

SGML: A Symmetric Graph Metric Learning Framework for Efficient Hyperspectral Image Classification

Yunsong Li ¹, Member, IEEE, Bobo Xi ², Graduate Student Member, IEEE, Jiaojiao Li ³, Member, IEEE, Rui Song ⁴, Yuchao Xiao ⁵, and Jocelyn Chanussot ⁶, Fellow, IEEE

Abstract—Recently, the semi-supervised graph convolutional network (SSGCN) has been verified effective for hyperspectral image (HSI) classification. However, constrained by the limited training data and spectral uncertainty, the classification performance is remained to be further improved. Moreover, attribute to the massive data, the SSGCN with complex computation is generally too time- and resource-consuming to be applicable in real-time needs. To conquer these issues, we propose an efficient symmetric graph metric learning (SGML) framework by incorporating metric learning into the SSGCN paradigm. Specifically, we first conduct multilevel pixel-to-superpixel projection (P-SP) on the HSI to investigate the multiscale spatial information, where the suitable superpixel numbers are adaptively determined. Then, to extract more expressive representations, we design a new structure denoted as GSvolution, comprising the graph convolution (G-Conv) and a novel self-channel-enhanced convolution (S-Conv), to propagate the labeled and unlabeled graph node information and simultaneously enhance the critical intranode channel features. Finally, the superpixel node features are reprojected to the pixel level (SP-P) so that the distilled multistream features can be integrated to obtain the final decision. Noticeably, this ingenious symmetric mechanism (P-SP and SP-P) can alleviate the spectral variability and facilitate the framework to be an efficient model. Furthermore, in the metric learning module, we propose an innovative metric loss function to enhance the discrimination of the embedding features, i.e., inter class far apart and intra-class close. In the experiments, we demonstrate that the classification capacity of the proposed SGML can surpass the comparators on three benchmark data sets.

Index Terms—Graph convolutional network, HSI classification, metric learning, multilevel superpixel segmentation.

I. INTRODUCTION

HYPERSPECTRAL image (HSI) can distinguish various land-covers owing to its refined reflection/radiation information, attracting momentous vast industry and agriculture applications, such as medical diagnosis, mineral exploration, and environmental monitoring, to name a few [1]–[6]. Notably, HSI classification (HSIC), identifying the pixel-level attributes over the whole scenario, is a significant scene parsing task in HSI processing [7], [8].

Recently, the HSIC approaches based on deep learning (DL) have flourished, which implement deep feature extraction and classification in an end-to-end configuration [9]–[11]. For instance, Li *et al.* [12] presented a three-dimensional convolutional neural network (3-D-CNN), which can explore the spectral and spatial features synchronously and obtain promising classification results. Similarly, the patch-based spectral–spatial models, mitigating the spectral uncertainty problem by taking the spatially neighbored pixels into the calculation, have become the mainstream in HSIC. For example, Zhong *et al.* [13] introduced the shortcut connection architecture into HSIC and proposed a spectral–spatial residual network (SSRN). Moreover, a fast dense spectral–spatial convolutional (FDSSC) network was presented to further enhance the representations by dense skip-connections of low-level and high-level features [14]. Notably, a diverse region CNN (DR-CNN) [15] was proposed to precisely exploit the wealthy spatial context features by utilizing various neighborhoods of the central target pixel.

Nonetheless, the generalization capability of these supervised deep models dramatically downgrades with insufficient labeled samples. To alleviate this issue, novel DL architectures like squeeze-and-excitation network (SENet) were introduced to extract more significant features leveraging the attention mechanism [16]–[18]. In addition, Haut *et al.* [19] presented an active learning (AL) approach associated with a Bayesian CNN, which is reinforced by the acquisition of new hard unlabeled samples to obtain robust identification results. Chen *et al.* [20] realized heterogeneous transfer learning by utilizing the model pretrained on the ImageNet data set to perform the HSIC tasks. Moreover, Liu *et al.* adopted the prototypical network to achieve deep

Manuscript received June 15, 2021; revised August 18, 2021 and October 22, 2021; accepted December 2, 2021. Date of publication December 15, 2021; date of current version January 5, 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2018AAA0102702, in part by the National Nature Science Foundation of China under Grant 61901343, Grant 61571345, Grant 61671383, Grant 91538101, Grant 61501346, and Grant 61502367, in part by Science and Technology on Space Intelligent Control Laboratory under Grant ZDSYS-2019-03, in part by the Fundamental Research Funds for the Central Universities under Grant JB190107, in part by the 111 Project under Grant B08038, and in part by the Innovation Fund of Xidian University under Grant 10221150004 and Grant 5001-20109215456. (Corresponding authors: Bobo Xi; Jiaojiao Li.)

Yunsong Li, Bobo Xi, Jiaojiao Li, and Rui Song are with the State Key Laboratory of Integrated Service Networks, School of Telecommunications Engineering, Xi'an 710071, China (e-mail: ysli@mail.xidian.edu.cn; xibobo1301@foxmail.com; jjli@xidian.edu.cn; ruiscientific@gmail.com).

Yuchao Xiao is with the School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: ycxiao@stu.xidian.edu.cn).

Jocelyn Chanussot is with the Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-Lab, 38000 Grenoble, France, and also with the Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100094, China (e-mail: jocelyn@hi.is).

Digital Object Identifier 10.1109/JSTARS.2021.3135548

few-shot learning for HSIC, which accomplishes knowledge transformation through the episode-based training strategy and the meta-learning paradigm [21], [22].

On the other hand, more and more research focuses on semi-supervised learning (SSL), which can not only make full use of the labeled data, but also take the information of the unlabeled samples into account as an auxiliary. Remarkably, the semi-supervised graph convolutional network (SSGCN) has been demonstrated to be one of the most effective SSL methods for HSIC, due to that the filter parameters are approximated by the truncated Chebyshev polynomials and can be trained in an end-to-end manner [23]. For instance, concerning the specific application of HSIC, a spectral–spatial GCN (S^2GCN) was proposed in Ref. [24], which obtains improved classification performance than that of naive GCN [23]. Furthermore, Mou *et al.* [25] proposed a special nonlocal GCN, which incorporates the self-attention mechanism into constructing the adjacency matrix to acquire more compact classification maps.

However, two problems remain to be resolved.

1) In the small sample size (SSS) regime for HSIC, the above GCN-based models would not effectively propagate the labels to the entire topological graph, which likely leads to inferior classification results. The reason can be inferred as follows: Labels are generally harnessed as “anchors” to force the learning models to fit the labeled samples with certain confidence [26]. By this means, the information extracted from them can be reliably propagated to the unlabeled samples. However, when the labeled samples are insufficient, the models tend to exhibit a significant test variance even though the training bias is slight, i.e., overfitting phenomenon. Additionally, the high intraclass and low interclass variability, i.e., spectral uncertainty inherent to HSI data [7], may produce that semantically identical pixels may be differently grouped and consequently mislabeled.

2) Attribute to the large amount of pixel-level data, the adjacency matrix of the graph has a relatively massive size. Thus, the completion of the above methods demands too much time and computing resources, which is typically unacceptable in real-time applications [27]. Therefore, the efficient GCN-based model is worthy of further exploration.

Concerning the first issue, note that humans are exceptional learners with a solid capability to generalize their learned knowledge even to novel concepts and learn from very few examples through contrastive analysis [21]. Motivated by this founding, we skillfully incorporate the metric learning paradigm into the GCN framework. Namely, apart from exploiting the connectivity patterns between the labeled and unlabeled samples, we seek to transfer as much valuable knowledge as possible from the limited labeled samples to the excessive unlabeled ones in the embedding space. Specifically, we devise a metric loss function to enhance the discrimination of the embedding features, which can push the interclass features apart and pull the intraclass features close.

To alleviate the second deficiency, Wan *et al.* [28] presented a multiscale dynamic GCN (MDGCN), which involves superpixels into the multihop graph learning to reduce the computational complexity and save the elapsed training time. However, utilizing the spatially multihop graph structure to integrate the

multiscale spatial information may bias the classifier. It is because that the superpixel segmentation method is essentially based on the assumption that one superpixel belongs to the same material, implying that the multihop (two- or three-hop) adjacent distant superpixels have a great possibility to be the different land-cover types. To overcome this drawback, we propose to directly impose multilevel superpixel segmentation on the HSI data set. After that, the pixel-level data is projected to the superpixel level (P-SP), and only the adjacent SPs (one-hop) are connected at each level. Compared to MDGCN, the consistency probability of the connected nodes in the initial adjacency matrices is promoted in this manner. Correspondingly, before the decision phase, the deep node features extracted from the multilevel superpixels are reprojected to the pixel level (SP-P) so that they can be assembled in the same dimension. Due to this intelligent symmetric structure, the calculation of the network is sharply decreased, thus achieving high-precise HSIC efficiently.

Prominently, owing to the multiscale contextual information contained in the multilevel from undersegmentation to oversegmentation superpixels, the problems resulted from the spectral uncertainty can be commendably conquered. Moreover, it is superior to the aforementioned patch-based scheme because the spatial context feature is aggregated adaptive to the physical land-cover content. Furthermore, to extract more representative deep spectral–spatial features in each scale, we present a new GSvolution structure comprising successive graph convolution and a novel self-channel-enhanced convolution. The latter can automatically emphasize the vital channel information and boost the classification performance.

Specifically, the contributions of this study can be summarized in the following.

- 1) A novel SGML network is presented for HSIC. The calculation amount is compressed dramatically due to the innovative symmetric framework involving P-SP projection and SP-P reprojection. Thus, the training and test processes can be accomplished efficiently.
- 2) To precisely investigate the multiscale spatial information, we impose multilevel superpixel segmentation on the HSI data leveraging its inherent attributes to determine the superpixel numbers heuristically.
- 3) Unlike the general GCN, we devise a GSvolution structure comprising graph and self-channel-enhanced convolutions to extract the deep representative spectral–spatial features, which can not only propagate various labeled and unlabeled node features, but also explicitly strengthen the informative intranode channel characteristics.
- 4) In the decision stage of the metric learning module, a special metric loss function is designed, which can force the model to distill more discriminative features, i.e., interclass separable and intraclass compact. Abundant experiment results show that the proposed SGML can perform better than some prevalent and advanced classification methods.

The rest of this article is organized as follows. Section II presents the related works. Section III details our proposed method. Section IV shows the experiment results and analysis. Finally, Section V concludes this article.

II. RELATED WORKS

A. Superpixel Segmentation

The most intuitive explanation for superpixel is to aggregate several pixels with similar properties into a more prominent and representative element [29]. This new element will serve as the basic unit for the subsequent image processing algorithms. In this way, the feature dimension is remarkably reduced to decrease the calculation burden. Hence, there have been abundant HSIC methods utilizing the superpixel segmentation as a pre-processing technique, which has been demonstrated effective [7]. For instance, Li *et al.* [30] presented a subpixel–pixel–superpixel-based multiview network combined with the AL strategy, which jointly exploits the characteristics of HSI, such as the spectral mixture, spectral discrimination, and spectral–spatial structure. Jia *et al.* [31] incorporated the superpixel into the label propagation method for HSIC, which also obtains encouraging classification performance and saves the computational time. Specifically, in all these algorithms, entropy rate segmentation (ERS) [32] is adopted to generate superpixels from an image due to its effectiveness. Concisely, as a graph-based method, the ERS can be translated into solving the following objective function:

$$\arg \max_P E(P) + \gamma A(P) \quad (1)$$

where $E(P)$ represents the entropy rate restriction to generate homogeneous clusters. $A(P)$ refers to the balance constraint to force the clusters to have a similar spatial size and decrease the number of unbalanced superpixels. $\gamma \geq 0$ is a weight coefficient of the balance term.

