Fractional Gabor Convolutional Network for Multisource Remote Sensing Data Classification

Xudong Zhao, *Student Member, IEEE*, Ran Tao, *Senior Member, IEEE*, Wei Li, *Senior Member, IEEE*, Wilfried Philips, *Senior Member, IEEE*, and Wenzhi Liao, *Senior Member, IEEE*

***Abstract*— Remote sensing using multisensor platforms has been systematically applied for monitoring and optimizing human activities. Several advanced techniques have been developed to enhance and extract the spatially and spectrally semantic information in the hyperspectral image (HSI) and light detection and ranging (LiDAR) data processing and analysis. However, an abundance of redundant information and sometimes a lack of discriminative features reduce the efficiency and effectiveness of multisource classification methods. This article proposes a fractional Gabor convolutional network (FGCN), focusing on efficient feature fusion and comprehensive feature extraction. First, the proposed FGCN uses Octave convolution layers to perform multisource information fusion and preserve discriminative information. Second, fractional Gabor convolutional (FGC) layers are proposed to extract multiscale, multidirectional, and semantic change features. The completeness and discrimination of the multisource features using different FGC kernels are improved, which yield robust feature extraction against semantic changes. Finally, the fractional Gabor feature and spectral feature are combined with two weighting factors which can be learned during the network training. Experimental results and comparisons with state-of-the-art multisource classification methods indicate the effectiveness of the proposed FGCN. With the FGCN, we can obtain an 89.90% overall accuracy on the challenging Muufl Gulfport (MUUFL) data set, with an improvement of 3% over state-of-the-art methods.**

***Index Terms*— Fractional Gabor convolutional network (FGCN), hyperspectral image (HSI), light detection and ranging (LiDAR), multisensor data fusion.**

Manuscript received September 1, 2020; revised January 18, 2021; accepted March 8, 2021. Date of publication March 23, 2021; date of current version December 13, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61922013, and in part by the Beijing Natural Science Foundation under Grant JQ20021. The work of Xudong Zhao was supported by the China Scholarship Council under Grant 201906030007. *(Corresponding author: Ran Tao.)*

Xudong Zhao is with the School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China and also with the Image Processing and Interpretation, IMEC Research Group, Ghent University, 9000 Ghent, Belgium (e-mail: zhaoxudong@bit.edu.cn).

Ran Tao and Wei Li are with the School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China and also with the Beijing Key Laboratory of Fractional Signals and Systems, Beijing 100081, China (e-mail: rantao@bit.edu.cn; liwei089@ieee.org).

Wilfried Philips is with Image Processing and Interpretation, IMEC Research Group, Ghent University, 9000 Ghent, Belgium (e-mail: wilfried.philips@ugent.be).

Wenzhi Liao is with Sustainable Materials Management, Flemish Institute for Technological Research (VITO), 2400 Mol, Belgium and also with the Department of TELIN, Ghent University, 9000 Ghent, Belgium (e-mail: wenzhi.liao@vito.be).

Digital Object Identifier 10.1109/TGRS.2021.3065507

I. Introduction

With the rapid development of Earth observation techniques, there have been ever-increasing amounts of multimodal data acquired from different platforms, such as airplanes, satellites, autonomous vehicles, and unmanned aerial vehicles (UAVs), for different remote sensing applications [1], [2]. Remote sensing using multisensor platforms has been systematically applied for monitoring land-use and land-cover classification, and environmental changes, such as rampant urban sprawling and land degradation [3]–[5]. Many remote sensing approaches have been proposed for land cover classification, but most consider only one modality, e.g., RGB images, hyperspectral images (HSI), light detection and ranging (LiDAR), or infrared images. This is in part due to the differences in structure among the modalities that complicate their joint analysis. Specifically, HSI can provide detailed spectral information for potential material identification [6] while LiDAR data provide elevation information about the area under investigation at any time of the day and under adverse weather conditions [7]–[9].

Recently, new techniques have been developed for joint classification of HSI and LiDAR from information modeling to data fusion [10]–[13]. For instance, Rasti *et al.* [14] proposed a technique for the fusion of hyperspectral and LiDAR data called Sparse and Low-Rank Component Analysis (SLRCA) fusion. Kang *et al.* [15] first proposed an extended random walker-based effective probability optimization method for classification of HSIs, which achieved promising performance. Then, a Hierarchical Random Walk Network (HRWN) was developed to fuse deep features extracted from HSI and LiDAR for precise land-cover classification [16]. In [17], a Semisupervised Graph Fusion (SSGF) was used to project the spectral, elevation, and spatial features onto a lower subspace to obtain the new features of LiDAR and HSI. In [18], an effective multiview edge-preserving filtering method was developed for material identification, which greatly improves classification performance. Nevertheless, how to extract joint features containing comprehensive and complemented information remains challenging.

One of the most significant tasks in multisource remote sensing is to model data from different sensors and fuse spatial, spectral, radiation, and shape information. Enormous efforts have been made to extract features of HSI and LiDAR sensors, develop data fusion techniques for reconstructing synthetic

data that have the advantages of different sensors [19]–[22]. To mine spatial information of HSI, morphological profiles have been applied to model multilevel features [23], [24]. In [23], Orthogonal Total Variation Component Analysis (OTVCA) was proposed to fuse extinction profiles modeling spatial and elevation information from HSI and rasterized LiDAR features. In [25], LiDAR data were used for scene segmentation, and then hyperspectral data were classified based on segmentation results, which showed a significant improvement over single-source data classification. Most of the aforementioned algorithms involve modalities with tradeoffs in spatial, spectral, and temporal resolutions. However, Liao *et al.* [26], [27] pointed out that simple feature splicing or stacking operations are highly susceptible to redundant information stacking. Khodadzadeh *et al.* [3] further pointed out that feature splicing increases the dimensionality of feature extraction and exacerbating the Hughes effect. The fusion of HSI and LiDAR information by coupling reducing the redundant details and preserving the discriminative geometrical features remains challenging.

