–2996, Jul. 2016.
[74] B. B. Gao, X.-S. Wei, J. Wu, and W. Lin, "Deep spatial pyramid: The devil is once again in the details," *CoRR*, vol. abs/1504.05277, 2015.
[75] L. Wang, S. Guo, W. Huang, and Y. Qiao, "Places205-VGGNet models for scene recognition," *CoRR*, vol. abs/1508.01667, 2015.
[76] M. Cimpoi, S. Maji, I. Kokkinos, and A. Vedaldi, "Deep filter banks for texture recognition, description, and segmentation," *Int. J. Comput. Vis.*, vol. 118, no. 1, pp. 65–94, 2016.

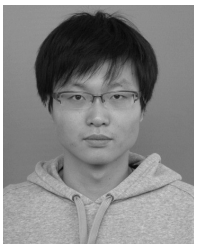
Zhe Wang received the B.Eng. degree from the Beijing University of Posts and Telecommunications, Beijing, China, in 2014. He is currently pursuing the Ph.D. degree with the University of California at Irvine, Irvine, USA. He was a Research Assistant with The Chinese University of Hong Kong and the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, from 2014 to 2016. His current research interests include computer vision and deep learning. He was the winner at the ChaLearn Challenge 2015 in Action/Interaction Recognition and Culture Event Classification, and at ActivityNet Large Scale Activity Recognition Challenge 2016 in video classification.



Limin Wang received the B.Sc. degree from Nanjing University, Nanjing, China, in 2011, and the Ph.D. degree from The Chinese University of Hong Kong, Hong Kong, in 2015. He is currently a Post-Doctoral Researcher with the Computer Vision Laboratory, ETH Zürich. His current research interests include computer vision and deep learning. He was the first runner-up at the ImageNet Large Scale Visual Recognition Challenge 2015 in scene recognition and the winner at the ActivityNet Large Scale Activity Recognition Challenge 2016 in video classification.



Yali Wang received the Ph.D. degree in computer science from Laval University in 2014. He is currently an Assistant Professor with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences. His research interests are machine learning and statistics, computer vision, and deep learning.



Bowen Zhang received the B.Eng. degree major in computer science and technology and minor in mathematics from Tongji University, Shanghai, China, in 2014, where he is currently pursuing the M.Eng. degree in computer engineering. In 2016, he was a Visiting Student with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China. His current research interests include computer vision and deep learning.

Yu Qiao (SM'13) received the Ph.D. degree from the University of Electro-Communications, Japan, in 2006. He was a JSPS Fellow and a Project Assistant Professor with The University of Tokyo from 2007 to 2010. He is currently a Professor with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences. He has authored over 110 papers. His research interests include computer vision, deep learning, multimedia, and robotics. He received the Lu Jiaxi Young Researcher Award from the Chinese Academy of Sciences in 2012.