



Article

Multiple Superpixel Graphs Learning Based on Adaptive Multiscale Segmentation for Hyperspectral Image Classification

Chunhui Zhao ^{1,2}, Boao Qin ^{1,2}, Shou Feng ^{1,2,*} and Wenxiang Zhu ^{1,2}

¹ College of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China; zhaochunhui@hrbeu.edu.cn (C.Z.); qinboao@hrbeu.edu.cn (B.Q.); zhuwenxiang@hrbeu.edu.cn (W.Z.)

² Key Laboratory of Advanced Marine Communication and Information Technology, Ministry of Industry and Information Technology, Harbin Engineering University, Harbin 150001, China

* Correspondence: fengshou@hrbeu.edu.cn

Abstract: Hyperspectral image classification (HSIC) methods usually require more training samples for better classification performance. However, a large number of labeled samples are difficult to obtain because it is cost- and time-consuming to label an HSI in a pixel-wise way. Therefore, how to overcome the problem of insufficient accuracy and stability under the condition of small labeled training sample size (SLTSS) is still a challenge for HSIC. In this paper, we proposed a novel multiple superpixel graphs learning method based on adaptive multiscale segmentation (MSGLAMS) for HSI classification to address this problem. First, the multiscale-superpixel-based framework can reduce the adverse effect of improper selection of a superpixel segmentation scale on the classification accuracy while saving the cost to manually seek a suitable segmentation scale. To make full use of the superpixel-level spatial information of different segmentation scales, a novel two-steps multiscale selection strategy is designed to adaptively select a group of complementary scales (multiscale). To fix the bias and instability of a single model, multiple superpixel-based graphical models obtained by constructing superpixel contracted graph of fusion scales are developed to jointly predict the final results via a pixel-level fusion strategy. Experimental results show that the proposed MSGLAMS has better performance when compared with other state-of-the-art algorithms. Specifically, its overall accuracy achieves 94.312%, 99.217%, 98.373% and 92.693% on Indian Pines, Salinas and University of Pavia, and the more challenging dataset Houston2013, respectively.

Keywords: hyperspectral image classification; superpixel segmentation; graph learning; sparse representation; multiscale fusion



Citation: Zhao, C.; Qin, B.; Feng, S.; Zhu, W. Multiple Superpixel Graphs Learning Based on Adaptive Multiscale Segmentation for Hyperspectral Image Classification. *Remote Sens.* **2022**, *14*, 681. <https://doi.org/10.3390/rs14030681>

Academic Editor: Naoto Yokoya

Received: 3 January 2022

Accepted: 28 January 2022

Published: 31 January 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Hyperspectral images (HSIs) contain hundreds of bands [1–3], which provide rich spectral information and spatial information [4,5]. The HSI classification (HSIC) has always been an activate research topic. HSIC is widely used in the fields of agriculture [6,7], environment monitoring [8,9] and smart city design [10]. The representations of HSIs are spatially sparse while spectrally correlated; unlike the object-level images classification, HSIC needs to perform dense classification or regression over the whole image, hence how to extract pixel-level spectral features while utilizing local spatial information of HSIs is the core to process HSIs. Recently, computer vision has become a mainstream research direction and has been applied in many fields, such as agronomy [11,12], plant science [13], remote sensing and so on. After deep learning (DL) is applied to HSI classification, the detailed spectral–spatial features can be extracted [14–17]. The authors in [18] proposed a 1D+2D HSIC method, which uses a one-dimensional (1D) convolution kernel to extract spectral features and a two-dimensional (2D) convolution kernel to extract spatial features. Hamida et al. [19] directly used the three dimensional (3D) convolution neural networks (3D-CNN) to extract spatial–spectral features simultaneously. Considering the local details of HSIs, a framework based on kernel-guide deformable convolution and double-window

joint bilateral filter is proposed to capture spatial–spectral features flexibly [20]. These DL-based methods achieved excellent performance but often require a large number of training samples. The limited number of training samples will obviously reduce the classification accuracy of these DL-based methods. However, manually annotating a large number of accurate training samples is extremely labor-intensive and time-consuming in a real scenario [21–23], and thus, the small labeled training sample size (STLSS) problem is always a challenge for HSIC [24,25].

Recently, superpixel (SP) segmentation has been widely applied to HSIC [26,27]. SP-based methods can generate an irregular homogeneous region with similar spatial information [28,29]. The local spatial similarity within an SP can easily enlarge the size of training samples, so the SP segmentation can be utilized to tackle the STLSS problem [30,31]. The most commonly used SP segmentation methods are simple linear iterative clustering (SLIC) [32,33] and entropy rate SP (ERS) [34]. Initially, the support vector machine (SVM) and SP segmentation was first combined together for HSIC by [35]. Then, Zheng et al. [36] proposed an SP-guided training sample enlargement and the distance weighted linear regression-based classification (STSE_DWLR) to solve STLSS problem. Liu et al. [37] applied SP segmentation to deep learning and employed the graph convolutional networks (GCN) to extract global features. They proposed the CNN-enhanced GCN (CEGCN) to integrate the complementary advantages of CNN and GCN, and achieved the pixel- and superpixel-level feature fusion. Sellars et al. [38] introduced a superpixel contracted graph based transductive learning framework, named superpixel graph learning (SGL). The SGL applies the advantages of SP segmentation to HSIC and has good classification performance. These SP-based methods have usually applied the SP segmentation as a pretreating technique and have achieved excellent performance, however, how to select a suitable segmentation scale (the number of SPs), which will affect the results of subsequent methods, still need to be fixed. Some researches have demonstrated that different segmentation scales will seriously affect the classification accuracy of SP-based methods [39–41]. To tackle with this issue, Jia et al. [42] proposed an effective framework to calculate the optimal single segmentation scale and select multiscale, but this method did not consider the spatial complementarity between different segmentation scales. Although multiscale fusion can cope with the manual dependence dilemma [43–46], these non-adaptive multiscale selection methods will select the unwanted fusion scale and cause a negative contribution to the classification accuracy. Later, a novel adaptive multiscale segmentations (AMSs) framework which can select scales through an intra-inter discrimination index $f(s_i)$ was provided by [47]. AMSs adopted a two-step framework that first selects the basic scale and then selects the fusion scales according to the basic scale. The intra-inter discrimination index $f(s_i)$ has satisfactory performance in small scale intervals, however, for hyperspectral data sets with complex spatial distribution, such as that of the University of Pavia which requires large scale to segment, this index may lose its effect.

To reduce the bias and variance of classification results when the number of training samples is very small and select a group of multi-scale superpixel maps with complementary information, a multiple superpixel graphs learning method based on adaptive multiscale segmentation (MSGGLAMS) is proposed for HSIC in this paper. In the proposed method, the multiscale superpixel segmentation maps through SLIC are generated first, and an optimal reference scale (ORS) selection algorithm (ORSSA) is used to select an ORS map as the basic scale. Then, a multiscale selection algorithm based on sparse representation is used to select fusion scale maps that have positive contribution to supplement the spatial information of the ORS. Finally, the classification results of a different fusion scale is obtained by SGL and these results are fused by voting to obtain the final result. The main contributions of this paper are:

- (1) The proposed MSGGLAMS adopts the multiscale-superpixel-based framework, which can reduce the adverse effect of improper selection of a superpixel segmentation scale on the classification accuracy while saving the cost to manually seek a suitable segmentation scale.

- (2) To make full use of the spatial information of different segmentation scales, a novel two-steps multiscale selection strategy is designed to adaptively select a set of complementary multiscales. Specifically, an optimal reference scale selection algorithm (ORSSA) which can select the single basic scale suitable for different data sets is proposed to select an optimal reference scale (ORS). Then, the multiscale selection method base on sparse representation (MSSR) is proposed to select fusion scales that have positive contribution to supplement the spatial information of ORS.
- (3) Multiple superpixel-based graphical models (SGL-based model), which are created via constructed superpixel contracted graph of determined scales (fusion scales pool), are adopted to jointly predict the final classification results, that is, pixel-level labels are determined via the voting results of these different models. This Boosting-like fusion strategy can significantly reduce the bias and instability of the final results, and keep the similar inductive bias of models.

2. Proposed Method

In this section, the proposed HSIC framework is explained in detail. As shown in Figure 1, for an HSI, the principal component analysis (PCA) [30] is first applied to extract top three principal components (PCs), and the candidate scale pool is constructed according to the size of the HSI. Then, the SLIC algorithm was used to segment HSI with different scales to obtain multiscale segmentation maps. A novel optimal reference scale (ORS) selection algorithm (ORSSA) is proposed to select the ORS suitable for different datasets as the basic scale, and the classification results of the ORS map are used to construct a sparse dictionary. Next, a multiscale selection method base on sparse representation (MSSR) is used to select fusion scales that can supplement the spatial information of enlarged samples of ORS map. Finally, multiple SGL models are obtained via constructed superpixel contracted graph with different scales (fusion scales), and the final results are jointly predicted via the voting results of these models.

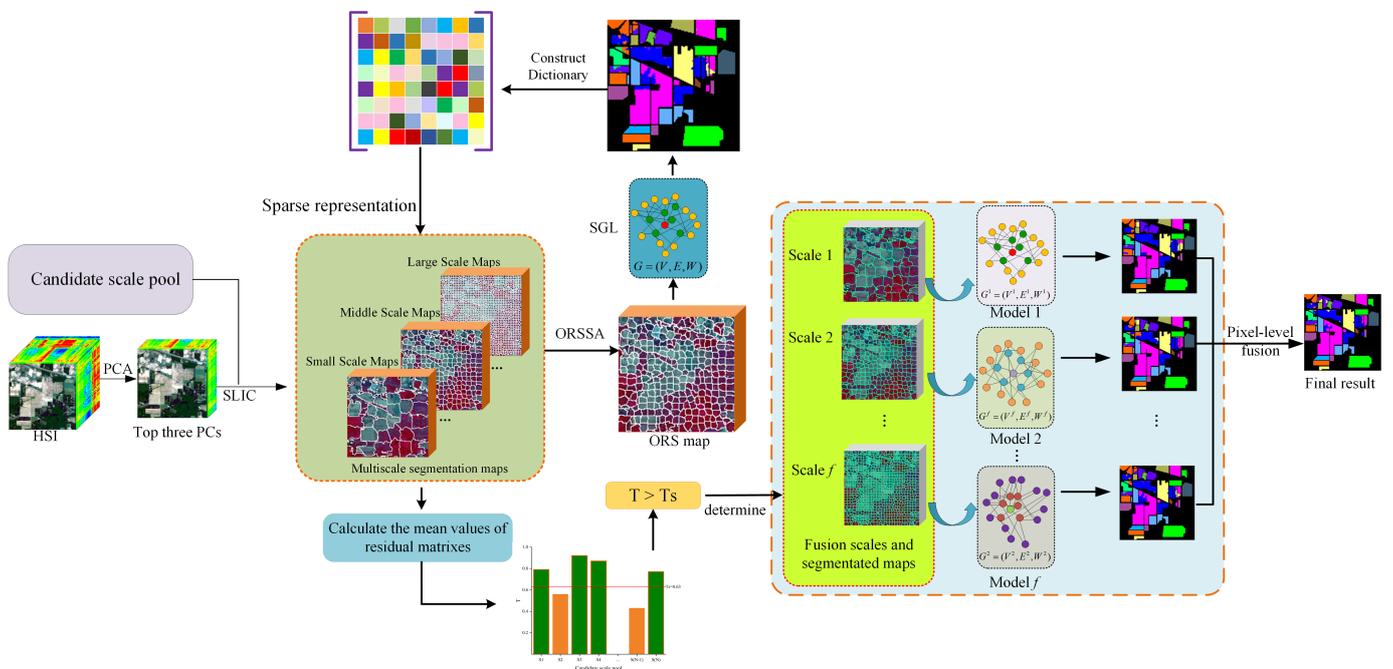


Figure 1. The flowchart of the proposed MSGLAMS method. PCA: principal component (PC) analysis; SLIC: simple linear iterative clustering; ORSSA: optimal reference scale (ORS) selection algorithm; SGL: superpixel graph learning.

