Effective and Efficient Midlevel Visual Elements-Oriented Land-Use Classification Using VHR Remote Sensing Images

Gong Cheng, Junwei Han, Lei Guo, Zhenbao Liu, Shuhui Bu, and Jinchang Ren

***Abstract*—Land-use classification using remote sensing images covers a wide range of applications. With more detailed spatial and textural information provided in very high resolution (VHR) remote sensing images, a greater range of objects and spatial patterns can be observed than ever before. This offers us a new opportunity for advancing the performance of land-use classification. In this paper, we first introduce an effective midlevel visual elements-oriented land-use classification method based on "partlets," which are a library of pretrained part detectors used for midlevel visual elements discovery. Taking advantage of midlevel visual elements rather than low-level image features, a partlets-based method represents images by computing their responses to a large number of part detectors. As the number of part detectors grows, a main obstacle to the broader application of this method is its computational cost. To address this problem, we next propose a novel framework to train coarse-to-fine shared intermediate representations, which are termed "sparselets," from a large number of pretrained part detectors. This is achieved by building a single-hidden-layer autoencoder and a single-hidden-layer neural network with an $L0$-norm sparsity constraint, respectively. Comprehensive evaluations on a publicly available 21-class VHR land-use data set and comparisons with state-of-the-art approaches demonstrate the effectiveness and superiority of this paper.**

***Index Terms*—Autoencoder, land-use classification, midlevel visual elements, part detectors, remote sensing images.**

I. Introduction

LAND-use classification plays an important role for a wide range of applications, such as natural geological hazard detection [1], land-use/land-cover (LULC) determination and visual categorization [2]–[14], geospatial object detection [12], [15]–[19], vegetation types mapping [20], urban planning, etc. In the past decades, although significant efforts have been made in developing various image features and classification methods to infer land usage from satellite and aerial images, the effective interpretation of these images remains one of the most challenging problems faced for remote sensing image analysis.

Most previous works [5]–[11], [17] mainly focus on classifying pixels or superpixels (or, rather, the grouping of local homogeneous pixels) into their thematic class by extracting low-level image features (e.g., the texture feature [6], [17], the color feature [5], spatial and spectral information [7], [8], or their hybrids [9]–[11]) for classification. For example, Bhagavathy and Manjunath [17] used texture features to model and classify compound objects with spatially recurrent patterns, such as harbors and golf courses. Li *et al.* [5] presented a new land-cover analysis method by adopting an improved color structure code for segmentation and a support vector machine (SVM) for classification using high-resolution QuickBird data. Chen *et al.* [8] proposed a novel nonlinear technique for hyperspectral image classification by representing each pixel via a kernel sparse representation in the spatial and spectral feature space.

The aforementioned methods have demonstrated impressive performance for a few number of LULC classes; however, pixels, or even superpixels, carry little semantic meanings, which severely limits the descriptive power of the image representation derived. For an automated understanding of the meanings and contents of a remote sensing image, such pixel-level or superpixel-level land-use classification methods are potentially not enough. With the rapid development of remote sensing technology, improvements in the spatial resolution of optical sensors open novel opportunities for advancing the field of land-use classification. In particular, in recent years, in contrast to previous works using pixel-level or superpixel-level approaches for remote sensing image classification, scene-level land-use classification using very high resolution (VHR) remote sensing images has attracted increasing attention [2]–[4], [21], [22].

One typical approach for scene-level land-use classification is the bag of visual words (BOVW) model [23] and its variations [1], [3], [4]. The BOVW model treats each image as a collection of unordered local features (e.g., a scale-invariant feature transform [24] descriptor), quantizes them into a set of visual words, and then computes a compact histogram representation for scene classification. This is found to be robust against spatial variations but ignores the spatial layout of the features. To overcome this drawback, Lazebnik *et al.* [25] proposed a spatial pyramid matching (SPM) kernel and extended the BOVW model by partitioning the image into increasingly

Manuscript received October 10, 2014; accepted December 11, 2014. This work was supported in part by the National Science Foundation of China under Grant 61401357, Grant 61473231, Grant 61333017, and Grant 61202185, and in part by the China Postdoctoral Science Foundation under Grant 2014M552491. *(Corresponding author: Junwei Han.)*

G. Cheng, J. Han, and L. Guo are with the School of Automation, Northwestern Polytechnical University, Xi'an 710072, China (e-mail: junweihan2010@gmail.com).

Z. Liu and S. Bu are with the School of Aeronautics, Northwestern Polytechnical University, Xi'an 710072, China.

J. Ren is with the Department of Electronic and Electrical Engineering, Faculty of Engineering, University of Strathclyde, Glasgow G1 1XW, U.K.

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/TGRS.2015.2393857

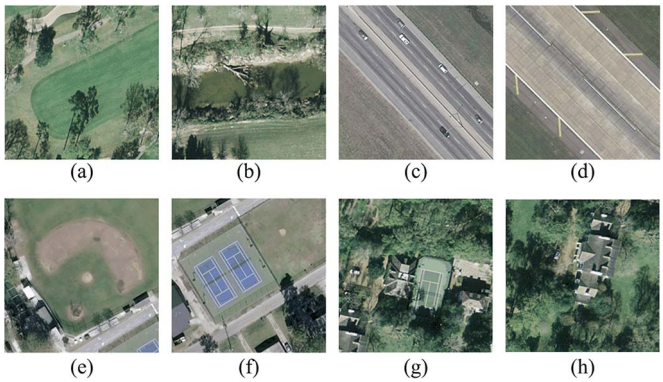

Fig. 1. Eight VHR images from the LULC class. (a) Golf course. (b) River. (c) Freeway. (d) Runway. (e) Baseball diamond. (f) and (g) Tennis courts. (h) Sparse residential.

finer spatial subregions and by computing the histograms of local features from each subregion. Yang and Newsam further extended SPM to a spatial cooccurrence kernel [3] and a spatial pyramid cooccurrence kernel (SPCK) [4] by considering the relative spatial arrangement of the visual words and both the absolute and relative spatial layouts of an image. In addition, rather than directly using low-level features, Cheriyadat [2] explored an unsupervised feature learning method for the scene classification of VHR images, in which sparse feature representations were derived by encoding low-level features in terms of a prelearned basis function set that was generated in an unsupervised manner. Zheng *et al.* [21] presented a novel framework for land-use classification using multifeature joint sparse coding (MFJSC) with a spatial relation constraint.

Although these approaches have produced good results for land-use classification, as the land-use scene classification task becomes more challenging, the description capability of low-level image features is extremely limited or even impoverished. We use eight VHR images in Fig. 1 as examples, which are from a publicly available 21-LULC-class data set [4]. An image classification method based on low-level image features, such as a texture, spectral, or color histogram, would easily misclassify images (a) and (b), (c) and (d), (e) and (f), and (g) and (h) as the same LULC class. Even introducing some spatial layout information would do little to differentiate them accurately. However, humans would classify them as belonging to different LULC classes based on discriminative visual elements (e.g., putting green, water, cars, runway markings, grass lines, a tennis court, buildings, etc.). This example and our visual experiences suggest that a straightforward way to classify challenging image scenes would be a discriminative visual elements-oriented method.

Recently, part-model-based methods have achieved state-of-the-art results for object detection [26] and image classification [27]–[29] on natural scene images. Their success is largely owed to the introduction of the notion of a part detector, which is a linear SVM classifier that can explicitly capture the locations and appearances of some discriminative visual elements. These distinctive visual elements can better complement or substitute low-level image features. Nowadays, VHR remote sensing images have been providing us more detailed spatial and textural information, in which a greater range of objects and recurring spatial patterns can be observed than ever before. With the fine resolution, more and more visual elements, such as cars, trees, buildings, etc., become recognizable and can be separately identified. This provides us a new opportunity to train a great deal of part detectors to further advance the performance of land-use classification by adopting a discriminative visual elements-oriented image representation scheme.

Guided by this observation and motivated by the idea of using part detectors as the basic representation of images, in this paper, we introduce an effective land-use classification method using a library of part detectors that is called "partlets" hereafter. These part detectors are used to detect midlevel visual elements that are more informative than low-level visual words [1], [3], [4], [23] and are meanwhile easier to detect than high-level semantic objects [30], [31]. In the land-use classification scheme, we use discriminative midlevel visual elements rather than individual pixels as attributes to represent images by computing their responses to a large number of pretrained part detectors. This new image representation method could capture much of the high-level meaning and contents of the image, making it more suitable for a complex real-word land-use classification task.

