Unsupervised Feature Learning Via Spectral Clustering of Multidimensional Patches for Remotely Sensed Scene Classification

Fan Hu, Gui-Song Xia, *Member, IEEE*, Zifeng Wang, Xin Huang, *Senior Member, IEEE*, Liangpei Zhang, *Senior Member, IEEE*, and Hong Sun, *Member, IEEE*

***Abstract*—Scene classification plays an important role in the interpretation of remotely sensed high-resolution imagery. However, the performance of scene classification strongly relies on the discriminative power of feature representation, which is generally hand-engineered and requires a huge amount of domain-expert knowledge as well as time-consuming hand tuning. Recently, unsupervised feature learning (UFL) provides an alternative way to automatically learn discriminative feature representation from images. However, the performances achieved by conventional UFL methods are not comparable to the state-of-the-art, mainly due to the neglect of locally substantial image structures. This paper presents an improved UFL algorithm based on spectral clustering, *named* UFL-SC, which cannot only adaptively learn good local feature representations but also discover intrinsic structures of local image patches. In contrast to the standard UFL pipeline, UFL-SC first maps the original image patches into a low-dimensional and intrinsic feature space by linear manifold analysis techniques, and then learns a dictionary (e.g., using K-means clustering) on the patch manifold for feature encoding. To generate a feature representation for each local patch, an explicit parameterized feature encoding method, i.e., triangle encoding, is applied with the learned dictionary on the same patch manifold. The holistic feature representation of image scenes is finally obtained by building a bag-of-visual-words (BOW) model of the encoded local features. Experiments demonstrate that the proposed UFL-SC algorithm can extract efficient local features for image scenes and show comparable performance to the state-of-the-art approach on open scene classification benchmark.**

***Index Terms*—Bag-of-visual-words (BOW) model, linear manifold analysis, scene classification, spectral clustering, unsupervised feature learning (UFL).**

Manuscript received September 30, 2014; revised May 31, 2015; accepted June 03, 2015. Date of publication July 15, 2015; date of current version July 20, 2015. This work was supported in part by the National Natural Science Foundation of China under the Contract 91338113, and in part by the National Basic Research Program (973-Program) of China under Grant 2011CB707105. *(Corresponding author: Gui-Song Xia.)*

F. Hu and H. Sun are with the State Key Laboratory for Information Engineering in Surveying, Mapping, and Remote Sensing (LIESMARS), Wuhan University, Wuhan 430072, China, and also with the Electronic Information School, Wuhan University, Wuhan 430072, China (e-mail: hfmelizabeth@gmail.com; hongsun@whu.edu.cn).

G.-S. Xia, Z. Wang, X. Huang, and L. Zhang are with the State Key Laboratory for Information Engineering in Surveying, Mapping, and Remote Sensing (LIESMARS), Wuhan University, Wuhan 430072, China (e-mail: guisong.xia@whu.edu.cn; zifeng.wang@whu.edu.cn; xhuang@whu.edu.cn; zlp62@whu.edu.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/JSTARS.2015.2444405

I. Introduction

In RECENT years, with the rapid development of satellite imaging techniques, a huge amount of high-resolution satellite images are available, which are provided by some special satellite sensors, e.g., WorldView-1/2 and GeoEye-1. The resulting remote-sensing imageries provide more accurate object observation, which have the surface spatial resolution of half meters. Nevertheless, they bring many challenging problems in image understanding and information mining [1]–[3] as well. For instance, the data volume of images grows sharply with the spatial resolution increasing. Classical approaches that have been proved to be effective for interpreting low-resolution satellite images [4] become inapplicable for analyzing high-resolution satellite images with complex content. Thus, it is highly desirable to develop intelligent and automatic methods for interpreting such massive remotely sensed images with high spatial resolution.

This paper addresses the problem of scene classification for high spatial resolution remote-sensing (HSR-RS) images. The “scene,” in the interpretation of HSR-RS images, usually refers to local areas in images that contain clear semantic information on the surface [2]–[6], e.g., the *residential area, commercial area, farmland, green land, and bare land*. Scene classification can provide an overall layout for HSR-RS images containing various types of complicated land covers and object-oriented scenes. However, the complexities of HSR-RS images make scene classification a challenging task. For instance, some objects contained in the same category of scenes frequently appear at different scales and orientations. If satellite images are taken under different weather conditions, there are probably radiometrical changes between images of the same type. Actually, “scene classification” is challenging, not only because of the high diversity of the appearances and geometries of objects but also due to the complexities in scene semantics.

1) A scene of HSR-RS images often consists of different specific thematic classes.
2) Different types of scenes in HSR-RS images share some identical thematic classes.

For example, *tree*, *road,* and *building,* which exist in the residential area, may also appear in the commercial area. Therefore, most of previous approaches for high-resolution image classification, e.g., in [7] and [8], which dedicate to classifying pixels by extracting local textural and structural features, may lead to serious confusions among similar scenes in the scene classification task. Thus, it is important to encode the spatial configuration of images for deriving an efficient global feature representation of scene for HSR-RS images.

One practicable way is to establish a holistic scene representation by aggregating local image information, so-called *low-level features*, such as key-points [9], [10], texture [11]–[13], and color [14]. Bag-of-visual-words (BOW) model [15], which encodes an image by an unordered collection of local features, has been reported to be an effective method to generate a global image representation and has been widely used in scene classification [16]–[18]. However, it is worth noticing that the methods based on BOW model strongly rely on the extraction of the low-level features. In the past decades, tremendous investigations have been devoted to design novel low-level features with both strong discriminative power and nice invariant prosperities to geometrical and radiometrical changes in images. However, designing those hand-engineered features needs much prior knowledge and expertise in related fields; hence, the process of building new features based on domain-expertise and numerous trials is extraordinarily difficult and time-consuming, especially in the case of handling massive images.

Recently, much work [19]–[23] has focused on automatically learning "good"[1] feature representations from a large amount of unlabeled data by incorporating different unsupervised learning algorithms as a "black box" module within, and these typical feature learning frameworks can be included in the scope of unsupervised feature learning (UFL) algorithms [19]. The UFL methods assume that there are rich useful structures behind unlabeled data. For natural images, it has been reported that UFL methods can discover low-level structures (e.g., edges) as well as mid-level ones (e.g., shapes). Given these powerful learned features, images of different categories can be better separated in a supervised classification framework. In other words, these features obtained by UFL methods can be suitable or even better alternatives to the hand-crafted features in some image classification tasks.

An initial motivation of our study is trying to demonstrate whether the features generated by UFL method can be more powerful and robust compared with the typical hand-crafted features in the conventional scene classification pipeline (e.g., with BOW model), which still remains unclear. Note that in image classification applications with UFL methods [19], [22], the local features are all extracted directly from raw image patches (i.e., pixel intensities). In general settings of UFL, both training model parameters and the feature encoding stage involve large quantities of image patches which are relatively high-dimensional vectors in the raw pixel space and contain great redundant information. To some extent, the standard UFL schemes not only result in extremely high computational cost but also cannot discover the intrinsic information hidden in original image patches. Therefore, we can further investigate the natural statistical properties of image patches, and propose a better UFL scheme according to these beneficial properties.

This paper presents an improved UFL algorithm based on spectral clustering, named UFL-SC, which cannot only automatically learn good feature representations but also discover intrinsic structures of local image patches. In contrast to traditional UFL methods, UFL-SC first maps the original image patches embedded in the high-dimensional image space into a low-dimensional and intrinsic feature space by linear manifold analysis techniques, and then learns a dictionary (e.g., using k-means clustering) on the image patch manifold for feature encoding. To generate a feature representation for each local patch, an explicit parameterized feature encoding method, i.e., triangle encoding, is applied with the learned dictionary to the same patch manifold. The traditional BOW model is introduced to generate the holistic scene representations, where the features extracted by our UFL-SC method are encoded. The whole scene classification pipeline is totally free of any kind of handcrafted features. We evaluate the proposed method on two aerial scene datasets and a large-scale satellite image for classification and annotation task, respectively. The experiments demonstrate that the proposed UFL-SC algorithm can generate representative local features for image scenes and achieve encouraging performance with a low-computational cost. The short version of this work has appeared in [24].

The main contributions of this paper are as follows.

1) We do an in-depth investigation on the manifold of local image patches and have discovered some attracting properties that are helpful for the feature extraction stage.
2) We propose an improved UFL method with spectral clustering, *named* UFL-SC, which learns model parameters and encodes features on low-dimensional patch manifold during feature extraction. In this way, we can both speed up the traditional UFL method and extract powerful low-level features for subsequent scene classification task.
3) We build a BOW model for scene classification based on the local features extracted by the proposed UFL-SC method instead of hand-crafted features, and to our best knowledge, it is the first attempt to apply features generated by the UFL method to the BOW scheme.

This paper is organized as follows. In Section II, we briefly review some related works on UFL algorithms, manifold learning algorithms, and high-resolution satellite scene classification. In Section III, we first introduce the feature extraction scheme of image scene by traditional UFL method. Then, we study the image patch manifold in detail and present the improved UFL-SC method in Section IV. In Section V, we describe our proposed scene classification framework. Details of our experiments and results are presented in Section VI. Finally, Section VII concludes this paper with some remarks.

II. Related Work

Recently, how to learn feature representations in an unsupervised way has become a hot research issue and many efforts have been devoted to efficient learning algorithms. Roughly speaking, there are two main trends in the literatures.

[1] The "goodness" of a feature depends on the context, here a "good" feature means a feature that is discriminative enough for recognition.

A. From Feature Engineering to UFL

A lot of works [25]–[30] have focused on learning hierarchical representations from unlabeled data. With the greedy layer-wise unsupervised pretraining scheme, many novel frameworks by stacking intermediate layers to build deep architectures have been proposed. These deep architectures attempt to learn multilevel structures and have achieved promising results in progressively learning simple and complex concepts hidden in the unlabeled data. For each layer of the deep-learning algorithms, a single-layer model is built via a typical unsupervised learning method. Coates *et al.* [19] make an indepth study into many of the factors that can affect performance to understand what makes a UFL system work well. In [19], a detailed UFL pipeline for extracting locally connected and convolutional features of images is presented and it is shown that even the k-means clustering algorithm is able to achieve state-of-the-art performance on some widely used datasets when the model parameters are chosen properly. In [31], sparse coding is demonstrated to be a universally effective nonlinear encoding scheme in the UFL pipeline whatever approaches the dictionary is generated by. In [3], the UFL approach is successfully applied to aerial scene classification, where sparse coding is exploited as the unsupervised learning method to generate sparse features. In contrast to [19], local dense low-level features rather than pixel values are extracted for training a dictionary in the sparse coding scheme. However, these approaches performed the dictionary learning and feature encoding stage in the high-dimensional space, which makes the entire process of extracting features very time-consuming.