B. Graph Convolutional Network

In addition to the non-Euclidean data domain, the semi-supervised GCN has been demonstrated to be one of the most effective SSL methods for processing the Euclidean image data [33]. Especially, Kipf *et al.* [23] utilized the Chebyshev polynomials to approximate the graph convolution (G-Conv) kernels and designed an efficient layer-wise propagation rule, which can simultaneously encode the local graph structure and node features to achieve a more stable state. Supposing to utilize the Chebyshev polynomial $C_k(x)$ up to K th order to approximate the convolutional filter parameterized by θ , i.e., g_θ , the G-Conv can be expressed as

$$g_\theta \star \mathbf{x}_g \approx \sum_{k=0}^K \theta_k C_k(\widehat{\mathbf{L}}) \mathbf{x}_g \quad (2)$$

where \mathbf{x}_g refers to the graph signal and θ_k represents the k th Chebyshev coefficients. The shifted $\widehat{\mathbf{L}} = \frac{2}{\lambda_{\max}} \mathbf{L} - \mathbf{I}$, where $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} = \mathbf{U} \mathbf{\Lambda} \mathbf{U}^T$ means the normalized graph Laplacian [34] and λ_{\max} denotes the maximum eigenvalue of \mathbf{L} . Besides, \mathbf{I} refers to the identity matrix with the suitable size. \mathbf{D} is the degree matrix of the graph and \mathbf{A} is the corresponding adjacency matrix. $\mathbf{\Lambda}$ is a diagonal matrix filled with the eigenvalues of \mathbf{L} and \mathbf{U} is the eigenvector matrix of \mathbf{L} , respectively.

Based on (2), Ref. [23] further approximated λ_{\max} to 2 and constrained the layer-wise convolution operation to $K = 1$. Then, the calculation can be represented as

$$g_{\theta'} \star \mathbf{x}_g \approx \theta'_0 \mathbf{x}_g + \theta'_1 (\mathbf{L} - \mathbf{I}) \mathbf{x}_g = \theta'_0 \mathbf{x}_g - \theta'_1 \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \mathbf{x}_g \quad (3)$$

where θ'_0 and θ'_1 are two free parameters shared throughout the whole graph. After constraining $\theta = \theta'_0 = -\theta'_1$, (3) can be simplified as

$$g_\theta \star \mathbf{x}_g \approx \theta \left(\mathbf{I} + \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \right) \mathbf{x}_g. \quad (4)$$

Additionally, to get rid of the numerical instabilities and gradients exploding/vanishing, renormalization is carried out by $\mathbf{I} + \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \rightarrow \widehat{\mathbf{D}}^{-\frac{1}{2}} \widehat{\mathbf{A}} \widehat{\mathbf{D}}^{-\frac{1}{2}}$, where $\widehat{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ and $\widehat{\mathbf{D}}_{ii} = \sum_j \widehat{\mathbf{A}}_{ij}$. Finally, for the graph signal $\mathbf{X} \in \mathbb{R}^{N \times C}$ (N nodes), the G-Conv is expressed as

$$\mathbf{G}_o = \widehat{\mathbf{D}}^{-\frac{1}{2}} \widehat{\mathbf{A}} \widehat{\mathbf{D}}^{-\frac{1}{2}} \mathbf{X} \Theta \quad (5)$$

where $\Theta \in \mathbb{R}^{C \times F}$ represents the trainable convolutional parameters, and F is the kernel number. $\mathbf{G}_o \in \mathbb{R}^{N \times F}$ denotes the output result of the G-Conv. According to (5), it can be ratiocinated that the GCN propagates the node information between the labeled and unlabeled samples to obtain the distinguishing features. However, the intranode information is not explicitly explored. Therefore, to further enhance the representativeness of the node features, we devise a novel self-channel-enhanced convolution (S-Conv), which can adaptively model the intranode dependence and promote the classification performance. In brief, the G-Conv and S-Conv operations are together abbreviated as GSvolution for the embedding feature extraction.

C. Deep Metric Learning

Metric learning methods aim to optimize the embedding features through learning a distance-based prediction rule over the low-dimensional latent space. With the increasingly mature applications of DL to HSIC, deep metric learning (DML) also comes into being for HSIC, which generates the discriminative embedding space by deep neural networks [35]. For example, Liu *et al.* [36] introduced the Siamese network to extract the spectral–spatial underlying features, which obtains better classification performance than the conventional manner. Deng *et al.* [37] proposed a DML method based on the triplet loss for HSIC, achieving impressive classification results under the SSS conditions. However, in the above methods, only limited labeled samples are exploited in the embedding network, restricting the potential utility of the abundant unlabeled data. Thus, in this work, we introduce the semi-supervised GCN to serve as the deep embedding network for the DML and hence named a graph metric learning framework. To the best of our knowledge, we are the first to incorporate the DML into GCN for the HSIC task. In this way, both the labeled and unlabeled data can be put to good utilization, and the model decision boundaries that reflect the data distribution more closely can be determined by using the data itself [38]. Notably, a new loss function, which comprises a joint cross-entropy and metric term, is devised to optimize the proposed model to address the within-class spectral variance

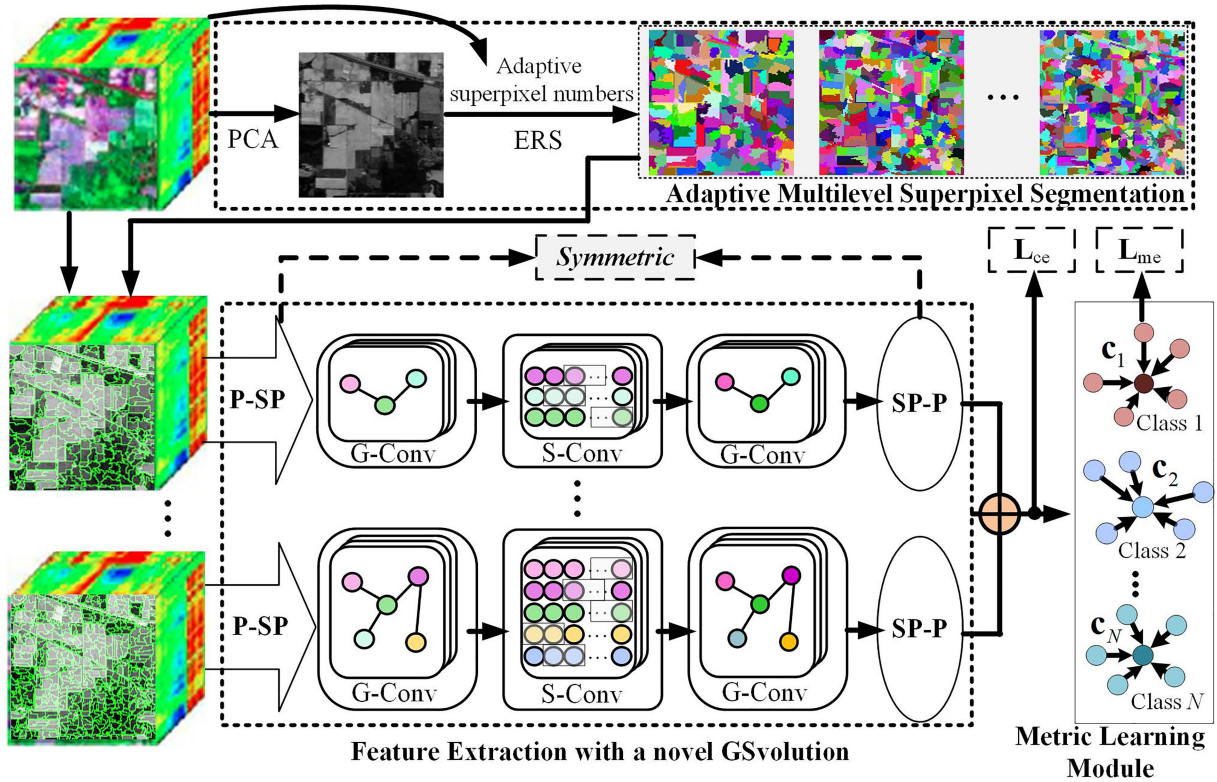


Fig. 1. Structural schematic diagram of the proposed SGML. The framework is composed of three parts: adaptive multilevel superpixel segmentation, feature extraction with a novel GSvolution, and a metric learning module.

and between-class spectral similarity, i.e., spectral uncertainty inherent to HSI scenes.

III. PROPOSED METHOD

A. Architecture of the Proposed SGML

As can be seen in Fig. 1, the proposed SGML mainly comprises three portions, including adaptive multilevel superpixel segmentation, feature extraction with a novel GSvolution, and a metric learning module. First, we segment the original HSI into multilevel superpixels by combining the Principal Component Analysis (PCA) and ERS methods, where the superpixel numbers are adaptively determined. Then, after projecting the pixel into the superpixel pattern (P-SP), we investigate the representative SP feature with a novel GSvolution, detailed in Section III-C. Next, the obtained in-depth features are reprojected to the pixel level (SP-P) and then aggregated by the summation operation. Finally, the assembled information is parsed by the metric learning module into the semantic category space. In the following, the above three parts are described minutely.

B. Multilevel Superpixel Segmentation

Compared with superpixel segmentation at a specific level, multilevel segmentation can better exploit different information of various shapes and sizes for the land-covers. In detail, we first apply PCA to the HSI and then perform the superpixel segmentation on the first PC. In this way, the calculation amount

is substantially decreased, and the segmentation result will be more compact because of the more discriminative features in the first PC. Here, we employ the ERS method to realize the superpixel segmentation at each level due to its ease of use. Most importantly, the correct number of superpixels should be set for multilevel superpixel segmentation to avoid inappropriate undersegmentation or redundant oversegmentation. This work adopts a heuristic approach to adaptively compute the number of superpixels P_m by using the spatial resolution res and spatial size $H \times W$ of the data set. In the formula, the method can be depicted as

$$P_m = \text{floor} \left(\frac{H \times W}{\text{floor}(100 \times 0.7^{\sqrt{res}}) \times 2^{m-1}} \right) \quad (6)$$

where $m = \{1, 2, \dots, M\}$ and M represents the number of the scales. floor diminishes the value to the nearest integer. This strategy has been validated effective in Refs. [39] and [40]. For instance, Fig. 2 shows the multilevel superpixel segmentation maps obtained from the public Pavia University¹ data set (PaviaU) via this method. The superpixel numbers at multilevel are 3142, 1571, and 785, respectively.