With synthetic data that have the advantages of different sensors, another significant issue is feature extraction of the multisensor image to achieve robust and accurate joint classification of HSI and LiDAR data. Recent deep collaboration frameworks for multisource data have shown efficient performance in spatial-spectral feature extraction [16], [28], [29]. In [30], a Contextual Convolutional Neural Network (CCNN) was developed to extract deeper and wider spatial features for HSI classification. In [31], a Convolutional Recurrent Neural Network (CRNN) model was proposed to effectively analyze hyperspectral pixels as sequential data and then determine information categories via network reasoning. However, simple concatenation or stacking of features may be limited in individual feature extraction [3], [32].

To extract robust and high-level spatial-spectral features by deep networks, Convolutional Neural Networks (CNNs)-based methods were proposed to combine HSI and LiDAR data using multibranch architectures [33]–[35]. In [36], a CNN in combination with a Markov Random Field (CNNMRF) was proposed to classify pixel vectors in a way fully taking spatial and spectral information into account. In [33], a Two-Branch CNN framework (TBCNN) is developed to extract spectral-spatial features from HSI and LiDAR data sets. In [37], an end-to-end fusion module is proposed for efficient joint feature extraction and classification. To solve the problem caused by the unbalance between different features, decision-fusion methods for HSI and LiDAR classification have been presented [27], [38].

Although these features and decision-fusion-based approaches have shown excellent performance in local and global feature extraction, they still fail to accurately describe the orientation and semantic changes information [39]. Gabor operators guide the CNN to acquire spatial features that are multiscale and multidirectional [40], [41]. More significantly, the features only extracted from the spatial domain are relatively limited in representation ability and diversity, particularly for the complex scenes including various spectral variabilities. Therefore, feature extraction in the frequency domain [42], [43] or fractional domains [44]is beneficial to enrich the diversity of the features. In [44], the fractional Fourier transform (FrFT) was proven to be desirable for noise removal and can enhance the discrimination between anomalies and background.

This article focuses on efficient feature fusion and comprehensive and discriminative feature extraction of multisensor remote sensing data. It proposes a fractional Gabor convolutional network (FGCN). First, the proposed FGCN uses Octave convolutional layers to decompose multisource remote sensing data to the low-frequency and high-frequency components, and reduce the redundant low-frequency information. Then, corresponding frequency components of the two sources are fused preserving the discriminative features. Experimental results show improvements in discrimination. Second, fractional Gabor convolutional (FGC) layers are used to extract features at multiple scales, directions, and multiple chirp rates. The FGC layers add feature diversity and discrimination, improve the completeness of the multisource features. Finally, fractional Gabor feature and spectral feature are combined with two weighted factors which can be learned during the network training, and the classification map is obtained by softmax classifier. Experimental results of three multisource data sets are used to assess and compare the classification accuracies of state-of-the-art multisource classification methods, which indicate the effectiveness of the proposed FGCN.

The main contributions are highlighted as follows.

1) The Octave convolutional layers are used to fuse information from different data sources, which can reduce data redundancy and improve discrimination. Because of the frequency decompositions by Octave convolution, the volume of parameters and complexity can be reduced compared with the common convolutional layers.
2) The FGC layers are designed to improve the completeness and discrimination of the multisource features, which yield robust feature extraction. The FGCN architecture is proposed to integrate comprehensive multiscale spatial features, directional texture features, frequency variation features, and spectral features for accurate multisource joint classification.

The remainder of this article is organized as follows. The proposed framework is introduced in Section II. In Section III, experimental results and analysis are presented. Finally, Section IV summarizes with some concluding remarks.

II. Proposed Classification Framework

The FGCN framework is designed for pixel-level classification by fusing multisource remote sensing images. The framework of the proposed FGCN is illustrated in Fig. 1. It consists of three parts: 1) the multisource frequency decomposition and fusion by Octave convolutional layers (Part I); 2) the FGC layers extracting features at multiple scales, directions, and multiple chirp rates (Part II); and 3) the classifier using comprehensive features to obtain classification map (Part III).

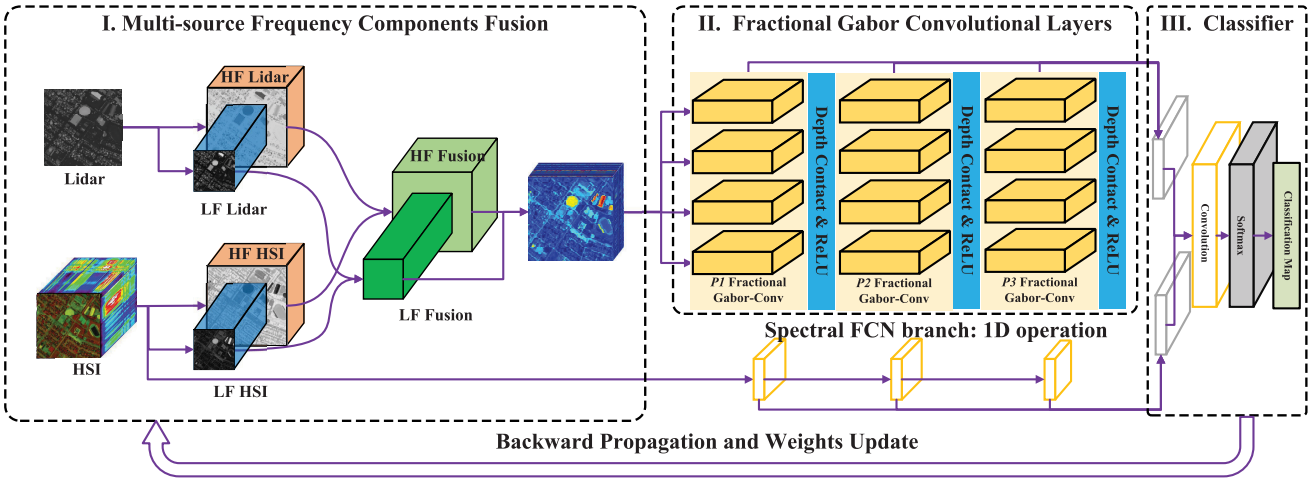

Fig. 1. Proposed FGCN classification framework.