A large number of nonlinear techniques [32]–[35] for dimensionality reduction have been proposed in the last decade. In contrast to the classical linear techniques, e.g., principal component analysis (PCA), the nonlinear techniques have the ability to discover the intrinsic structure of natural data lying on a low-dimensional manifold embedded in a high-dimensional space. Laplacian eigenmaps (LE) [33] as a typical nonlinear method compute a low-dimensional representation of the training data by using the notion of graph Laplacian, and can optimally preserve local neighborhood information to some extent. A linear variant of LE, called locality preserving projections (LPPs), is presented in [36]. LPP not only shares the similar property of preserving local structure of the dataset but also has several advantages over LE. In nonlinear manifold techniques such as ISOMAP [34], local linear embedding (LLE), and LE, each training sample has its own set of free low-dimensional embedding coordinates. However, these methods do not directly learn a parameterized function applicable to evaluate the map for new testing samples, which limits their application for large dataset. By learning a linear mapping matrix, LPP can be simply applied to any new testing samples to compute the embedding coordinates in the same reduced representation space. Much recent works with respect to the classification of remotely sensed images have explored some nonlinear manifold techniques. Bachmann *et al.* [37] present a data-driven approach, which can achieve full-scene global manifold coordinates while removing artifacts and may be limited by the use of landmarks, to represent the nonlinear structure of large-scale hyperspectral scenes. Huang *et al.* [38] propose a hierarchical manifold learning approach for supervised classification of high-resolution remote-sensing images which can sufficiently take advantage of both class-label information and local geometric information.

B. From Low-Level Features to Mid-Level Image Representation

The BOW methods have shown impressive performance in scene classification and other tasks [39]–[42]. In typical BOW representation methods, the visual words are first generated by clustering a large number of local low-level image features, and then the histogram representing the frequency of the visual words for each image is computed statistically. Spatial pyramid matching (SPM) kernel [16] is a classical approach as an extension to BOW. SPM computes local histogram of visual words for each subregions of the image and concatenates all the histograms in a spatial pyramid way. Yang and Newsam [40] compute the co-occurrences of visual words with respect to spatial predicates over a hierarchical spatial partitioning of an image. They show that extended spatial co-occurrence kernel (SPCK++) can outperform the traditional BOW and SPM methods on a high-resolution aerial scene dataset. In addition, [43] and [44] represent satellite image scenes as a finite mixture over some underlying semantic classes learnt by applying the latent Dirichlet allocation (LDA) model [45], which is a generative probabilistic model to the visual words histograms.

III. Unsupervised Feature Learning

This section briefly recalls the basics of UFL in the context of learning features from image patches. In principle, the main goal of UFL is to automatically learn a general feature representation $\Phi(X)$, which can reveal the important qualities of images and can be used to encode other unknown image examples Y [19], from a large set of unlabeled image examples X.

To achieve such a feature representation $\Phi(X)$, one can parameterize $\Phi(X)$ by a set of parameters Θ, i.e., $\Phi(X;\Theta)$. Most of UFL algorithms can be regarded as a hybrid framework composed of two main components:

1) an unsupervised learning algorithm that optimally estimates the model parameter Θ by relying on unlabeled input X;
2) encoding unknown image examples Y by $\Phi(Y;\Theta)$ to yield feature vectors ϕ.

A. Learning Feature Representation by Unsupervised Learning Methods

To train the parameters set Θ associated with the parameterized feature representation (or feature mapping function) $\Phi(x,\Theta)$, a few notable unsupervised learning algorithms are available. Some typical methods come to be, e.g., Gaussian mixture model (GMM), K-means clustering, sparse coding, and auto-encoder [25]. In this paper, K-means clustering is applied to all experiments as an unsupervised learning algorithm, since it is computationally efficient and easy to implement, scales well, does not require any parameter tuning, and furthermore, learns centroids that have similar effect with Gabor filters.

Given the input vector $x^{(i)} \in \mathbb{R}^n$, $i = 1, 2, \ldots, m$, K-means clustering learns a dictionary $D \in \mathbb{R}^{n \times K}$ containing K cluster centers and assignments $c^{(i)} \in \{1, 2, \ldots, K\}$ of the training samples $x^{(i)}$ to clusters that minimize the distance between data points and their cluster centers. Specifically, the algorithm optimizes the following objective:

$$\min_{D,c} \sum_i \|Dc^{(i)} - x^{(i)}\|_2^2 \tag{1}$$

$$\text{s.t. } \|D^{(k)}\|_2 = 1 \; \forall k \text{ and } \|c^{(i)}\|_0 \leq 1 \forall i. \tag{2}$$

This can be optimized by alternating iteration over dictionary D and cluster assignment c. In K-means case, the model is parameterized by $\Theta = \{D\}$. After using k-means to train model parameters Θ from the unlabeled data X, we should define a mapping from a new data to a corresponding feature vector ϕ. The natural and frequently used encoding methods for K-means are hard assignment, which makes each encoding feature only one nonzero element, e.g., the element ϕ_i is equal to one if input vector belongs to cluster center $D^{(i)}$ and the other elements are set to zero. Although hard assignment is a common mapping choice for K-means clustering, the mapping functions $\Phi(x; \Theta)$ may be chosen arbitrarily.

In our work, an alternative method called triangle encoding is introduced. Given the dictionary D, the encoding feature ϕ for a new input $x \in \mathbb{R}^n$ is defined by

$$\phi_k = \max\left\{0, \frac{1}{K}\sum_{k=1}^{K} d_k - d_k\right\} \tag{3}$$

where $d_k = \|x - D^{(k)}\|_2$. This mapping function outputs 0 for any features where the distance from x to the centroid $D^{(k)}$ is above average. This implies that approximate half of the features will be set to 0 in fact. Therefore, this can be thought of as a simple form of competition between features. In addition, it is easy to see from (3) that the length of an encoding feature vector depends on the number of cluster in K-means, i.e., $\phi \in \mathbb{R}^K$.

B. Image Feature Extraction by Standard UFL Scheme

Here, we present a brief scheme of extracting dense local features of images by means of K-means and triangle encoding. As with the standard UFL pipeline described above, to learn a dictionary from unlabeled samples, we need to perform following three steps first: 1) extracting large quantities of small image patches from random locations in unlabeled training images [46]; 2) performing brightness and contrast normalization as well as zero component analysis (ZCA) whitening to the patches as a preprocessing phase; and 3) training a dictionary D from the preprocessed patches using K-means clustering according to (1) and (2). Note that each patch has dimension r-by-r and has c channels (for natural images, there are only R, G, B channels), so each r-by-r patch can be represented as a vector in $\mathbb{R}^n$ of pixel intensity values, with $n = r \cdot r \cdot c$.

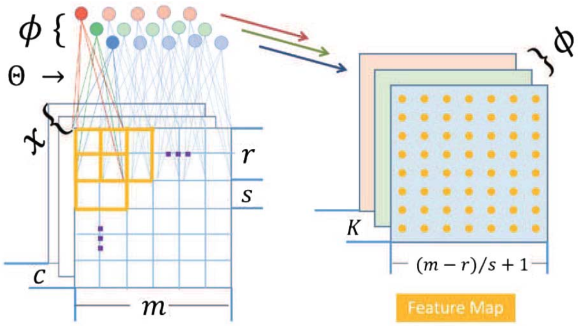


Fig. 1. Illustration of convolutionally extracting local features for images by the UFL approach.

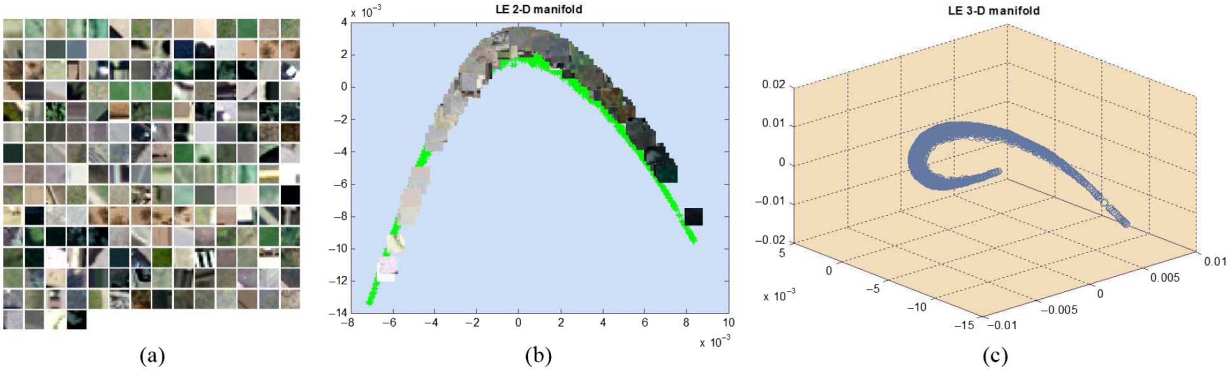
Given the learned dictionary D, a corresponding feature vector ϕ for a new patch x can be achieved by the triangle encoding $\Phi(x; D) : \mathbb{R}^n \mapsto \mathbb{R}^K$. We are now capable of mapping any r-by-r pixel patch to a K-dimensional feature vector ϕ. For a given image of size m-by-m (with c channels), we then divide the image into a number of square patches of size r-by-r pixels, separated by s pixels each. Rather than learning different dictionaries for each patch of the image, we just simply reuse the same $\Phi(x; D)$ to extract features for each patch. This trick relies on the assumption that any r-by-r patches in an image set have the similar statistic structure. Finally, all the feature mapping operation at every location of the input image can yield a resulting $((m - r)/s + 1)$-by-$((m - r)/s + 1)$-by-K dimensional feature representation. The illustration of local feature extraction is shown in Fig. 1. In the special case where the step size is equal to one pixel, this evolves into a convolutional architecture resembling the convolutional neural networks (CNNs) [47], and the dictionary D can be regarded as filter banks convolved with the input image.

IV. UFL-SC of Multidimensional Patches

In this section, we will investigate the low-dimensional structure of tiny image patches, and present a modified UFL pipeline, which is inspired by the spectral clustering, for the tractability of image feature extraction. It has been reported that spectral clustering often outperforms traditional clustering approaches, such as K-means, by considering the structures of data. Spectral clustering is easy to implement and can be solved efficiently by standard linear algebra methods. Moreover, spectral clustering can be implemented efficient for large dataset, as long as we make sure that the similarity graph is sparse.

A. Discovering Low-Dimensional Manifold of Image Patches

As demonstrated in [33], LE, a nonlinear manifold learning method, has a natural connection to spectral clustering. Since LE attempts to preserve local structures of data by utilizing the notion of graph Laplacian matrices, which is well known in spectral graph theory, LE implicitly reflects the natural clustering attributes of the data. From this viewpoint, a



(a) (b) (c)

Fig. 2. 2-D and 3-D image patch manifold learned by LE. (a) Training patch samples of 10-by-10 pixels. (b) 2-D image patch manifold, all the training patches are displayed according to their low-dimensional coordinates. (c) 3-D image patch manifold.

brief implementation of spectral clustering consists of only two steps:

1) computing a low-dimensional representation of the input data by means of LE;
2) applying K-means clustering to find the cluster centroid in the low-dimensional feature space.