However, as the number of part detectors grows increasingly, a major obstacle to the broader application of the partlets-based method is its computational cost. In this situation, how to share a large number of part detectors is highly desirable due to the potential for gains in the computational cost. Recently, the notion of "sparselets" [32] was introduced as a novel shared intermediate representation for multiclass object detection with a deformable part model (DPM) [26]. In this application, the sparselets are defined as a universal set of shared parts learned from a number of part filters in a sparse coding framework, where each sparselet is regarded as a generic part that is shared between all object classes. With this representation, the part responses of a DPM can be reconstructed as sparse combinations of the sparselets with their corresponding activation vectors. However, the method proposed by Song *et al.* [32] for learning sparselets and activation vectors is obviously brittle, in which sparselets and activation vectors were approximately obtained by using greedy algorithms such as the orthogonal matching pursuit (OMP) algorithm [33], [34] without exploiting the discriminative information hidden in training samples. Although these sparse-coding-based sparselets led to a great computational saving, they also resulted in a substantial loss in the detection accuracy.

Based on partlets, in this paper, we propose a novel and extremely effective framework to train coarse-to-fine sparselets for efficient midlevel visual elements-oriented land-use classification. Specifically, we first train coarse sparselets using a single-hidden-layer autoencoder (SA) [35]–[39]. Then, we simultaneously train fine sparselets and activation vectors using a single-hidden-layer neural network (SNN). In order to adequately explore the discriminative information hidden in the training samples and to make the learned activation vectors sparse, we propose to optimize a new objective function by imposing an $L0$-norm sparsity constraint on the activation vectors.

To sum up, the principal contribution of this paper is threefold. First, we introduce an effective midlevel visual elements-oriented land-use classification method based on a library of pretrained part detectors that is called partlets. By taking advantage of discriminative visual elements rather than low-level image features as attributes, the partlets-based image representation method could capture much of the high-level meaning and contents of an image, making it extremely effective for a challenging land-use classification task. Second and most importantly, to save computational costs while preserving the desired accuracy, we propose a novel and effective framework to train shared intermediate representations, i.e., sparselets, from a large number of pretrained part detectors. This is achieved by training a SA and an SNN with an $L0$-norm sparsity constraint, respectively. The proposed framework can adequately explore the information hidden in the training samples to make the learned activation vectors extremely discriminative, hence yielding efficient VHR image land-use classification with a maximum of fivefold speedup but almost no decrease in the classification accuracy. Third, comprehensive evaluations on a challenging 21-class VHR LULC data set and comparisons with state-of-the-art approaches demonstrate the effectiveness and superiority of this paper. To the best of our knowledge, this result is the best on this data set.

The rest of this paper is organized as follows. Section II introduces partlets and the partlets-based land-use classification method. Section III describes the coarse-to-fine sparselet training in detail. Section IV presents comparative experimental results on a publicly available 21-class VHR LULC data set. Finally, conclusions are drawn in Section V.

II. Partlets

A. Partlets Overview

We use the notion of partlets to represent a library of pretrained part detectors that are trained by taking advantage of the technology of midlevel visual elements discovery [27]–[29] in a weakly supervised learning scheme where only image class labels are required. The trained partlets are used to detect discriminative visual elements and then to compute a visual elements-oriented image representation. This new image representation method could capture much of the high-level meaning and contents of an image, making it extremely effective for a challenging land-use classification task.

Fig. 2 gives an overview of the partlets-based land-use classification method. It is mainly composed of two stages, i.e., partlet training and land-use classification. In the first stage, given an image database, we first train a set of class-specific part detectors for each image class, in the histogram of oriented gradients (HOG) [40] feature space, from the visual clusters of image patches that have a consistent scale, viewpoint, and appearance. Then, the part detectors from all image classes are combined to obtain partlets. In the second stage, we first run the trained partlets to detect discriminative visual elements from each image. Then, we represent the image by computing its response to partlets. Finally, we perform classification by using a simple off-the-shelf classifier such as a linear SVM classifier.

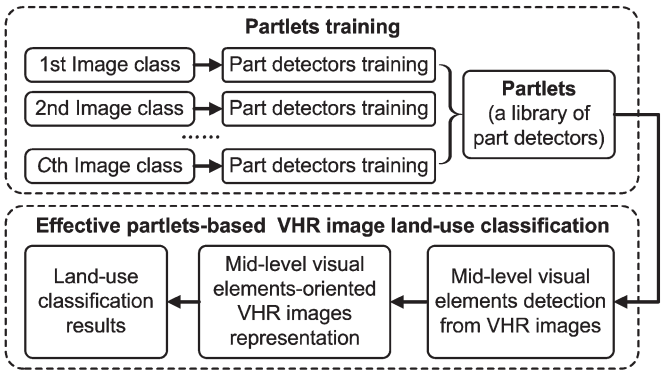

Fig. 2. Overview of the partlets-based VHR image land-use classification method.

B. Partlets Training

Let $\omega = \{\omega_1, \omega_2, \ldots, \omega_C\}$ be a set of C image classes, $\Gamma = \{\Gamma^{(1)}, \Gamma^{(2)}, \ldots, \Gamma^{(C)}\}$ denote partlets for ω, and $\Gamma^{(c)} = \{\Gamma_{c,1}, \Gamma_{c,2}, \ldots, \Gamma_{c,J_c}\}(c = 1, \ldots, C)$ denote a set of class-specific part detectors for image class ω_c, where J_c is the total number of part detectors of $\Gamma^{(c)}$. The training of $\Gamma^{(c)}$ is performed in terms of the following steps [12], [26]–[29].

1) Construct positive training set P_c and negative training set N_c. P_c is composed of the images of class ω_c, and N_c is composed of the images of classes $(\omega - \omega_c)$.
2) Randomly sample a large number (10 000 in this paper) of image patches with the size of $R \times R$ pixels from all images in P_c at ten different scales, and discard highly overlapped patches within the same image scale.
3) Set the initial cluster number to be the value of the total number of refined image patches divided by T, and perform standard k-means clustering over these patches in the HOG feature space; remove the clusters with less than ten members. The value of T should be small because we do not trust that k-means can generalize well [29].
4) Train a part detector $\Gamma_{c,j} = (w_{c,j}, b_{c,j})(j = 1, \ldots, J_c)$ for each cluster by optimizing the following objective function:

$$\arg\min_{(w_{c,j}, b_{c,j})} \left\{ \begin{array}{c} \frac{1}{2}\|w_{c,j}\|^2 + \kappa \sum\limits_{x^+ \in X_{c,j}^+} h\left(w_{c,j}^T \psi(x^+) + b_{c,j}\right) \\ + \kappa \sum\limits_{x^- \in X_{c,j}^-} h\left(-w_{c,j}^T \psi(x^-) - b_{c,j}\right) \end{array} \right\} \tag{1}$$

where $X_{c,j}^+$ and $X_{c,j}^-$ denote the sets of positive examples and negative examples of the jth cluster, respectively, which correspond to the image patches within this cluster and all the hard negative examples of negative training set N_c, respectively. $\psi(x^+)$ and $\psi(x^-)$ denote the HOG feature vectors of positive example x^+ and negative example x^-, respectively. $h(\tau) = \max(0, 1 - \tau)$ is the standard hinge loss function [41] that makes the objective function convex, and κ is a regularization parameter set to 0.1 by following the works in [29] and [42].
5) Run detector $\Gamma_{c,j}$ on P_c to update its cluster by selecting the top n high-scoring detections as its new members.