A. Multisource Frequency Components Fusion

As illustrated in Fig. 1, Part I reflects multiple Octave convolutional layers. Octave convolution has been used for HSI classification with more efficient convolution layers [45], [46]. Simple feature splicing or stacking operations are highly susceptible to redundant information stacking [26], [27]. The fusion of HSI and LiDAR information using classical feature extraction and fusion methods reduce part of redundancy [23], [26], [47], but still have redundancy in low-frequency parts. Furthermore, the discrimination of objects in the data sets is not improved leading challenges in classification. To reduce redundancy in low-frequency components of data and improve discrimination of different objects, the Octave convolution layers are used in the proposed FGCN. The motivation of Octave layers including fusing multisource information and improving discrimination between classes.

We first use an Octave convolutional layer to decompose the input images into a multiresolution representation, which makes it easier to reduce spatial redundancy [48]. In this step, a HSI image is given as $\mathbf{X}_h \in \mathbb{R}^{r\times c\times b_h}$ with b_h bands by $r \times c$ pixels, and a LiDAR image covering the same area is denoted as $\mathbf{X}_l \in \mathbb{R}^{r\times c\times b_l}$ with b_l bands. We use OctConv to explicitly factorize $\mathbf{X}_h$ and $\mathbf{X}_l$ along the channel dimension into $\mathbf{X}_h = \{\mathbf{X}_h^H, \mathbf{X}_h^L\}$ and $\mathbf{X}_l = \{\mathbf{X}_l^H, \mathbf{X}_l^L\}$, where the high-frequency convolutional output features $\mathbf{X}_h^H \in \mathbb{R}^{r\times c\times(1-\alpha)b_h}$ and $\mathbf{X}_l^H \in \mathbb{R}^{r\times c\times(1-\alpha)b_l}$ capture fine details. The low-frequency convolutional output feature $\mathbf{X}_h^L \in \mathbb{R}^{(r/2)\times(c/2)\times\alpha b_h}$ and $\mathbf{X}_l^L \in \mathbb{R}^{(r/2)\times(c/2)\times\alpha b_l}$ vary slower in the spatial dimensions, which indicates the spatial redundancy in low-frequency components. $\alpha \in [0, 1]$ denotes the ratio of channels allocated to the low-frequency part and the spatial resolution of the low-frequency feature maps is reduced by an octave (a division of the spatial dimensions by 2^1).

The Octave feature representation of HSI and LiDAR data reduces the spatial redundancy and is more compact than the normal convolution [48]. Three Octave convolutional layers are designed to decompose the input HSI and LiDAR data into components $\mathbf{X}_h = \{\mathbf{X}_h^H, \mathbf{X}_h^L\}$ and $\mathbf{X}_l = \{\mathbf{X}_l^H, \mathbf{X}_l^L\}$, reducing the redundancy in low-frequency components and fuse the frequency features as compact and discriminative outputs. The goal of the three-layer Octave convolution design is to effectively process the low and high frequencies in their corresponding frequency components and also enable efficient fusion or interfrequency communication of multiple sources. As Fig. 2 shown, the input and output frequency components of Octave convolution layer are denoted by $\mathbf{X}_h = \{\mathbf{X}_h^H, \mathbf{X}_h^L\}$, $\mathbf{X}_l = \{\mathbf{X}_l^H, \mathbf{X}_l^L\}$,$\mathbf{Y}_h = \{\mathbf{Y}_h^H, \mathbf{Y}_h^L\}$, and $\mathbf{Y}_l = \{\mathbf{Y}_l^H, \mathbf{Y}_l^L\}$. Specifically, taking the output of a HSI $\mathbf{Y}_h = \{\mathbf{Y}_h^H, \mathbf{Y}_h^L\}$ as an example

$$\begin{aligned}\mathbf{Y}_h^H &= \mathbf{Y}_h^{H\to H} + \mathbf{Y}_h^{L\to H}\\ \mathbf{Y}_h^L &= \mathbf{Y}_h^{L\to L} + \mathbf{Y}_h^{H\to L}\end{aligned} \tag{1}$$

where $\mathbf{Y}^{A\to B}$ denotes the convolutional update from feature map group A to group B. To compute these outputs, the convolutional kernel $\mathbf{W}$ is separated into two components $\mathbf{W} = [\mathbf{W}^H, \mathbf{W}^L]$, which response for $\mathbf{X}_h^H$ and $\mathbf{X}_h^L$, respectively. Specifically, $\mathbf{W}_h^H$ and $\mathbf{W}_h^L$ can be further divided into intrafrequency and interfrequency parts $\mathbf{W}_h^L = [\mathbf{W}_h^{L\to L}, \mathbf{W}_h^{H\to L}]$, $\mathbf{W}_h^H = [\mathbf{W}_h^{H\to H}, \mathbf{W}_h^{L\to H}]$. Then the output low-frequency feature map at location (x, y) can be computed as