Put simply, spectral clustering just add a "special dimensionality reduction" processing stage compared with K-means. Hence, we first discuss the performance of LE on image patches before the spectral clustering is introduced into the feature extraction pipeline.

1) Learning With LE on Image Patches: Image patches can be considered as vectors in a high-dimensional space which is spanned by the individual pixel values at each location in the image patch, e.g., 10-by-10 pixel images patches (with R, G, and B channels) can model the image space of 300 dimensions. Many investigations have revealed that images of interesting types reside on a low-dimensional manifold embedded in a high-dimensional space, such as human faces and hand-written digital numbers. Especially in [48], Shi and Zhu characterize the image patches by explicit manifold and implicit manifold. Here, we do not pay much effort to model image patches via manifold, but make an assumption that image patches indeed lie on an intrinsic low-dimensional manifold. Practically, we apply LE not only to learn the low-dimensional embedding of image patches but also to explore the underlying geometric structure of the manifold that is helpful to guide the clustering process.

Given a large set of r-by-r pixel image patches (with c channels) $\{P_i\}_{i=1,\ldots,N}$ randomly extracted from unlabeled images, we span the patches into vectors $\{v_i \in \mathbb{R}^n\}_{i=1,\ldots,N}$ in the high-dimensional space, where $n = r \cdot r \cdot c$. Then, LE conducts the following four steps to learn the low-dimensional embedding $\{y_i\}_{i=1,\ldots,N}$ of the input vectors:

1) constructing the undirected adjacency graph $\mathcal{G}$ in a k-nearest neighbors way;
2) weight assignment for each adjacent vector pair. Weight matrix W can be selectively defined as

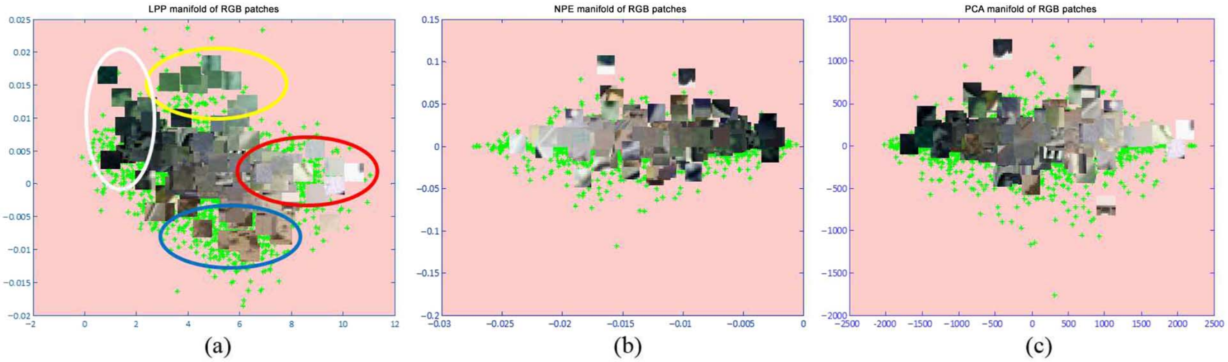
$$W_{ij} = \begin{cases} e^{\left(-\|v_i - v_j\|_2^2/t\right)}, & \text{if } v_i \in N(v_j) \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

where $N(v_j)$ denotes k nearest neighbors set of the vector v_j, and t is the weight decay constant.
3) compute the Laplacian matrix L. The formulation is expressed as $L = S - W$, where S is diagonal weight matrix, $S_{ii} = \sum_j W_{ji}$;
4) solve the generalized eigenvector problem $L\mathbf{y} = \lambda S\mathbf{y}$, and use the first h eigenvectors corresponding to eigenvalues (leave out eigenvalue 0) sorted by ascending order for embedding $\{y_i \in \mathbb{R}^h\}_{i=1,\ldots,N}$ in the h-dimensional Euclidean space. The theoretical justification details of the algorithm are presented in [33].

After accomplishing the procedures above, we now obtain a new low-dimensional representation for the raw image patches. A toy example of two-dimensional (2-D) and three-dimensional (3-D) manifold structure is shown in Fig. 2. We can discover that the image patches appear to be regular geometrical curves in both 2-D and 3-D manifolds. Besides, it is noted that along the curve in 2-D intrinsic space, gray level and structural complexity of image patches vary gradually. In this case, the gray level and structural complexity are two degrees of freedom (DOF) for image patches. Hence, the two DOFs form an intuitive partition of the training image patches in low-dimensional manifold, which testify that LE has the natural clustering property in a sense.

Following the spectral clustering pipeline, K-means clustering is then applied to the low-dimensional data in Euclidean space. At this step, K-means performs quite efficiently and we probably get a better clustering result under the guide of LE than the clustering performance directly using K-means in the original high-dimensional space.

So far, as a nonlinear feature embedding approach, LE has been demonstrated to be helpful to yield satisfactory clusters for image patches. We readily come up with an idea that in the UFL pipeline, the dictionary learning and triangle encoding stage can be done on an intrinsically low-dimensional manifold. Concretely speaking, we apply LE to the original vector space spanned by all the training image patches before K-means clustering in the dictionary learning stage; in the feature extraction stage, all the patches are embedded into the low-dimensional manifold defined by LE in the preceding stage so as to obtain low-dimensional coordinates, then the features for these patches are generated via the triangle encoding scheme on the basis of mapping coordinates. But actually, this method is inapplicable because LE is defined only on the training data and it is very difficult to evaluate the mapping



(a) (b) (c)

Fig. 3. 2-D linear patch manifold structure of the sample patches learned by different linear techniques. The patches are located according to the low-dimensional coordinates. (a) LPP. (b) NPE. (c) PCA. Circles in (a) show the better natural clustering property of patches when embedded in 2-D manifold by applying LPP.

coordinates for new testing data (actually, mapping coordinate can be computed using Nystrom approximation, but this remains cumbersome and computationally expensive), so this gives rise to the fact that we are incapable of extracting features for new image patch vectors without its low-dimensional coordinates of the identical space on which the training vectors are embedded. As a result, an improved approach that can be efficiently applied to new testing vectors is preferable. Some linear dimensionality reduction algorithms become suitable alternatives to LE under this circumstance, since they all can provide a linear projective map M to any input vectors rather than just the training vectors. Next, we will discover several typical linear techniques and their performance on image patches.

2) Learning a Linear Manifold on Image Patches: PCA is a classical and most popular linear dimensionality reduction technique that embeds the data into a linear subspace where the amount of variance in the data is maximal. However, PCA is incapable of discovering the nonlinear structure of the data manifold; thus, an alternative linear method called LPP [36] is proposed. Although LPP is a linear algorithms, it shares the locality preserving properties of LE. In other words, LPP gathers advantages from both PCA and LE.

LPP is considered as a linear approximation to the nonlinear LE. The algorithmic procedure is the same as LE except the last step, so to learn low-dimensional representation $\{y_i \in \mathbb{R}^d\}_{i=1,\ldots,N}$ of input patch vectors $\{v_i \in \mathbb{R}^n\}_{i=1,\ldots,N}$, we just need to slightly modify the generalized eigenvalue problem, described as follows:

$$VLV^\top \mathbf{m} = \lambda VDV^\top \mathbf{m} \tag{5}$$

where $V = (v_1, v_2, \ldots, v_N)$. L is the Laplacian matrix, λ is the eigenvalue, and D is the diagonal matrix, defined identically in LE. We sort the eigenvectors $m_0, m_1, \ldots, m_{d-1}$ according to their eigenvalues in ascend order. As a result, the low-dimensional embedding is described as a linear projective map

$$y_i = M^\top v_i, M = (m_0, m_1, \ldots, m_{d-1}) \tag{6}$$

where $M \in \mathbb{R}^{n\times d}$ denotes the linear mapping matrix. As theoretically demonstrated in [36], LPP is based on the same variational principle that gives rise to LE. Otherwise, LPP has more discriminating power and is less sensitive to outliers than PCA. Another linear dimensionality reduction algorithm is called neighborhood preserving embedding (NPE) [49], which is a linear approximation to locally linear embedding (LLE) and shares some similar properties with LPP. The detailed theoretical derivation is seen in [49].

As we described in preceding chapter, image patch vectors actually reside on a low-dimensional manifold closely related with some specific DOF (e.g., gray level and structural complexity); therefore, LPP is supposed to obtain better low-dimensional representation of image patch than PCA due to its preserving properties of local manifold structure. A toy example shown in Fig. 3 verifies this assumption, which shows that, in dimensionality reduced space, image patches can be naturally grouped into explicit distinct clusters by LPP mapping, and the clusters reveal the local neighborhood relationship of image patch manifold structure very well. However, PCA is less capable of grouping image patches in low-dimensional space. As a result, LPP is a preferred method for mapping image patches into low-dimensional representation rather than PCA.

B. UFL-SC: Image Feature Extraction on Patch Manifold

Note that LPP not only has similar properties with LE but also provides a linear map for new testing vectors, so the LPP can be appropriately introduced in the UFL pipeline. The proposed two-stage UFL pipeline, i.e., UFL-SC, comprised of dictionary learning and feature encoding is illustrated in Fig. 4. In dictionary learning stage, the raw image patch vectors (as training vectors) $\{v_i \in \mathbb{R}^n\}_{i=1,\ldots,N}$ are first embedded into a low-dimensional Euclidean space using a linear manifold technique, then K-means clustering is performed on the low-dimensional representation $\{y_i \in \mathbb{R}^d\}_{i=1,\ldots,N}$ so as to obtain a dictionary $D \in \mathbb{R}^{d\times K}$ (K is the number of centroids) on the image patch manifold, as well as the linear mapping matrix $M \in \mathbb{R}^{n\times d}$. Fig. 4 depicts the algorithm flow in details. This stage is motivated from traditional spectral clustering and specifically designed for the following feature encoding stage. In the feature encoding phase, the low-dimensional coordinates $Y \in \mathbb{R}^d$ for new input patch vectors $X \in \mathbb{R}^n$ are generated by simply multiplying them by the mapping matrix M. This operation embeds these input patch vectors into the identical low-dimensional space which is learnt on training patch vectors