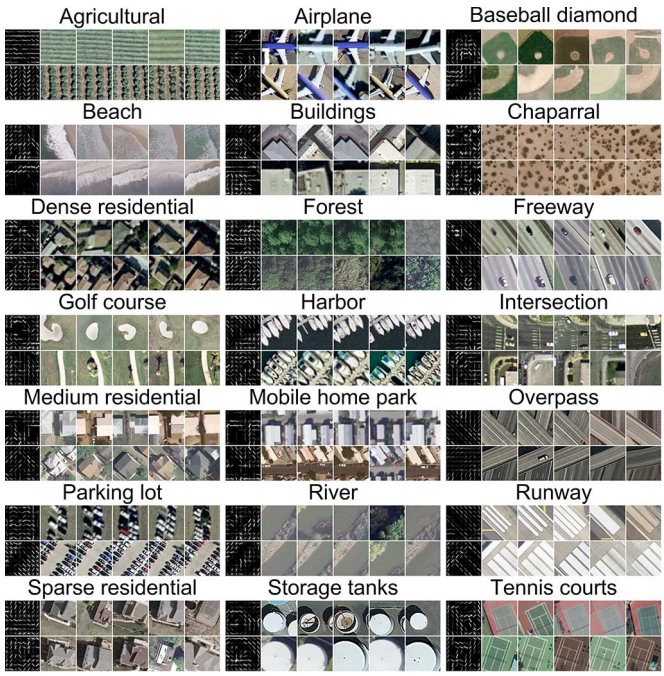

Fig. 3. Visualization of two randomly selected part detectors from each image class and their corresponding top 5 high-scoring detections.

It can be noted that the setting of parameter n is very important, i.e., a smaller value of n makes the detector have a poor generalization ability, whereas a bigger value makes the cluster members less homogeneous.

6) Repeat steps 4 and 5 until convergence is reached to obtain final part detectors $\Gamma^{(c)}$ for image class ω_c.

Fig. 3 shows the visualization of two randomly selected detectors for each image class, which are trained on a publicly available 21-LULC-class data set [4], and their corresponding top 5 high-scoring patches. It is very interesting to see that the detectors can capture many informative contents of the images that seem very intuitive to us, making it more suitable for complex real-word visual recognition tasks. For example, the detectors for the "airplane" class capture the airplanes with different orientations and sizes; the detectors for the "intersection" class capture the turnings and the zebra crossings.

C. Partlets-Based Land-Use Classification

The core of partlets-based land-use classification is to represent images by using discriminative visual elements as attributes, which can be achieved by computing their response to a set of part detectors. Specifically, given an image, for each location S with HOG feature $\psi(S)$, we first run partlets $\Gamma = \{\Gamma^{(1)}, \Gamma^{(2)}, \ldots, \Gamma^{(C)}\}$ on it to compute its response S_{Response} and its corresponding part detector label S_{Label}. S_{Response} is defined as the maximum response of all part detectors to $\psi(S)$ as follows:

$$S_{\text{Response}} = \max_{(w,b)\in\Gamma} \left(w^T\psi(S) + b\right). \quad (2)$$

S_{Label} is defined as the label of the part detector with the maximum response. Next, the top M high-scoring detections, together with their responses and their corresponding part detector labels, are selected. Then, the responses are normalized to [0, 1], and the image is represented as a feature vector by accumulating all normalized responses to their corresponding part detectors. Finally, we train a linear one-against-all SVM classifier for each image class. An unlabeled test image is assigned to a label of the classifier with the highest response.

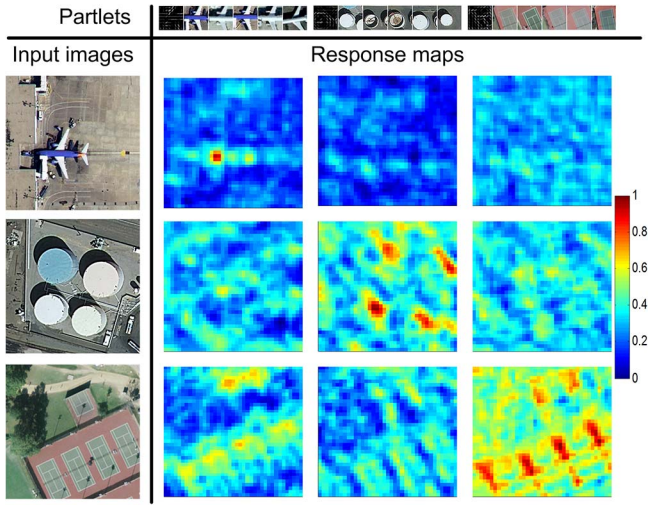

Fig. 4. Response maps of three randomly chosen part detectors applied to three input images, where dark red represents the highest response, and dark blue denotes the lowest response.

Fig. 4 illustrates the response maps of three randomly chosen part detectors applied to three input images. As shown in Fig. 4, these three detectors can accurately fire at their corresponding image patches (an airplane, storage tanks, and tennis courts), which demonstrates that our trained partlets are effective for discriminative visual elements detection.

III. Sparselets

A. Sparselets Overview

Sparselets were first introduced by Song *et al.* [32] as a new shared intermediate representation for accelerating multiclass object detection. In brief, sparselets are a generic dictionary $\mathbf{D} = [\boldsymbol{d}_1, \boldsymbol{d}_2, \ldots, \boldsymbol{d}_K] \in \Re^{m\times K}$ learned from a number of part detectors $\Gamma = [\Gamma_1, \Gamma_2, \ldots, \Gamma_N] = [w_1, w_2, \ldots, w_N] \in \Re^{m\times N}$ (to simplify the formulation, we omit bias term b), where each column $\boldsymbol{d}_k \in \Re^m (k = 1, 2, \ldots, K)$ is called a sparselet, K is the dictionary size, and N is the total number of part detectors, with $N = \sum_{c=1}^{C} J_c$, in this paper. Denoting the HOG feature pyramid of an image as ψ, the computational bottleneck of object detection is the convolution of ψ with a set of detectors, but in the framework of sparselets, the response for each detector $\Gamma_i (i = 1, 2, \ldots, N)$ can be approximated as a sparse linear combination of sparselets $\mathbf{D}$ by

$$\psi^*\Gamma_i \approx \psi^*(\mathbf{D}\alpha_i) = \psi^*\left(\sum_{k=1}^{K}\alpha_{ik}\boldsymbol{d}_k\right) = \sum_{k=1}^{K}\alpha_{ik}(\psi^*\boldsymbol{d}_k) \quad (3)$$

where $\alpha_i = [\alpha_{i,1}, \alpha_{i,2}, \ldots, \alpha_{i,K}]^T \in \Re^K$ is an activation vector of Γ_i with only a few nonzero elements.

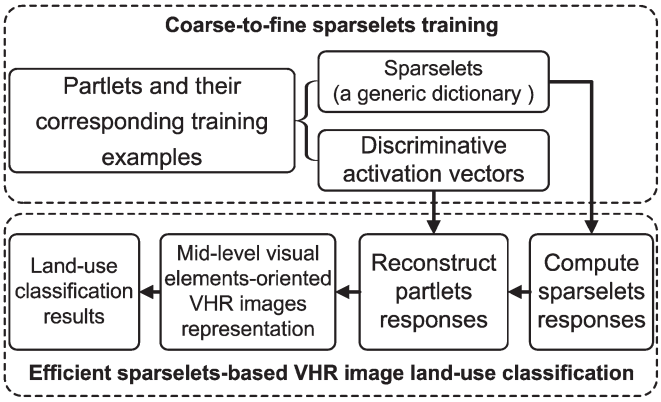

Fig. 5. Overview of the sparselets-based land-use classification method.

We define the speedup factor as the ratio between the time needed to perform partlets-based land-use classification and the time needed for sparselets-based land-use classification. For sparselet dictionary size K, the total number of detectors N, and sparselet dimensionality m, let λ_0 denote the average number of nonzero elements in $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_N]^T \in \Re^{N\times K}$; the partlets-based detection scheme approximately requires Nm operations, and the sparselets-based method only approximately requires $Km + N\lambda_0$ operations, where the first term is from the convolution with sparselets, and the second term is the average activation level from the sparse reconstruction. Consequently, if we ignore the time for low-level feature extraction, the theoretical speedup factor η provided by the sparselet model against the partlet model can be written as

$$\eta = \frac{Nm}{(Km + N\lambda_0)}. \tag{4}$$

To enlarge the speedup factor, dictionary size K should be much smaller than the total number of detectors N, and the average number of nonzero coefficients λ_0 should be much less than sparselet size K. Note that Km is independent of the number of detectors and only depends on the dictionary size, which is fixed. As the number of detectors grows, the cost of computing sparselet responses becomes fully amortized, which leads to a maximum theoretical speedup of m/λ_0 [32].