2.1. Construction of the Candidate Scale Pool

Different hyperspectral datasets usually reflect different ground distribution. It is necessary to construct candidate scale pools $[S^{lower}, \dots, S^{upper}]$ suitable for different datasets,

because the meaningless SP segmentation scale is not helpful to improve the classification accuracy and will waste computing time. As shown in Figure 2, when the scales of SP exceed the appropriate range, it is a meaningless segmentation. Small-scale segmentation maps cannot preserve the local detailed information of the HSI and can't ensure that the pixels within the SP have similar land-cover categories (undersegmentation). The large-scale segmentation maps will lose large amount of spatial information (oversegmentation) [20,48].

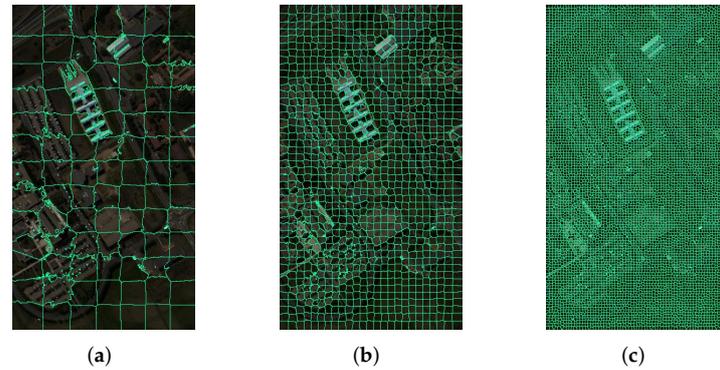


Figure 2. Segmentation maps with different superpixel scale S on University of Pavia. (a) $S = 91$. (b) $S = 1376$. (c) $S = 6856$. (b) is a meaningful segmentation scale map. (a) is an undersegmentation map. (c) is an oversegmentation map.

Therefore, it is important to construct a candidate scale pool that can exclude the meaningless segmentation scale and include the indispensable segmentation scales. A candidate scale pool $[S^{lower}, \dots, S^{upper}]$ in [42] was defined as

$$\text{Interval}(S) = [C \quad C/R] \quad (1)$$

where C is the real number of land cover classes and R is the texture ratio of the HSI. C is usually much smaller than the lower boundary of meaningful segmentation scale, especially for datasets such as that of Pavia University with large image sizes and a large number of edge pixels. In addition, the texture ratio R is easily affected by algorithm of texture detectors. The ideal interval is determined only by the dataset, so the lower boundary and upper boundary are defined as

$$\begin{aligned} S^{lower} &= \text{floor}\left(\frac{L \times W}{K_1}\right) \\ S^{upper} &= S^{lower} \times C \end{aligned} \quad (2)$$

where L and W are the length and width of the HSI. $K_1 = \arg \min(L, W)$. The S^{lower} and S^{upper} defined by this method will not miss meaningful segmentation scales.

If the scale interval is defined as $[S^{lower} : k : S^{upper}]$, the definition of S^{lower} and S^{upper} will only affect the runtime of the algorithm, but the setting of k will affect the selection of fusion scale. As shown in Figure 3, the number of SP cluster centers and the segmentation scale are not linear. If the setting of k is too small, the final fusion scale pool will contain many scales with same segmentation (i.e., non-contribution scales). If k is too large, it is easy to lose the important fusion scale (i.e., positive-contribution scales). So the candidate scale pool is divided into small scale pool $[S^{lower} : k^s : S^M]$, middle scale pool $[S^M : k^m : S^L]$ and large scale pool $[S^L : k^l : S^{upper}]$. Assuming $\Delta = S^{upper} - S^{lower}$, S^M is defined as

$$S^M = S^{lower} + \frac{\Delta}{6} \quad (3)$$

S^L is defined as

$$S^L = S^M + \frac{\Delta}{3}. \quad (4)$$

In addition, $k^s = k/2$, $k^m = 2k$ and $k^l = 3k$, which $k = \Delta/N_S$, N_S is the number of multiscale segmentation maps (generally set to 30). Such a candidate scale pool can retain positive-contribution scales and exclude the meaningless segmentation maps.

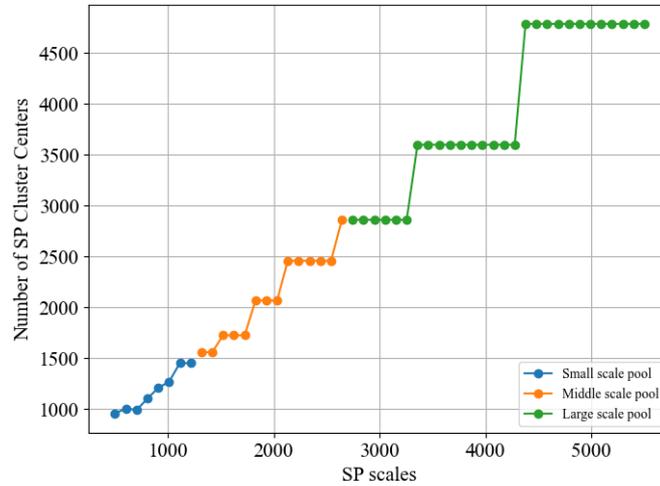


Figure 3. The number of cluster centers corresponding to different SP scale maps obtained by SLIC on University of Pavia. The same number of cluster centers means the same segmentation map.

2.2. Optimal Reference Scale Selection

To select multiple fusion scales, an ORS should be selected as the basic scale first. As shown in Figure 4, the influence of the SP scale on the overall accuracy (OA) is not linear. The small scale SP contains more similar spectral features, while the large scale SP contains richer spatial information. It is difficult to choose a single segmentation scale with good classification performance. Therefore, the optimal reference scale selection algorithm (ORSSA) is designed to choose the optimal single scale suitable for different datasets by calculating the segmentation evaluation indexes of different scale maps.

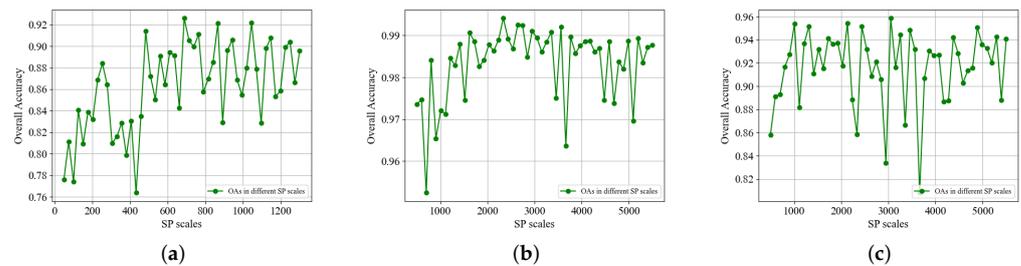


Figure 4. OAs of the SGL in different scales on three HSI datasets. (a) Indian Pines. (b) Salinas (c) University of Pavia.

Small scale maps that contain pure spectral features can obtain high-quality enlarged samples. Large scale maps that contain better spatial information can obtain more enlarged samples. The optimal single scale generally contains similar spectral features and rich spatial features, so the optimal single scale can be only sought with balanced spatial–spectral features in the middle scale pool.

Assuming that the middle scale pool is $S = \{S^1, S^2, \dots, S^K\}$, $S \subseteq [S^M : k^m : S^L]$. The k th scale has M SP regions, $S^k = \{S^k_1, S^k_2, \dots, S^k_M\}$. Then, $p_i \in S^k_m$ is the i th pixel within the m th SP of k th scale, p_c is the corresponding cluster center of S^k_m , (x_i, y_i) and (x_c, y_c) are the spatial coordinates of p_i and p_c , respectively. In order to ensure the similar spectral features of the enlarged sample, the optimal single scale should contain small spectral differences between pixels within an SP, so the spectral evaluation index of the m th SP is defined as

$$f_{\text{spectral}}(S_m^k) = \frac{\sum_{i=1}^{I_m} \cos^{-1}\left(\frac{p_i^T p_c}{\|p_i\| \cdot \|p_c\|}\right)}{I_m} \quad (5)$$

where I_m is the number of pixel within the m th SP. Then, the overall spectral evaluation index of k th scale can be expressed as

$$F_{\text{spectral}}(S^k) = \frac{\sum_{m=1}^M f_{\text{spectral}}(S_m^k)}{M}. \quad (6)$$

In addition, each SP should contain a larger number of pixels, so that SPs with poor spatial information will not easily be randomly selected to enlarge training samples. In order to ensure the size of enlarged samples, we define the spatial evaluation index of m th SP as

$$f_{\text{spatial}}(S_m^k) = \frac{\sum_{i=1}^{I_m} (\sqrt{(x_i - x_c)^2 + (y_i - y_c)^2} / d)}{I_m} \quad (7)$$

where d is the length of the diagonal of m th SP S_m^k . The overall spatial evaluation index of k th scale can be expressed as

$$F_{\text{spatial}}(S^k) = \frac{\sum_{m=1}^M f_{\text{spatial}}(S_m^k)}{M}. \quad (8)$$

The final segmentation evaluation index of k th scale is presented as

$$F(S^k) = (1 - \lambda)F_{\text{spectral}}(S^k) + \lambda F_{\text{spatial}}(S^k) \quad (9)$$

where $F_{\text{spectral}}(S^k)$ and $F_{\text{spatial}}(S^k)$ are the normalized results of (6) and (8), and λ is a spatial–spectral balance factor. The optimal reference scale $\text{ORS} = \arg \max(F(S^k))$. The pseudocode of the ORSSA is given in Algorithm 1.

Algorithm 1 ORSSA

Require: The middle scale pool, $S = \{S^1, S^2, \dots, S^K\}$, $S \subseteq [S^M : k^m : S^L]$; Number of SPs in each scale, $\mathbf{M} = \{M_1, M_2, \dots, M_K\}$; parameter λ ;

Ensure: The optimal reference scale ORS;

- 1: Obtain a series of middle scale maps through SLIC;
 - 2: **for** $k = 1$ to K **do**
 - 3: **for** $m = 1$ to M_k **do**
 - 4: $f_{\text{spectral}}(S_m^k)$ can be calculated by (5);
 - 5: $f_{\text{spatial}}(S_m^k)$ can be calculated by (7);
 - 6: **end for**
 - 7: $F_{\text{spectral}}(S^k)$ can be calculated by (6);
 - 8: $F_{\text{spatial}}(S^k)$ can be calculated by (8);
 - 9: Compute the segmentation evaluation index of k th scale $F_{\text{spectral}}(S^k)$ by (9);
 - 10: **end for**
 - 11: ORS = $\arg \max(F(S^k))$;
 - 12: **return** ORS;
-

2.3. Superpixel Graph Learning

The superpixel graph learning (SGL) proposed by [38] is an SP contracted graph-based transductive learning framework for semi-supervised HSI classification. The SGL can be split into two major tasks: graph construction and label propagation.