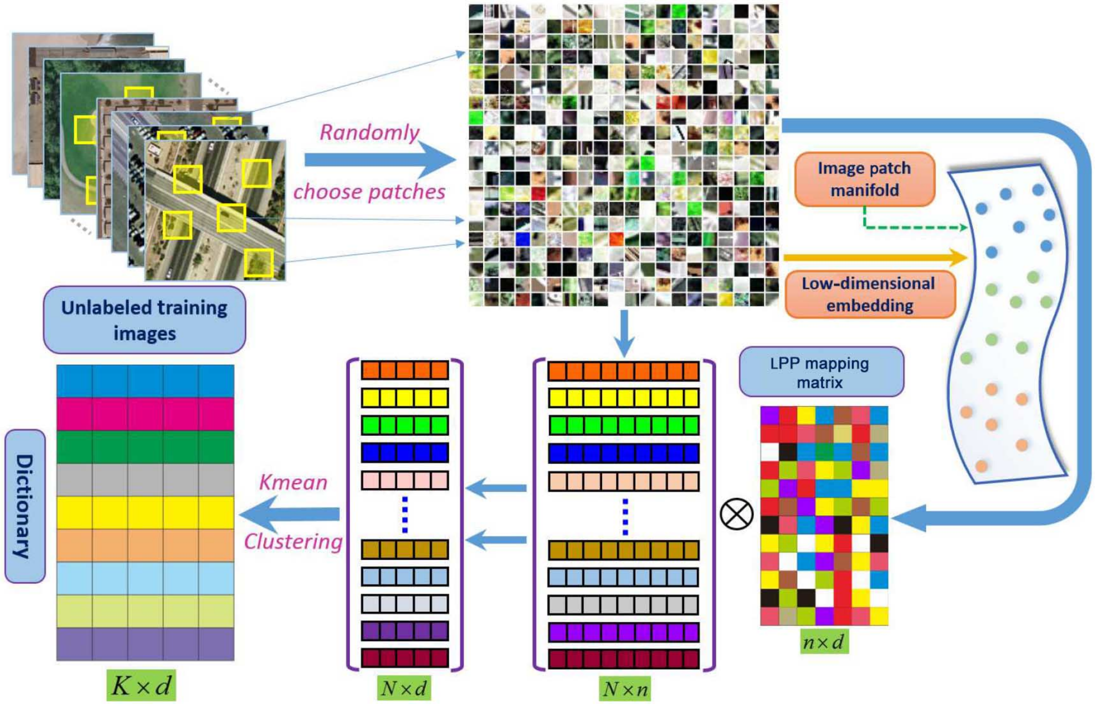


Fig. 4. Illustration of learning dictionary on patch manifold. At this stage, the dictionary and a linear mapping matrix, which are the model parameters for feature encoding stage, are generated simultaneously.

by the linear manifold technique. After this, the triangle encoding $\Phi(Y; D) : \mathbb{R}^d \mapsto \mathbb{R}^K$ is applied as a feature extractor to achieve a feature vector ϕ for each input patch vector. This can be essentially considered as extracting features on the patch manifold, which may lead to some interesting properties. Our UFL-SC pipeline is given in Algorithm 1. Any linear manifold technique, which can provide the linear mapping matrix (e.g., LPP, PCA, NPE), is applicable to the UFL-SC pipeline. Here, we adopt the LPP as the linear manifold method in our algorithm flow, as an example.

Algorithm 1. Feature Extraction via UFL-SC with LPP.

```
Input:
    The original image dataset, $\mathcal{S}$;
    New image patch vectors, $X$
Output:
    The dictionary on patch manifold, $D$;
    The linear mapping matrix, $M$;
    The feature vectors for input patch vectors, $\phi$;
Extract a large number of raw image patches vectors $V$ from random location of images in the dataset $\mathcal{S}$;
Apply brightness and contrast normalization as well as ZCA whitening to the patches;
Compute the linear mapping matrix $M$ by LPP;
Generate low-dimensional representation $Y$ of the patch vectors, according to Eq. 6;
Train a dictionary $D$ on the low-dimensional representation $Y$ via K-means clustering;
Compute the feature vectors $\phi$ for new patch vectors $X$ via triangle encoding, according to Eq. 3;
return $M$, $D$, $\phi$;
```

It is worth noting from Algorithm 1 that our UFL-SC method can reduce the computational cost in contrast to the standard UFL method, which is concluded from the general analysis of computational complexity. Concretely, In dictionary learning phase, the computational cost of UFL-SC is on the order of $O(Kd)$, while the computational cost of standard UFL is on the order of $O(Kn)$; in the triangle encoding phase, the computational cost of UFL-SC is on the order of $O(Kd^2)$, while the computational cost of standard UFL is on the order of $O(Kn^2)$.

Actually, triangle encoding is a typical nonlinear feature coding method that projects the input data points into feature space with a parametric mapping function. In other words, triangle encoding can be interpreted as another form of "manifold learning." The nonlinear triangle encoding method aims to find a feature space (not necessary a low-dimensional space) where the embedded features of image patches have more discriminative power. As shown in Fig. 5, it is very interesting that the geometric structures of 2-D and 3-D feature vectors generated by triangle encoding on the image patch linear manifold are very similar to low-dimensional manifold structure learnt by LE. This observation means that in the proposed UFL-SC method, triangle encoding on linear patch manifold has similar power of feature representation as some nonlinear manifold learning methods, but can be applied in a wider range of application.

V. Global Feature Representation for Scenes and Scene Classification Framework

So far, we have proposed an improved UFL-SC pipeline for local feature extraction of images. Given an input image scene, the local feature descriptors for image patches extracted densely

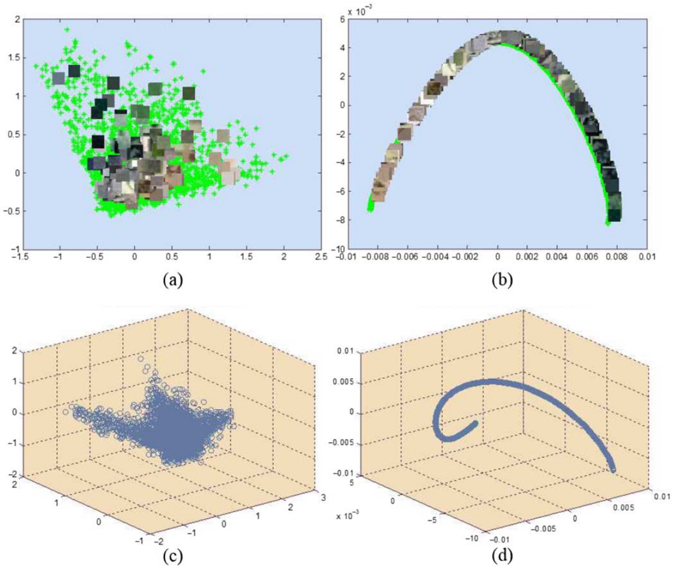


Fig. 5. Toy low-dimensional manifold structure of encoded features sample patches. (a) and (c) show the 2-D and 3-D manifold structures of encoded features generated by triangle encoding in original patch space [(a) triangle encoding 2-D manifold; (c) triangle encoding 3-D manifold]. (b) and (d) show the 2-D and 3-D manifold structures of encoded features generated by triangle encoding on the linear patch manifold [(b) triangle encoding 2-D manifold after LPP; (d) triangle encoding 3-D manifold after LPP]. Here, LPP is used as the linear manifold technique.

on a regular grid are encoded via the proposed method. Then, we follow the traditional BOW scheme to finally represent each image scene with a histogram in which each bin counts the occurrence frequency of features on a codeword. The BOW scheme is implemented via two steps: 1) learning codebook; and 2) coding features. The codewords are typically generated by unsupervised clustering algorithms (e.g., k-means). For instance, a plenty of local features randomly sampled from image base are clustered via k-means and the cluster centers form the codewords. Considering that this intuitive method of learning codewords leaves out the category information that is helpful to subsequent supervised classification task, we slightly enhance this method by generating codewords for each class. Let the $\mathcal{F}_p^{(m)} = \{f^{(i)} \in \mathbb{R}^K\}_{i=1,\ldots,N}$ be N K-dimensional features extracted by an image I_p belonging to class m. By applying K-means to the training features sampled from feature set $\mathcal{F}^{(m)} = \{\mathcal{F}_1^{(m)}, \mathcal{F}_2^{(m)}, \ldots, \mathcal{F}_p^{(m)}\}$, the codebook $C^{(m)} \in \mathbb{R}^{K\times L}$ with respect to class m is then generated; L is the number of codewords. As a result, we construct a joint discriminative codebook C^J by simply concatenating the codebook of each class, i.e., the joint codebook is represented as $C^J = [C^{(1)}, C^{(2)}, \ldots, C^{(B)}] \in \mathbb{R}^{K\times BL}$; B denotes the number of class in the image base. Given any testing image I_j, we assign each local feature to its closest codeword and obtain a statistical histogram $H_j \in \mathbb{R}^{BL}$ for all the assignments. The resulting global feature representation for image I_j is described with the histogram H_j which is subsequently fed into supervised classifier.

The scene classification flowchart based on our UFL-SC method is depicted in Fig. 6. To summarize, the whole scene classification process can be divided into three separate parts, i.e., local feature extraction, holistic feature representation, and SVM classification. Concretely, at the beginning of local feature extraction stage, we randomly sample a large number of image patches from images in a dataset at any location, and then a certain linear manifold method is performed to yield the low-dimensional representation of the training image patch vectors, as well as the linear mapping matrix. Next, we train a dictionary used for triangle encoding by performing K-means clustering on the image patch manifold. These several steps are implemented offline and prepared for the feature encoding step. Following the instructions of Algorithm 1, local image patches of each image scene densely extracted on a grid with a fixed step are first projected into the low-dimensional image patch manifold by the linear mapping matrix, and are then encoded into feature vectors via triangle encoding method. After the feature extraction, a discriminative joint codebook is built according to the improved method described above, and thereby holistic histogram features are generated given the learnt codebook. Both training and testing image scenes undergo the same feature extraction and feature representation procedure. At the last stage, an SVM classifier is trained with the training image scenes and predicts the category labels for all testing image scenes. The detailed UFL-SC-based scene classification framework is seen in Algorithm 2. It is important to note that throughout the overall scene classification pipeline, K-means clustering is performed twice but plays a totally different role in each case.