As aforementioned in the introduction, although the sparse-coding-based sparselets [32] led to a great computational saving, they also resulted in a large decrease in the task performance. In order to adequately explore the discriminative information hidden in the training samples, in the remainder of this section, we propose a novel framework to train coarse-to-fine sparselets for efficient land-use classification. Fig. 5 gives an overview of the proposed sparselets-based land-use classification method, which consists of two stages, i.e., sparselet training and land-use classification. In the first stage, given a library of part detectors and their corresponding training examples, we train sparselets and discriminative activation vectors, respectively. In the land-use classification stage, we first compute the sparselet responses for each image, and then, we approximately reconstruct partlet responses as sparse combinations of the sparselet responses with their corresponding discriminative activation vectors. Finally, images are represented and classified by using the same way as the partlets-based land-use classification method.

B. Coarse-to-Fine Sparselet Training

Fig. 6 gives the framework of our proposed coarse-to-fine sparselet training. We first train coarse sparselets using a SA [35]–[39] while using our pretrained part detectors as a validation set to prevent overfitting. Then, we simultaneously train fine sparselets and discriminative activation vectors using an SNN with an $L0$-norm sparsity constraint to explore the discriminative information hidden in the training samples.

1) Coarse Sparselet Training Based on Autoencoder: The coarse sparselet training is based on a SA, which is a learning architecture used to pretrain neural networks, as shown in Fig. 6(a). Specifically, suppose that we have N part detectors $\Gamma = [\Gamma_1, \Gamma_2, \ldots, \Gamma_N]$ and that each detector has n training examples obtained in Section II-B; the input of the SA is Nn m-dimensional HOG features of the training examples. Let $\boldsymbol{\Phi} = [\Phi^{(1)}, \Phi^{(2)}, \ldots, \Phi^{(Nn)}] \in \Re^{m\times Nn}$ denote the Nn input, where $\Phi^{(r)} = [\phi_{r,1}, \phi_{r,2}, \ldots, \phi_{r,m}]^T \in \Re^m (r = 1, 2, \ldots, Nn)$ denote an input of $\boldsymbol{\Phi}$, and $\hat{\Phi}^{(r)} = [\hat{\phi}_{r,1}, \hat{\phi}_{r,2}, \ldots, \hat{\phi}_{r,m}]^T \in \Re^m$ denote the reconstruction of $\Phi^{(r)}$; our purpose is to learn weights $\mathbf{D} = [\boldsymbol{d}_1, \boldsymbol{d}_2, \ldots, \boldsymbol{d}_K] \in \Re^{m\times K}$ and $\mathbf{D}' = [\boldsymbol{d}'_1, \boldsymbol{d}'_2, \ldots, \boldsymbol{d}'_K] \in \Re^{m\times K}$ to make the output of the reconstruction layer close to the input layer by minimizing the following objective function with an activation sparsity constraint to the hidden layer:

$$F_1(\mathbf{D}, \mathbf{D}'; \Phi) = \frac{1}{2Nn}\sum_{r=1}^{Nn}\left\|\Phi^{(r)} - \hat{\Phi}^{(r)}\right\|^2 + \beta\sum_{k=1}^{K}\mathrm{KL}(\rho\|\hat{\rho}_k) \tag{5}$$

$$\hat{\Phi}^{(r)} = \frac{\mathbf{D}'}{1 + \exp\left(-\mathbf{D}^T\Phi^{(r)}\right)} \tag{6}$$

$$\mathrm{KL}(\rho\|\hat{\rho}_k) = \rho\log\frac{\rho}{\hat{\rho}_k} + (1-\rho)\log\frac{1-\rho}{1-\hat{\rho}_k} \tag{7}$$

$$\hat{\rho}_k = \frac{1}{Nn}\sum_{r=1}^{Nn}\left(1 + \exp\left(-\boldsymbol{d}_k^T\Phi^{(r)}\right)\right)^{-1} \tag{8}$$

where $\mathbf{D}$ is the coarse sparselets to be learned subjected to $\|d_k\|_2 = 1$, $\mathbf{D}'$ is a reconstruction weight that reconstructs the input layer from the hidden layer, K is the number of neurons in the hidden layer corresponding to the sparselet dictionary size, β is the weight of the sparsity penalty, ρ is the target average activation of the hidden nodes, and $\hat{\rho}_k$ is the average activation of the kth hidden node over the Nn training data. The Kullback–Leibler divergence $\mathrm{KL}(\cdot)$ is a standard function providing the sparsity constraint. Here, we set $\beta = 3$ and $\rho = 0.05$, as suggested in [36].

We can easily see that the objective function given by (5) mainly measures an average reconstruction error between input $\Phi^{(r)}$ and reconstruction $\hat{\Phi}^{(r)}$. If the model achieves a good reconstruction, we can be sure that the sparselets have preserved most of the information of the part detectors. In practice, we solve this optimization problem using the limited-memory

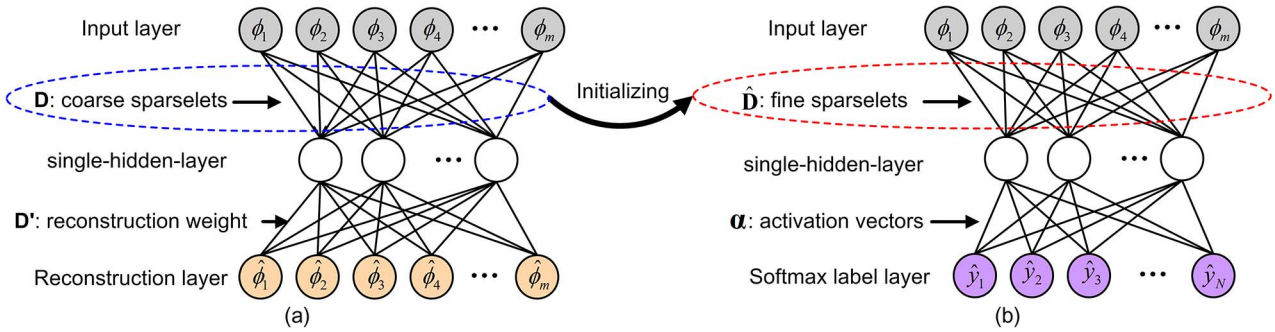

Fig. 6. Framework of our presented coarse-to-fine sparselet training. (a) Coarse sparselet training based on a SA. (b) Fine sparselet and discriminative activation vector training based on an SNN.

Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) algorithm [43] that addresses large-scale data with limited memory.

2) Fine Sparselet and Discriminative Activation Vector Training Based on Neural Network: Notice that we have a large number of training examples with confident labels. In order to incorporate this information to explore the discriminative information hidden in the training examples, we further train fine sparselets $\hat{\mathbf{D}} = [\hat{\boldsymbol{d}}_1, \hat{\boldsymbol{d}}_2, \ldots, \hat{\boldsymbol{d}}_K] \in \Re^{m\times K}$ and simultaneously learn discriminative activation vectors $\boldsymbol{\alpha}$ by building an SNN with an $L0$-norm sparsity constraint, as illustrated in Fig. 6(b). The architectures between the input layers and single-hidden-layers in Fig. 6(a) and (b) are completely identical; thus, we initialize fine sparselets $\hat{\mathbf{D}}$ to be the same as coarse sparselets $\mathbf{D}$. Different from the reconstruction layer in Fig. 6(a), the output layer is now a binary vector with a softmax unit that allows one element to be 1 out of N dimensions for an N-way classification problem. Sparselets $\hat{\mathbf{D}}$ are now learned not only from reconstructing the input data but also from a classifier predicting the labels. Let $y^{(r)} = [y_{r,1}, y_{r,2}, \ldots, y_{r,N}]^T \in \Re^N$ denote the actual label of the rth input and $\hat{y}^{(r)} = [\hat{y}_{r,1}, \hat{y}_{r,2}, \ldots, \hat{y}_{r,N}]^T \in \Re^N$ denote the predicted label of $y^{(r)}$; our new discriminative objective function with an $L0$-norm sparsity constraint can be rewritten as

$$F_2(\hat{\mathbf{D}}, \boldsymbol{\alpha}; \Phi, y) = \frac{1}{2Nn}\sum_{r=1}^{Nn}\left\|y^{(r)} - \hat{y}^{(r)}\right\|^2 + Z(\boldsymbol{\alpha}) \quad (9)$$

$$Z(\boldsymbol{\alpha}) = \frac{\lambda_1}{2}\sum_{i=1}^{N}\|\alpha_i\|_2^2 \text{ subject to } \|\alpha_i\|_0 \le \lambda_0 \quad \forall i = 1, 2, \ldots, N \quad (10)$$

$$\hat{y}^{(r)} = \text{softmax}\left(\boldsymbol{\alpha}\hat{\mathbf{D}}^T\Phi^{(r)}\right) \quad (11)$$

where $\text{softmax}(a_i) = \exp(a_i)/\sum_{i'}\exp(a_i)(i = 1, \ldots, N; a \in \Re^N)$, $\hat{\mathbf{D}}$ is the fine sparselets we attempt to learn subjected to $\|\hat{d}_k\|_2 = 1$, $\boldsymbol{\alpha}$ is the discriminative activation vectors to be learned, λ_1 is a weight decay parameter that controls the relative importance of the two terms, which is set to be 0.001 as suggested in [36], and λ_0 is the number of nonzero elements in each activation vector.