$$\begin{aligned}\mathbf{Y}_{x,y}^L &= \mathbf{Y}_{x,y}^{L\to L} + \mathbf{Y}_{x,y}^{H\to L}\\ &= \sum_{i,j\in\mathcal{N}_k} \mathbf{W}_{i+\frac{k-1}{2},j+\frac{k-1}{2}}^{L\to L\ T} \mathbf{X}_{x+i,y+j}^L\\ &\quad + \sum_{i,j\in\mathcal{N}_k} \mathbf{W}_{i+\frac{k-1}{2},j+\frac{k-1}{2}}^{H\to L\ T} \mathbf{X}_{(2*x+0.5+i,2*y+0.5+j)}^H\end{aligned} \tag{2}$$

where k is the convolution kernel size, $\mathcal{N}_k = \{(i, j) : i = \{-((k-1)/2), \ldots, ((k-1)/2)\}, j = \{-((k-1)/2), \ldots, ((k-1)/2)\}\}$ defines a local convolution neighborhood. Similarly, the high-frequency feature map can be computed as

$$\begin{aligned}\mathbf{Y}_{x,y}^H &= \mathbf{Y}_{x,y}^{H\to H} + \mathbf{Y}_{x,y}^{L\to H}\\ &= \sum_{i,j\in\mathcal{N}_k} \mathbf{W}_{i+\frac{k-1}{2},j+\frac{k-1}{2}}^{H\to H\ T} \mathbf{X}_{x+i,y+j}^H\\ &\quad + \sum_{i,j\in\mathcal{N}_k} \mathbf{W}_{i+\frac{k-1}{2},j+\frac{k-1}{2}}^{L\to H\ T} \mathbf{X}_{(\lfloor\frac{x}{2}\rfloor+i,\lfloor\frac{y}{2}\rfloor+j)}^L\end{aligned} \tag{3}$$

where $\lfloor\cdot\rfloor$ denotes the floor operation. As shown in Fig. 2, the first layer factorizes input HSI and LiDAR data with $\alpha_{\text{in}} = 0, \alpha_{\text{out}} = \alpha$, outputs four parts of HSI and LiDAR data $\mathbf{Y}_h^H, \mathbf{Y}_h^L, \mathbf{Y}_l^H$ and $\mathbf{Y}_l^L$. The middle layer cascades $\mathbf{Y}_h^H, \mathbf{Y}_l^H$ and $\mathbf{Y}_h^L, \mathbf{Y}_l^L$ and outputs two fused multifrequency components $\mathbf{Y}_f^H, \mathbf{Y}_f^L$. The last Octave convolution layer synthesizes $\mathbf{Y}_f^H, \mathbf{Y}_f^L$ and outputs the fused multisource feature $\mathbf{Y}$ with $\alpha_{\text{in}} = \alpha, \alpha_{\text{out}} = 0$. The fused $\mathbf{Y}$ combines detailed spectral signatures of HSI and elevation information of LiDAR, reducing the spatial redundancy in low-frequency components, while utilizing the discriminative information. In the following subsection, the improved discrimination of different land-covers is shown to prove the effectiveness of Octave convolutional layers.

B. FGC Layers

The Octave convolution layers utilize both spatial-spectral information of HSI and elevation information of LiDAR. Furthermore, to extract the textural and semantic change features, the FGC layers are employed with their learning objective as Part II in Fig. 1.

For remote sensing scenes, not all the objects in a specific class have same shape without any change in orientations. In these scenes, CNN lack the ability for describing the directional information and geometric change, and the fixed convolution kernel structure in convolutional layers resulting in a single feature scale. Gabor filters can guide CNN to obtain multiscale and multidirectional spatial features, thereby combining Gabor filters with CNNs can improve the performance of CNN models. The Gabor wavelet can exhibits different scales and directions under different wavelength parameters [40], which can be seen as a product of a Gaussian function and a sinusoidal plane wave

$$\begin{aligned} g_{f,\theta}(x, y) &= \frac{1}{2\pi}\exp(-(\alpha^2 x'^2 + \beta^2 y'^2))\exp(j2\pi\omega x') \\ x' &= \left(x - \frac{m+1}{2}\right)\cos\theta + \left(y - \frac{n+1}{2}\right)\sin\theta \\ y' &= \left(x - \frac{m+1}{2}\right)\sin\theta + \left(y - \frac{n+1}{2}\right)\cos\theta \end{aligned} \tag{4}$$

where m, n denote the size of Gabor filters, θ is the rotation of Gaussian function, α and β means the sharpness of Gaussian function. The Gabor wavelet decomposes the data according to both spatial location and frequency content. However, complex scenes contain semantic changes with different frequency change rates. The Gabor kernel can extract semantic changes at different directions and scales, while fractional Fourier kernel can extend this ability to semantic changes with different frequency change rates [49]. Features between spatial and frequency domain (fractional domain) contain significant local semantic change information. The FrFT analyzing p order domain features is defined as

$$X_p(u) = \int_{-\infty}^{+\infty} f(x)K_p(u, x)dx \tag{5}$$

where p is the transform order. When $p = (\pi/2)$, the fractional transform is the Fourier transform. $K_p(u, x)$ is the kernel of transform, which is defined as

$$K_p(x, u) = \begin{cases} \sqrt{\dfrac{1 - j\cot p}{2\pi}}\exp\left(j\dfrac{x^2 + u^2}{2}\cot p - \dfrac{jxu}{\sin p}\right) \\ \qquad\qquad p \neq n\pi \\ \delta(x - u), \quad p = 2n\pi \\ \delta(x + u), \quad p = (2n \pm 1)\pi. \end{cases} \tag{6}$$