Algorithm 2. UFL-SC-Based Scene Classification Framework.

```
Input:
    The training image scene set and the corresponding ground truth labels, $\mathcal{I}^{tr}$, $\mathcal{L}^{tr}$;
    The testing image scene set, $\mathcal{I}^{te}$;
    The dictionary on patch manifold, $D$;
    The linear mapping matrix, $M$;
Output:
    The predicted labels for testing image scenes, $\mathcal{L}^{te}$;
1: For each image scene in the training set $\mathcal{I}^{tr}$, densely extract local image patches on a grid;
2: Compute the feature vectors $\mathcal{F}$ for all image patches via Algorithm 1;
3: For each class $m$, learn a codebook $C^{(m)}$ by applying K-means clustering on the feature vectors randomly sampled from feature subset $\mathcal{F}^{(m)}$;
4: Build the joint codebook $C^J$ by concatenating the codebook of each class, $C^J \Leftarrow [C^{(1)}, C^{(2)}, \ldots, C^{(B)}]$
5: For each image scene $\mathcal{I}_i^{tr}$ in the training set $\mathcal{I}^{tr}$, assign each local feature to its closest codeword in $C^J$, and compute a statistical histogram $H_i^{tr}$ for all assignments;
6: Train model parameters of SVM classifier using the holistic histogram representations $H^{tr}$;
7: For each image scene $\mathcal{I}_i^{te}$ in the testing set $\mathcal{I}^{te}$, compute the histogram $H_i^{te}$ by repeating the steps 1, 2 and 5;
8: Predict class labels for all testing image scenes by the trained SVM classifier with $H_i^{te}$;
9: return $\mathcal{L}^{te}$;
```

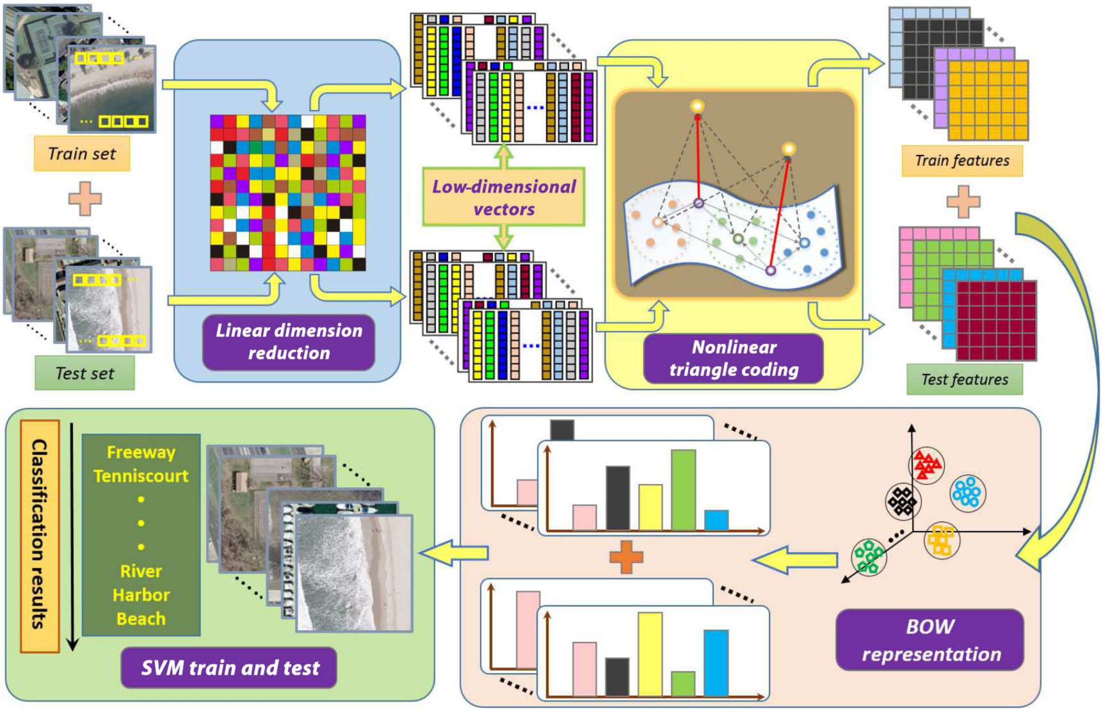


Fig. 6. Proposed overall scene classification framework.

When training the SVM classifier, the histogram intersection kernel (HIK) [50] is adopted, which is suitable for measuring similarity between vectors of histogram, and is defined as

$$I\left(H_i, H_j\right) = \sum_{k=1}^{BL} \min\left(H_i(k), H_j(k)\right) \tag{7}$$

where H_i and H_j are histogram features with BL bins for image i and j, and $H_i(k)$ denotes the count of the k^{th} bin of H_i.

VI. Analysis of Experimental Results

In this section, we evaluate our proposed method on two different high-resolution land-use data sets as well as a large satellite image, and present detailed experimental setup and reasonable analysis.

A. Experimental Dataset

1) UC Merced Land-Use Dataset: The UC Merced land use dataset (UCM) [51] consists of 21 scene categories which are manually extracted from large aerial orthoimagery with the pixel resolution of one foot. Each class contains 100 images with size of 256 × 256 pixels, and two typical examples of each class are shown in Fig. 7. In fact, the term "land use" is used here to refer to the set of hybrid classes which even contain some land-cover (e.g., forest, agricultural) and object classes (e.g., airplane, tennis courts). Note that this dataset covers various typical scene categories with great intraclass variability. Moreover, a few identical texture structures and objects that differ in spatial patterns and density distribution are more or less shared among some of scene categories, such as the freeway, runway, and overpass.

2) WHU-RS Dataset: The WHU-RS dataset [1] consists of 19 satellite scene classes and each class contains 50 images collected from Google Earth (Google Inc.) with size of 600 × 600 pixels. Fig. 8 displays one sample of each class. We note that complexity of scene layout, illumination variation, and changes of location and scale of both textons and objects in these scenes increase the diversity among the same classes (e.g., port and park), as well as the ambiguity of the different classes (e.g., industrial area, commercial area, and residential area). Hence, this dataset appears to be a more challenging one than the UCM dataset.

3) Large Satellite Image of Grenoble, France: We perform the scene annotation experiment on a large satellite image of urban area in Grenoble, France. The large satellite image captured from Google Earth has the size 6000 × 6000 pixels. Since there are only several main scene categories included in the image, we specially select training examples belonging to six scene classes: forest (FOR), industrial area (IA), meadow (MEA), parking (PAR), residential area (RA), and water (WA). The training examples, which are all collected nearby the geographical position of the large image, are of size 150 × 150 pixels with 30 examples per class. The subimage-level ground truth for this large satellite image and training examples are shown in Fig. 9(b) and (c), respectively. Note that there are some unknown scene classes which obviously cannot be annotated as any of the six classes in the large image.

B. Experimental Setup

Considering that there are a number of parameters that may influence the final classification performance, we try to set reasonable parameters for different datasets. For UCM dataset, we

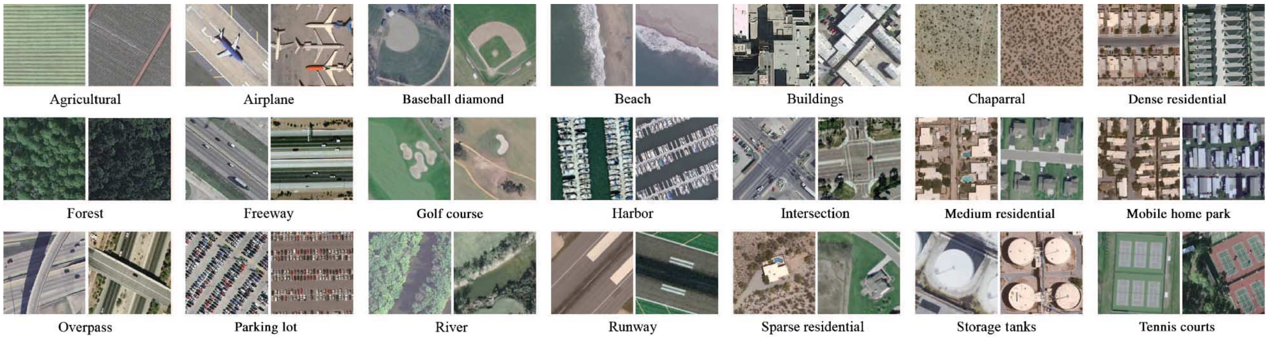


Fig. 7. Two examples of each scene category in UC Merced dataset.

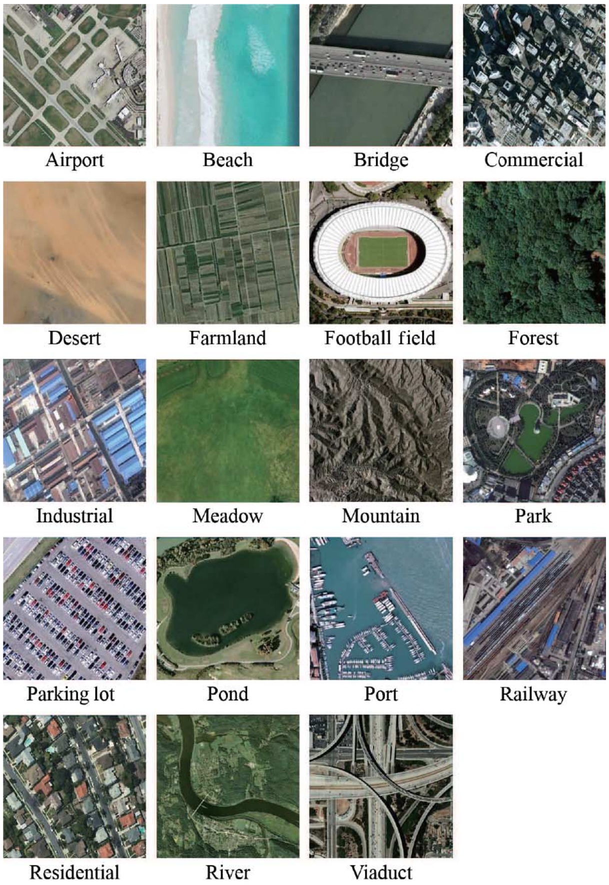


Fig. 8. Examples of each category in WHU-RS dataset.

randomly sample 100 patches per image scene and fix the patch size to 10 × 10 pixels (with R, G, B channels). All these patches are used to learn the linear patch manifold, as well as the dictionary. During feature extraction stage, the sampling step of image patch is set to 5 pixels. We randomly select 80 images per class as training set to train an SVM classifier with HIK and the rest as testing set. Classification performance is quantitatively evaluated by the classification accuracy defined as

$$\mathcal{A} = \mathcal{N}_c / \mathcal{N}_t \tag{8}$$

where $\mathcal{N}_c$ denotes the number of samples correctly classified in testing samples and $\mathcal{N}_t$ denotes the total number of testing samples. The classification experiment is repeated 100 times to yield a mean accuracy $\overline{\mathcal{A}}$. In addition, three important free parameters, which are: 1) the dimensionality d of the low-dimensional space where image patches are mapped; 2) the length K of encoded feature vectors; and 3) the number of codewords per class L, are evaluated to report how these parameters impact classification performance. The evaluation of these three parameters can provide a convincing guidance to properly set parameters for new datasat. On WHU-RS dataset, we follow the basic settings above, except that all image scenes are rescaled to the size of 300 × 300 pixels for computational efficiency, and the train/test proportion for SVM is fixed to be 4:1. In annotation experiments, we divide the original large image into 1600 nonoverlapping subimages with size of 150 × 150 pixels. Each subimage is considered as an image scene. Since the train images are prepared beforehand, testing images are all 1600 subimages. We perform the classification experiment with HIK SVM [52] once to obtain the final classification accuracy. Otherwise, we do ZCA whitening preprocessing for image patches in the first place whether in dictionary-learning phase

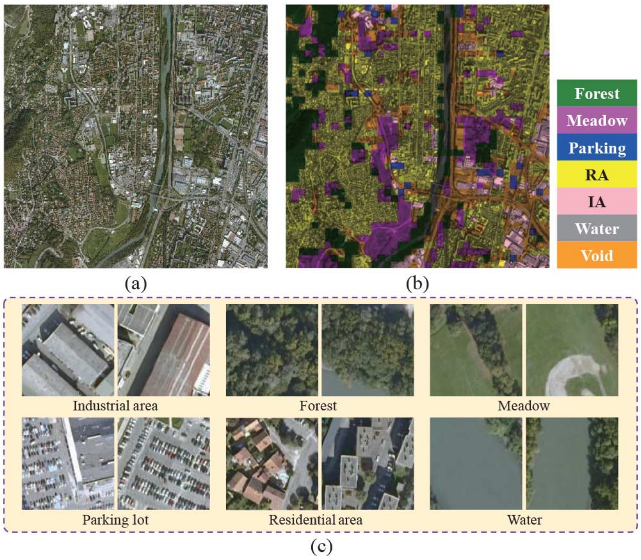


Fig. 9. Large satellite image of Grenoble, France. (a) Original image. (b) Subimage-level ground truth of the large image. (c) Training examples of each category. All training samples are collected from somewhere nearby the location of the large satellite images. We select six most common scene categories appearing in the urban area; hence, there exist some unknown scene classes in our experimental image.