In the discriminative objective function of (9), the first term represents the supervised goal, ensuring that the learned sparselets are also good for discriminating between different part detectors. The second term is a regularization term that tends to decrease the magnitude of the activation vectors and helps prevent overfitting while with the $L0$-norm sparsity constraint to ensure their sparsity. Similar to coarse sparselet training, we solve this optimization problem by using the L-BFGS algorithm [43]. However, as the second term is not a convex optimization problem, we adopt an alternative method to approximately minimize it by employing a two-step process. To be specific, in the first step, based on the learned coarse sparselets $\mathbf{D}$, we initialize the activation vectors by minimizing the average reconstruction error between all part detectors and their reconstruction approximation via the following formulation:

$$\min \frac{1}{N}\sum_{i=1}^{N}\|\Gamma_i - \mathbf{D}\alpha_i\| \text{subject to} \|\alpha_i\|_0 \le \lambda_0 \quad \forall i = 1, 2, \ldots, N. \quad (12)$$

Here, we use the OMP algorithm [33], [34] implemented in the SPArse Modeling Software (SPAMS) package [34] to optimize (12). In the second step, the initialization of nonzero variables is fixed, which leads to the satisfaction of the sparsity constraint and results in a convex optimization problem to solve. We then discriminatively learn the selected variables according to (9).

C. Sparselets-Based Land-Use Classification

Once we have finished the training of sparselets and activation vectors, given an image with its HOG feature pyramid denoted as ψ, we can precompute its convolutions with all sparselets. Then, we can use the activation vectors trained for each part detector to approximately reconstruct the response obtained from the convolution with the original detector, as formulated in (3). In brief, we can recover the individual part detector response via a sparse matrix multiplication, with the activation vector replacing the exhaustive convolution operation as follows:

$$\psi^*\Gamma = \begin{bmatrix}\psi^*\Gamma_1\\ \psi^*\Gamma_2\\ \vdots\\ \vdots\\ \psi^*\Gamma_N\end{bmatrix} \approx \begin{bmatrix}\alpha_1\\ \alpha_2\\ \vdots\\ \vdots\\ \alpha_N\end{bmatrix}\begin{bmatrix}\psi^*\boldsymbol{d}_1\\ \psi^*\boldsymbol{d}_2\\ \vdots\\ \psi^*\boldsymbol{d}_K\end{bmatrix} = \boldsymbol{\alpha}\mathbf{S} \quad (13)$$

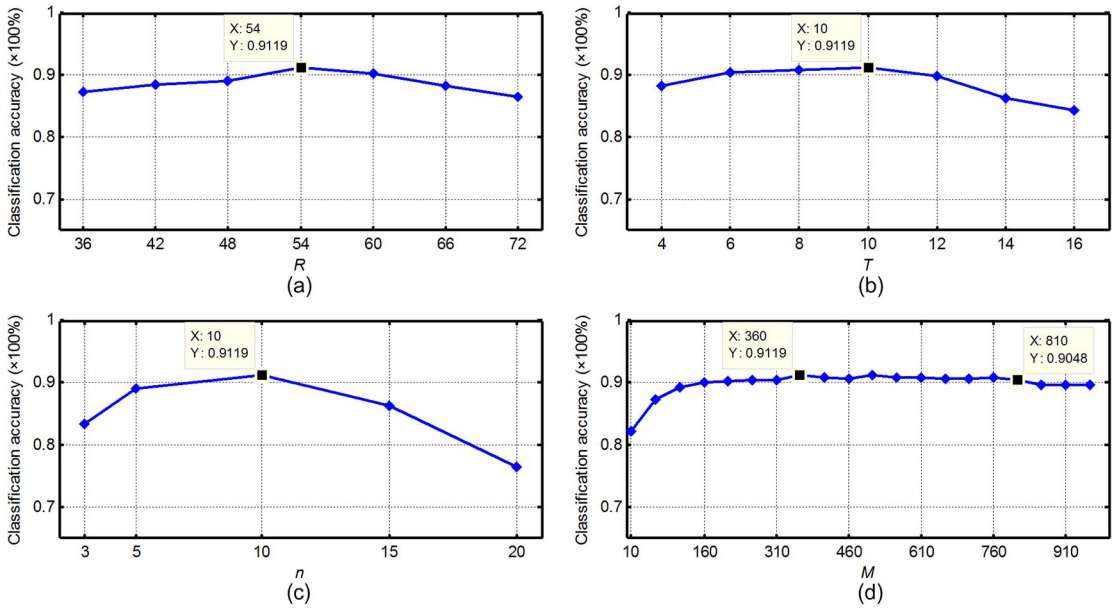

Fig. 7. Classification performance under variation in various parameters. (a) Varying the dimension of the image patch. (b) Varying the parameter used for initializing the number of part detectors. (c) Varying the number of top high-scoring detections of each part detector when updating its corresponding cluster. (d) Varying the number of top high-scoring detections of an image when computing its feature representation.

where $\mathbf{S}$ is the matrix of all sparselet responses, and $\boldsymbol{\alpha}$ is the matrix of sparse activation vectors. Note that the summation is only over the nonzero elements of sparse vector α_i, which could be efficiently implemented as sparse matrix multiplications. Finally, images are represented and classified by using the same way as the partlets-based image classification method, as described in Section II-C.

IV. Experiments

A. LULC Data Set Description

We comprehensively evaluate the performance of our method on a publicly available LULC data set [4]. The data set is composed of the following 21 LULC classes: agricultural, airplane, baseball diamond, beach, buildings, chaparral, dense residential, forest, freeway, golf course, harbor, intersection, medium-density residential, mobile home park, overpass, parking lot, river, runway, sparse residential, storage tanks, and tennis courts. Each class consists of 100 images measuring 256×256 pixels, with a pixel resolution of 30 cm in the red–green–blue color space. This data set has been widely used in [2]–[4], [21], and [22] for evaluating their land-use classifiers.

B. Experimental Setting

When detecting discriminative visual elements, each image is represented by a multilevel HOG feature pyramid, and each octave contains five levels. We follow the construction in [40] to extract the 36-dimensional HOG feature and then project it onto a lower 31-dimensional space, as described in [15], [26], and [29]. We use a 31-dimensional HOG feature because it includes both contrast-sensitive information and contrast-insensitive information. As demonstrated in [26], this 31-dimensional feature can improve the object detection performance than the original HOG feature, where 27 dimensions correspond to different orientation channels (9 contrast insensitive and 18 contrast sensitive), and 4 dimensions are used to capture the overall gradient energy. Suppose the minimum image patch that each part detector could detect is $R \times R$ pixels; the total number of feature pyramid levels is $\lfloor 5 \log_2 \min(\text{rows},\text{cols})/R \rfloor + 1$, where rows and cols denote the image sizes in pixels in the row and the column, respectively. This results in the maximum image patch that each part detector could detect being as large as a LULC image chip.

To make a comprehensive comparison with state-of-the-art methods [2]–[4], [21], [22] that have been evaluated on the LULC data set, two different experimental settings are considered.

1) Setting One: We evaluate the approach using the same fivefold cross-validation methodology in [2]–[4]. To be specific, the images of each class are randomly divided into five equal nonoverlapping sets. For each LULC class, we select four sets for training and evaluated on the held-out set. The classification accuracy is the fraction of the held-out images of 21 classes that are correctly labeled.

2) Setting Two: Following the experimental setup in [21] and [22], the data set was randomly split into 50% for training and 50% for testing. To obtain reliable results, we repeated the experimental process ten times and averaged the results.