2.3.1. Graph Construction

An HSI is denoted as $I \in \mathbf{R}^{L \times W \times B}$ and the reduced image after PCA is denoted as $\hat{I} \in \mathbf{R}^{L \times W \times A}$. Assuming that the i th SP S_i contains n_i connected pixels, $S_i = \{p_{i,1}, \dots, p_{i,n_i}\}$, three meaningful features need to be extracted from each SP ready for graph construction first. The mean features vector of i th SP \vec{S}_i^m is defined as

$$\vec{S}_i^m = \frac{\sum_{j=1}^{n_i} \hat{I}(p_{i,j})}{n_i}. \quad (10)$$

The weighted feature vector \vec{S}_i^w can be expressed as

$$\vec{S}_i^w = \sum_{j=1}^J w_{i,z_j} \vec{S}_{z_j}^m \quad (11)$$

where the weight between adjacent SPs w_{i,z_j} is defined as

$$w_{i,z_j} = \frac{\exp\left(-\|\vec{S}_{z_j}^m - \vec{S}_i^m\|_2^2/h\right)}{\sum_{j=1}^J \exp\left(-\|\vec{S}_{z_j}^m - \vec{S}_i^m\|_2^2/h\right)}. \quad (12)$$

The final centroidal location of each SP \vec{S}_i^p is calculated by

$$\vec{S}_i^p = \frac{\sum_{j=1}^{n_i} p_{i,j}}{n_i}. \quad (13)$$

Next, a weighted undirected graph $G = (V, E, W)$ can be created from these previously discussed features and SP node set. The weight between two connected SPs S_i and S_j is constructed based on two Gaussian kernels and is given as

$$w_{ij} = s_{ij}l_{ij} \quad (14)$$

where

$$s_{ij} = \exp\left(\frac{(\beta - 1)\|\vec{S}_i^w - \vec{S}_j^w\|_2^2 - \beta\|\vec{S}_i^m - \vec{S}_j^m\|_2^2}{\sigma_s^2}\right) \quad (15)$$

$$l_{ij} = \exp\left(\frac{-\|\vec{S}_i^p - \vec{S}_j^p\|_2^2}{\sigma_l^2}\right) \quad (16)$$

where β is the balance factor between the mean and weighted features, and the width of the Gaussian kernels are determined by σ_s, σ_l . The edge set is constructed using k -nearest neighbors. Therefore, the edge weights are defined as

$$W_{ij} = \begin{cases} w_{ij}, & \text{if } i \text{ is one of the } k \text{ nearest neighbor of } j \\ & \text{or vice versa} \\ 0, & \text{otherwise} \end{cases} \quad (17)$$

2.3.2. Label Propagation

A set of labeled spectral pixels are first randomly selected from the original HSI. The final labeling can be specified using a matrix $F \in \mathbf{R}^{K \times c}$. The cost function associated with the matrix F is given by

$$\mathcal{Q}(F) = \frac{1}{2} \sum_{i,j=1}^n W_{ij} \left\| \frac{F_i}{\sqrt{D_{ii}}} - \frac{F_j}{\sqrt{D_{jj}}} \right\|^2 + \frac{\mu}{2} \sum_{i=1}^n \|F_i - Y_i\|^2 \quad (18)$$

where $\mu > 0$ is a regularization parameter, $Y \in R^{K \times C}$ is a labeling matrix contained the labels of K superpixels. \mathcal{F} denotes the set of $n \times c$ matrices with non-negative entries. The labeling matrix is given by $F^* = \operatorname{argmin}_{F \in \mathcal{F}} \mathcal{Q}(F)$. The above cost function has a closed form solution which reads: $F^* = \beta \left(I - \alpha D^{-1/2} W D^{-1/2} \right) Y$, where $\beta = \mu / (1 + \mu)$ and $\alpha = 1 - \beta$. The final labeling of the nodes is then computed as: $y_i = \operatorname{argmax}_{j \leq c} F_{ij}$. The final classification result map of the SP map can be obtained by calculating the label of each SP.

2.4. Multiscale Selection Based on Sparse Representation

As shown in Figure 4, the classification accuracy of a single scale is greatly affected by the selection of SP scale. When the number of samples is limited, the random selection of samples will seriously affect the spectral–spatial features of the enlarged samples after SP segmentation and thus affect the final classification results. The results shown in Figure 5 indicate that choosing a good single scale can improve the accuracy, but the robustness of the single scale is poor. The accuracy of the same single scale under different iterations is also very different because each iteration will randomly selects different samples to obtain corresponding enlarged samples. After using a multiscale fusion strategy, complementary spatial information can improve the classification accuracy under STLSS conditions, and complementary spectral features of enlarged samples can improve the robustness of classification accuracy. Hence, the multiscale fusion strategy has more advantages than the single-scale SP classification algorithm under the condition of STLSS. Hence, the accuracy and robustness of classification can be obviously improved by multiscale fusion.

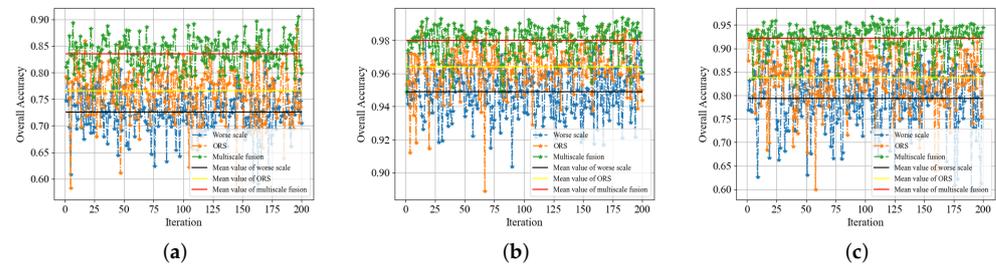


Figure 5. OAs of different scales for 200 iterations on three HSI datasets. The ORS selected by ORSSA, worse single scale, and randomly selected multiscale fusion are labeled in orange ‘*’, blue ‘*’ and green ‘*’, respectively. The three horizontal lines are the mean values of 200 iterations, that is, the bias of different manners. The ‘*’ curves can reflect the variances of different manners. The number of training samples are 3 per class. (a) Indian Pines. (b) Salinas (c) University of Pavia.

The sparse representation can select pixels similar to the spectral features of the sample used to construct the dictionary [49–52], so when the classification result of ORS map is used to construct the dictionary, the fusion scale maps similar to ORS map can be selected by the representation residual. Consequently, the multiscale selection algorithm based on sparse representation (MSSR) can be exploited to select fusion scale maps that have positive contribution to supplement the spatial information of ORS.

The traditional sparse representation assumes that a pixel in HSI is represented as $y \in R^B$, where B is the number of spectral bands. The structured dictionary is denoted by $X = [X^1, X^2, \dots, X^K] \in R^{B \times D}$, where K is the total number of classes and D is the number of training samples. In this algorithm, we suppose the HSI have K distinct classes, and structured dictionary $X = [X^1, X^2, \dots, X^K] \in R^{B \times N}$, where N is the number of all pixels of the ORS map. Assuming the classification result of ORS maps has N_j pixels of j th class, the

j th class subdictionary is $\mathbf{X}^j = [\mathbf{X}_1^j, \mathbf{X}_2^j, \dots, \mathbf{X}_{N_j}^j]$. The superpixel \mathbf{S} can be represented by a linear combination of these reference pixels as

$$\mathbf{S} \approx \alpha_1^j \mathbf{X}_1^j + \alpha_2^j \mathbf{X}_2^j + \dots + \alpha_{N_j}^j \mathbf{X}_{N_j}^j. \tag{19}$$

The m th SP in the i th scale map can be represented as

$$S_m^i = \mathbf{X} \alpha_m^i + \mathbf{N}_m^i \tag{20}$$

where α_m^i is the sparse coefficient matrix. The sparse coefficient matrix α_m^i can be calculated by

$$\hat{\alpha}_m^i = \arg \min \left\| S_m^i - \mathbf{X} \alpha_m^i \right\|_F \quad \text{s.t.} \quad \left\| \alpha_m^i \right\|_0 \leq T \tag{21}$$

where $\|\cdot\|_F$ is the Frobenius norm and the OMP [53] is implemented to solve this problem. Once $\hat{\alpha}_m^i$ is obtained, and the j th class reconstruction residual error which is the difference between original SP and reconstructed SP can be calculated by

$$r^j(S_m^i) = \left\| S_m^i - \mathbf{X}^j \cdot \hat{\alpha}_m^i \right\|_2, j = 1, 2, \dots, K. \tag{22}$$

The residual matrix \mathfrak{R} can be expressed by calculating the minimum residual error of each SP. Mean value of residual matrix of i th scale map can be calculated by

$$\mathbf{E}(S^i) = \text{mean}(\mathfrak{R}(S^i)) = \frac{\sum_{m=1}^M \arg \min(r^j(S_m^i))}{N_c} \tag{23}$$

where M is the number of SPs in the S^i th scale map. Small mean value of \mathfrak{R} means that the segmentation information is more similar to the ORS map. The finally fusion scale pool is determined by

$$\mathbf{S}^F = \left\{ S^1, S^2, \dots, S^f \right\} \quad \text{s.t.} \quad \forall S^i \in \mathbf{S}^F, \mathbf{E}(S^i) \leq T_s \tag{24}$$

where T_s is a threshold and T_s can be presented as

$$T_s = \mathbf{E}_{min} + \frac{\mathbf{E}_{max} - \mathbf{E}_{min}}{\kappa}, \kappa = 1, 2, \dots, K_s \tag{25}$$

where \mathbf{E}_{max} and \mathbf{E}_{min} are the maximum and minimum values of the $\mathbf{E}(S_i)$, respectively. $\kappa = 1$ means that all scales are selected for fusion, and $T_s = \mathbf{E}_{min}$ means that only the ORS map selected by ORSSA is used for fusion. The classification result maps of S^F is obtained by the SGL. Subsequently, these result maps are fused to obtain the final classification result.

2.5. A Pixel-Level Fusion Strategy for Multiple Graphical Models

In general, superpixel-based methods assume that the pixels within the same superpixel share the same label. These superpixel-level predictions can be sub-optimal for dense prediction tasks due to the discrepancy between superpixel-level and pixel-level prediction. Moreover, the instability and bias of the single-scale superpixel based graph classifier comes from the insufficient generalization ability of the single graph model, that is, the single graph cannot well represent the distribution of a real HSI. To fix this gap, a pixel-level fusion strategy for multiple graphical models is proposed in this paper. Similar to Boosting [54], multiple graphical models, which are generated via constructed superpixel contracted graphs of determined scales $\mathbf{S}^F = \left\{ S^1, S^2, \dots, S^f \right\}$, are adopted to jointly predict the label of each pixel. Thereby, assume the f constructed graphs as

$$G^1 = (V^1, E^1, W^1), G^2 = (V^2, E^2, W^2), \dots, G^f = (V^f, E^f, W^f). \tag{26}$$

For the pixel p_i , the labels predicted by different models can be presented as

$$y_i^1 = \operatorname{argmax}_{j \leq c} F_{ij}^1, y_i^2 = \operatorname{argmax}_{j \leq c} F_{ij}^2, \dots, y_i^f = \operatorname{argmax}_{j \leq c} F_{ij}^f \quad (27)$$

where F_{ij} can be obtained by Equation (18) in Section 2.3.2. The final label can be obtained by the majority voting rule, which is presented as

$$y_i = \operatorname{Voting}\{y_i^1, y_i^2, \dots, y_i^f\}. \quad (28)$$

After predicting the label of each pixel, the final classification map is generated. The pseudocode of the MSGLAMS is given in Algorithm 2.