C. Feature Extraction With a Novel GSvolution

Next, we introduce how to explore the spectral–spatial features involved in the multilevel superpixels. For a brief, we will

¹[Online]. Available: <https://rslab.ut.ac.ir/data>

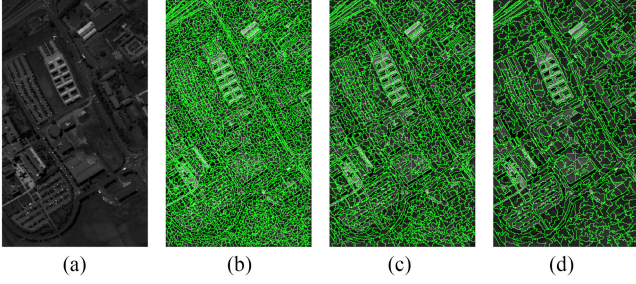


Fig. 2. First PC and adaptive multilevel superpixel segmentation maps obtained from the Pavia University data set. The figures are the superpixel numbers at different levels. (a) first PC (b) 3142 (c) 1571 (d) 785

describe the procedure of one stream of the network and the others keep consistent.

1) *P-SP*: As described in Section II-B, superpixels can offer a good description of the land-covers since their shapes and sizes can be automatically adjusted according to the HSI content; thus, we employ SP to facilitate the subsequent graph learning. Here, the SPs are obtained by weighted averaging the pixels included in one superpixel $V^{\text{SP}} = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_J\}$, which can be formulated as

$$\mathbf{x}^{\text{SP}} = \sum_{j=1}^J \mathbf{w}_{\mathbf{v}_j, \mathbf{v}_c} * \mathbf{v}_j \quad (7)$$

$$\mathbf{w}_{\mathbf{v}_j, \mathbf{v}_c} = \frac{\exp(-\varepsilon \|\mathbf{v}_j - \mathbf{v}_c\|_2^2)}{\sum_{j=1}^J \exp(-\varepsilon \|\mathbf{v}_j - \mathbf{v}_c\|_2^2)} \quad (8)$$

where $\mathbf{v}_c = \frac{1}{J} \sum_{j=1}^J \mathbf{v}_j$ denotes the average pixel of V^{SP} along the spatial dimension and ε is a scalar factor. $\exp(\cdot)$ means the exponential function.

2) *Deep SP Feature Extraction*: After P-SP, suppose that the labeled and unlabeled SPs are $\mathbf{X}_l^{\text{SP}} \in \mathbb{R}^{S_1 \times L} = \{\mathbf{x}_l^1, \mathbf{x}_l^2, \dots, \mathbf{x}_l^{S_1}\}$ and $\mathbf{X}_{ul}^{\text{SP}} \in \mathbb{R}^{S_2 \times L} = \{\mathbf{x}_{ul}^1, \mathbf{x}_{ul}^2, \dots, \mathbf{x}_{ul}^{S_2}\}$, respectively, where L denotes the length of each SP. S_1 and S_2 denote the number of superpixels containing the labeled and unlabeled samples, respectively, and $S_1 + S_2 = S$. Additionally, the corresponding labels of \mathbf{X}_l^{SP} are determined by majority voting of the contained pixels. Then, all SPs are adopted to construct the graph $G = (\mathbf{X}, \mathbf{A})$, where $\mathbf{X} \in \mathbb{R}^{S \times L}$ denotes the SP features of the graph nodes. \mathbf{A} is the adjacency matrix, which is calculated by

$$A_{i,j} = \begin{cases} e^{-\beta \|\mathbf{x}^i - \mathbf{x}^j\|_2^2}, & \text{if } \mathbf{x}^i \text{ adj } \mathbf{x}^j \\ 0, & \text{else} \end{cases} \quad (9)$$

where *adj* means the two SPs are adjacent and β represents a steepness hyperparameter affecting the weights.

Next, the graph signal is deeply exploited by the proposed GSvolution, which is fundamentally comprised of two G-Convs interacting with a novel S-Conv. In the G-Conv layer, we set K to 2 in (2) as same as Ref. [24]. Then, the operation can be represented as

$$\mathbf{g}_\phi * \mathbf{x}_g \approx \phi \left(\mathbf{I} + \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} + \left(\mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \right)^2 \right) \mathbf{x}_g \quad (10)$$

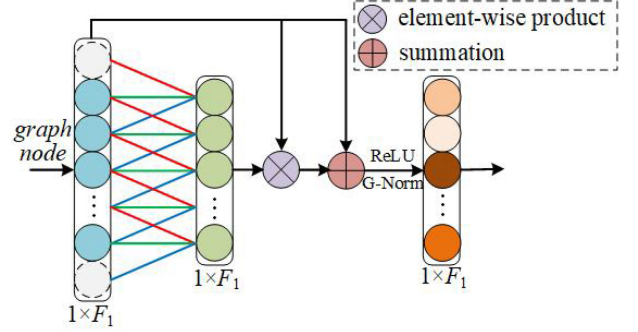


Fig. 3. Implementation of the proposed efficient self-channel-enhanced convolution (S-Conv).

where the graph signal \mathbf{x}_g is the feature in the same dimension overall the graph, differing from one arbitrary node feature \mathbf{x} (one SP vector in the first G-Conv). And then, the output of the first G-Conv layer can be expressed as

$$\mathbf{H}_1 = \left(\mathbf{I} + \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} + \left(\mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \right)^2 \right) \mathbf{X} \Phi_1 \quad (11)$$

where $\Phi_1 \in \mathbb{R}^{L \times F_1}$ represents the learnable filter parameters and $\mathbf{H}_1 \in \mathbb{R}^{S \times F_1}$.

Inspired by the effective structured CNNs, i.e., convolution, batch normalization [41], and ReLU [42] activation function, we design a unified structure for our SGML, i.e., G-Conv, graph normalization (G-Norm), and ReLU. That means, the obtained \mathbf{H}_1 is further processed by

$$\hat{\mathbf{H}}_1 = \psi(\text{G-Norm}(\mathbf{H}_1)) \quad (12)$$

where ψ denotes the ReLU function, and G-Norm can be formulated as

$$\hat{\mathbf{h}}_{1g} = \frac{\mathbf{h}_{1g} - \mu_{1g}}{\sigma_{1g}} \quad (13)$$

$$\mu_{1g} = \frac{1}{S} \sum_{i=1}^S \mathbf{h}_{1g}^{(i)}, \quad \sigma_{1g} = \sqrt{\frac{1}{S} \sum_{i=1}^S (\mathbf{h}_{1g}^{(i)} - \mu_{1g})^2} \quad (14)$$

where $\mathbf{h}_{1g} \in \mathbb{R}^{S \times 1}$ denotes one graph signal after the first G-Conv. Similar to CNN, the G-Norm can decrease the internal covariance shift of the graph signal, accelerate the convergence, and shorten the training procedure.

Considering the GCN theory, the G-Conv primarily propagates the information across different node features, regardless of the importance of the intranode relationships. However, especially for the HSI, the local and nonlocal characteristic spectrum corresponding to the graph node plays an essential role in discrimination. Thus, we devise a novel S-Conv in the GSvolution structure. Visually, the implementation of the proposed S-Conv is illustrated in Fig. 3.

Assuming that $\hat{\mathbf{h}}_1 \in \mathbb{R}^{F_1 \times 1}$ is an arbitrary hidden graph node, we first operate the light-weight 1-D-Conv on it and utilize the sigmoid function to scale the output into the scope of 0 and 1. This result is regarded as the importance factor of different channels of the graph node, and then it is imposed on the input graph node feature by element-wise product. Besides, to avoid

undesired information loss, we sum the original feature with the above result. In this self-supervised manner, the important channels are enhanced while the insignificant features are relatively constrained. In the formula, the S-Conv can be represented as

$$\mathbf{h}_2 = \hat{\mathbf{h}}_1 \odot (\sigma(\mathbf{w} * \hat{\mathbf{h}}_1) + 1) \quad (15)$$

where \mathbf{w} is the weight of the 1-D-Conv. \odot means the element-wise product operation. Afterward, we adopt ReLU to promote the nonlinear fitting ability of GSvolution and G-Norm to regularize the obtained features.

After getting the output $\hat{\mathbf{H}}_2$ of the S-Conv, the second G-Conv equipped with the G-Norm is carried out to produce the deep SP feature, which is denoted as $\hat{\mathbf{H}}_3 \in \mathbb{R}^{S \times N}$ and N equals to the number of the classes.

3) *SP-P*: As illustrated in Fig. 1, the numbers of the graph nodes for the SP feature are varying for different streams since the superpixels are in multiscale. Thus, to aggregate the extracted deep SP features, we reproject the SP in each scale into the pixel level and then sum them up to obtain the assembled information. Finally, the softmax function is utilized to transfer the output to the probability vectors $\mathbf{H}_4 \in \mathbb{R}^{S \times N}$ reflecting the semantic attributes.

D. Metric Learning Module

In the last stage, deep representative features are encouraged into the low-dimensional manifolds. However, the features may be less discriminative if we only adopt the naive cross-entropy (CE) loss. The softmax-based CE loss mainly focuses on enlarging the interclass distance, but less concerning the intraclass variance, which may result in inevitable misclassifications [43]–[45].

To conquer this deficiency, we design a metric learning module to restrict the discrepancy between each node representation and the centroid of each class. Specifically, the class-specified centroid in the semantic space can be computed by averaging the labeled embedding features of each class

$$\mathbf{c}_k = \frac{1}{|\mathbf{V}_k|} \sum_{(\mathbf{x}_k, \mathbf{y}_k) \in \mathbf{V}_k} \mathbf{H}_4^{(\mathbf{x}_k, \mathbf{y}_k)} \quad (16)$$

where \mathbf{V}_k is the collection of the k th class labeled samples. $\mathbf{H}_4^{(\mathbf{x}_k, \mathbf{y}_k)}$ means the corresponding embedding features. Based on the centroids, the proposed metric loss can be formulated as

$$L_{\text{me}} = - \sum_{(\mathbf{x}_l, \mathbf{y}_l) \in \mathbf{V}_l} \log \frac{\exp[-dis(\mathbf{h}_l, \mathbf{c}_{y_l})]}{\sum_{n=1, n \neq l}^N \exp[-dis(\mathbf{h}_l, \mathbf{c}_{y_n})]} \quad (17)$$

where dis calculates the distance of the latent features and the centers. We can observe that, through minimizing L_{me} , the interclass distance can be enlarged while the intraclass distance is encouraged to be compact. Thus, the embedding features can be separated into the accurate category. Finally, we can summarize that the loss function in our proposed SGML is

$$L = L_{\text{ce}} + \alpha L_{\text{me}} \quad (18)$$

where L_{ce} is the commonly used CE loss function. α is a hyperparameter influencing the weight of the metric loss. For

Algorithm 1: Pseudo-Code of Proposed SGML.

Input: The HSI (\mathbf{D}), true labels of the labeled data, locations of the labeled and unlabeled data.

Parameters: H, W, res ; the number of the scales (M), $\varepsilon, \alpha, \beta$; *epoch number, learning rate*.

Output: Predicted labels of the unlabeled data.