C. Parameter Optimization

In the implementation of partlets-based land-use classification, there are four critical parameters associated with the classification performance, i.e., the dimension of the image patch (R in partlet training), the parameter used for initializing the number of part detectors (T in partlet training), the number of top high-scoring detections of each part detector when updating

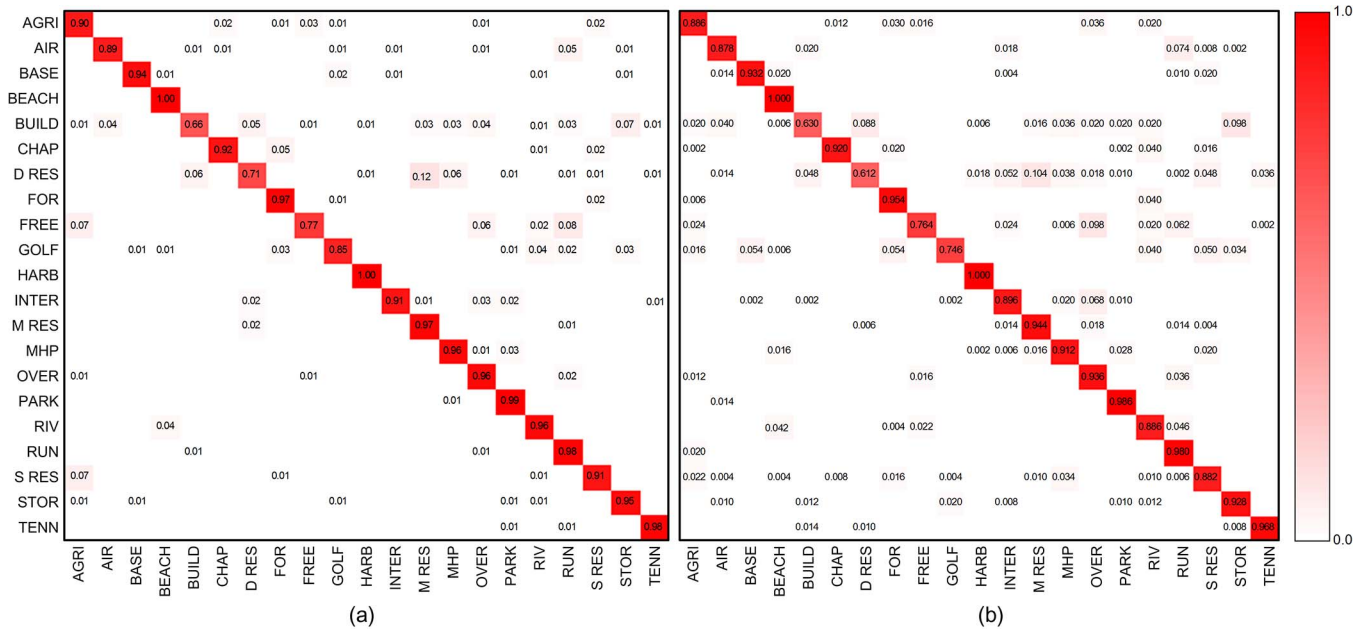

Fig. 8. Confusion matrix for experimental settings one and two. (a) Setting one: 80 training samples per class. (b) Setting two: 50 training samples per class.

TABLE I
AVERAGE CLASSIFICATION ACCURACY (IN PERCENTAGE) OF EIGHT DIFFERENT METHODS

Methods	Number of training samples per class	
	80 (setting one)	50 (setting two)
MFJSC [21]	--	77.33
GRID-FLH [22]	--	79.2
Method of [2]	81.67±1.23	--
BOVW [3]	76.81	--
Color-HLS [3]	81.19	--
Texture [3]	76.91	--
SPCK++ [4]	77.38	--
Partlets-based method	**91.33±1.11**	**88.76±0.79**

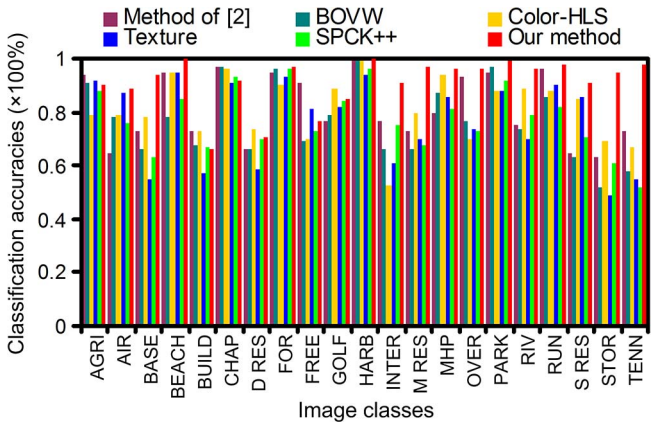

Fig. 9. Per-class classification accuracy of six different methods for setting one.

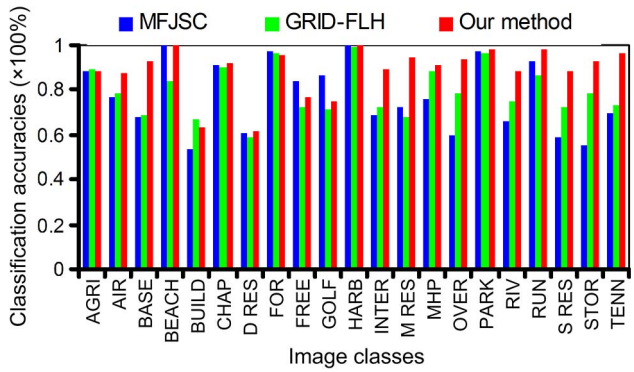

Fig. 10. Per-class classification accuracy of three different methods for setting two.

TABLE II
AVERAGE CLASSIFICATION ACCURACY (IN PERCENTAGE) OF THE PARTLETS-BASED METHOD WITHOUT AND AFTER IMAGE ROTATION

	Number of training samples per class	
	80 (setting one)	50 (setting two)
Without image rotation	91.33±1.11	88.76±0.79
After image rotation	90.61±1.02	88.12±0.67

its corresponding cluster (n in partlet training), and the number of top high-scoring detections of an image when computing its feature representation (M in image feature vector computing). Consequently, we first investigate how the classification performance is affected by these parameters by designing optimization experiments on the first held-out set of setting one. In our implementation, we set $R = \{36, 42, 48, 54, 60, 66, 72\}$, $T = \{4, 6, 8, 10, 12, 14, 16\}$ and $n = \{3, 5, 10, 15, 20\}$, and we varied M from 10 to 960 with a stride of 50. Fig. 7 shows the classification performance under variation in various parameters. As can be seen, the classification performance was improved and then dropped off with the increase in R, T, and n, and the classification performance was improved and then stabilized in a certain range with the increase in M. In particular, when we fixed R, T, and n to be 54, 10, and 10, respectively,