As shown in Fig. 3, the fractional Gabor transform (FGT) is used to modulate the CNN convolutional kernel, which is designed to extract multiscale and multidirection features with multiple chirp rates. The FGT is

$$\begin{aligned} &\text{FGT}_{px,py}(x, y, u, v) \\ &\quad = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x, y)K_{px,py}(x, y, u, v)h(x', y')dxdy \end{aligned} \tag{7}$$

where $f(x, y)$ is the image, x, y is spatial location u, v is the location in fractional domain. $K_{px,py}(x, y, u, v)$ is the 2-D fractional kernel with px, py being the horizontal and the vertical fractional transform orders, $h(x', y')$ is the Gaussian function with rotation θ in (4). For 2-D image, the discrete FGT is

$$\begin{aligned} &\text{FGT}_{px,py}\left(x, y, \frac{u}{UT_1}, \frac{v}{VT_2}\right) \\ &\quad = \sum_{x=0}^{M-1}\sum_{u=0}^{U-1}\left[\sum_{n=0}^{N-1}\sum_{v=0}^{V-1} f(x, y)K_{py}\left(y, \frac{v}{VT_2}\right)h(y')\right] \\ &\qquad \cdots K_{px}\left(x, \frac{u}{UT_1}\right)h(x') \end{aligned} \tag{8}$$

where U, V are numbers of samples in the fractional domain, T_1, T_2 are the sampling interval, M, N are the size of the input image.

To interpret complex scenes containing semantic changes, the FGT matrices are used to modulate the CNN kernels. The modulated FGC kernels have different receptive fields and characteristics of directionality in each convolutional layer. As shown in Fig. 3, there are three FGC layers and each layer has four branches. The FGC kernels are defined by modulating the classical CNN kernel with the fractional Gabor filters as

$$\text{FGC}_{px,py,i,o} = C_{i,o} * \text{FGT}_{px,py,u,v,\theta_o} \tag{9}$$

where $\text{FGC}_{px,py,i,o}$ denotes the ith modulated FGC kernel of the oth output channel, $C_{i,o}$ is the original convolution kernel, $\text{FGT}_{px,py,u,v,\theta_o}$ denotes the FGT matrix with fractional frequency u, v and direction θ_o in px, py fractional domain. For each channel of CNN kernels, 2-D FGT matrix is used to modulate the weight matrix. Then the fractional convolution layer is

$$\mathbf{Y}_{px,py,o} = \sum_{i}^{I} \mathbf{Y}_i * \text{FGC}_{px,py,i,o} \tag{10}$$

where $\mathbf{Y}_{px,py,o}$ denotes the oth feature of the convolution layer, I is the bands of input. In our method, the horizontal order px and the vertical order py are set to p_l in three fractional

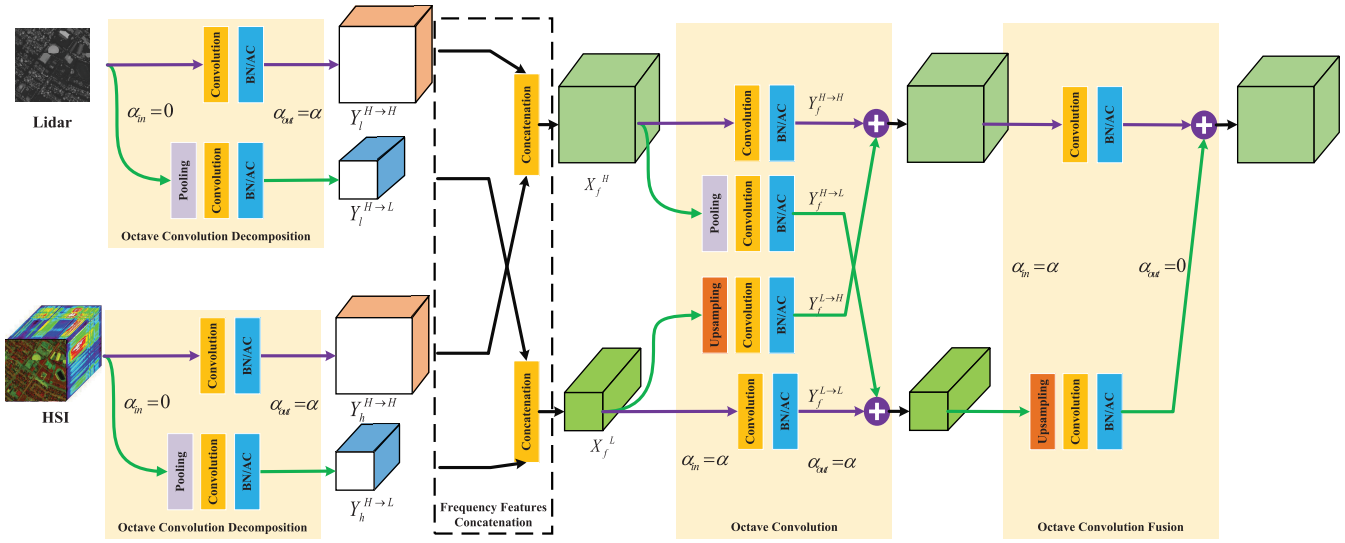

Fig. 2. Details of the designed multisource frequency components fusion (Part I in Fig. 1). Green lines denote interfrequency Octave convolution, purple lines for intrafrequency and black lines for cross-source components concatenation.