Step1: Perform PCA on \mathbf{D} to obtain the first PC;

Step2: Conduct ERS on the first PC to obtain the m th ($m = 1, 2 \dots M$)-level superpixel maps by using (6);

Step3: Project the pixel-level data to the m th level superpixels according to (7) and (8);

Step4: Construct multiscale adjacency matrices by (9);
bf for $i = 1$ to epoch number **do**

Step5: Calculate the embedding features $\hat{\mathbf{H}}_3$ of the m th level superpixels according to (10)–(15);

Step6: Reproject $\hat{\mathbf{H}}_3$ to the pixel-level and then obtain the probability vectors $\hat{\mathbf{H}}_4$;

Step7: Compute the class centroid \mathbf{c}_k by (16);

Step8: Optimize the network parameters driven by the loss function (18);

Step9: $i = i + 1$

end

better understanding, the pseudo-code of the proposed SGML is depicted in Algorithm 1.

IV. EXPERIMENT RESULTS AND ANALYSIS

To demonstrate the effectiveness and high-efficiency of the proposed SGML, we conduct sufficient experiments on three benchmark HSIC data sets, such as Indian Pines (IndianP), PaviaU, and Kennedy Space Center (KSC). Specifically, some prevalent and state-of-the-art DL networks based on CNN and GCN are taken as the comparisons, i.e., 3-D-CNN [12], SSRN [13], FDSSC [14], DR-CNN [15], GCN [23], S² GCN [24], and MDGCN [28]. All the algorithms are conducted on the NVIDIA GeForce GTX 1080 Ti GPU and completed with TensorFlow DL framework. For the quantitative evaluation of the performance, we adopt the widely used class-specific accuracy (CA), average accuracy (AA), overall accuracy (OA), and Kappa coefficient (Kappa). Additionally, we depict the corresponding classification maps for visual and qualitative analysis.

A. Experiment Data Sets

1) *Indian Pines*: The first benchmark HSI data set named IndianP was captured by the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) over Northwestern Indiana. The spatial resolution is 20 m and the spatial size is 145×145 . In the spectral view, it recorded the reflection values of 220 bands from 0.4 to 2.5 μm . In our experiments, 200 bands are preserved after removing the noisy and water-absorbed ones. The R-G-B composite image is shown in Fig. 4(a). Besides, Fig. 4(b) and (c) depicts the train and test samples, respectively. The specific numbers of the train, test, and total samples for each class are

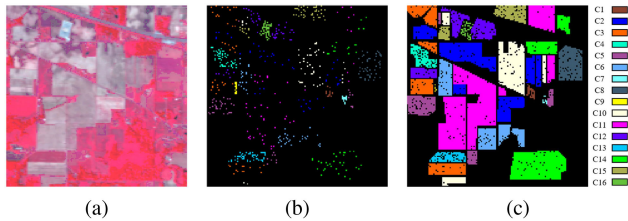


Fig. 4. Indian Pines. (a) R-G-B composite image (R-50th, G-27th, and B-17th bands). (b) Standard train samples. (c) Test samples. Best zoomed-in view.

TABLE I
LABELED LAND-COVER TYPES AND NUMBERS OF THE TRAIN, TEST,
AND TOTAL SAMPLES FOR INDIANP DATA SET

No.	Land-Cover Type	Train	Test	Total
C1	Alfalfa	15	39	54
C2	Corn-notill	50	1384	1434
C3	Corn-mintill	50	784	834
C4	Corn	50	184	234
C5	Grass-pasture	50	447	497
C6	Grass-trees	50	697	747
C7	Grass-pasture-mowed	15	11	26
C8	Hay-windrowed	50	439	489
C9	Oats	15	5	20
C10	Soybean-notill	50	918	968
C11	Soybean-mintill	50	2418	2468
C12	Soybean-clean	50	564	614
C13	Wheat	50	162	212
C14	Woods	50	1244	1294
C15	Buildings-Grass-Trees-Drives	50	330	380
C16	Stone-Steel-Towers	50	45	95
Total		695	9671	10366

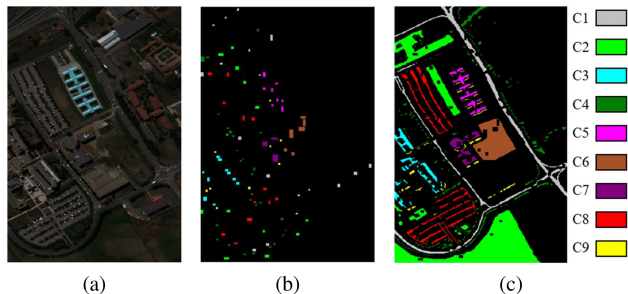


Fig. 5. Pavia University. (a) R-G-B composite image (R-30th, G-25th, and B-18th bands). (b) Standard train samples. (c) Test samples. Best zoomed-in view.

listed in Table I. Since IndianP has much noise and spectral variability, it is challenging and widely used to evaluate the newly proposed methods in the HSIC community.

2) *Pavia University*: The second PaviaU scene was obtained by the Reflective Optics System Imaging Spectrometer (ROSIS) facility over an urban region. It comprises 610×430 pixels with the spatial resolution of 1.3 m. In the spectral dimension, 103 bands are remained in the wavelength from 0.43 to $0.86 \mu\text{m}$ after abandoning the noisy bands. Fig. 5(a)–(c) shows its R-G-B composite image, train map, and test map, respectively. In detail, the numbers of the train, test, and total samples for each class are detailed in Table II.

3) *Kennedy Space Center*: The last data set, KSC, is also collected by the AVIRIS sensor over Kennedy Space Center,

TABLE II
LABELED LAND-COVER TYPES AND NUMBERS OF THE TRAIN, TEST,
AND TOTAL SAMPLES FOR PAVIAU DATA SET

No.	Land-Cover Type	Train	Test	Total
C1	Asphalt	548	6304	6852
C2	Meadows	540	18146	18686
C3	Gravel	392	1815	2207
C4	Trees	524	2912	3436
C5	Painted metal sheets	265	1113	1378
C6	Bare Soil	532	4572	5104
C7	Bitumen	375	981	1356
C8	Self-Blocking Bricks	514	3364	3878
C9	Shadows	231	795	1026
Total		3921	40002	43923

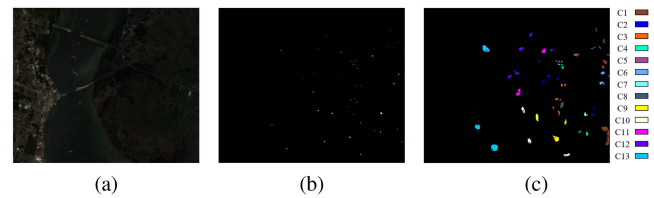


Fig. 6. Kennedy Space Center. (a) R-G-B composite image (R-50th, G-30th, and B-20th bands). (b) Controlled random sampling train samples. (c) Test samples. Best zoomed-in view.

TABLE III
LABELED LAND-COVER TYPES AND NUMBERS OF THE TRAIN, TEST,
AND TOTAL SAMPLES FOR KSC DATA SET

No.	Land-Cover Type	Train	Test	Total
C1	Scrub	30	731	761
C2	Willow swamp	30	213	243
C3	CP hammock	30	226	256
C4	CP/Oak	30	222	252
C5	Slash pine	30	131	161
C6	Oak/Broadleaf	30	199	229
C7	Hardwood swamp	30	75	105
C8	Graminoid marsh	30	401	431
C9	Spartina marsh	30	490	520
C10	Cattail marsh	30	374	404
C11	Salt marsh	30	389	419
C12	Mud flats	30	473	503
C13	Water	30	897	927
Total		390	4821	5211

Florida. It includes 224 bands in the wavelength from 0.4 to $2.5 \mu\text{m}$. In the experiments, 176 bands are utilized after discarding the low SNR and water absorption channels. For the spatial dimension, the image comprises 614×512 pixels with a geographic resolution of 18 m. The R-G-B composite image, train, and test samples are demonstrated from Fig. 6(a) to (c), respectively, while the numbers of the train, test, and total samples for each class are displayed in Table III.

Noticeably, to ensure fair comparisons and reliable conclusion, the train and test samples are consistent with the recent HSIC works for the first two data sets, which is regarded as a standard data split manner [25], [46]. For the KSC data set, different from the random sampling way, we utilize the

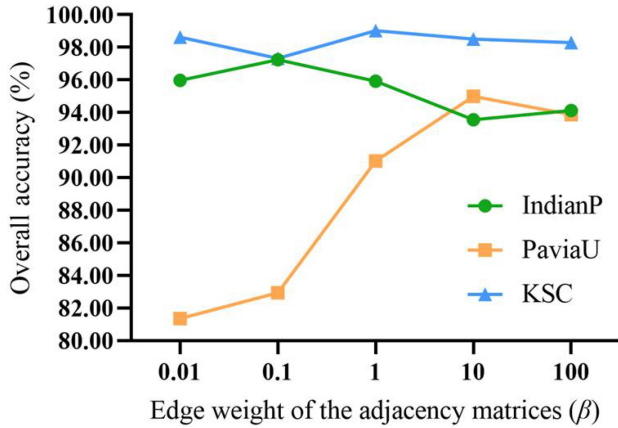


Fig. 7. Influence of the edge weight of the adjacency matrices.

controlled random sampling strategy to acquire the training data [47]–[49]. In this way, the training samples are gathered locally and distributed globally, which is closer to the real applications, and more realistic performance of the spectral–spatial classifier can be accessed [10]. Moreover, among the training data, we randomly select 10% samples as the validation set to tune the hyperparameters of the proposed frameworks.

B. Experimental Settings

To maximize the utility of the proposed SGML, we carry out elaborate experimental settings.

1) *Superpixel Segmentation*: First, three levels of superpixel segmentation are utilized, which correspond to three spatial scales in the afterward feature extraction process.

2) *Gsvolution*: According to the validation accuracy, the scale factor ε in (8) is set to 1 for the IndianP and KSC data set, and 100 for the PaviaU scene. Then, we investigate the influence of the edge weight parameter of the adjacency matrices on the three data sets. The grid search strategy is adopted here and the β varies in $\{0.01, 0.1, 1, 10, 100\}$. Fig. 7 depicts the obtained OAs. We can find that the classification results achieve the best when β equals to 0.1, 10, and 1 for the IndianP, PaviaU, and KSC data sets, respectively. Thus, the β is experimentally assigned to 0.1, 10, and 1 for the three data sets, respectively.

Additionally, we analyze the influence of the output dimension of the first G-Conv. In detail, the number of the hidden nodes varies in $\{16, 32, 64, 128\}$ and the acquired OAs are shown in Fig. 8. It can be observed that the classification performance reaches the highest point when the output dimension is 32. Thus, we set it to 32 in our experiments. Moreover, the kernel size of the S-Conv is empirically set to 1×3 . Regarding the parameter optimization, we adopt the efficient Adam algorithm with the full-batch gradient descent method. The learning rate is experimentally set to $5 \times e^{-4}$ for IndianP and KSC data sets, and $5 \times e^{-5}$ for PaviaU. Besides, the epoch number is assigned to 500.

3) *Distance Metric*: For the distance metric in the metric learning module, we test the Euclidean and Cosine distance. The classification accuracies for the experimental data sets are

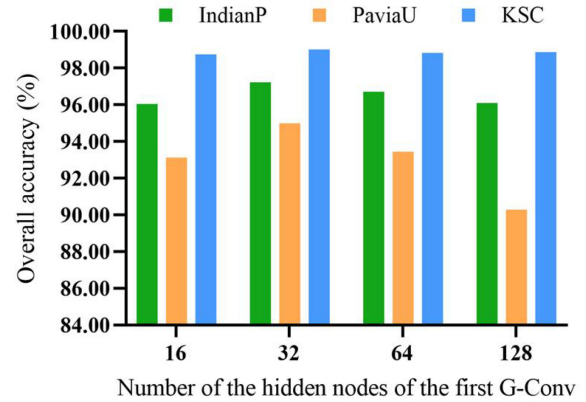


Fig. 8. Influence by the number of the hidden nodes of the first G-Conv.