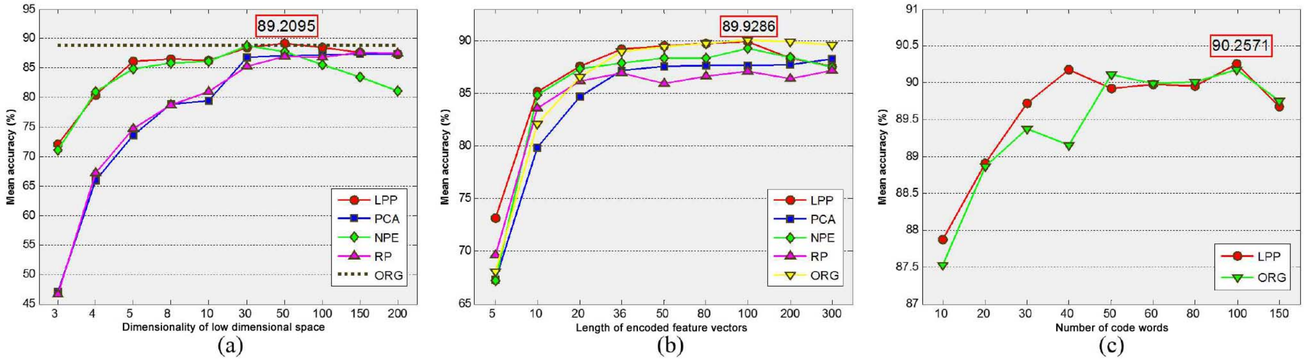


Fig. 10. Classification performance comparison under different parameter settings on UCM dataset. (a) Dimensionality of low-dimensional space where the patches are embedded. (b) Length of encoded feature vectors, i.e., the number of clusters in K-means at the dictionary learning stage. (c) Number of codewords while building BOW model. The dash line "ORG" represents the standard UFL pipeline in which both dictionary learning and feature encoding are performed in the original image patch space, rather than a low-dimensional patch manifold. In this case, ORG generates a baseline accuracy with 88.96%.

or in the feature extraction phase. In particular, the number of nearest neighbors is also an important parameter to LPP and NPE, since it has a close relationship to the construction of adjacency graph. Therefore, the number of nearest neighbors is empirically fixed to 12 throughout all our experiments.

All approaches in our work are implemented in MATLAB on the Windows 7 platform, with a 2.93-GHz Intel Xeon CPU.

C. Experimental Results

1) UC Merced Dataset: As described above, we focus our evaluation on three key hyper-parameters: d, K, and L. When we discuss how the three free parameters impact final classification performance, respectively, we evaluate single one (as a variable) of the three and keep the other two to be constants. Now that we extract image local features on a low-dimensional patch manifold; the dimensionality d of the space where all image patches are embedded becomes a dominant role in our method. Due to the patch size of 10 × 10 pixels, the length of original patch vectors is 300, which limits the size of d within 300. Fig. 10(a) shows the overall mean classification accuracy of UCM dataset under different d ranging from 3 to 200. In this case, K and L is set to 36 and 50, respectively, by experience. Four typical linear manifold learning algorithms which can learn patch manifold and generate linear mapping matrix are tested in the proposed UFL-SC method. Random projection (RP) [53] is a simple yet effective dimensionality reduction method. It provides the linear mapping matrix M where the entries of M obey zero-mean, unit-variance normal distribution independently. It shows that RP is a comparable method to PCA under different d. The average classification precision increases along with d growing in both PCA and RP case. In contrast, bigger d do not lead to better performance, while LPP or NPE is applied to learn patch manifold. The value of d that makes classification accuracy reach the peak is 50 in LPP case and 30 in NPE case. It is noted that when d is relatively small, say, less than 10, LPP and NPE perform far better than PCA and RP. This is mainly because LPP and NPE have the ability of preserving locality of raw data and loss less information while projecting raw image patches into a low-dimensional space, which makes the dictionary learning on the low-dimensional patch manifold more reliable. In addition, LPP performs better than NPE, in general, with d varying, especially when d is greater than 30. On the whole, LPP is the most stable and effective one of the four linear manifold methods in our experimental pipeline, and even yields the classification accuracy with 89.2% beyond the performance with the standard UFL pipeline (learning dictionary and encoding features in the original image patch space).

TABLE I
MEAN CLASSIFICATION ACCURACY COMPARISON ON UCM DATASET

Methods	Classification accuracy (%)
BOVW [40]	71.86
SPM [16]	74
SCK [51]	72.52
SPCK++ [40]	77.38
SC+Pooling [3]	81.67 ± 1.23
SG+UFL [54]	82.72 ± 1.18
COPD [55]	91.33 ± 1.11
HMFF [56]	**92.38 ± 0.62**
Bag of SIFT [56]	85.37 ± 1.56
Bag of colors [56]	83.46 ± 1.57
Bag of DisLBP [56]	82.52 ± 2.75
Ours	**90.26 ± 1.51**

Now that triangle encoding method is applied to extract features for local image patches, the quality of local features depends on the dictionary D. It is worth noting that, as discussed in Section III, the length of feature vectors is equal to the number of clusters; hence, the number of clusters K when using K-means to train a dictionary D becomes the only parameter that influences the resulting encoded local features. The length of encoded feature vectors not only directly relates to the computational efficiency of feature extraction but also has a big impact on the subsequent scene classification performance. Fig. 10(b) shows the overall classification accuracy under different K. In this group of experiments, L and d are fixed at 50 and 50 guided by experiments above. As a very similar situation to (a), classification performance do not grow consistently along with K in LPP and NPE case. Despite this, LPP still outperforms other three linear methods. The proposed UFL-SC pipeline with LPP is comparable to the standard UFL and

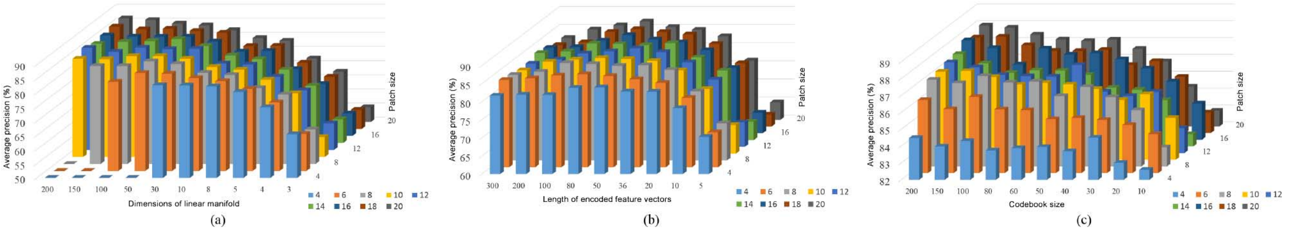


Fig. 11. Classification performance comparison under different patch size on WHU-RS dataset. (a) Dimensions of low-dimensional space where the patches are embedded. (b) Length of encoded features. (c) Size of codebook in BOW representation.

performs better when K is less than 36. One possible explanation for this observation is that clustering on patch manifold has more advantages over clustering on the original patch space due to the natural clustering attribute of LPP when the number of clusters is small.

The average classification performance with varying L that plays an important role in the holistic feature representations is shown in Fig. 10(c). Once again, we compare our UFL-SC method with the standard one. In this case, d and K is fixed to 50 and 100, respectively. Given that, under UFL-SC pipeline, LPP outperforms other threes linear techniques according to the results above, only LPP is tested here as the linear manifold learning method. It is encouraging that the UFL-SC pipeline achieves comparable performance with the standard UFL pipeline, and produces the best average accuracy of 90.25% when $L = 100$. Our results also illustrate that neither too small nor too large size of codebook is beneficial to yield satisfactory classification performance, the reason for which is that small size of codebook lacks adequate representation ability, while large size of codebook leads to high-dimensional global histogram features which probably lead to overfitting problem during training SVM classifier. In our experiments on UCM dataset, the appropriate L should range between 40 and 100.

We also compared our proposed UFL-SC method with some off-the-shelf scene classification methods that have reported classification accuracy on the UCM dataset as shown in Table I. Cheriyadat [3] explores a similar UFL pipeline to ours, in which dictionary is learnt via sparse coding and features are encoded in a soft-encoding strategy. We can see that our UFL-SC method outperforms far better than the method of [3]. Zhang *et al.* [54] present a saliency-guided UFL method in which the standard UFL pipeline is explored with the sparse autoencoder (an unsupervised symmetrical neural network) used to learn parameters of network and encode features, while the proposed UFL-SC not only improves the standard UFL but also adopts simpler methods for model learning and feature encoding than the method in [54]. Cheng *et al.* [55] very recently proposed an object-oriented scene classification framework, which utilizes many pretrained part detectors to discovery distinctive visual parts from images, and encodes these visual parts to represent the images, while our UFL-SC-based classification frame is totally local patch oriented. In spite of a little better performance than ours, the computation cost of method in [55] is limited by the number of part detectors. Shao *et al.* [56] report results of basic BOW model with three typical hand-crafted features (SIFT, Colors, LBP). It is obvious that features automatically learnt via our UFL-SC can result in much better classification performance than the three well-designed features under the BOW-based scene classification framework on UCM dataset. This result comparison means that features extracted by UFL-SC is superior to these hand-crafted features in aspect of robustness and discriminative power. In addition, Shao *et al.* [56] propose a hierarchical multiple feature fusion (HMFF) approach that produces the state-of-the-art performance on UCM dataset. In contrast to our method which is free of any hand-crafted features, the HMFF extracts carefully selected hand-crafted features and focuses on the hierarchical fusion of multiple features. Therefore, the better performance produced by HMFF is to be expected, because of its complex fusion strategy and well-designed classification framework. However, the two methods put emphasis on completely different aspects: our UFL-SC focuses on automatically feature extraction, while HMFF focuses on fusion strategy of multifeatures. In other words, we can naturally introduce the features extracted by UFL-SC into the HMFF method. To summarize, our UFL-SC method achieves impressive and comparable classification performance with appropriate parameter settings on the UCM dataset.