Algorithm 2 The proposed MSGLAMS

Require: Raw HSI dataset $I \in \mathbf{R}^{L \times W \times B}$ with B bands; Training set $A \in \mathbf{R}^{B \times m}$ with m labeled samples; Parameter λ and κ ;

Ensure: A final classification map after multiscale fusion;

- 1: Define the interval of candidate scale pool S^{lower} and S^{upper} by Equation (2);
 - 2: Construct a candidate scale pool composed of the small scale pool, middle scale pool and large scale pool by Equations (3) and (4);
 - 3: Reduced image $\hat{I} \in \mathbf{R}^{L \times W \times A}$ is obtained by PCA.
 - 4: Apply SLIC algorithm on candidate scale pool to create multiscale segmentation maps.
 - 5: Select the ORS by Algorithm 1;
 - 6: Obtain the classification result of ORS $(l_1, l_2, \dots, l_{L \times W})$ by SGL;
 - 7: Construct a sparse dictionary $\mathbf{X} = [\mathbf{X}^1, \mathbf{X}^2, \dots, \mathbf{X}^K] \in \mathbf{R}^{B \times D}$ by the classification results of ORS;
 - 8: **for** $S^i = S^{\text{lower}}$ to S^{upper} **do**
 - 9: Calculate the $\mathbf{E}(S^i)$ through Equation (23);
 - 10: **end for**
 - 11: Determine the final fusion scale pool \mathbf{S}^F by Equation (24);
 - 12: Create multiple graphical models of \mathbf{S}^F by Equation (26).
 - 13: Fuse multiple scale maps to obtain the final classification result by Equations (27) and (28).
 - 14: **return** Final classification result;
-

3. Experimental Results and Analysis

In this section, the details of three experimental datasets and experimental settings are first introduced. Then, the results of the proposed method and other state-of-the-art methods on three HSI datasets are compared. Finally, the effect of the ORASS algorithm and the MSSR algorithm, and the parameter analysis of the proposed MSGLAMS is discussed.

3.1. Datasets

Three HSI datasets are selected for experiments, which are those of the Indian Pines, Salinas and University of Pavia. These datasets are collected by different sensors over different land covers, and have been selected for experiments by many related papers. The details of these datasets are given as follows:

1. The dataset was collected by an Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor over an agricultural site in Indiana. The dataset consists of 145×145 pixels, 200 spectral channels in the wavelength range of 0.4–2.45 μm . The spatial resolution of the data is 20 m. The ground truth map utilizes 10,249 labeled pixels with 16 different surface classes.
2. This image was also collected by the AVIRIS sensor over Salinas Valley, California. It contains 512×217 pixels with a spatial resolution of 3.7 m and 200 spectral channels over 0.4–2.5 μm . The dataset has a spatial resolution of 3.7 m/pixel. Its ground truth contains 54,129 labeled pixels and 16 classes.

3. This dataset was acquired by the Reflective Optics System Imaging Spectrometer (ROSIS) sensor over the urban area surrounding the University of Pavia, Italy. The spatial size of the data is 610×340 , and the spatial resolution is as high as 1.3 m/pixel. The image contains 115 spectral channels from 0.43 to 0.86 μm and has a spatial resolution of 1.3 m. Its ground truth contains 42,776 labeled pixels and 9 classes.
4. The Houston 2013 dataset was captured by an airborne spectrographic image sensor, which covers the area of University of Houston and its neighboring urban area. Houston 2013 dataset include 349×1905 pixels with a spatial resolution of 2.5 m and consists of 144 spectral channels ranging from 0.38 to 1.05 μm , and the ground truth map utilizes 15,029 labeled pixels with 15 different land-cover classes. It adopted the standard training and testing sets given by the 2013 GRSS Data Fusion Contest.

Detailed information of different land cover classes and the corresponding false colors of Indian Pines, Salinas, and University of Pavia are listed in Tables 1–4, respectively.

Table 1. background color and number of land cover classes on indian pines.

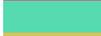
Color	Land Cover Type	Numbers
	Stone-Steel-Towers	93
	Hay-windrowed	478
	Corn-mintill	830
	Soybean-notill	972
	Alfalfa	46
	Soybean-clean	593
	Grass-pasture	483
	Woods	1265
	Buildings-Grass-Tree-Drives	386
	Grass-pasture-mowed	28
	Corn	237
	Oats	20
	Corn-notill	1428
	Soybean-mintill	1257
	Grass-trees	1257
	Wheat	205
Total Number		10,249

Table 2. background color and number of land cover classes on salinas.

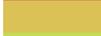
Color	Land Cover Type	Numbers
	Brocoli_green_weeds_1	2009
	Brocoli_green_weeds_2	3726
	Fallow	1976
	Fallow_rough_plow	1394
	Fallow_smooth	2678
	Stubble	3959
	Celery	3579
	Grapes_untrained	11,271
	Soil_vinyard_develop	6203
	Corn_senesced_green_weeds	3278
	Lettuce_romaine_4wk	1068
	Lettuce_romaine_5wk	1927
	Lettuce_romaine_6wk	916
	Lettuce_romaine_7wk	1070
	Vinyard_untrained	7268
	Vinyard_vertical_trellis	1807
Total Number		54,129

Table 3. background color and number of land cover classes on Pavia university.

Color	Land Cover Type	Numbers
	Asphalt	6631
	Meadows	18,649
	Gravel	2099
	Trees	3064
	Painted metal sheets	1345
	Bare Soil	5029
	Bitumen	1330
	Self-Blocking Bricks	3682
	Shadows	947
Total Number		42,776

Table 4. background color and number of land cover classes on Houston2013.

Color	Land Cover Type	Numbers
	Healthy grass	1251
	Stressed grass	1254
	Synthetic grass	697
	Trees	1244
	Soil	1242
	Water	5029
	Residential	126
	Commercial	1244
	Road	1252
	Highway	122
	Railway	1235
	Parking Lot 1	1233
	Parking Lot 2	469
	Tennis Court	428
	Running Track	660
Total Number		15,029

3.2. Experimental Settings

In order to verify the effectiveness of the proposed MSGLAMS comprehensively, the experiment can be divided into three parts. First, the comparison between MSGLAMS and other state-of-the-art methods is conducted. Second, as two key components of the the proposed MSGLAMS, the performances of ORASS and MSSR are also compared with their counterparts, respectively. Finally, the influence of key parameters of the MSGLAMS is analyzed. For the evaluation metrics, overall accuracy (OA), average accuracy (AA) and the kappa coefficient are adopted to quantify the classification performance. All the experiments are repeated ten times to obtain the final average performance. The hardware platform is Intel Xeon Silver 4210 2.4 GHz twelve-core processor CPU, GeForce RTX 3090 GPU and 32G memory.

3.2.1. Experimental Settings for Comparing with Other State of the Arts

In order to validate the performance of our proposed framework MSGLAMS, seven state-of-the-art algorithms are selected for comparison. These methods include three HSI classification methods based on SP segmentation: CEGCN [37], STSE_DWLR [36], and SGL [38]. Two HSI classification methods to solve small sample problems: KDCDWBF [20] and 3D-CNN [19]. Two common HSI classification methods: 1D+2D [18] and SVM. For the Houston, we still can't provide the memory needed for GCN-M method even if we have adopted a decent hardware platform(CPU: Intel Xeon Silver 4210 2.4 GHz twelve-core processor; GPU: GeForce RTX 3090; Memory: 32G).

3.2.2. Experimental Settings of ORSSA

The purpose of proposed ORASS is to select an ORS map with best classification performance as basic scale. The three mainstream single scale selection algorithms are used to compare with ORSSA. First, the texture ratio was proposed in [55] to calculate the optimal single scale (TRS). The ORS calculated by TRS are 706, 1603 and 2765 for the Indian Pines, Salinas Scene, and University of Pavia, respectively. Second, the resolution of the dataset is used to calculate the optimal single scale (RES) in [42], and the RES are obtained as 1051, 3142 and 2216 for the Indian Pines, Salinas Scene, and University of Pavia. Finally, the ORS can be adaptively selected through inter-intra parameters (ASS) in [47], and the parameter setting is consistent with [47]. The spatial-spectral balance factor λ of ORSSA is set to 0.3 on the three datasets.

3.2.3. Experimental Settings of MSSR

The multi-scale comparison experiment is to compare the impact of different multiscale selection methods on the classification results of the same classifier SGL. The multiscale selection comparison methods include the multiscale selection formula (MSF) in [42], the multiscale SP number selection formula (MSNSF) in [56] and the adaptive multiscale selection (AMS) method proposed by [47]. The number of multiscales K in MSF and MSNSF is the same as the number of fusion scales of MSSN and the basic scale of MSF and MSNSF are both set to the ORS. The parameter κ in Equation (25) of MSSR is set to {4, 4, 7} for the Indian Pines, Salinas Scene, and University of Pavia, respectively. The influence of κ will be analyzed in detail in the Section 3.5.

3.3. Comparison of Results of Different Methods

In this section, seven state of the arts are compared with the proposed MSGLAMS framework. Tables 5–8 report the classification accuracy of MSGLAMS and seven methods on the Indian Pines, Salinas and University of Pavia dataset, and the cells with a green background represent the highest accuracy. The classification maps obtained by these methods are illustrated in Figures 6–9. It is easy to find that the accuracy of MSGLAMS far exceeds the common classification methods SVM and 1D+2D because the common classification methods cannot solve the problem of STLSS. Furthermore, the classification accuracy of MSGLAMS also far exceeds that of the small sample classification methods KDCDWBF and 3D-CNN. By comparing three small sample classification methods based on SP segmentation GNC-M, STSE_DWLR and SGL, the OA of MSGLAMS exceeds the best performing comparison method SGL 5.66%, 4.65% on the university of Pavia and the Indian Pines datasets, respectively. Kappa exceeds 7.23%, 4.57% than SGL on these two datasets. GNC-M, STSE_DWLR, SGL and our MSGLAMS perform well on the Salinas dataset, because the texture of Salinas dataset is not complicated. For the more challenging dataset Houston2013, MSGLAMS yields 92.906% overall accuracy, which outperforms the KDCDWBF and SGL by a large margin, i.e., 3.665% and 6.845%. It can be inferred that MSGLAMS also can achieve excellent performance even if the spatial distribution of HSI is complex. It should be noted that all accuracies except stressed grass and synthetic grass reached the top-1. Overall, these experimental results suggest that the proposed method can overcome the problem of unstable classification performance under small sample conditions and thus effectively improve the classification accuracy.

Table 5. Classification accuracy for the Indian Pines image with ten labeled samples per class.

Class Name	SVM	3D-CNN	1D+2D	GCN-M	KDCDWBF	STSE_DWLR	SGL	MSGLAMS
Alfalfa	0.49571	0.65940	0.50457	0.64092	1.00000	0.44264	1.00000	1.00000
Corn-notill	0.46833	0.29684	0.27424	0.93793	0.87593	0.61650	0.73484	0.93067
Corn-mintill	0.43615	0.35215	0.30760	0.85950	0.93172	0.69343	0.95595	0.99578
Corn	0.45338	0.42414	0.38454	0.39823	0.95157	0.70895	0.95595	0.99578
Grass-pasture	0.70857	0.63286	0.48650	0.86174	0.90497	0.55580	0.95983	0.84058
Grass-trees	0.84465	0.84096	0.70611	0.99067	0.99034	0.86224	0.98194	0.99452
Grass-pasture-mowed	0.43256	0.35618	0.25098	0.44103	0.88891	0.46240	1.00000	1.00000
Hay-windrowed	0.92244	0.95833	0.90342	0.99914	1.00000	0.97639	1.00000	1.00000
Oats	0.18511	0.21189	0.60169	0.32545	1.00000	0.32312	1.00000	1.00000
Soybean-notill	0.32139	0.40187	0.33047	0.84658	0.85973	0.62763	0.81393	0.85185
Soybean-mintill	0.49923	0.50666	0.56832	0.94420	0.75716	0.77789	0.88630	0.95723
Soybean-clean	0.34218	0.21864	0.24435	0.76209	0.75133	0.53653	0.95026	0.81619
Wheat	0.87283	0.83830	0.82577	0.99494	1.00000	0.93417	0.97949	1.00000
Woods	0.80549	0.88882	0.80326	0.99723	0.99928	0.80735	1.00000	0.99921
Buildings-Grass-Trees-Drives	0.38141	0.35919	0.31423	0.73869	0.99735	0.37927	1.00000	1.00000
Stone-Steel-Towers	0.86495	0.82275	0.97096	0.85339	1.00000	0.48141	1.00000	1.00000
OA	0.54644	0.53484	0.49926	0.89850	0.88372	0.70582	0.89662	0.94312
AA	0.56652	0.54806	0.52981	0.74946	0.93177	0.63661	0.94362	0.95839
Kappa	0.49372	0.48120	0.43718	0.88360	0.86803	0.66948	0.88228	0.92802

Table 6. Classification accuracy for the Salinas image with ten labeled samples per class.