TABLE IV
CLASSIFICATION ACCURACY WITH DIFFERENT DISTANCE METRICS IN THE METRIC LEARNING MODULE

Data	Metric	AA \uparrow	OA \uparrow	Kappa \uparrow
IndianP	Euclidean	0.9772	0.9698	0.9655
	Cosine	0.9784	0.9724	0.9684
PaviaU	Euclidean	0.9589	0.9463	0.9286
	Cosine	0.9649	0.9499	0.9334
KSC	Euclidean	0.9787	0.9849	0.9831
	Cosine	0.9833	0.9902	0.9890

TABLE V
CLASSIFICATION RESULTS OF THE ABLATION STUDIES ON PAVIAU DATA SET

Variants	S-Conv?	Metric loss?	AA \uparrow	OA \uparrow	Kappa \uparrow
Var. 1	×	×	0.9304	0.9262	0.9022
Var. 2	✓	×	0.9588	0.9463	0.9285
Var. 3	×	✓	0.9368	0.9350	0.9138
Var. 4	✓	✓	0.9649	0.9499	0.9334

detailed in Table IV. It can be observed that the classification performance obtained by the Cosine distance is always slightly better than that of the Euclidean distance. Thus, we choose Cosine distance in the metric loss. Furthermore, the weight parameter α is experimentally set to 1 for PaviaU and 0.1 for the other two data sets, respectively.

C. Ablations Study

In this section, ablation studies are conducted to demonstrate the contributions of the proposed S-Conv in the Gsvolution and metric loss in the metric learning module. In specific, there are four variants in our experiment. Var.1 to Var.3 represent the SGML without both, without metric loss, and without S-Conv, respectively, while Var.4 denotes the SGML itself. Table V provides the classification results on PaviaU data set. It can be found that Var.4 can achieve the optimal performance, and the accuracies decrease either S-Conv or metric loss is disabled. Concretely, the AA of Var.4 surpasses Var.1 to Var.3 by 3.45%, 0.61%, and 2.81%, respectively; the OA of Var.4 exceeds Var.1 to Var.3 by 2.37%, 0.36%, and 1.49%, respectively; the

TABLE VI
CLASSIFICATION PERFORMANCE OF VARIOUS METHODS FOR INDIANP DATA SET

No.	3-D-CNN [12]	SSRN [13]	FDSSC [14]	DR-CNN [15]	GCN [23]	S ² GCN [24]	MDGCN [28]	SGML
1	100.00±0.00	71.79±1.88	<u>97.44±0.34</u>	100.00±0.00	100.00±0.00	100.00±0.00	84.90±0.09	100.00±0.00
2	66.11±1.48	77.60±1.22	85.26±0.48	82.17±0.62	82.95±0.33	81.29±1.03	98.09±0.12	<u>93.82±0.20</u>
3	79.59±1.23	96.68±0.65	99.11±0.13	84.36±0.89	80.61±0.37	87.76±0.24	100.00±0.00	<u>99.62±0.11</u>
4	100.00±0.00	100.00±0.00	100.00±0.00	85.65±0.99	95.65±0.10	100.00±0.00	95.53±0.02	<u>98.44±0.03</u>
5	<u>97.99±0.35</u>	96.42±0.33	91.72±1.37	91.60±0.48	93.29±0.29	97.09±0.38	96.99±0.06	98.90±0.05
6	90.24±0.78	97.85±0.67	98.13±0.49	100.00±0.00	92.25±0.18	91.68±0.14	<u>99.09±0.06</u>	97.87±0.08
7	100.00±0.00	100.00±0.00	100.00±0.00	91.67±0.63	100.00±0.00	100.00±0.00	92.59±0.45	93.75±0.15
8	96.81±0.41	100.00±0.00	100.00±0.00	100.00±0.00	97.95±0.37	99.32±0.06	97.85±0.02	100.00±0.00
9	100.00±0.00	100.00±0.00	100.00±0.00	100.00±0.00	100.00±0.00	100.00±0.00	<u>95.74±0.03</u>	100.00±0.00
10	72.22±1.01	70.70±1.07	79.85±1.35	82.42±0.88	89.11±0.96	93.46±0.34	<u>98.77±0.13</u>	99.35±0.01
11	75.43±0.62	<u>96.69±0.31</u>	79.78±1.21	95.70±0.09	79.20±1.34	85.03±0.97	100.00±0.00	96.66±0.05
12	79.08±0.31	89.01±1.32	98.05±0.44	80.32±1.34	86.35±0.31	84.22±0.25	100.00±0.00	89.46±0.06
13	100.00±0.00	100.00±0.00	100.00±0.00	98.18±0.14	100.00±0.00	100.00±0.00	100.00±0.00	<u>99.42±0.08</u>
14	94.61±0.84	99.04±0.12	99.12±0.15	99.66±0.19	95.98±0.16	97.11±0.24	100.00±0.00	<u>99.68±0.06</u>
15	94.24±0.17	97.27±0.25	99.70±0.09	77.62±2.38	92.73±0.67	93.94±0.41	100.00±0.00	98.52±0.17
16	100.00±0.00	100.00±0.00	<u>82.22±1.74</u>	78.95±1.26	100.00±0.00	100.00±0.00	100.00±0.00	100.00±0.00
AA (%) ↑	90.40±1.03	93.32±0.87	94.40±0.93	90.52±0.71	92.88±0.42	94.43±0.40	<u>97.47±0.04</u>	97.84±0.02
OA (%) ↑	81.67±0.79	91.62±0.62	90.00±0.63	90.01±0.55	87.15±0.27	89.70±0.25	95.71±0.13	99.20±0.14
Kappa × 100 ↑	79.15±0.89	90.35±0.52	88.65±0.81	88.61±0.49	85.35±0.35	88.24±0.21	<u>95.07±0.18</u>	96.84±0.16

TABLE VII
CLASSIFICATION PERFORMANCE OF VARIOUS METHODS FOR PAVIAU DATA SET

No.	3-D-CNN [12]	SSRN [13]	FDSSC [14]	DR-CNN [15]	GCN [23]	S ² GCN [24]	MDGCN [28]	SGML
1	82.61±1.62	95.11±0.27	98.81±0.19	98.62±0.17	84.38±0.55	82.22±0.48	90.74±0.21	89.94±0.09
2	81.54±1.33	77.19±1.98	75.06±1.74	99.87±0.01	76.20±1.01	82.60±0.77	<u>97.17±0.09</u>	94.69±0.06
3	57.19±2.34	43.14±2.19	74.55±1.08	84.51±1.23	83.58±1.51	79.06±0.71	<u>88.37±0.16</u>	99.73±0.05
4	98.73±0.21	89.32±0.33	96.57±0.09	96.40±0.14	94.40±0.06	<u>97.70±0.10</u>	96.36±0.03	91.38±0.02
5	99.55±0.14	100.00±0.00	98.02±1.22	98.92±0.65	99.91±0.07	100.00±0.00	96.86±0.02	96.21±0.01
6	92.98±1.35	100.00±0.00	98.05±1.27	67.99±2.74	94.73±0.98	95.32±0.16	86.20±0.06	100.00±0.00
7	81.86±1.47	99.39±0.21	<u>84.00±1.88</u>	91.16±0.94	95.11±0.54	98.78±0.33	82.57±0.28	100.00±0.00
8	97.24±0.41	99.82±0.18	95.54±0.22	71.44±2.21	95.01±0.51	94.59±0.14	96.46±0.03	96.95±0.02
9	94.21±1.43	99.12±0.26	96.86±1.33	97.17±0.85	96.60±0.23	100.00±0.00	97.36±0.02	<u>99.53±0.01</u>
AA (%) ↑	87.32±1.30	89.23±0.71	90.83±0.45	89.56±0.77	91.10±0.45	92.25±0.39	<u>92.45±0.04</u>	96.49±0.01
OA (%) ↑	85.25±1.44	85.48±1.14	85.99±1.57	89.95±0.83	84.38±0.50	87.17±0.47	<u>94.02±0.03</u>	94.99±0.02
Kappa × 100 ↑	80.77±1.41	81.22±1.65	81.98±1.39	86.78±1.05	79.93±0.77	83.31±0.41	<u>91.91±0.06</u>	93.34±0.02

Kappa of Var.4 surmounts Var.1 to Var.3 by 3.12%, 0.49%, and 1.96%, respectively. These results indicate that the S-Conv can enhance the node feature representation by assigning different importance to the channels while the metric loss can promote the feature discrimination, which cooperatively improve the classification performance.

D. Performance Evaluation

In this section, we evaluate the proposed SGML compared with some advanced methods. In specific, the 3-D-CNN [12], SSRN [13], FDSSC [14], and DR-CNN [15] are highly cited approaches in the literature, which are all CNN-based spectral-spatial classifiers, and the DR-CNN also involves multiscale spatial features into a multistream fashion like the proposed networks. Additionally, GCN [23] is the first work introducing the convolutional operation directly on the graphs. Thus, we employ it as one baseline method. As for other GCN-based methods, S²GCN [24] first takes the spatial information of the HSI into account; MDGCN [28] employs the superpixel segmentation technique and utilizes dynamic graph construction in different graph convolutional layers, which can represent the state-of-the-art GCN-related algorithms. The hyperparameters of these compared methods are assigned as the original papers. In our experiments, all methods are tested 10 times on the three

benchmarks. Minutely, the average CA, AA, OA, and Kappa with the standard deviations are exhibited in Tables VI–VIII. The highest and the second-highest values of each line are highlighted in bold and underline, respectively.

Table VI displays the classification results of the IndianP data set, where the SGML achieves the best performance concerning AA, OA, and Kappa. For the CA, the SGML ranks the first and second for five and six categories, respectively. Among the CNN-based methods, SSRN and FDSSC acquire better results due to the residual and dense connections in the networks. Although GCN only leverages the spectral features, it surpasses 3-D-CNN a lot in AA, OA, and Kappa, proving the potential of the graph-related HSIC methods to some degree. Furthermore, MDGCN acquires impressive classification performance, of which seven classes are predicted perfectly. However, for all the small categories with fewer training samples, i.e., the first, seventh, and ninth classes, the proposed SGML achieves higher accuracies than the competitive MDGCN. It implies that SGML can effectively handle the SSS problem, which is much valuable in practical applications.