2) WHU-RS Dataset: For this dataset, we concentrate on another important parameter—the image patch size r that may affect final classification accuracy because of its close relationship to patch manifold learning and feature encoding phase. On UCM dataset, we fix the patch size to 10×10 pixels in consideration of computational cost; thus, the length of patch vectors remain unchanged in above experiments. Even though this patch size worked well in our experiments, it remains unclear whether it is the best choice. When the patch size changes, the structure of patch manifold changes accordingly and a different dictionary is learned on the manifold correspondingly. On WHU-RS dataset, we investigated the classification performance under different patch size r with d, K, and L varying. For other settings, we use LPP to learn linear patch manifold and set a sampling step of five pixels. It is supposed that the larger patches, which contain more complex spatial patterns, can allow us to learn more helpful features. However as shown in Fig. 11(a) and (b), an interesting observation is that different patch sizes lead to nearly consistent classification performance under any identical parameter configurations. There does not exist an optimal patch size working better than the others through all the experiments. It is probably because in our UFL-SC pipeline, original patch vectors of any size are embedded into a low-dimensional manifold where the most intrinsic features in original patches are preserved, and

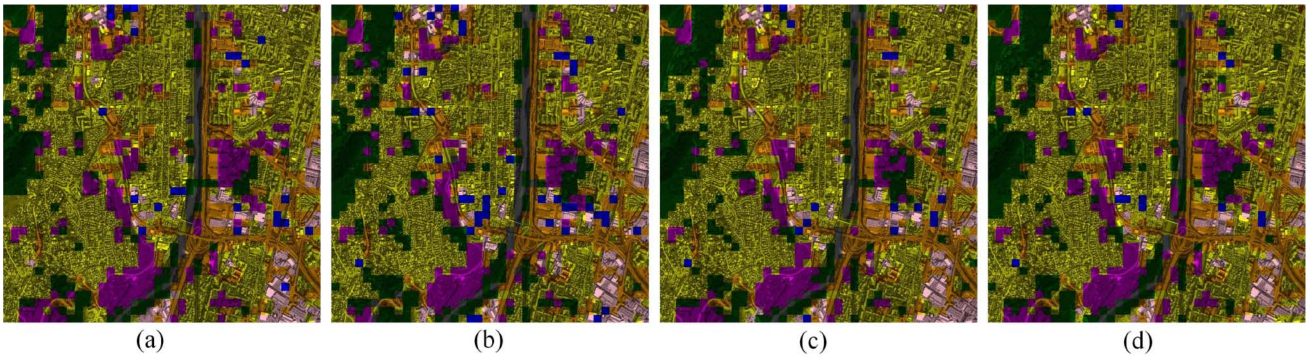


Fig. 12. Annotation results comparison with different linear manifold techniques. (a) LPP (89.20%). (b) PCA (84.21%). (c) RP (85.65%). (d) ORG (84.74%).

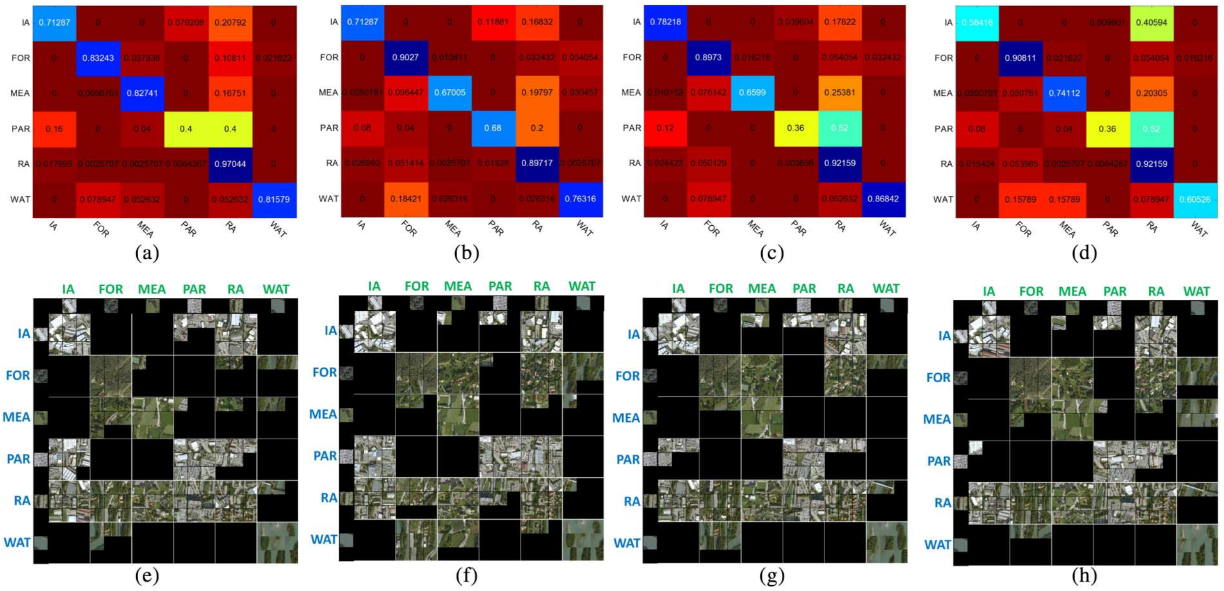


Fig. 13. Confusion matrices of annotation performance on the Grenoble large satellite image. (a)–(d) Confusion matrices under different methods, the rows and columns of matrices denote true and predicted classes [(a) LPP (89.20%). (b) PCA (84.21%). (c) RP (85.65%). (d) ORG (84.74%)]. (e)–(h) Visualized version of corresponding matrices shown in (a)–(d), the rows and columns of matrices denote predicted and true classes [(e) LPP. (f) PCA. (g) RP. (h) ORG].

the low-dimensional representation of image patches of different size shows similar properties. In addition, we note from Fig. 11(c) that too small patch size (e.g., when patch size is set to 4 or 6) generally leads to worse performance than large patch size when the size of codebook is relatively large. It can be concluded that when patch size is above 6, it does not have critical influence on final classification performance. According to our results, the patch size r can be assigned to be any value between 8 and 20. In practical use, we suggest to choose small patch size because small patch size can speed up both the linear patch manifold learning and the feature encoding phase.

3) Large Satellite Scene of Grenoble: For this annotation task, we set the parameter d, K, and L to be 50, 36, and 50, respectively, following previous experimental experience. Three linear manifold learning methods, which are LPP, PCA, and RP, are tested under UFL-SC pipeline on this large image. The final annotation performance is shown in Fig. 12. LPP still outperforms PCA and RP a lot as we expected, and leads to the highest accuracy 97% for the RA that is the dominant class in the large image. It is surprising that the proposed UFL-SC pipeline with RP that is a very simple and intuitive method even performs better than the standard UFL under this parameter settings. The confusion matrices showing classification performance are presented in Fig. 13, where we meanwhile show a visualized version for each confusion matrix. Visualized confusion matrix has the capability of reporting the corresponding class label for every specific image scene visually. We note that although RP leads to a satisfactory overall classification accuracy, it generates serious misclassification on the MEA and PAR classes. It is obvious that a majority of confusion occurs between the RA and PAR, IA, and RA classes. On one hand, this is due to the fact that both of the two classes are mainly composed of some similar thematic classes such as road, cars, and buildings. On the other hand, we can see from the visualized confusion matrices that a number of test scenes do not purely contain only one predefined scene class. For these "ill-formed" scenes, it is hard to give them a ground-truth label properly; hence, it leads to misclassification of the SVM classifier. This bad case is inevitable because of our grid partition operation on the large image in contrast to the accurate segmentation.

VII. Conclusion

In this paper, the low-level local features of image scenes are extracted automatically through two major steps: dictionary learning and feature encoding. Our proposed UFL-SC pipeline, which performs the two steps on the low-dimensional image patch manifold learnt by any proper linear manifold learning technique, discovers the intrinsic space of image patches and makes dictionary learning and feature encoding more effective than the standard UFL pipeline. Experimental results show that the encoded features generated by the proposed method not only have comparable discriminative power to the features generated by the standard UFL pipeline with a lower computational cost but also lead to better performance than some typical hand-designed features within the BOW-based scene classification framework. Moreover, our scene classification framework based on the UFL-SC method and the basic BOW representation outperforms a majority of off-the-shelf approaches on the UCM dataset. From the careful evaluation of some parameters in our model, it can be concluded that the dimensionality of the image patch manifold and the length of encoding feature vectors are two key parameters. Both of them can significantly affect the final classification accuracy, and the optimal values of the two parameters vary when different linear manifold learning methods are applied. Otherwise, it is also interesting that the size of image patches appears not to affect resulting performance very much.

The holistic feature representations via the basic BOW model ignore the spatial and semantic information of complex scene. To better understand the scene, one extension to our method would be building the "deep" architecture to learn more discriminative high-level features which can represent some meaningful structures or patterns. In another way, we plan to extract multiscale local features and perform multiresolution analysis on the feature space.

References

[1] G.-S. Xia, W. Yang, J. Delon, Y. Gousseau, H. Sun, and H. Maitre, "Structrual high-resolution satellite image indexing," in *Proc. Int. Soc. Programm. Remote Sens. Symp. TC VII A: 100 Years ISPRS—Adv. Remote Sens. Sci.*, vol. 38, pp. 298–303, Jul. 2010.

[2] W. Shao, W. Yang, and G.-S. Xia, "Extreme value theory-based calibration for multiple feature fusion in high-resolution satellite scene classification," *Int. J. Remote Sens.*, vol. 34, no. 3, pp. 8588–8602, 2013.

[3] A. Cheriyadat, "Unsupervised feature learning for aerial scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 1, pp. 439–451, Jan. 2014.

[4] J. A. Richards and X. Jia, *Remote Sensing Digital Image Analysis: An Introduction*, 4th ed. New York, NY, USA: Springer, 1999.

[5] S. Aksoy, K. Koperski, C. Tusk, G. B. Marchisio, and J. C. Tilton, "Learning bayesian classifiers for scene classification with a visual grammar," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 581–589, Mar. 2005.

[6] W. Yang, X. Yin, and G.-S. Xia, "Learning high-level features for satellite image classification with limited labeled samples," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 8, pp. 4472–4482, Aug. 2015.

[7] M. Pesaresi and A. Gerhardinger, "Improved textural built-up presence index for automatic recognition of human settlements in arid regions with scattered vegetation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 4, no. 1, pp. 16–26, Mar. 2011.

[8] I. A. Rizvi and B. K. Mohan, "Object-based image analysis of high-resolution satellite images using modified cloud basis function neural network and probabilistic relaxation labeling process," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 12, pp. 4815–4820, Dec. 2011.

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

[10] G.-S. Xia, J. Delon, and Y. Gousseau, "Accurate junction detection and characterization in natural images," *Int. J. Comput. Vis.*, vol. 106, no. 1, pp. 31–56, 2014.

[11] T. Ojala, M. Pietikainen, and T. Maenpaa, "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.

[12] G.-S. Xia, J. Delon, and Y. Gousseau, "Shape-based invariant texture indexing," *Int. J. Comput. Vis.*, vol. 88, no. 3, pp. 382–403, 2010.

[13] G. Liu, G.-S. Xia, W. Yang, and L. Zhang, "Texture analysis with shape co-occurrence patterns," in *Proc. of 22nd Int. Con. Pattern Recog.*, 2014, pp. 1627–1632.