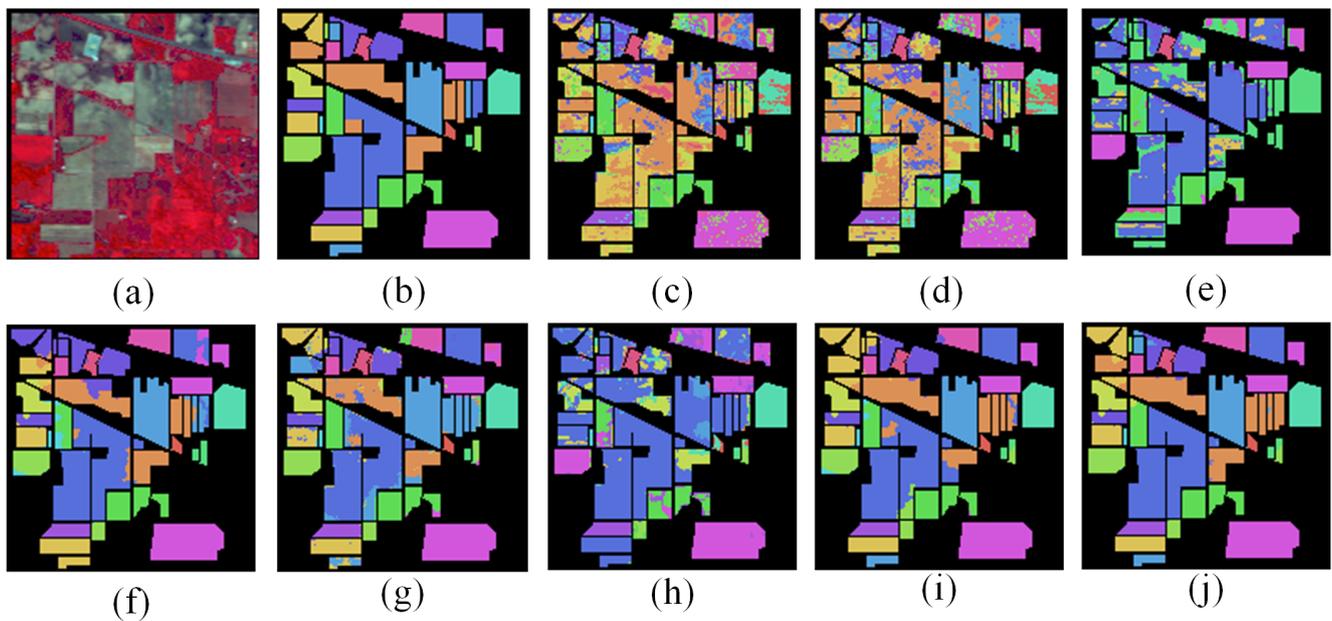
Class Name	SVM	3D-CNN	1D+2D	GCN-M	KDCDWBF	STSE_DWLR	SGL	MSGLAMS
Brocoli_green_weeds_1	0.97213	0.77190	0.89575	1.00000	1.00000	1.00000	0.99350	1.00000
Brocoli_green_weeds_2	0.97979	0.83700	0.88249	1.00000	0.99899	0.98541	1.00000	1.00000
Fallow	0.92227	0.69037	0.38318	1.00000	1.00000	0.99931	1.00000	1.00000
Fallow_rough_plow	0.98131	0.96011	0.99348	0.99957	0.99931	0.99373	0.99783	0.99498
Fallow_smooth	0.96958	0.88974	0.94161	0.97818	0.98884	0.99860	0.98726	0.98208
Stubble	0.99469	0.99141	0.99750	0.99980	0.99926	0.99913	0.99873	1.00000
Celery	0.98829	0.89513	0.92519	0.99872	0.99727	0.94286	0.98963	0.99888
Grapes_untrained	0.62056	0.68778	0.70029	0.87680	0.99156	0.87804	0.98703	0.98501
Soil_vinyard_develop	0.97219	0.90053	0.17241	1.00000	1.00000	0.99840	1.00000	1.00000
Corn_senesced_green_weeds	0.82560	0.80295	0.75480	0.93511	0.96025	0.95372	0.97980	0.98383
Lettuce_romaine_4wk	0.79953	0.62583	0.53860	0.99659	0.98363	0.90000	0.96408	0.99906
Lettuce_romaine_5wk	0.97146	0.88920	0.83733	1.00000	1.00000	0.99715	1.00000	0.99637
Lettuce_romaine_6wk	0.97738	0.80832	0.83215	0.99668	0.97578	0.99730	0.98675	0.98908
Lettuce_romaine_7wk	0.90863	0.90656	0.94550	0.99396	0.98775	0.97158	0.94434	0.94393
Vinyard_untrained	0.55898	0.25129	0.55553	0.93584	0.87010	0.79336	0.99559	0.99106
Vinyard_vertical_trellis	0.97332	0.57847	0.86472	0.98697	1.00000	0.82473	1.00000	1.00000
OA	0.83018	0.74808	0.70606	0.95995	0.97665	0.93139	0.99175	0.99217
AA	0.90098	0.78041	0.76378	0.98121	0.98467	0.95208	0.98903	0.99152
Kappa	0.81221	0.71955	0.68856	0.95546	0.97395	0.92355	0.99082	0.99128

Table 7. Classification accuracy for the university of Pavia image with ten labeled samples per class.

Class Name	SVM	3D-CNN	1D+2D	GCN-M	KDCDWBF	STSE_DWLR	SGL	MSGLAMS
Asphalt	0.70013	0.84315	0.82234	0.97317	0.84083	0.87721	0.85153	0.97240
Meadows	0.79182	0.74326	0.60375	0.89858	0.87875	0.84716	0.87167	0.99507
Gravel	0.33954	0.81594	0.60163	0.94129	0.99712	0.77953	0.85543	0.92758
Trees	0.64275	0.70643	0.64811	0.95690	0.77147	0.82180	0.89293	0.93570
Painted metal sheets	0.98206	1.00000	1.00000	1.00000	1.00000	0.98873	0.99551	0.99926
Bare Soil	0.43017	0.61936	0.36057	0.99016	1.00000	0.74031	1.00000	1.00000
Bitumen	0.46258	0.64829	0.62409	0.99955	1.00000	0.68226	0.99545	1.00000
Self-Blocking Bricks	0.71294	0.90115	0.80587	0.98829	0.98156	0.87564	0.88589	0.98099
Shadows	0.99965	0.99553	0.99634	0.98975	0.52196	0.99931	0.99146	0.99894
OA	0.69445	0.75116	0.63356	0.94319	0.89384	0.83138	0.89591	0.98373
AA	0.67352	0.80812	0.71808	0.97085	0.88797	0.84577	0.92665	0.97888
Kappa	0.60372	0.69162	0.55293	0.92615	0.86203	0.78824	0.86523	0.97843

Table 8. Classification accuracy for the Houston2013 image with ten labeled samples per class.

Class Name	SVM	3D-CNN	1D+2D	KDCDWBF	STSE_DWLR	SGL	MSGLAMS
Healthy grass	0.86978	0.69503	0.89193	0.95822	0.93328	0.95414	0.97161
Stressed grass	0.82756	0.89545	0.96160	0.96331	0.68559	0.87345	0.77028
Synthetic grass	0.32008	0.83721	0.98606	0.99064	0.92484	0.99574	0.99748
Trees	0.90607	0.80034	0.92557	0.95275	0.47288	0.78560	0.89952
Soil	0.81166	0.84134	0.95273	0.97953	0.72748	1.00000	1.00000
Water	0.66780	0.48551	0.91662	0.86703	0.93313	0.89380	0.98230
Residential	0.46402	0.75479	0.73228	0.86080	0.51978	0.72018	0.93766
Commercial	0.51150	0.39113	0.64877	0.80451	0.33407	0.72156	0.81905
Road	0.59123	0.18508	0.61895	0.83688	0.26036	0.70592	0.84684
Highway	0.34081	0.35746	0.66846	0.79076	0.98727	0.98455	1.00000
Railway	0.45040	0.48471	0.57222	0.87527	0.88303	0.87547	0.98275
Parking Lot 1	0.20157	0.35746	0.45568	0.84590	0.89429	0.83414	0.95451
Parking Lot 2	0.13862	0.06155	0.24215	0.84586	0.92604	0.83069	0.93196
Tennis Court	0.76142	0.59532	0.88101	0.94617	0.98210	0.99610	1.00000
Running Track	0.97148	0.94607	0.96709	0.96433	0.96319	0.95363	0.99248
OA	0.59387	0.59493	0.75340	0.89350	0.71886	0.86091	0.92906
AA	0.59464	0.57904	0.76141	0.89880	0.75586	0.84985	0.93905
Kappa	0.55917	0.56117	0.73349	0.88541	0.69744	0.87520	0.92339

**Figure 6.** Classification maps of different methods on Indian Pines dataset; (a) Composite RGB image. (b) Groundtruth map. (c) SVM; (d) 3D-CNN; (e) 1D+2D; (f) GCN-M; (g) KDCDWBF; (h) STSE_DWLR; (i) SGL; (j) MSGLAMS (Proposed).

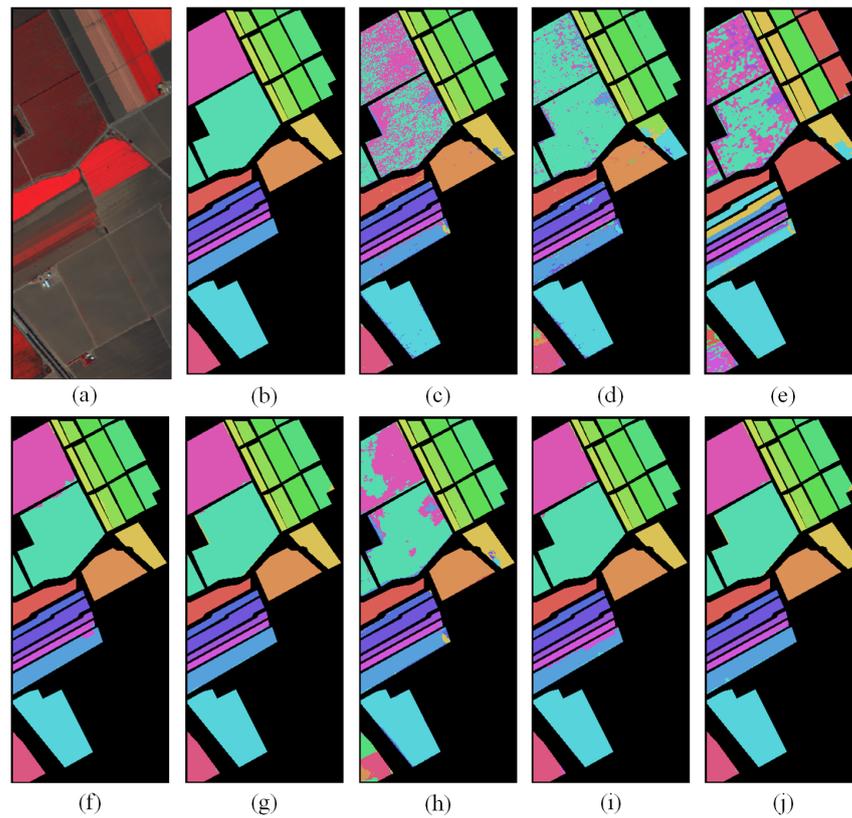


Figure 7. Classification maps of different methods on Salinas dataset; (a) Composite RGB image. (b) Groundtruth map. (c) SVM; (d) 3D-CNN; (e) 1D+2D; (f) GCN-M; (g) KDCDWBF; (h) STSE_DWLR; (i) SGL; (j) MSGLAMS (Proposed).