From Table VII, it can be observed that the proposed SGML acquires the optimal classification performance on PaviaU data set. Different from the results on IndianP, the DR-CNN achieves a relatively better performance here. The reason can be speculated that PaviaU has more irregular boundaries and scatter

TABLE VIII
CLASSIFICATION PERFORMANCE OF VARIOUS METHODS FOR KSC DATA SET

No.	3-D-CNN [12]	SSRN [13]	FDSSC [14]	DR-CNN [15]	GCN [23]	S ² GCN [24]	MDGCN [28]	SGML
1	97.95±0.98	99.73±0.15	100.00±0.00	100.00±0.00	97.26±0.65	95.21±0.11	100.00±0.00	99.45±0.14
2	87.32±1.45	93.43±0.61	100.00±0.00	100.00±0.00	91.08±0.61	92.02±0.52	100.00±0.00	100.00±0.00
3	89.38±1.29	100.00±0.00	100.00±0.00	98.23±0.88	96.02±0.41	99.56±0.29	100.00±0.00	99.58±0.06
4	44.34±2.95	92.31±1.81	75.11±2.42	95.71±0.49	77.83±0.62	88.24±0.79	83.71±0.19	99.10±0.09
5	48.85±2.87	94.66±0.65	98.47±0.43	62.33±3.22	83.21±1.06	72.52±1.43	46.56±1.68	84.17±0.57
6	83.42±0.54	65.83±1.86	36.18±2.99	100.00±0.00	74.87±1.62	84.42±1.29	98.49±0.08	100.00±0.00
7	66.67±1.87	82.67±0.86	90.67±0.79	60.00±2.78	92.00±1.06	89.33±0.77	100.00±0.00	100.00±0.00
8	88.53±1.31	100.00±0.00	96.01±0.69	82.85±1.30	96.01±1.01	96.76±0.81	100.00±0.00	100.00±0.00
9	84.90±0.56	93.47±1.49	97.55±1.33	100.00±0.00	78.37±1.28	91.22±0.66	100.00±0.00	100.00±0.00
10	81.23±1.71	100.00±0.00	100.00±0.00	100.00±0.00	89.28±0.72	95.44±0.64	98.12±0.19	99.73±0.08
11	87.92±1.25	89.20±1.09	95.37±0.83	100.00±0.00	100.00±0.00	88.95±0.09	100.00±0.00	100.00±0.00
12	76.11±1.37	100.00±0.00	100.00±0.00	97.53±0.64	95.35±0.51	98.94±0.42	99.79±0.02	96.27±0.69
13	99.55±0.17	100.00±0.00	100.00±0.00	100.00±0.00	95.88±0.16	99.22±0.12	100.00±0.00	100.00±0.00
AA (%) ↑	79.71±1.30	93.18±0.51	91.49±0.23	92.05±0.21	89.78±0.59	91.68±0.22	94.36±0.12	98.33±0.10
OA (%) ↑	86.14±1.28	95.95±0.41	95.08±0.19	95.64±0.33	91.78±0.51	94.15±0.39	97.57±0.16	99.02±0.15
Kappa × 100 ↑	84.47±1.44	95.46±0.39	94.48±0.25	95.12±0.20	90.82±0.34	93.46±0.29	97.28±0.14	98.90±0.16

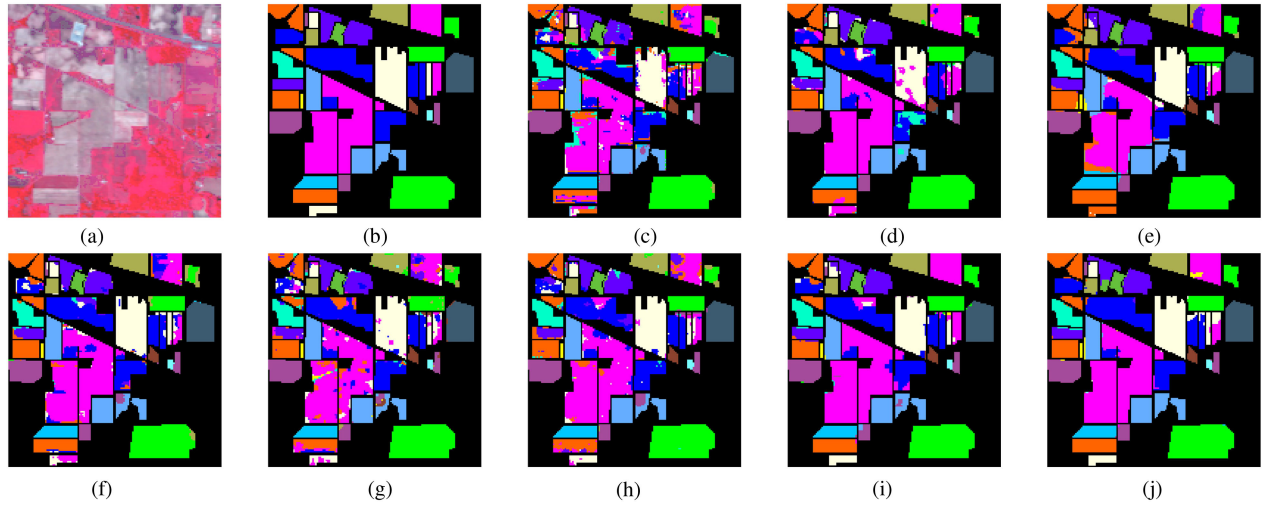


Fig. 9. Classification maps produced by different methods by using IndianP data set. Best zoomed-in view. (a) Composite image (b) Groundtruth (c) 3-D-CNN (d) SSRN (e) FDSSC (f) DR-CNN (g) GCN (h) S²GCN (i) MDGCN (j) SGML

regions than IndianP scene, where the diverse region can play a more critical role in exploring sophisticated spectral–spatial features. Preferably, the multilevel superpixel segmentation in SGML can work more effectively to perceive the complicated spatial features. Besides, although the SGML acquires slightly better OA and Kappa than the MDGCN, the AA of SGML outperforms MDGCN by a substantial margin of 4.04%, which explicitly demonstrates its advancement.

In Table VIII of the KSC data set, we can observe that the presented SGML surmounts the second-best method MDGCN by 3.97%, 1.45%, and 1.62% about AA, OA, and Kappa, respectively. In addition, the SSRN obtains the competitive results benefited from its powerful spectral–spatial feature extraction capability. Furthermore, the MDGCN exhibits an expressive classification performance on KSC for predicting 8/13 land-cover types perfectly. Most significantly, among the graph-related methods, the accuracy of the fifth category (Slash pine) is relatively low while the SGML achieves the best. It indicates that the multilevel superpixel segmentation, GSvolution, and metric learning module jointly make

the designed framework an outstanding and stable graph model.

With respect to qualitative analysis, Figs. 9–11 draw the classification maps produced by different methods by using IndianP, PaviaU, and KSC data sets. For convenient comparisons, the R-G-B composite image and the groundtruth are also delineated. From Figs. 9 to 11, we can observe that the proposed SGML achieves the most excellent performance since its classification maps are closest to the groundtruth, with fewer misclassifications than the comparisons. Generally, the classification maps obtained by the 3-D-CNN, SSRN, FDSSC, and DR-CNN are compact due to the utilized spatially neighborhood samples, even oversmooth for several results, such as FDSSC for IndianP shown in Fig. 9(e) and SSRN for PaviaU drawn in Fig. 10(d). By contrast, many scattered salt-and-pepper noises exist in the maps of GCN, which only exploits the spectral characteristics. Particularly, it is observed that the proposed SGML can achieve precise results at the edge of the regions, where the boundary samples that are difficult to distinguish are mainly distributed. It implies that the proposed SGML can

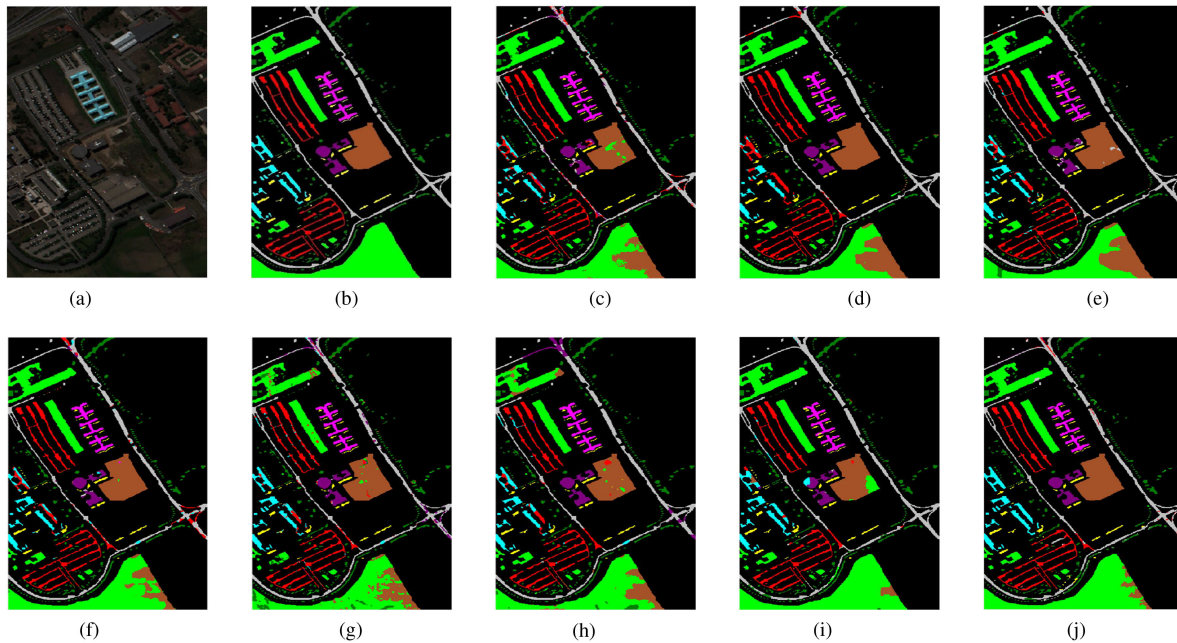


Fig. 10. Classification maps produced by different methods by using PaviaU data set. Best zoomed-in view. (a) Composite image (b) Groundtruth (c) 3-D-CNN (d) SSRN (e) FDSSC (f) DR-CNN (g) GCN (h) S^2GCN (i) MDGCN (j) SGML

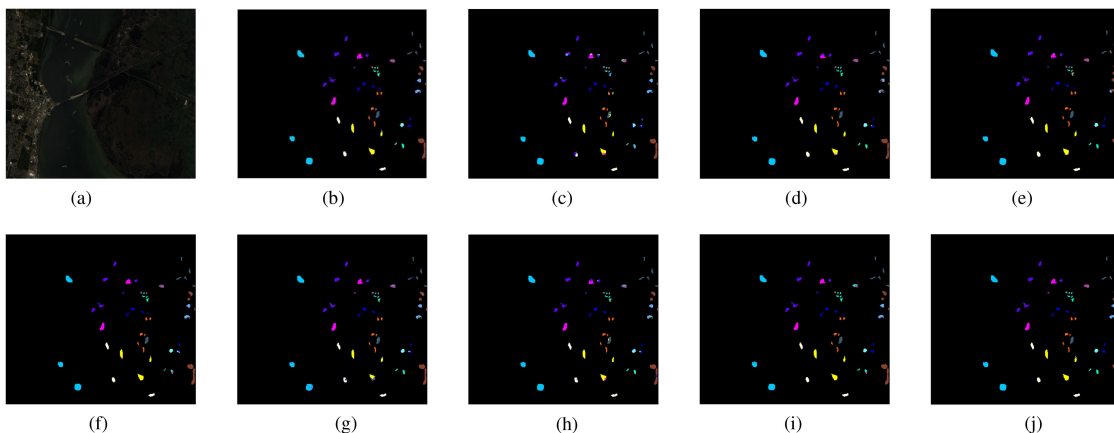


Fig. 11. Classification maps produced by different methods by using KSC data set. Best zoomed-in view. (a) Composite image (b) Groundtruth (c) 3-D-CNN (d) SSRN (e) FDSSC (f) DR-CNN (g) GCN (h) S^2GCN (i) MDGCN (j) SGML

effectively extract the expressive and discriminative features to identify various land-covers even with similar spectral-spatial information.