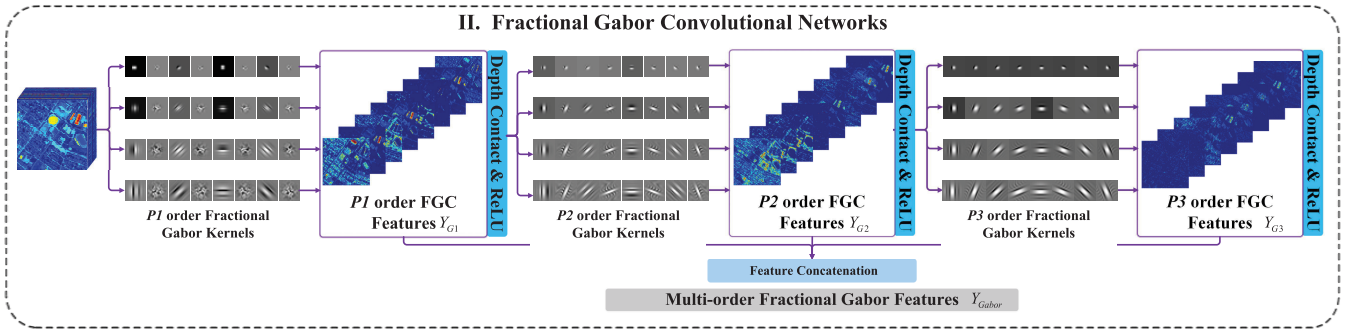

Fig. 3. Details of FGC layers (Part II in Fig. 1). Parts of the directional filters and feature maps are shown for illustration.

layers, where l is the number of convolution layer. For each branch in a single layer, the fractional frequencies are the same in different directions. As shown in Fig. 3, the three output FGC features are concatenated as the output $\mathbf{Y}_G$ and then the rectified linear units (ReLU) function $f(x) = \max(0, x)$ is utilized as activation function [50]. The input feature of the first FGC layer is $\mathbf{Y}$ computed in Section II-A, the output $\mathbf{Y}_{G1}$ is the input of the next layer. To this end, the FGC features $\mathbf{Y}_{G1}, \mathbf{Y}_{G2}, \mathbf{Y}_{G3}$, are concatenated to obtain the FGC features $\mathbf{Y}_{\text{Gabor}}$.

C. Classifier of FGCN Framework

CNNs have shown great potential in spectral feature extraction of HSI data. In the proposed FGCN framework, the fully convolutional layers are used to learn spectral features from original HSI data. The 1×1 convolutional kernels are used to ensure that the spectral features are extracted and the size of the feature maps keeps consistent with the original image. The convolutional layer is

$$\mathbf{Y}^i = f\left(\sum_j (\mathbf{W}^i \mathbf{X}^j) + \mathbf{B}^i\right) \tag{11}$$

where $\mathbf{Y}^i$ is the feature obtained by ith channel, $\mathbf{W}^i$ is the convolutional kernel, $\mathbf{X}^j$ is the jth channel of the previous layer, $\mathbf{B}^i$ is the corresponding bias term. Then ReLU function [50] is utilized as the activation function. To reduce the information loss during the convolutional layer getting deeper, the first three layers are concatenated to obtain the spectral feature $\mathbf{Y}_{\text{spec}}$.

Up to now, we have the spectral feature $\mathbf{Y}_{\text{spec}}$ of HSI images and the FGC feature $\mathbf{Y}_{\text{Gabor}}$ from the fused feature by Octave convolution. These comprehensive features are combined by a weighted addition layer as

$$\mathbf{Y}_{\text{joint}} = \lambda_{\text{Gabor}} \mathbf{Y}_{\text{Gabor}} + \lambda_{\text{Spec}} \mathbf{Y}_{\text{Spec}} \tag{12}$$

where λ_{Gabor} and λ_{Spec} are the weight parameters. The final joint feature $\mathbf{Y}_{\text{joint}}$ is fed into two convolutional layers and a softmax classify layer to predict the probability distribution

$$P(y(u, v) = k) = \frac{e^{\mathbf{Y}^k_{\text{joint}}(u,v)}}{\sum_{k=1}^{N} \mathbf{Y}^k_{\text{joint}}(u, v)} \tag{13}$$

where $y(u, v)$ is the pixel label, N is the number of class. Then the final labeling result $\mathbf{R}$ can be obtained. These algorithm steps are summarized in **Algorithm 1**. As the overall framework shown in Fig. 4, the FGCN is consist

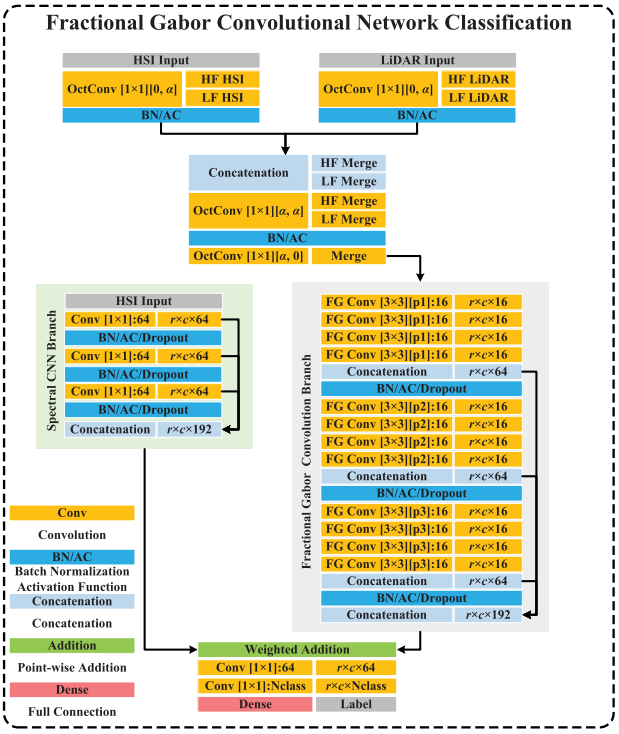

Fig. 4. Overall parameter configuration of the designed FGCN.