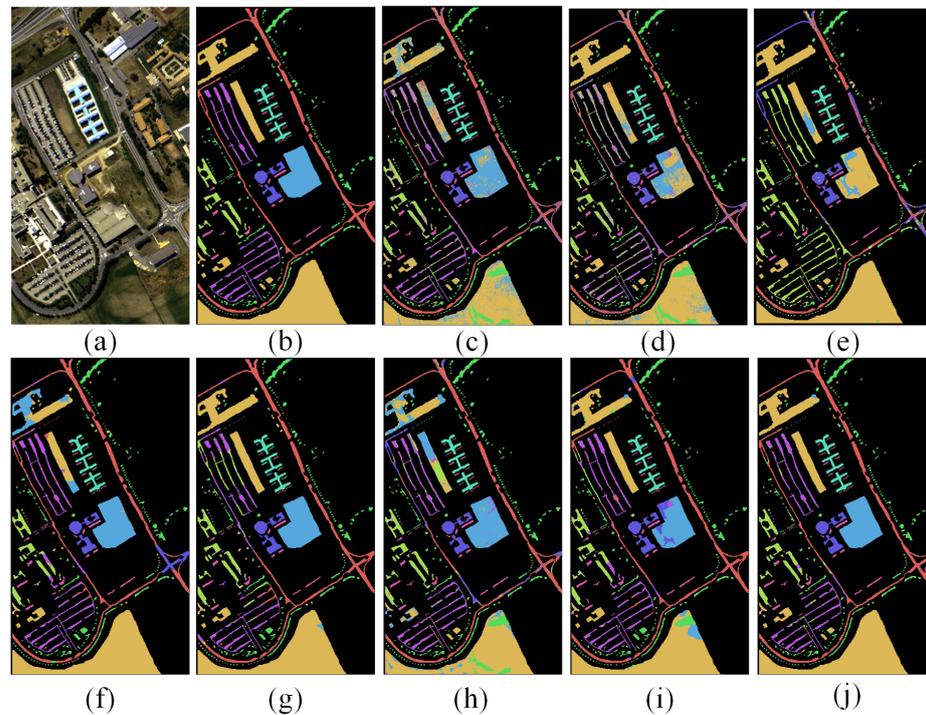


Figure 8. Classification maps of different methods on University of Pavia dataset; (a) Composite RGB image. (b) Groundtruth map. (c) SVM; (d) 3D-CNN; (e) 1D+2D; (f) GCN-M; (g) KDCDWBF; (h) STSE_DWLR; (i) SGL; (j) MSGLAMS (Proposed).

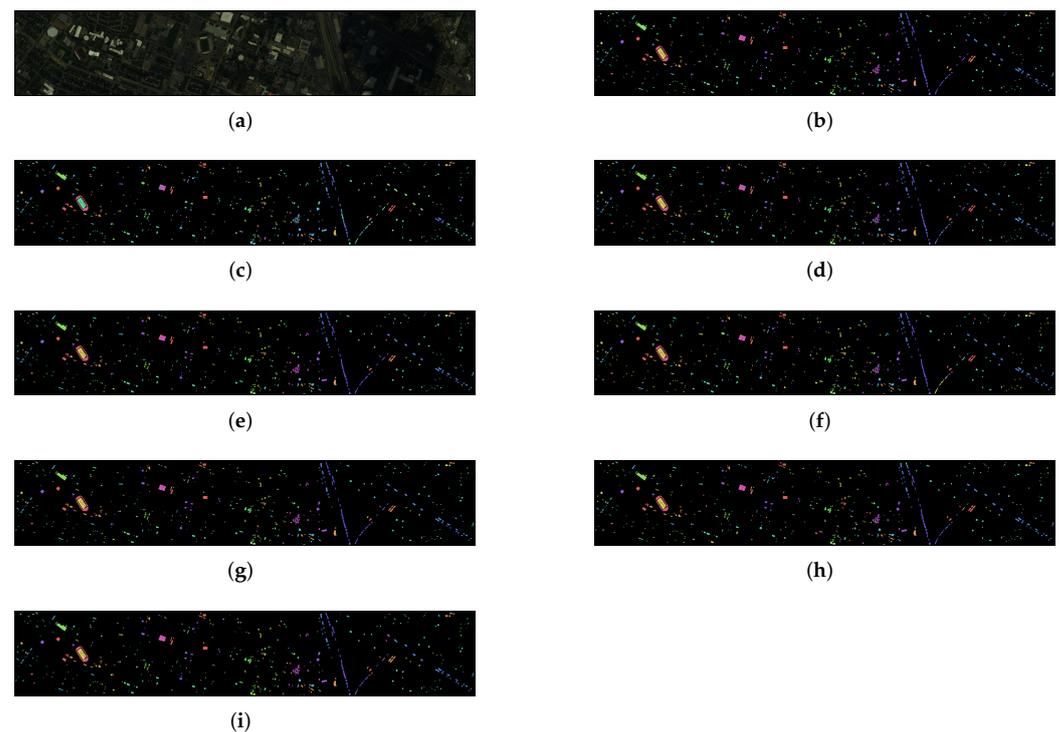


Figure 9. Classification maps of different methods on Houston2013 dataset; (a) Composite RGB image. (b) Groundtruth map. (c) SVM; (d) 3D-CNN; (e) 1D+2D; (f) KDCDWBF; (g) STSE_DWLR; (h) SGL; (i) MSGLAMS (Proposed).

3.4. Analysis of Experimental Results of the ORASS and MSSR

To better display the effect and function of these two key components of the the proposed MSGLAMS, the performance of the ORASS and MSSR are compared with other single scale and multiscale selection methods, respectively. The experimental results of single-scale and multiscale selection methods are put in the same tables for comparison. Tables 9–11 show the classification accuracy OA of different scales on the three datasets. The bolded results represent the highest accuracy in the single-scale selection methods, and the cells with green background represent the highest accuracy in both single-scale selection methods and multiscale selection methods.

Table 9. OAs obtained by different single scale and multiscale on Indian Pines dataset with different number of training samples per class.

Samples per Class	Single-Scale Selection Methods				Multiscale Selection Methods			
	TRS	RES	ASS	ORASS	MSF	AMS	MSNSF	MSSR
3	0.75198	0.74120	0.72434	0.79413	0.83637	0.81529	0.77793	0.86604
5	0.82860	0.80873	0.76507	0.84099	0.85296	0.85374	0.82847	0.87032
7	0.85636	0.84646	0.76472	0.86877	0.87306	0.86711	0.89814	0.92643
10	0.86539	0.85094	0.81504	0.87135	0.91843	0.90945	0.89975	0.94214
15	0.88001	0.88390	0.85503	0.91358	0.95443	0.92107	0.92955	0.95648

It can be seen from the results that among all single-scale selection algorithms, the single scale selected by ORASS has better performance. ASS is also an adaptive single-scale selection method, but because the training sample size is small, the intra-inter indexes $f(s_i)$ of the SP maps is not must positively related to the OAs. Therefore, the optimal single scale selected by the maximum value of the intra-inter index $\text{argmax}(f(s_i))$ may not have the best performance. TRS and RES do not have the ability to adaptively select scales according

to different SP segmentation methods, so the fixed scale chosen by TRS and RES may not be suitable for the SGL classifier.

Table 10. OAs of different single scale and multiscale on Salinas dataset with different number of training samples per calss.

Samples per Class	Single-Scale Selection Methods				Multiscale Selection Methods			
	TRS	RES	ASS	ORASS	MSF	AMS	MSNSF	MSSR
3	0.94556	0.95464	0.95322	0.96771	0.98116	0.96434	0.96227	0.99048
5	0.96358	0.95996	0.96966	0.97008	0.98501	0.98013	0.97572	0.99188
7	0.96379	0.97652	0.97121	0.98809	0.98877	0.98553	0.98108	0.99408
10	0.97297	0.98419	0.98137	0.99043	0.99154	0.98619	0.98423	0.99477
15	0.98973	0.98697	0.98441	0.99061	0.99161	0.99012	0.98683	0.99547

Table 11. OAs of different single scale and multi-scale fusion on university of Pavia dataset with different number of training samples per calss.

Samples per Class	Single-Scale Selection Methods				Multiscale Selection Methods			
	TRS	RES	ASS	ORASS	MSF	AMS	MSNSF	MSSR
3	0.82787	0.81473	0.83604	0.84376	0.90945	0.87703	0.82612	0.94128
5	0.83131	0.83793	0.84895	0.86153	0.91769	0.90020	0.88813	0.94588
7	0.85337	0.84037	0.86587	0.87798	0.92202	0.90798	0.92776	0.95404
10	0.87185	0.88042	0.88078	0.89451	0.93877	0.92767	0.94735	0.96327
15	0.91360	0.91487	0.91669	0.93349	0.95532	0.93246	0.95841	0.96454

In addition, the classification accuracy of HSI has been significantly improved through a multiscale fusion strategy. Especially for the dataset of the University of Pavia with complex texture, the classification accuracy of multiscale fusion is much higher than single scale. However, choosing different fusion scales will supplement different spatial information of ORS and effect the final classification accuracy. The negative-contribution scales will destroy the quality and quantity of the enlarged samples and reduce the classification accuracy. For example, the OA of MSF and MSNSF is lower than MSSR, because the fixed fusion scales selected by MSF and MSNSF may contain scales with negative contribution. AMS has the ability to adaptively select multiple scales, but it still relies too much on intra-inter indexes $f(s_i)$. The $f(s_i)$ is not must positively related to the OAs, so AMS does not have the best performance. As shown in Tables 9–11, the OA of MSSR is higher than other multiscale selection methods and much higher than single-scale SP segmentation. Because under the condition of STLSS, the fusion scales selected by MSSR have complementary enlarged samples information, which makes a positive contribution to improving the classification accuracy. Finally, with the increase of the number of samples, the OAs of different scales are very close, especially on the dataset Salinas with simple texture.

3.5. Parameter Analysis

In this section, the influence of three key parameters of the MSGLAMS method are analyzed in detail. First, we jointly analyzed the influence of the spatial–spectral balance parameter λ on the ORS selection and the influence of κ in Equation (25) on the classification results. Then, the effect of the number of training samples ($\{3, 5, 7, 10, 15\}$ randomly selected samples per class) on several classifiers is examined on the sets of the Indian Pines, Salinas, and University of Pavia images.