E. Performance With Various Number of Training Samples

The classification performance under the SSS conditions is critical in real applications for HSIC. Therefore, we exploit the capacity of the SGML compared to the other methods with smaller sizes of the training set. All experiments are executed 10 times with the training samples randomly generated from Figs. 4(b) to 6(b) in the percentage from 20% to 100%, respectively. Meanwhile, the test sets are kept same with Figs. 4(c) to 6(c). Concretely, the average OAs are delineated from Fig. 12(a) to (c) for the three data sets.

From Fig. 12, it can be observed that the proposed SGML always occupies the first place for different data sets with varying training samples. The classification performance of 3-D-CNN, SSRN, FDSSC, DR-CNN, GCN, and S^2GCN dramatically drops with the training sample decreases. By contrast, the accuracies of the MDGCN and SGML downgrade slightly. Especially for the PaviaU of SGML, the OA can reach 93.80% with only 20% samples of the training set, while that is 94.99% with the full training set. The results demonstrate the stability of the proposed SGML. The reason can be inferred as two aspects: First, the proposed GSvolution can facilitate the network to distill more representative features contained in the small-size training set, and second, the presented metric learning module can force the features to be more discriminative, which further enhances the prediction capability of the classifier.

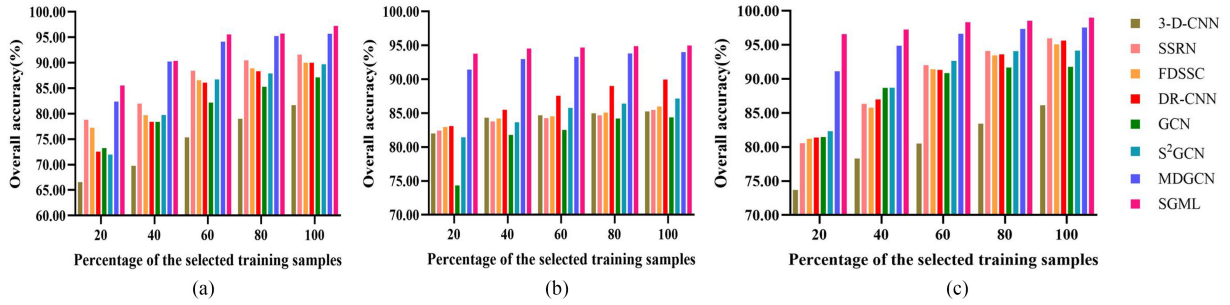


Fig. 12. Overall accuracy of 3-D-CNN, SSRN, FDSSC, DR-CNN, GCN, S^2 GCN, MDGCN, and the proposed SGML with different number of training samples. (a) IndianP. (b) PaviaU. (c) KSC.

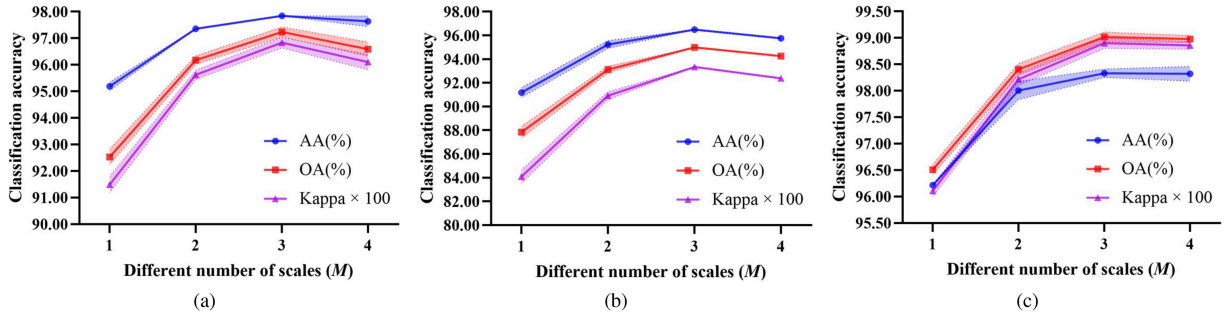


Fig. 13. Classification performance with different number of scales (M). (a) IndianP. (b) PaviaU. (c) KSC.

F. Influence of Different Number of Scales

The adaptive multilevel superpixel segmentation contributes to the afterward multiscale P-SP process, which assists the SGML to precisely exploit the spatial information in various scales. In this part, we validate the significance of different number of considered scales. Specifically, we keep the network settings fixed while only change the number of the streams according to how many scales are adopted. The experiments are repeated ten times on IndianP, PaviaU, and KSC data sets with 1–4 scales. The average classification accuracies and standard deviations are depicted in Fig. 13. We can observe that the classification performance is promoted as the scale increases, and the accuracies reach the top when three scales are utilized. It is attributed to that the spectral uncertainty is effectively suppressed when the multiscale spatial relationship of the samples is sufficiently exploited. However, when four scales are used, the accuracy no longer improves and even decreases significantly on IndianP and PaviaU data sets. The reason is that the undersegmentation for the fourth scale is too coarse due to the superpixel number being the smallest among these four scales. Consequently, the large-size superpixels will bring unexpected disturbance to the SPs, leading to undesired misclassifications.

G. Computational Cost

Benefiting from the symmetric P-SP and SP-P structure, the multiscale training and test processes of the proposed SGML can be efficiently done, which is one of the most important novelties in this work. Therefore, we access the computational cost of the SGML compared with other graph-based baseline methods, i.e., GCN, S^2 GCN, and MDGCN. In detail, the training and testing time on the experimental data sets are detailed in Table IX.

TABLE IX
ELAPSED TIME (S) OF THE TRAINING AND TESTING PROCESSES OF THE PROPOSED AND THE COMPARED GRAPH-RELATED METHODS

Methods		IndianP	PaviaU	KSC
GCN	Training time	192.16	329.45	42.57
	Testing time	0.02	0.06	0.01
S^2 GCN	Training time	251.61	595.09	50.98
	Testing time	0.03	0.10	0.02
MDGCN	Training time	62.54	350.47	24.75
	Testing time	1.11	5.84	0.13
SGML	Training time	4.61	9.69	15.96
	Testing time	0.15	0.31	0.13

It can be observed that the S^2 GCN is time-consuming due to the adjacency matrix construction searching for the K -nearest neighbors [24]. Significantly, the SGML is the fastest approach. Although the MDGCN obtains appropriate classification accuracies to the SGML, the SGML is almost 40 times faster than the MDGCN on the PaviaU data set. It is mainly because the training epoch number is 500 for SGML, while it is 5000 for the MDGCN, implying the proposed framework is indeed an efficient and rational model for HSIC.

V. CONCLUSION

In this work, a novel symmetric graph metric learning (SGML) framework is designed for efficient HSIC. In the presented SGML, we first conduct adaptive multilevel superpixel segmentation on the HSI data set to exploit the multiscale spatial information. After projecting the pixels to the superpixel pattern (P-SP), we propose a new GSvolution to sufficiently investigate

the deep representative spectral–spatial features. In the GSvolution, an innovative S-Conv is put forward to automatically enhance the significant node features while suppressing the trivial channels. Finally, the extracted features are reprojected to the pixel-level (SP-P) and transferred into the metric learning module to parse the semantic annotations. Notably, the symmetric P-SP and SP-P structure can dramatically reduce the computational costs and accelerate the training and test processes. Moreover, the proposed metric loss in the metric learning module can enhance the discrimination of the extracted features, namely, pulling the same classes close and pushing the different classes far away, which further promotes the classification performance. Consequently, the experiment results on three benchmark data sets verify that the proposed SGML outperforms the compared prevalent and state-of-the-art HSIC algorithms in accuracy and efficiency, even under the severe SSS conditions.

ACKNOWLEDGMENT

The authors would like to thank Prof. P. Gamba for providing the Pavia University data set. They appreciate the GRSS Image Analysis and Data Fusion Technical Committee for releasing the standard training-test sets of the benchmark data set for developing new algorithms.² The authors would also like to thank the editors and anonymous reviewers for their valuable comments and suggestions to help improve the quality of this manuscript.