Algorithm 1 FGCN

Require: HSI data $\mathbf{X}_h$, LiDAR data $\mathbf{X}_l$, training samples $\mathbf{S}$, training epochs *epochs*.

Ensure: Classification map $\mathbf{R}$.

1: Initialize all weights and bias terms
2: **for** *epoch* < *epochs* **do**
3: Extract multifrequency components of multisource data as (1) and fuse features as $\mathbf{Y}$.
4: Extract multiorder fractional Gabor features $\mathbf{Y}_{Gabor}$ using fused feature $\mathbf{Y}$ and spectral feature $\mathbf{Y}_{spec}$ from $\mathbf{X}_h$.
5: Combined $\mathbf{Y}_{Gabor}$ and $\mathbf{Y}_{spec}$ as (12) with two weighting factors and obtain the classification map by softmax classifier as 13.
6: Train the FGCN as shown in Fig. 4 using training samples $\mathbf{S}$
7: **end for**
8: Obtain probability distribution by (13) and obtain classification map $\mathbf{R}$.

of three parts including 1) Octave convolutional layers for multisource frequency components decomposition and fusion; 2) the FGC layers for multidirection, semantic change features extraction; and 3) classifier using spectral feature and FGC feature.

D. Motivations and Effectiveness Analysis of the Proposed FGCN

The HSI can distinguish and detect ground targets with powerful diagnostic ability due to the high spectral resolution, narrow bandwidth, and a large amount of information.

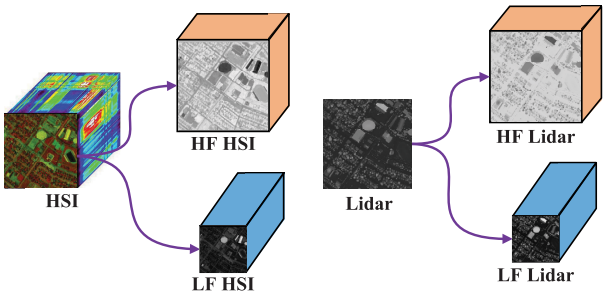

Fig. 5. Visualization of example HSI and LiDAR data decomposed by Octave convolutional layer.

However, its poor spatial contrast between objects limits its performance in spatial feature presentation. The LiDAR data possesses clear boundaries between objects with different elevations and homogeneous regions inside real objects with the same elevations. Different from semantic labeling of the single source remote sensing image, an effective and accurate joint classification algorithm usually depends on jointly modeling spatial, spectral, elevation, and textural information from different sensors. With this motivation, the proposed FGCN considers the spatially structural, textural, and elevation information in the form of semantic patches, and the detailed spectral signatures of land-covers. Taking the sampled HSI and LiDAR data sets as an example, the motivations and effectiveness of each step are analyzed as follows.

1) Octave Convolutional Layers: In Section II-A is used to fuse spatial, spectral and elevation information. Simple feature splicing or stacking operations are highly susceptible to redundant information stacking [26]. As shown in Fig. 5, the low- and high-frequency components of sampled HSI and LiDAR are obtained by Octave convolutional layer. Compared with the normal convolutional layers, the Octave convolutional layers separate the high- and low-frequency components of both HSI and LiDAR images (from the spatial aspect). While the redundancy in low-frequency components of the Octave convolutional output is reduced, the compression of low-frequency components can further reduce the volume of parameters and computational complexity compared with the normal convolutional networks. Although there is still redundancy in both high- and low-frequency components, the efficiency of convolution is improved.

To fuse HSI and LiDAR data effectively, classical methods like Gram–Schmidt Pan Sharpening (GS-merge) [47], Kernel Principal Component Analysis (KPCA) [27], and OTVCA [23] are proposed to model features from multiple sources. However, the fusion of HSI and LiDAR information still has lots of redundant details while the discrimination of data sets not improved, leads to challenges in classification. The t-distributed Stochastic Neighbor Embedding (t-SNE) is proposed for the visualization of similarity data, which is capable of retaining the local structure of the data and revealing global structure such as clusters at multiple scales [51]. In Fig. 6, the t-SNE feature visualization algorithm is used to visualize the original HSI data, fused features by GS-merge [47], Principle Component Analysis (PCA), KPCA, OTVCA, and

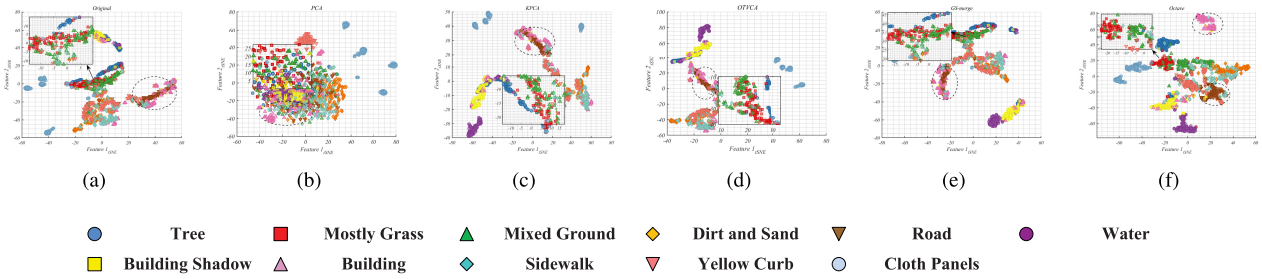

Fig. 6. Visualization of example data by competitive information fusion methods and the proposed Octave convolutional fusion strategy. (a) Original data. (b) PCA fusion. (c) Kernel PCA fusion. (d) OTVCA fusion. (e) G-S fusion. (f) Octave fusion.

the processed feature by Octave convolutional layers in the proposed FGCN. Among these multisource feature fusion methods, Octave operators can not only fuse information from multiple sources but also increase the discriminatory. As shown by the dashed circles in Fig. 6, the building and road categories with different elevations but the similar material are easily confused in the original HSI images. After information fusion of HSI and LiDAR by competitive methods, the objects are still difficult to distinguish. In contrast, the objects are more discriminative after Octave convolution layers. In the magnified area, categories of similar height and different materials, such as mostly grass and mixed ground, have better discriminatory after combining spectral information with elevation information using Octave convolution layers. These facts demonstrate that the Octave convolutional layers can fuse multisource information and increase discriminatory.