3.5.1. Parameter Analysis of λ and κ

The setting of the spatial–spectral balance parameter λ determines the selection of ORS which used to construct the sparse dictionary, and ultimately affects the fusion scale pool. The interval of λ is set to $[0 : 0.1 : 1]$. When $\lambda = 0$, ORSSA only selects the ORS based

on the spectral evaluation index F_{spectral} , i.e., $\text{ORS} = S^{\text{upper}}$, because S^{upper} contains the smallest spectral difference between pixels within the same SP. Conversely, $\lambda = 1$ means that ORASS only selects ORS based on the spatial evaluation index F_{spatial} , i.e., $\text{ORS} = S^{\text{lower}}$, because each SP in S^{lower} scale map contains the largest number of pixels. The interval of κ in the fusion threshold T_S is set to $[1 : 2 : 11]$. When $\kappa = 1$, $T_S = E_{\text{max}}$, representing all scales fusion. For $\kappa = 11$ or higher, the performance of MSGLAMs is close to ORSSA. Figure 10 shows the influence of λ and κ on the overall accuracy on the three datasets, and deeper color corresponds to higher performance. For the Indian Pines dataset, MSGLAMs suggests the best performance when $\lambda = 0.3$ and $\kappa = 5$. In addition, it can be seen from the result of $\kappa = 11$, the ORS selected by ORASS has the best performance when the $\lambda = 0.3$. When the κ value is small, especially $\kappa = 5$, OA of MSGLAMs is higher than other κ values. For the Salinas dataset, MSGLAMs shows the best performance when $\lambda = 0.7$ and $\kappa = 5$, and when $\lambda = 0.4$, the ORS selected by ORSSA achieved the best performance. Finally, for the University of Pavia dataset, MSGLAMs has the best performance when $\lambda = 0.3$ and $\kappa = 7$, and when $\lambda = 0.3$, ORASS yield the best performance.

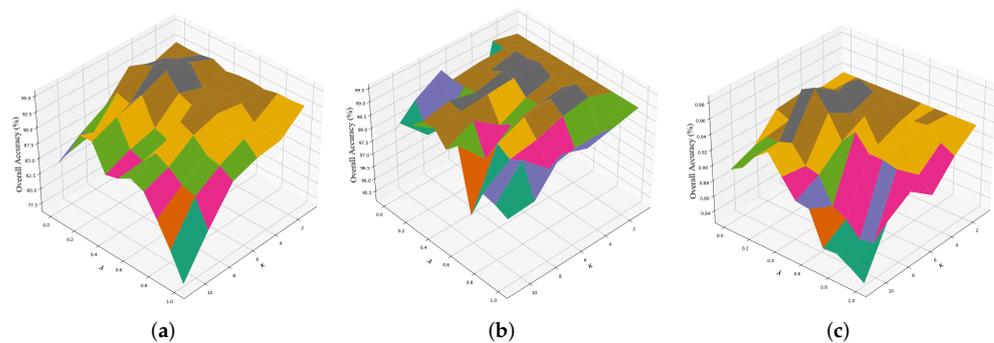


Figure 10. Influence of two parameters on classification accuracy on the three HSI data sets. (a) Indian Pines. (b) Salinas. (c) University of Pavia. λ is the spatial–spectral balance parameter, which determines the selection of the optimal reference scale (ORS). κ is the fusion threshold parameter, which determines the number of the multiscale maps. Deeper colors represent better performance. Generally, MSGLAMs can achieve excellent performance when the $\lambda = 0.3$ and $\kappa = 5$.

Two conclusions can be drawn from the above analysis. First, the small spatial–spectral balance factor λ (i.e., small spectral difference between pixels within the same SP) will get a better performing ORSSA. Consequently, in order to obtain a best-performance ORS to construct a sparse dictionary, λ can be set to 0.3. Second, fusing worse scales can reduce the accuracy (e.g., when $\kappa = 1$ or $\kappa = 3$), and too few fusion scales may not capture complete spatial information of multiscale (e.g., when $\kappa = 9$ or $\kappa = 11$). For normal HSIs, κ can be set to 5, but for datasets with complex textures (such as University of Pavia), setting $\kappa = 7$ will achieve better classification results.

3.5.2. Effect of Different Number of Training Samples

The effect of the number of training samples on several classifiers is examined on the Indian Pines, Salinas, and University of Pavia images in this section. The parameters for all the classifiers are kept the same as that in Section 3.3, and randomly select {3, 5, 7, 10, 15} samples per class as training samples. Figure 11 shows the OA of different classifiers when selecting randomly different numbers of training samples. It can be seen from the results in Figure 11 that the OA obtained by the eight classifiers will increase with the increasing of the number of training samples. In addition, MSGLAMs has the best performance under different training sample sizes. When the number of samples selected is smaller, the advantage of MSGLAMs is more obvious. Therefore, when the number of training samples is small (e.g., three or five training samples per class), the OA of MSGLAMs is significantly higher than other comparison methods. This result proves the obvious advantages of our proposed MSGLAMs under the condition of STLSS.

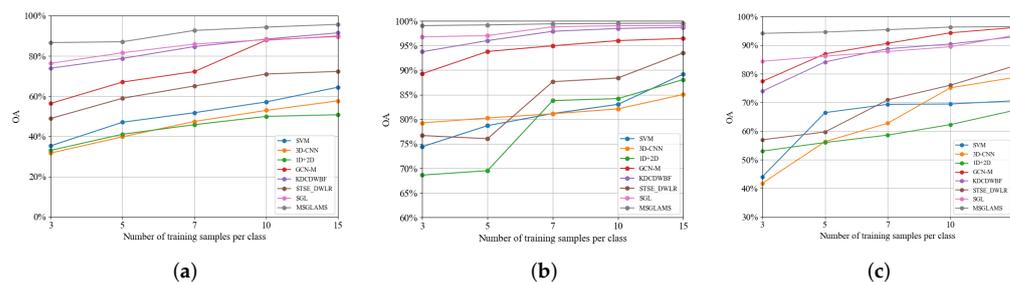


Figure 11. Classification accuracy values of different numbers of labeled samples on three datasets. (a) Indian Pines; (b) University of Pavia; (c) Salinas.

3.5.3. Running Time Comparison

To comprehensively evaluate our proposed method, the corresponding experiments are conducted on four datasets. Table 12 presents the runtime results of different methods. For the small-size datasets, such as Indian Pines, Salinas and University of Pavia, SVM and lightweight networks (1D+2D and 3D-CNN) yield shorter runtime. For the large-size dataset, Houston2013, the results suggest that all methods require more runtime. Compared with the first three methods, STSE_DWLR and SGL require more computation time. The key of GCN-M is the graph convolution network, which needs to feed the whole HSI into network, hence the GCN-M requires much more calculated cost. KDCDWBF also requires more runtime due to its extra post-processing procedure. Our proposed method also requires more computational cost because MSGLAMS needs spend a lot of time to build multiple superpixel contracted graphs. Especially when the size of an HSI is large, it requires more time to construct the large-scale superpixel maps.

Table 12. runtime result (seconds) comparison on four data sets.

Data Set	SVM	3D-CNN	1D+2D	GCN-M	KDCDWBF	STSE_DWLR	SGL	MSGLAMS
Indian Pines	3.39	21.837	9.16	110.49	131.34	40.317	64.65	95.033
Salinas	20.55	50.31	20.04	750.28	672.35	103.08	163.79	569.90
University of Pavia	4.14	56.57	17.61	608.22	714.35	70.05	351.76	590.53
Houston2013	34.56	101.42	42.70	-	1157.96	70.05	520.31	874.59

4. Conclusions

This paper proposes a multiple superpixel graphs learning method based on adaptive multiscale segmentation for HSI classification (MSGLAMS). The MSGLAMS can be split into two major section: the selection of multiscale, and creating multiple superpixel-based graphical models to jointly predict the classification results via a pixel-level fusion. Aiming at the election of a multiscale, a novel two-steps multiscale selection strategy is designed to adaptively select a set of complementary multiscales. Initially, an optimal reference scale selection algorithm (ORSSA) is proposed to select the optimal reference scale (ORS) as a basic scale from the built candidate scale pool. Then, a multiscale selection algorithm based on sparse representation (MSSR) is developed to select the fusion scales, which has positive contribution to supplementing spatial information of ORS. Finally, multiple superpixel-based graphical models are obtained via a constructed superpixel contracted graph of determined scales (fusion scales), and the final results are jointly predicted via the voting results of these models. The proposed MSGLAMS adopts the multiscale-superpixel-based framework, which can reduce the adverse effect of improper selection of a superpixel segmentation scale on the classification accuracy while saving the cost to manually seek suitable segmentation scale. To make full use of the spatial information of different segmentation scales, a novel two-steps multiscale selection strategy is designed to adaptively select a set of complementary multiscales. To fix the bias and instability of a single model, multiple superpixel-based graphical models obtained by

constructing superpixel contracted graph of fusion scales are developed to jointly predict the final results via a pixel-level fusion strategy. This Boosting-like prediction strategy can significantly improve the accuracy and robustness of the classification results, especially when the number of labeled samples is small. The experimental results on the three datasets indicate the superiority of the proposed MSGLAMS over several well-known classifiers in terms of qualitative and quantitative results, especially when the number of selected labeled samples is small.

The proposed method has achieved excellent performance on four datasets, but it needs a higher computational costs for HSIs with larger sizes. In the future, we will try to reduce computational complexity and adopt the CUDA to accelerate MSGLAMS.

Author Contributions: B.Q. and S.F. wrote and edited original draft preparation; C.Z. supervised the work and reviewed the manuscript; B.Q. and S.F. designed and implemented this framework. B.Q. and W.Z. tuned and evaluated this method by designing different experiments. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China Grant 62002083, 61971153, 61801142 and 62071136, and the Heilongjiang Provincial Natural Science Foundation of China Grant LH2021F012, and Heilongjiang Postdoctoral Foundation Grant LBH-Q20085, and the Fundamental Research Funds for the Central Universities Grant 3072021CF0801, 3072021CF0814, 3072021CF0807 and 3072021CF0808. (Corresponding author: Shou Feng).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Li, W.; Du, Q. Gabor-Filtering-Based Nearest Regularized Subspace for Hyperspectral Image Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 1012–1022. [[CrossRef](#)]
- Jafarzadeh, H.; Mahdianpari, M.; Gill, E.; Mohammadimanesh, F.; Homayouni, S. Bagging and Boosting Ensemble Classifiers for Classification of Multispectral, Hyperspectral and PolSAR Data: A Comparative Evaluation. *Remote Sens.* **2021**, *13*, 4405. [[CrossRef](#)]
- Liu, Q.; Wu, Z.; Jia, X.; Xu, Y.; Wei, Z. From Local to Global: Class Feature Fused Fully Convolutional Network for Hyperspectral Image Classification. *Remote Sens.* **2021**, *13*, 5043. [[CrossRef](#)]
- Song, M.; Yu, C.; Xie, H.; Chang, C.I. Progressive Band Selection Processing of Hyperspectral Image Classification. *IEEE Geosci. Remote Sens. Lett.* **2020**, *17*, 1762–1766. [[CrossRef](#)]
- Xue, B.; Yu, C.; Wang, Y.; Song, M.; Li, S.; Wang, L.; Chen, H.M.; Chang, C.I. A Subpixel Target Detection Approach to Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 5093–5114. [[CrossRef](#)]
- Haboudane, D. Hyperspectral vegetation indices and novel algorithms for predicting green LAI of crop canopies: Modeling and validation in the context of precision agriculture. *Remote Sens. Environ.* **2004**, *90*, 337–352. [[CrossRef](#)]
- Merentitis, A.; Debes, C.; Heremans, R. Ensemble Learning in Hyperspectral Image Classification: Toward Selecting a Favorable Bias-Variance Tradeoff. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 1089–1102. [[CrossRef](#)]
- Ellis, R.J.; Scott, P.W. Evaluation of hyperspectral remote sensing as a means of environmental monitoring in the St. Austell China clay (kaolin) region, Cornwall, UK. *Remote Sens. Environ.* **2004**, *93*, 118–130. [[CrossRef](#)]
- Shao, Z.; Fu, H.; Li, D.; Altan, O.; Cheng, T. Remote sensing monitoring of multi-scale watersheds impermeability for urban hydrological evaluation. *Remote Sens. Environ.* **2019**, *232*, 111338. [[CrossRef](#)]
- Schneider, S.; Murphy, R.J.; Melkumyan, A. Evaluating the performance of a new classifier—The GP-OAD: A comparison with existing methods for classifying rock type and mineralogy from hyperspectral imagery. *ISPRS J. Photogramm. Remote Sens.* **2014**, *98*, 145–156. [[CrossRef](#)]
- Chen, M.; Tang, Y.; Zou, X.; Huang, Z.; Zhou, H.; Chen, S. 3D global mapping of large-scale unstructured orchard integrating eye-in-hand stereo vision and SLAM. *Comput. Electron. Agric.* **2021**, *187*, 106237. [[CrossRef](#)]
- Cao, X.; Yan, H.; Huang, Z.; Ai, S.; Xu, Y.; Fu, R.; Zou, X. A Multi-Objective Particle Swarm Optimization for Trajectory Planning of Fruit Picking Manipulator. *Agronomy* **2021**, *11*, 2286. [[CrossRef](#)]
- Wu, F.; Duan, J.; Chen, S.; Ye, Y.; Ai, P.; Yang, Z. Multi-Target Recognition of Bananas and Automatic Positioning for the Inflorescence Axis Cutting Point. *Front. Plant Sci.* **2021**, *12*, 705021. [[CrossRef](#)] [[PubMed](#)]
- Rasti, B.; Hong, D.; Hang, R.; Ghamisi, P.; Kang, X.; Chanussot, J.; Benediktsson, J.A. Feature Extraction for Hyperspectral Imagery: The Evolution From Shallow to Deep: Overview and Toolbox. *IEEE Geosci. Remote Sens. Mag.* **2020**, *8*, 60–88. [[CrossRef](#)]