REFERENCES

- [1] M. Fauvel, Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton, “Advances in spectral–spatial classification of hyperspectral images,” *Proc. IEEE*, vol. 101, no. 3, pp. 652–675, Mar. 2013.
- [2] P. Ghamisi, J. Plaza, Y. Chen, J. Li, and A. J. Plaza, “Advanced spectral classifiers for hyperspectral images: A review,” *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 1, pp. 8–32, Mar. 2017.
- [3] P. Ghamisi *et al.*, “Advances in hyperspectral image and signal processing: A comprehensive overview of the state of the art,” *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 37–78, Dec. 2017.
- [4] N. Audebert, B. Le Saux, and S. Lefevre, “Deep learning for classification of hyperspectral data: A comparative review,” *IEEE Geosci. Remote Sens. Mag.*, vol. 7, no. 2, pp. 159–173, Jun. 2019.
- [5] W. Sun *et al.*, “A multiscale spectral features graph fusion method for hyperspectral band selection,” *IEEE Trans. Geosci. Remote Sens.*, to be published, doi: [10.1109/TGRS.2021.3102246](https://doi.org/10.1109/TGRS.2021.3102246).
- [6] W. Sun, K. Ren, X. Meng, C. Xiao, G. Yang, and J. Peng, “A band divide-and-conquer multispectral and hyperspectral image fusion method,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–13, 2022, Art. no. 5502113, doi: [10.1109/TGRS.2020.3046321](https://doi.org/10.1109/TGRS.2020.3046321).
- [7] L. He, J. Li, C. Liu, and S. Li, “Recent advances on spectral–spatial hyperspectral image classification: An overview and new guidelines,” *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 3, pp. 1579–1597, Mar. 2018.
- [8] J. Peng *et al.*, “Low-rank and sparse representation for hyperspectral image processing: A review,” *IEEE Geosci. Remote Sens. Mag.*, to be published, doi: [10.1109/MGRS.2021.3075491](https://doi.org/10.1109/MGRS.2021.3075491).
- [9] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, and J. A. Benediktsson, “Deep learning for hyperspectral image classification: An overview,” *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6690–6709, Sep. 2019.
- [10] M. Paoletti, J. Haut, J. Plaza, and A. Plaza, “Deep learning classifiers for hyperspectral imaging: A review,” *ISPRS J. Photogrammetry Remote Sens.*, vol. 158, pp. 279–317, 2019.
- [11] J. Li, B. Xi, Y. Li, Q. Du, and K. Wang, “Hyperspectral classification based on texture feature enhancement and deep belief networks,” *Remote Sens.*, vol. 10, no. 3, p. 396, 2018.
- [12] Y. Li, H. Zhang, and Q. Shen, “Spectral–spatial classification of hyperspectral imagery with 3D convolutional neural network,” *Remote Sens.*, vol. 9, no. 1, p. 67, 2017.
- [13] Z. Zhong, J. Li, Z. Luo, and M. Chapman, “Spectral–spatial residual network for hyperspectral image classification: A 3-D deep learning framework,” *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 847–858, Feb. 2018.
- [14] W. Wang, S. Dou, Z. Jiang, and L. Sun, “A fast dense spectral–spatial convolution network framework for hyperspectral images classification,” *Remote Sens.*, vol. 10, no. 7, 2018, Art. no. 1068.
- [15] M. Zhang, W. Li, and Q. Du, “Diverse region-based CNN for hyperspectral image classification,” *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2623–2634, Jun. 2018.
- [16] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7132–7141.
- [17] B. Xi *et al.*, “Multi-direction networks with attentional spectral prior for hyperspectral image classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–15, Jan. 2021, Art. no. 5500915.
- [18] Y. Shi, J. Li, Y. Zheng, B. Xi, and Y. Li, “Hyperspectral target detection with RoI feature transformation and multiscale spectral attention,” *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 6, pp. 5071–5084, Jun. 2021.
- [19] J. M. Haut, M. E. Paoletti, J. Plaza, J. Li, and A. Plaza, “Active learning with convolutional neural networks for hyperspectral image classification using a new Bayesian approach,” *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 11, pp. 6440–6461, Nov. 2018.
- [20] X. He, Y. Chen, and P. Ghamisi, “Heterogeneous transfer learning for hyperspectral image classification based on convolutional neural network,” *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3246–3263, May 2020.
- [21] J. Snell, K. Swersky, and R. S. Zemel, “Prototypical networks for few-shot learning,” 2017, *arXiv:1703.05175*.
- [22] B. Liu, X. Yu, A. Yu, P. Zhang, G. Wan, and R. Wang, “Deep few-shot learning for hyperspectral image classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 2290–2304, Apr. 2019.
- [23] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” in *Proc. Int. Conf. Learn. Representations*, 2016, *arXiv: 1609.02907*.
- [24] A. Qin, Z. Shang, J. Tian, Y. Wang, T. Zhang, and Y. Y. Tang, “Spectral–spatial graph convolutional networks for semisupervised hyperspectral image classification,” *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 2, pp. 241–245, Feb. 2019.
- [25] L. Mou, X. Lu, X. Li, and X. X. Zhu, “Nonlocal graph convolutional networks for hyperspectral image classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 12, pp. 8246–8257, Dec. 2020.
- [26] W. Lin, Z. Gao, and B. Li, “Shoestring: Graph-based semi-supervised classification with severely limited labeled data,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 4174–4182.
- [27] Q. Liu, L. Xiao, J. Yang, and Z. Wei, “CNN-enhanced graph convolutional network with pixel- and superpixel-level feature fusion for hyperspectral image classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 10, pp. 8657–8671, Oct. 2021.
- [28] S. Wan, C. Gong, P. Zhong, B. Du, L. Zhang, and J. Yang, “Multiscale dynamic graph convolutional network for hyperspectral image classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3162–3177, May 2020.
- [29] F. Yang, Q. Sun, H. Jin, and Z. Zhou, “Superpixel segmentation with fully convolutional networks,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 13964–13973.
- [30] Y. Li, T. Lu, and S. Li, “Subpixel–pixel–superpixel-based multiview active learning for hyperspectral images classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 7, pp. 4976–4988, Jul. 2020.
- [31] S. Jia, X. Deng, M. Xu, J. Zhou, and X. Jia, “Superpixel-level weighted label propagation for hyperspectral image classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 7, pp. 5077–5091, Jul. 2020.
- [32] M.-Y. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa, “Entropy rate superpixel segmentation,” in *Proc. CVPR/IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 2097–2104.
- [33] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip, “A comprehensive survey on graph neural networks,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 4–24, Jan. 2021.
- [34] Y. Huang *et al.*, “An efficient graph-based algorithm for time-varying narrowband interference suppression on SAR system,” *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 10, pp. 8418–8432, Oct. 2021.

²[Online]. Available: <http://dase.grss-ieee.org/>

- [35] X. Cao, Y. Ge, R. Li, J. Zhao, and L. Jiao, "Hyperspectral imagery classification with deep metric learning," *Neurocomputing*, vol. 356, pp. 217–227, 2019.
- [36] B. Liu, X. Yu, P. Zhang, A. Yu, Q. Fu, and X. Wei, "Supervised deep feature extraction for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 1909–1921, Apr. 2018.
- [37] B. Deng, S. Jia, and D. Shi, "Deep metric learning-based feature embedding for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 2, pp. 1422–1435, Feb. 2019.
- [38] K. Allen, E. Shelhamer, H. Shin, and J. Tenenbaum, "Infinite mixture prototypes for few-shot learning," in *Int. Conf. Mach. Learn.*, 2019, pp. 232–241.
- [39] B. Xi, J. Li, Y. Li, R. Song, W. Sun, and Q. Du, "Multiscale context-aware ensemble deep KELM for efficient hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 6, pp. 5114–5130, Jun. 2021.
- [40] S. Jia, X. Deng, J. Zhu, M. Xu, J. Zhou, and X. Jia, "Collaborative representation-based multiscale superpixel fusion for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7770–7784, Oct. 2019.
- [41] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. ICML/IEEE Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
- [42] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif. Intell. Statist. JMLR Workshop Conf.*, 2011, pp. 315–323.
- [43] L. He, Z. Wang, Y. Li, and S. Wang, "Softmax dissection: Towards understanding intra- and inter-class objective for embedding learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 7, 2020, pp. 10 957–10 964.
- [44] B. Xi, J. Li, Y. Li, and Q. Du, "Semi-supervised graph prototypical networks for hyperspectral image classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2021, pp. 2851–2854.
- [45] B. Xi *et al.*, "Deep prototypical networks with hybrid residual attention for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 3683–3700, 2020.
- [46] D. Hong, L. Gao, J. Yao, B. Zhang, A. Plaza, and J. Chanussot, "Graph convolutional networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 7, pp. 5966–5978, Jul. 2021.
- [47] J. Liang, J. Zhou, Y. Qian, L. Wen, X. Bai, and Y. Gao, "On the sampling strategy for evaluation of spectral-spatial methods in hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 862–880, Feb. 2017.
- [48] C. Shi and C.-M. Pun, "Superpixel-based 3D deep neural networks for hyperspectral image classification," *Pattern Recognit.*, vol. 74, pp. 600–616, 2018.
- [49] S. Zeng, Z. Wang, C. Gao, Z. Kang, and D. Feng, "Hyperspectral image classification with global-local discriminant analysis and spatial-spectral context," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 12, pp. 5005–5018, Dec. 2018.



Yunsong Li (Member, IEEE) received the M.S. degree in telecommunication and information systems and the Ph.D. degree in signal and information processing from Xidian University, Xi'an, China, in 1999 and 2002, respectively.

Since 1999, he has been with the School of Telecommunications Engineering, Xidian University, where he is currently a Professor. He is also the Director of the State Key Laboratory of Integrated Service Networks, Image Coding and Processing Center. His research interests include image and video processing,

hyperspectral image (HSI) processing, and high-performance computing.



Bobo Xi (Graduate Student Member, IEEE) received the B.S. degree in information engineering, in 2017, from the School of Telecommunications Engineering, Xidian University, Xi'an, China, where he is currently working toward the Ph.D. degree with the State Key Laboratory of Integrated Service Networks.

His research interests include hyperspectral image (HSI) processing, machine learning, and deep learning.



Jiaojiao Li (Member, IEEE) received the B.E. degree in computer science and technology, the M.S. degree in software engineering, and the Ph.D. degree in communication and information systems from Xidian University, Xi'an, China, in 2009, 2012, and 2016, respectively.

She was an exchange Ph.D. Student with Mississippi State University, supervised by Dr. Qian Du. She is currently an Associate Professor with the State Key Laboratory of Integrated Service Networks, School of Telecommunications, Xidian University.

Her research interests include hyperspectral remote sensing image analysis and processing, pattern recognition, and data compression.



Rui Song (Member, IEEE) received the M.S. and Ph.D. degrees in signal and information processing from Xidian University, Xi'an, China, in 2006 and 2009, respectively.

He is currently a Professor with the State Key Laboratory of Integrated Service Networks, School of Telecommunications, Xidian University. His research interests include video coding algorithms, VLSI architecture design, and 3-D reconstruction.



Yuchao Xiao received the B.E. degree in communication engineering from the School of Telecommunications Engineering, Xidian University, Xi'an, China, in 2021. He is currently working toward the M.D. degree in information and communication engineering with the School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing, China.

Since 2019, he has been a Research Assistant with the State Key Laboratory of Integrated Service Network, Image Coding and Processing Center, Xidian University. His research interests include hyperspectral image (HSI) processing, target identification, and deep learning.



Jocelyn Chanussot (Fellow, IEEE) received the M.Sc. degree in electrical engineering from the Grenoble Institute of Technology (Grenoble INP), Grenoble, France, in 1995, and the Ph.D. degree in electrical engineering from the Université de Savoie, Annecy, France, in 1998.

Since 1999, he has been with Grenoble INP, where he is currently a Professor of Signal and Image Processing. He was a Visiting Scholar with Stanford University, Stanford, CA, USA, KTH Royal Institute of Technology, Stockholm, Sweden, and National University of Singapore, Singapore. Since 2013, he has been an Adjunct Professor with the University of Iceland, Reykjavik, Iceland. From 2015 to 2017, he was a Visiting Professor with the University of California, Los Angeles (UCLA), Los Angeles, CA, USA. His research interests include image analysis, hyperspectral remote sensing, data fusion, machine learning, and artificial intelligence.

Dr. Chanussot holds the AXA Chair in remote sensing and is an Adjunct Professor with the Chinese Academy of Sciences, Aerospace Information Research Institute, Beijing, China. He is the founding President of IEEE Geoscience and Remote Sensing French chapter (2007–2010), which received the 2010 IEEE GRSS Chapter Excellence Award. He has received multiple outstanding paper awards. He was the Vice-President of the IEEE Geoscience and Remote Sensing Society, in charge of meetings and symposia (2017–2019). He was the General Chair of the first IEEE GRSS Workshop on Hyperspectral Image and Signal Processing, Evolution in Remote Sensing (WHISPERS). He was the Chair during 2009–2011 and Co-Chair of the GRS Data Fusion Technical Committee during 2005–2008. He was a member of the Machine Learning for Signal Processing Technical Committee of the IEEE Signal Processing Society from 2006 to 2008 and the Program Chair of the IEEE International Workshop on Machine Learning for Signal Processing in 2009. He is an Associate Editor for IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, the IEEE TRANSACTIONS ON IMAGE PROCESSING, and *Proceedings of the IEEE*. He was the Editor-in-Chief of *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* from 2011 to 2015. In 2014, he was a Guest Editor for the *IEEE Signal Processing Magazine*. He is a member of the Institut Universitaire de France from 2012 to 2017 and a Highly Cited Researcher (Clarivate Analytics/Thomson Reuters, 2018–2019).