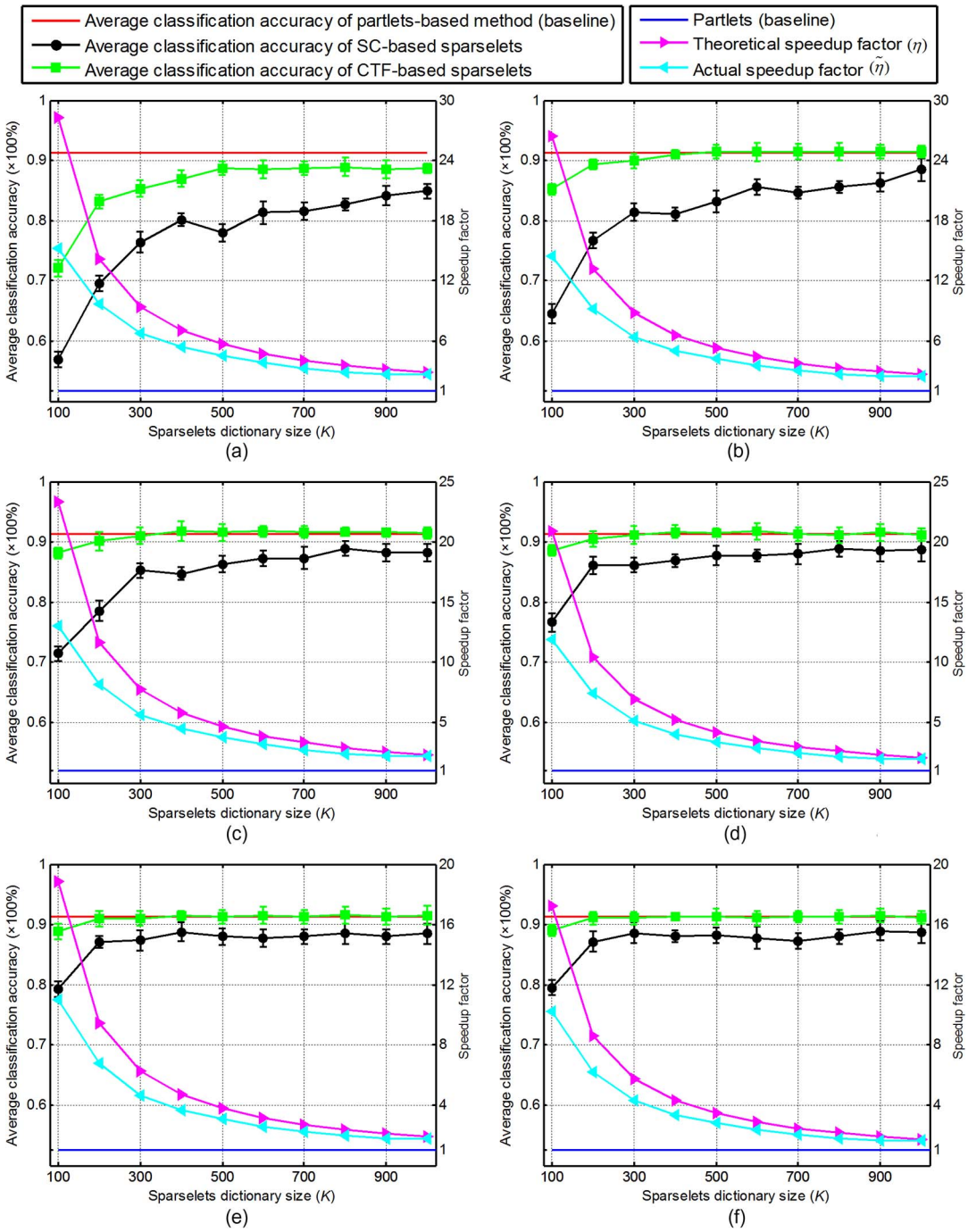

Fig. 11. Average classification accuracy, theoretical speedup factors, and actual speedup factors obtained with different methods, different sparselet dictionary sizes, and different sparsity levels. The parameters used for the baseline are $R = 54$, $T = 10$, $n = 10$, and $M = 360$. (a) Sparsity level: 0.95. (b) Sparsity level: 0.9. (c) Sparsity level: 0.8. (d) Sparsity level: 0.7. (e) Sparsity level: 0.6. (f) Sparsity level: 0.5.

and then changed M from 360 to 810, the classification accuracy only varies in the range of [0.9048, 0.9119]. Consequently, we empirically set $R = 54$, $T = 10$, $n = 10$, and $M = 360$ in all our subsequent land-use classification evaluations.

D. Partlets-Based Land-Use Classification Results and Comparisons

Fig. 8 gives the confusion matrix for experimental settings one and two, where the entry in the ith row and the jth column denotes the rate of test images from the ith class that is classified as the jth class. In Fig. 8, we observed that, for 16 out of 21 LULC classes, we have a classification rate higher than 90% and 88% for settings one and two, respectively. In particular, for the beach and harbor classes, the classification rate is 100%.

Table I presents the average classification accuracy over the 21 classes of our method and some state-of-the-art approaches, as detailed in [2]–[4], [21], and [22]. We also give the per-class classification accuracy for settings one and two in

Figs. 9 and 10, respectively. The comparison with state-of-the-art methods shows that our method can improve the best published results in [2]–[4], [21], and [22] by 9.56% and 9.66% for the two different experimental settings, respectively. To our best knowledge, this result is the best on this data set, which adequately shows the effectiveness and superiority of this partlets-based land-use classification approach.

To further demonstrate that our partlets-based method is robust against image rotation, we augmented the testing set (settings one and two) by simply rotating each testing image in the step of 90° from 0° to 360°. Thus, we can obtain four times of samples after the image rotation operation. Table II shows the average classification accuracy of our partlets-based method without and after image rotation. As shown in Table II, after image rotation, the average classification accuracy has a small decrease (0.72% and 0.64% for settings one and two, respectively), but considering the significant enlargement of testing samples, the reduction is reasonable.

E. Sparselets-Based Land-Use Classification Results and Comparisons

To illustrate the availability of the proposed sparselet training framework, we evaluate the baseline method [32] and our method based on our pretrained part detectors on the 21-LULC-class data set [3], [4] using experimental setting one. The total number of part detectors on all five held-out sets used for sparselet training is $N = \{3093, 3140, 3072, 3110, 2822\}$, respectively. In the rest of this paper, we will call the baseline sparselets learned by the sparse coding method as SC-based sparselets and the sparselets learned by our proposed coarse-to-fine training framework as CTF-based sparselets.

We define the sparsity level as the rate of zero entries in the matrix of sparse activation vectors $\boldsymbol{\alpha}$ and set it to be the set of $\{0.95, 0.9, 0.8, 0.7, 0.6, 0.5\}$, and we set sparselet size K to be the set of $\{100, 200, 300, 400, 500, 600, 700, 800, 900, 1000\}$. Fig. 11 shows the average classification accuracy, theoretical speedup factors, and actual speedup factors obtained with different methods, different sparselet dictionary sizes, and different sparsity levels, where the parameters used for our partlet baseline are $R = 54, T = 10, n = 10$, and $M = 360$. As shown in Fig. 11, first, our proposed CTF-based method is much better than the SC-based method under the same speedup factor at all sparsity levels. The difference in the classification accuracy between the SC-based method and the proposed method decreased and then approximately stabilized in a certain range as the value of K increases. Second, the average classification accuracy of our method was improved with the increase in dictionary size K, and then, it approximately stabilized in a certain level when K is bigger than an appropriate value. Third, the average classification accuracy of our method was improved with the decrease in the sparsity level (from 0.95 to 0.8), and then, it stabilized in a certain level when varying the sparsity level from 0.7 to 0.5. Fourth, there is a tradeoff between the average classification accuracy and the speedup factor. A smaller dictionary size and a higher sparsity level can obtain a bigger speedup factor but low average classification accuracy, and vice versa. Fifth, when the sparsity level is smaller than or equal to 0.8 and dictionary size K is equal to or bigger than 300, our method can obtain almost the same classification accuracy as the partlets-based method (the highest accuracy even outperformed the partlets-based method slightly) but with a maximum of fivefold speedup. Sixth, there is a disparity between the theoretical speedup factor and the actual speedup factor because of the unoptimized implementation of our sparse matrix multiplications. Consequently, how to efficiently optimize the sparse matrix multiplications is one of our future works.

The partlets-based method and the sparselets-based method were both implemented on a Windows system and the MATLAB R2010b platform. On a 24-core Lenovo server with an Intel Xeon central processing unit of 2.8 GHz, for a VHR remote sensing image chip with 256×256 pixels, our partlets-based discriminative image patch detection took 3.8428 s on average, and the subsequent image feature vector computation and image classification took 0.0036 s. However, with almost the same image classification accuracy (91.14%) as the partlet baseline (91.33%), by adopting our proposed sparselets-based method (when $K = 300$ and the sparsity level is equal to 0.7), the discriminative image patch detection task only took 0.7513 s. Considering the large-scale and high-coverage characteristic of VHR remote sensing images, the potential benefit of the computation gain is very considerable and appealing.

V. Conclusion

In this paper, we have first introduced an effective partlets-based land-use classification method using VHR remote sensing images, improving the best published results in [2]–[4], [21], and [22] by 9.56% and 9.66% for two different experimental settings, respectively. Then, we proposed and formulated a new framework to train sparselets to achieve efficient VHR image land-use classification, obtaining almost the same classification accuracy as the partlets-based method (the highest accuracy even outperformed the partlets-based method slightly) but with a maximum of fivefold speedup.

Our future works mainly include integrating more low-level features (e.g., spectral information, color features, contextual cues, etc.) with our framework to help further improve the classification accuracy, optimizing the implementation of our algorithm including the sparse matrix multiplications technique to increase the actual speedup factor, and testing our method on more visual recognition tasks in the field of remote sensing image analysis, such as multiclass geospatial object detection and geographic image retrieval.