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

[15] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2003 pp. 1470–1477.

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

[17] L. Fei-Fei and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2005 vol. 2, pp. 524–531.

[18] P. Quelhas, F. Monay, J.-M. Odobez, D. Gatica-Perez, and T. Tuytelaars, "A thousand words in a scene," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 9, pp. 1575–1589, Sep. 2007.

[19] A. Coates, A. Y. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," in *Proc. Int. Conf. Artif. Intell. Stat.*, 2011 pp. 215–223.

[20] K. Kavukcuoglu, M. Ranzato, R. Fergus, and Y. LeCun, "Learning invariant features through topographic filter maps," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2009 pp. 1605–1612.

[21] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce, "Learning mid-level features for recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010 pp. 2559–2566.

[22] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 9999, pp. 3371–3408, 2010.

[23] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng, "Self-taught learning: Transfer learning from unlabeled data," in *Proc. Int. Conf. Mach. Learn.*, 2007 pp. 759–766.

[24] F. Hu, G.-S. Xia, Z. Wang, L. Zhang, and H. Sun, "Unsupervised feature coding on local patch manifold for satellite image scene classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2014 pp. 1273–1276.

[25] Y. Bengio *et al.*, "Greedy layer-wise training of deep networks," *Proc. Adv. Neural Inf. Process. Syst.*, 2007 vol. 19, p. 153.

[26] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in *Proc. Int. Conf. Mach. Learn.*, 2009 pp. 609–616.

[27] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012 pp. 1106–1114.

[28] K. Kavukcuoglu, P. Sermanet, Y.-L. Boureau, K. Gregor, M. Mathieu, and Y. LeCun, "Learning convolutional feature hierarchies for visual recognition," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 1090–1098.

[29] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1915–1929, Aug. 2013.

[30] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, "What is the best multi-stage architecture for object recognition?" in *Proc. IEEE Int. Conf. Comput. Vis.*, 2009 pp. 2146–2153.

[31] A. Coates and A. Y. Ng, "The importance of encoding versus training with sparse coding and vector quantization," in *Proc. Int. Conf. Mach. Learn.*, 2011 pp. 921–928.

[32] L. K. Saul and S. T. Roweis, "Think globally, fit locally: Unsupervised learning of low dimensional manifolds," *J. Mach. Learn. Res.*, vol. 4, pp. 119–155, 2003.

[33] M. Belkin and P. Niyogi, "Laplacian Eigenmaps for dimensionality reduction and data representation," *Neural Comput.*, vol. 15, no. 6, pp. 1373–1396, 2003.

[34] J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[35] K. Q. Weinberger and L. K. Saul, "Unsupervised learning of image manifolds by semidefinite programming," *Int. J. Comput. Vis.*, vol. 70, no. 1, pp. 77–90, 2006.

[36] X. He and P. Niyogi, "Locality preserving projections," in *Proc. Adv. Neural Inf. Process. Syst.*, 2003 vol. 16, pp. 234–241.

[37] C. Bachmann, T. Ainsworth, and R. Fusina, "Improved manifold coordinate representations of large-scale hyperspectral scenes," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 10, pp. 2786–2803, Oct. 2006.

[38] H.-B. Huang, H. Huo, and T. Fang, "Hierarchical manifold learning with applications to supervised classification for high-resolution remotely sensed images," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 3, pp. 1677–1692, Mar. 2014.

[39] K. Grauman and T. Darrell, "The pyramid match Kernel: Discriminative classification with sets of image features," in *Proc. IEEE Int. Conf. Comput. Vis.*, 205, vol. 2, pp. 1458–1465.

[40] Y. Yang and S. Newsam, "Spatial pyramid co-occurrence for image classification," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011 pp. 1465–1472.

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

[42] B. Fernando, E. Fromont, and T. Tuytelaars, "Mining mid-level features for image classification," *Int. J. Comput. Vis.*, vol. 108, no. 3, pp. 186–203, 2014.

[43] M. Lienou, H. Maitre, and M. Datcu, "Semantic annotation of satellite images using latent Dirichlet allocation," *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 1, pp. 28–32, Jan. 2010.

[44] C. Vaduva, I. Gavat, and M. Datcu, "Latent Dirichlet allocation for spatial analysis of satellite images," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 5, pp. 2770–2786, May 2013.

[45] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, 2003.

[46] J. Hu, G.-S. Xia, F. Hu, and L. Zhang, "Dense v.s. sparse: A comparative study of sampling analysis in scene classification of high-resolution remote sensing imagery," Preprint-arXiv:1502.01097, Jan. 2015.

[47] 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.

[48] K. Shi and S.-C. Zhu, "Mapping natural image patches by explicit and implicit manifolds," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2007 pp. 1–7.

[49] X. He, D. Cai, S. Yan, and H.-J. Zhang, "Neighborhood preserving embedding," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2005 vol. 2, pp. 1208–1213.

[50] S. Maji, A. C. Berg, and J. Malik, "Classification using intersection kernel support vector machines is efficient," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2008 pp. 1–8.

[51] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proc. 18th SIGSPATIAL Int. Conf. Adv. Geogr. Inf. Syst.*, 2010 pp. 270–279.

[52] C.-C. Chang and C.-J. Lin. (2013). *LIBSVM: A Library for Support Vector Machines* [Online]. Available: http://www.csie.ntu.edu.tw/cjlin/libsvm/

[53] L. Liu and P. W. Fieguth, "Texture classification from random features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 574–586, Mar. 2012.

[54] F. Zhang, B. Du, and L. Zhang, "Saliency-guided unsupervised feature learning for scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 4, pp. 2175–2184, Apr. 2015.

[55] G. Cheng, J. Han, P. Zhou, and L. Guo, "Multi-class geospatial object detection and geographic image classification based on collection of part detectors," *ISPRS J. Photogramm. Remote Sens.*, vol. 98, pp. 119–132, 2014.

[56] W. Shao, W. Yang, G.-S. Xia, and G. Liu, "A hierarchical scheme of multiple feature fusion for high-resolution satellite scene categorization," in *Proc. Int. Conf. Comput. Vis. Syst.*, 2013 pp. 324–333.

Fan Hu received the B.S. degree in communication engineering from Wuhan University, Wuhan, China, in 2011. Currently, he is pursuing the Ph.D. degree in image understanding at Signal Processing Laboratory, Electronic Information School, Wuhan University.

His research interests include high-resolution image classification, machine learning, especially deep learning and their applications in remote sensing.

Gui-Song Xia (M'10) received the B.Sc. degree in electronic engineering and the M.Sc. degree in signal processing from Wuhan University, Wuhan, China, in 2005 and 2007, respectively, and the Ph.D. degree in image processing and computer vision from the CNRS LTCI, TELECOM ParisTech (also known as École Nationale Supérieure des Télécommunications), Paris, France, in 2011.

Since March 2011, he has been a Postdoctoral Researcher with the Centre de Recherche en Mathmatiques de la Decision (CEREMADE), CNRS, Paris-Dauphine University, Paris, France, for one and a half years. Currently, he is an Associate Professor with the State Key Laboratory of Information Engineering, Surveying, Mapping, and Remote Sensing (LIESMARS), Wuhan University. His research interests include mathematical image modeling, texture synthesis, image indexing and content-based retrieval, structures from motions, perceptual grouping, and remote-sensing imaging.

Zifeng Wang received the B.S. degree in science and technology of electronic information from Wuhan University, Wuhan, China, in 2014. Currently, he is pursuing the M.S. degree in computer vision at the State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing (LIESMARS), Wuhan University.

His research interests include modeling of image structures, high-resolution remotely sensed imaging, and active learning.

Xin Huang (M'13–SM'14) received the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2009.

He is currently a Full Professor with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan, China (LIESMARS). He has authored more than 50 peer-reviewed articles in international journals. His research interests include hyperspectral data analysis, high-resolution image processing, pattern recognition, and remote sensing applications.

Dr. Huang was a recipient of the Top-Ten Academic Star of Wuhan University, Wuhan, China, in 2009; the Boeing Award for Best Paper in Image Analysis and Interpretation from the American Society for Photogrammetry and Remote Sensing, in 2010; the New Century Excellent Talents in University from the Ministry of Education of China, in 2011; and the National Excellent Doctoral Dissertation Award of China, in 2012. In 2011, he was recognized by the IEEE Geoscience and Remote Sensing Society (GRSS) as a Best Reviewer of the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS. He was the winner of the IEEE GRSS 2014 Data Fusion Contest. Since 2014, he has been an Associate Editor of the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS.

Liangpei Zhang (M'06–SM'08) received the B.S. degree in physics from Hunan Normal University, Changsha, China, in 1982, the M.S. degree in optics from the Chinese Academy of Sciences, Xian, China, in 1988, and the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 1998.

He is currently the Head of the Remote Sensing Division, State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing, Wuhan University. He is also a Chang-Jiang Scholar Chair Professor appointed by the Ministry of Education of China. He is currently a Principal Scientist with the China State Key Basic Research Project (2011–2016) appointed by the Ministry of National Science and Technology of China to lead the remote sensing program in China. He has authored more than 410 research papers. He is the holder of 15 patents. His research interests include hyperspectral remote sensing, high-resolution remote sensing, image processing, and artificial intelligence.

Dr. Zhang is a Fellow of the Institution of Engineering and Technology, an Executive Member (Board of Governor) of the China National Committee of the International Geosphere-Biosphere Programme, and an Executive Member of the China Society of Image and Graphics. He regularly serves as a Cochair of the series SPIE Conferences on Multispectral Image Processing and Pattern Recognition, Conference on Asia Remote Sensing, and many other conferences. He edits several conference proceedings, issues, and Geoinformatics symposiums. He also serves as an Associate Editor of the *International Journal of Ambient Computing and Intelligence*, the *International Journal of Image and Graphics*, the *International Journal of Digital Multimedia Broadcasting*, the *Journal of Geo-spatial Information Science*, the *Journal of Remote Sensing*, and the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING.

Hong Sun (M'01) received the B.S., M.S., and Ph.D. degrees in communications and electrical systems from the Huazhong University of Science and Technology (HUST), Wuhan, China, in 1981, 1984, and 1995, respectively.

She has held teaching and research positions, including a Professor with HUST, from 1984 to 2000. She was a Visiting Scholar at the Conservatoire National des Arts et Métiers, Paris, France, in 1997, and a Visiting Professor at Ecole Nationale Supérieure des Télécommunications (ENST), Paris, France, in 1998, 1999, 2000, 2001, and 2007. Since 1998, when she was with ENST, her research focus has been on SAR and polarimetric SAR image processing and application techniques. Since 2001, she has been with the School of Electronic Information, Wuhan University, Wuhan, China, where she is currently a Professor and the Head of the Signal Processing Laboratory. Her research includes digital signal processing theory and its applications, including works on image interpretation, communication signal processing, and speech signal processing.

Prof. Sun is the Vice Chairman of the Signal Processing Society of the Chinese Institute of Electronics.