15. Liu, B.; Yu, A.; Yu, X.; Wang, R.; Gao, K.; Guo, W. Deep Multiview Learning for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 7758–7752. [[CrossRef](#)]
16. Yu, C.; Zhao, M.; Song, M.; Wang, Y.; Li, F.; Han, R.; Chang, C.I. Hyperspectral Image Classification Method Based on CNN Architecture Embedding with Hashing Semantic Feature. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2019**, *12*, 1866–1881. [[CrossRef](#)]
17. Liu, D.; Han, G.; Liu, P.; Yang, H.; Sun, X.; Li, Q.; Wu, J. A Novel 2D-3D CNN with Spectral-Spatial Multi-Scale Feature Fusion for Hyperspectral Image Classification. *Remote Sens.* **2021**, *13*, 4621. [[CrossRef](#)]
18. Audebert, N.; Le Saux, B.; Lefevre, S. Deep Learning for Classification of Hyperspectral Data: A Comparative Review. *IEEE Geosci. Remote Sens. Mag.* **2019**, *7*, 159–173. [[CrossRef](#)]
19. Ben Hamida, A.; Benoit, A.; Lambert, P.; Ben Amar, C. 3-D Deep Learning Approach for Remote Sensing Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 4420–4434. [[CrossRef](#)]
20. Zhao, C.; Zhu, W.; Feng, S. Hyperspectral Image Classification Based on Kernel-Guided Deformable Convolution and Double-Window Joint Bilateral Filter. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 1–5. [[CrossRef](#)]
21. Zhang, Z.; Crawford, M.M. A Batch-Mode Regularized Multimetric Active Learning Framework for Classification of Hyperspectral Images. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 6594–6609. [[CrossRef](#)]
22. Tu, B.; Zhang, X.; Kang, X.; Zhang, G.; Wang, J.; Wu, J. Hyperspectral Image Classification via Fusing Correlation Coefficient and Joint Sparse Representation. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 340–344. [[CrossRef](#)]
23. Hao, Q.; Li, S.; Kang, X. Multilabel Sample Augmentation-Based Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 4263–4278. [[CrossRef](#)]
24. Zhang, S.; Kang, X.; Duan, P.; Sun, B.; Li, S. Polygon Structure-Guided Hyperspectral Image Classification with Single Sample for Strong Geometric Characteristics Scenes. *IEEE Trans. Geosci. Remote Sens.* **2021**, 1–12. [[CrossRef](#)]
25. Imani, M.; Ghassemian, H. Band Clustering-Based Feature Extraction for Classification of Hyperspectral Images Using Limited Training Samples. *IEEE Geosci. Remote Sens. Letters* **2014**, *11*, 1325–1329. [[CrossRef](#)]
26. Zhang, Y.; Chen, Y. Multiscale Weighted Adjacent Superpixel-Based Composite Kernel for Hyperspectral Image Classification. *Remote Sens.* **2021**, *13*, 820. [[CrossRef](#)]
27. Zhao, Y.; Yan, F. Hyperspectral Image Classification Based on Sparse Superpixel Graph. *Remote Sens.* **2021**, *13*, 3592. [[CrossRef](#)]
28. Yang, X.; Zhao, M.; Gao, T.; Chen, J.; Zhang, J. Multiscale-Superpixel-Based SparseCEM for Hyperspectral Target Detection. *IEEE Geosci. Remote Sens. Lett.* **2021**, 1–5. [[CrossRef](#)]
29. Yu, H.; Gao, L.; Liao, W.; Zhang, B.; Pizurica, A.; Philips, W. Multiscale Superpixel-Level Subspace-Based Support Vector Machines for Hyperspectral Image Classification. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 2142–2146. [[CrossRef](#)]
30. Shi, C.; Pun, C.M. Multiscale Superpixel-Based Hyperspectral Image Classification Using Recurrent Neural Networks With Stacked Autoencoders. *IEEE Trans. Multimed.* **2020**, *22*, 487–501. [[CrossRef](#)]
31. Yang, C.; Bruzzone, L.; Zhao, H.; Tan, Y.; Guan, R. Superpixel-Based Unsupervised Band Selection for Classification of Hyperspectral Images. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 7230–7245. [[CrossRef](#)]
32. Zhang, Y.; Liu, K.; Dong, Y.; Wu, K.; Hu, X. Semisupervised Classification Based on SLIC Segmentation for Hyperspectral Image. *IEEE Geosci. Remote Sens. Lett.* **2020**, *17*, 1440–1444. [[CrossRef](#)]
33. Achanta, R.; Shaji, A.; Smith, K.; Lucchi, A.; Fua, P.; Susstrunk, S. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 2274–82. [[CrossRef](#)]
34. Liu, M.Y.; Tuzel, O.; Ramalingam, S.; Chellappa, R. Entropy rate superpixel segmentation. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; pp. 2097–2104. [[CrossRef](#)]
35. He, Z.; Shen, Y.; Zhang, M.; Wang, Q.; Wang, Y.; Yu, R. Spectral-spatial hyperspectral image classification via SVM and superpixel segmentation. In Proceedings of the 2014 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings, Montevideo, Uruguay, 12–15 May 2014; pp. 422–427. [[CrossRef](#)]
36. Zheng, C.; Wang, N.; Cui, J. Hyperspectral Image Classification with Small Training Sample Size Using Superpixel-Guided Training Sample Enlargement. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 7307–7316. [[CrossRef](#)]
37. Liu, Q.; Xiao, L.; Yang, J.; Wei, Z. CNN-Enhanced Graph Convolutional Network with Pixel- and Superpixel-Level Feature Fusion for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, 1–15. [[CrossRef](#)]
38. Sellars, P.; Aviles-Rivero, A.I.; Schonlieb, C.B. Superpixel Contracted Graph-Based Learning for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 4180–4193. [[CrossRef](#)]
39. Li, H.; Feng, R.; Wang, L.; Zhong, Y.; Zhang, L. Superpixel-Based Reweighted Low-Rank and Total Variation Sparse Unmixing for Hyperspectral Remote Sensing Imagery. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 629–647. [[CrossRef](#)]
40. Cui, B.; Xie, X.; Ma, X.; Ren, G.; Ma, Y. Superpixel-Based Extended Random Walker for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 3233–3243. [[CrossRef](#)]
41. Fang, L.; Li, S.; Duan, W.; Ren, J.; Benediktsson, J.A. Classification of Hyperspectral Images by Exploiting Spectral-Spatial Information of Superpixel via Multiple Kernels. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 6663–6674. [[CrossRef](#)]
42. Jia, S.; Deng, X.; Zhu, J.; Xu, M.; Zhou, J.; Jia, X. Collaborative Representation-Based Multiscale Superpixel Fusion for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 7770–7784. [[CrossRef](#)]
43. Liu, T.; Gu, Y.; Chanussot, J.; Dalla Mura, M. Multimorphological Superpixel Model for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 6950–6963. [[CrossRef](#)]

44. Lu, Q.; Wei, L. Multiscale Superpixel-Based Active Learning for Hyperspectral Image Classification. *IEEE Geosci. Remote Sens. Letters* **2021**, 1–5. [[CrossRef](#)]
45. Lu, T.; Li, S.; Fang, L.; Jia, X.; Benediktsson, J.A. From Subpixel to Superpixel: A Novel Fusion Framework for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 4398–4411. [[CrossRef](#)]
46. Jia, S.; Deng, B.; Zhu, J.; Jia, X.; Li, Q. Superpixel-Based Multitask Learning Framework for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 2575–2588. [[CrossRef](#)]
47. Leng, Q.; Yang, H.; Jiang, J.; Tian, Q. Adaptive MultiScale Segmentations for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 5847–5860. [[CrossRef](#)]
48. Jia, S.; Deng, B.; Zhu, J.; Jia, X.; Li, Q. Local Binary Pattern-Based Hyperspectral Image Classification with Superpixel Guidance. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 749–759. [[CrossRef](#)]
49. Li, J.; Du, Q.; Li, Y.; Li, W. Hyperspectral Image Classification with Imbalanced Data Based on Orthogonal Complement Subspace Projection. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 3838–3851. [[CrossRef](#)]
50. Prasad, S.; Labate, D.; Cui, M.; Zhang, Y. Morphologically Decoupled Structured Sparsity for Rotation-Invariant Hyperspectral Image Analysis. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 4355–4366. [[CrossRef](#)]
51. Dundar, T.; Ince, T. Sparse Representation-Based Hyperspectral Image Classification Using Multiscale Superpixels and Guided Filter. *IEEE Geosci. Remote Sens. Letters* **2019**, *16*, 246–250. [[CrossRef](#)]
52. Yu, H.; Shang, X.; Song, M.; Hu, J.; Jiao, T.; Guo, Q.; Zhang, B. Union of Class-Dependent Collaborative Representation Based on Maximum Margin Projection for Hyperspectral Imagery Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 553–566. [[CrossRef](#)]
53. Tropp, J.; Gilbert, A. Signal Recovery From Random Measurements via Orthogonal Matching Pursuit. *IEEE Trans. Inf. Theory* **2008**, *53*, 4655–4666. [[CrossRef](#)]
54. Su, H.; Yu, Y.; Du, Q.; Du, P. Ensemble Learning for Hyperspectral Image Classification Using Tangent Collaborative Representation. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 3778–3790. . [[CrossRef](#)]
55. Jiang, J.; Ma, J.; Wang, Z.; Chen, C.; Liu, X. Hyperspectral Image Classification in the Presence of Noisy Labels. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 851–865. [[CrossRef](#)]
56. Sun, L.; Ma, C.; Chen, Y.; Shim, H.J.; Wu, Z.; Jeon, B. Adjacent Superpixel-Based Multiscale spatial–spectral Kernel for Hyperspectral Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2019**, *12*, 1905–1919. [[CrossRef](#)]