References

[1] G. Cheng *et al.*, "Automatic landslide detection from remote-sensing imagery using a scene classification method based on BoVW and pLSA," *Int. J. Remote Sens.*, vol. 34, no. 1, pp. 45–59, Jan. 2013.

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

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

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

[5] H. Li, H. Gu, Y. Han, and J. Yang, "Object-oriented classification of high-resolution remote sensing imagery based on an improved colour structure code and a support vector machine," *Int. J. Remote Sens.*, vol. 31, no. 6, pp. 1453–1470, Mar. 2010.
[6] X. Huang, L. Zhang, and L. Wang, "Evaluation of morphological texture features for mangrove forest mapping and species discrimination using multispectral IKONOS imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 6, no. 3, pp. 393–397, Jun. 2009.
[7] N. Longbotham *et al.*, "Very high resolution multiangle urban classification analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 4, pp. 1155–1170, Apr. 2012.
[8] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Hyperspectral image classification via kernel sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 1, pp. 217–231, Jan. 2013.
[9] S. Moustakidis, G. Mallinis, N. Koutsias, J. B. Theocharis, and V. Petridis, "SVM-based fuzzy decision trees for classification of high spatial resolution remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 1, pp. 149–169, Dec. 2012.
[10] J. Munoz-Mari, D. Tuia, and G. Camps-Valls, "Semisupervised classification of remote sensing images with active queries," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 10, pp. 3751–3763, Sep. 2012.
[11] D. Tuia, M. Volpi, M. Dalla Mura, A. Rakotomamonjy, and R. Flamary, "Automatic feature learning for spatio-spectral image classification with sparse SVM," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 10, pp. 6062–6074, Oct. 2014.
[12] 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, Dec. 2014.
[13] L. Shao, L. Liu, and X. Li, "Feature learning for image classification via multiobjective genetic programming," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 7, pp. 1359–1371, Jun. 2014.
[14] J. Tang, L. Shao, and X. Li, "Efficient dictionary learning for visual categorization," *Comput. Vis. Image Understand.*, vol. 124, pp. 91–98, Jul. 2014.
[15] G. Cheng *et al.*, "Object detection in remote sensing imagery using a discriminatively trained mixture model," *ISPRS J. Photogramm. Remote Sens.*, vol. 85, pp. 32–43, Nov. 2013.
[16] J. Han *et al.*, "Efficient, simultaneous detection of multi-class geospatial targets based on visual saliency modeling and discriminative learning of sparse coding," *ISPRS J. Photogramm. Remote Sens.*, vol. 89, pp. 37–48, Mar. 2014.
[17] S. Bhagavathy and B. S. Manjunath, "Modeling and detection of geospatial objects using texture motifs," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 12, pp. 3706–3715, Dec. 2006.
[18] D. Zhang *et al.*, "Weakly supervised learning for target detection in remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 4, pp. 701–705, Apr. 2015.
[19] J. Han, D. Zhang, G. Cheng, L. Guo, and J. Ren, "Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 6, pp. 3325–3337, Jun. 2015.
[20] M. Kim, M. Madden, and T. A. Warner, "Forest type mapping using object-specific texture measures from multispectral Ikonos imagery: segmentation quality and image classification issues," *Photogramm. Eng. Remote Sens.*, vol. 75, no. 7, pp. 819–829, Jul. 2009.
[21] X. Zheng, X. Sun, K. Fu, and H. Wang, "Automatic annotation of satellite images via multifeature joint sparse coding with spatial relation constraint," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 4, pp. 652–656, Jul. 2013.
[22] 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, Jul. 2014.
[23] F. F. Li and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2005, pp. 524–531.
[24] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.
[25] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2006, pp. 2169–2178.
[26] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.
[27] Q. Li, J. Wu, and Z. Tu, "Harvesting mid-level visual concepts from large-scale Internet images," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 851–858.
[28] C. Doersch, A. Gupta, and A. A. Efros, "Mid-level visual element discovery as discriminative mode seeking," in *Proc. Conf. Adv. Neural Inf. Process. Syst.*, 2013, pp. 494–502.
[29] S. Singh, A. Gupta, and A. A. Efros, "Unsupervised discovery of mid-level discriminative patches," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 73–86.
[30] 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, Jun. 2014.
[31] F. Zhu and L. Shao, "Weakly-supervised cross-domain dictionary learning for visual recognition," *Int. J. Comput. Vis.*, vol. 109, no. 1/2, pp. 42–59, Aug. 2014.
[32] H. O. Song *et al.*, "Sparselet models for efficient multiclass object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 802–815.
[33] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.
[34] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proc. Int. Conf. Mach. Learn.*, 2009, pp. 689–696.
[35] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986.
[36] A. Ng, "CS294A lecture notes: Sparse autoencoder," Stanford Univ., Standford, CA, USA, 2010.
[37] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. Conf. Adv. Neural Inf. Process. Syst.*, 2008, pp. 1096–1103.
[38] L. Shao, D. Wu, and X. Li, "Learning deep and wide: a spectral method for learning deep networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 12, pp. 2303–2308, Dec. 2014.
[39] J. Han *et al.*, "Background prior based salient object detection via deep reconstruction residual," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.
[40] N. Dalal, and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2005, pp. 886–893.
[41] L. Rosasco, E. Vito, A. Caponnetto, M. Piana, and A. Verri, "Are loss functions all the same?," *Neural Comput.*, vol. 16, no. 5, pp. 1063–1076, May 2004.
[42] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. A. Efros, "What makes Paris look like Paris?," *ACM Trans. Graph.*, vol. 31, no. 4, pp. 101:1–101:9, Aug. 2012.
[43] J. Nocedal, "Updating quasi-Newton matrices with limited storage," *Math. Comput.*, vol. 35, no. 151, pp. 773–782, Jul. 1980.

Gong Cheng received the B.S. degree from Xidian University, Xi'an, China, in 2007 and the M.S. and Ph.D. degrees from Northwestern Polytechnical University, Xi'an, China, in 2010 and 2013, respectively.

He is currently a Postdoctoral Fellow with the School of Automation, Northwestern Polytechnical University, Xi'an, China. His main research interests include computer vision and remote sensing image analysis.

Junwei Han received the B.S. and Ph.D. degrees from Northwestern Polytechnical University, Xi'an, China, in 1999 and 2003, respectively.

He was a Research Fellow with Nanyang Technological University, Singapore; The Chinese University of Hong Kong, Shatin, Hong Kong; Dublin City University, Dublin, Ireland; and the University of Dundee, Dundee, U.K. He is currently a Professor with the School of Automation, Northwestern Polytechnical University. His research interests include computer vision and multimedia processing.

Lei Guo received the B.S. and M.S. degrees from Xidian University, Xi'an, China, in 1982 and 1986, respectively, and the Ph.D. degree from Northwestern Polytechnical University, Xi'an, China, in 1993.

He is currently a Professor with the School of Automation, Northwestern Polytechnical University. His recent research interest focuses on image processing.

Zhenbao Liu received the Ph.D. degree in computer science from the University of Tsukuba, Tsukuba, Japan in 2009.

In 2012, he was a Visiting Scholar with the GrUVi Laboratory, School of Computing Science, Simon Fraser University, Burnaby, BC, Canada. He is currently an Associate Professor with the School of Aeronautics, Northwestern Polytechnical University, Xi'an, China. His research interests include 3-D shape and scene analysis, computer vision, and remote sensing.

Shuhui Bu received the M.S. and Ph.D. degrees from the University of Tsukuba, Tsukuba, Japan, in 2006 and 2009, respectively.

From 2009 to 2011, he was an Assistant Professor with Kyoto University, Kyoto, Japan. He is currently an Associate Professor with the School of Aeronautics, Northwestern Polytechnical University, Xi'an, China. His research interests are concentrated on computer vision and robotics, including 3-D shape analysis, image processing, pattern recognition, 3-D reconstruction, and related fields.

Jinchang Ren received the B.E., M.Eng., and D.Eng. degrees from Northwestern Polytechnical University, Xi'an, China, and the Ph.D. degree in electronic imaging and media communication from Bradford University, Bradford, U.K.

He is currently with the Department of Electronic and Electrical Engineering, Faculty of Engineering, University of Strathclyde, Glasgow, U.K. His research interests focus on visual computing and multimedia signal processing, particularly on semantic content extraction for video analysis and understanding, and more recently, hyperspectral imaging.