2) FGC Layers: FGC layers are proposed in Section II-B to extract multiscale, multidirectional features, and semantic change features. For the difference of objects in remote sensing scenes, not all the objects in a specific class have the same shape without any change in orientations. The traditional 2-D convolutional operation can extract spatial information from images. But with fixed receptive fields, the extracted features are single scale and lack directional descriptions. The Gabor kernels at different frequencies and different directions are combined with CNNs and enable CNNs to obtain the features described by the multiscale and multidirection [40], [41], [52].

The combination of Gabor operators and CNNs can effectively improve the classification performance, but it is still hard to deal with the locally semantic change, such as scene composition, the relative position between objects, atmospheric effects, and material mixture. As shown in Fig. 7, the fractional Gabor kernel can extract semantic changes feature with multiscale information with different chirp rates. As the fractional-order varies, the energy of the image signal is concentrated on different scales. In our previous research, the FrFT is proven able to changes the spectral distribution of amplitude and increase the discrimination between the objects and background, and reduce the noisy components [44]. Through the proposed FGCN architecture, comprehensive features are combined for more accurate multisource joint classification.

III. Experiments and Analysis

In this section, three HSI and LiDAR data sets are used to validate the effectiveness of the proposed FGCN.

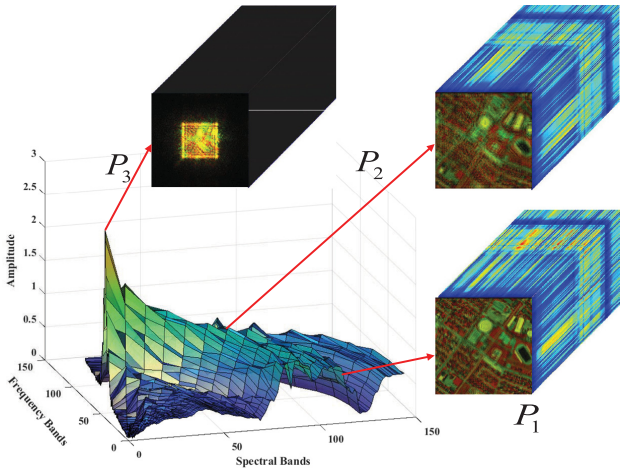

Fig. 7. Feature visualization of example data obtained with different orders of fractional transforms. An example spectral curve is used for illustration.

All the programs are implemented using Python 3.6. The networks are constructed using Tensorflow, which is an open-source software library for numerical computation using data flow graphs.[1] All experiments are conducted by a personal computer equipped with Ubuntu18.04 and NVIDIA GeForce RTX 2080 Ti. Three commonly used evaluation metrics, Overall Accuracy (OA), Average Accuracy (AA), and Kappa Coefficient (Kappa), are adopted to intuitively quantify the experimental results.

A. Data Description

1) MUUFL Data Set: Was acquired in November 2010 over the University of Southern Mississippi Gulfport Campus, Long Beach, Mississippi, USA [53]. The data set is composed of HSI and LiDAR-based digital surface model (DSM). HSI data are acquired by the ITRES Research Ltd. (ITRES) Compact Airborne Spectrographic Imager (CASI)-1500 sensor, which is composed of 325 × 220 pixels with 72 spectral channels (64 available bands) ranging from 375 nm to 1050 nm at a spectral sampling of 10 nm. The spatial resolution of HSI is 0.54 × 1.0 m, while that of LiDAR-based DSM is 0.60 × 0.78 m. LiDAR data are acquired by the gemini airborne laser terrain mapper (ALTM) LiDAR sensor. In this data set, 11 categories are investigated for the land cover classification task.

[1] http://tensorflow.org/

2) Houston Data Set: Was acquired by the National Science Foundation (NSF) Center in June 2012 over the University of Houston campus, Houston, Texas, USA [3]. The data set is composed of HSI and LiDAR-based DSM. HSI data are acquired by the ITRES CASI-1500 sensor, which is composed of 349×1905 pixels with 144 spectral channels ranging from 364 nm to 1046 nm at a spectral sampling of 10 nm. The spatial resolution of both HSI and LiDAR-based DSM is 2.5 m. In this data set, 15 categories are investigated for the land cover classification task.

3) Trento Data Set: Was acquired over a rural area in the south of the city of Trento, Italy [23]. The data set is composed of HSI and LiDAR-based DSM. HSI data is acquired by the AISA Eagle sensor, which is composed of 600×166 pixels with 63 spectral channels ranging from 0.40 μm to 0.98 μm. The spatial resolution of both HSI and LiDAR-based DSM is 1 m. LiDAR data are acquired by the Optech ALTM 3100EA sensor. In this data set, six categories are investigated for the land cover classification task.

B. Experimental Setup

1) Algorithm Configuration and Parameter Analysis: As a type of CNN, the proposed FGCN contains the basic parameter settings required [54]. According to the state-of-the-art CNN-based joint classification methods for HSI and LiDAR data, the basic parameters included in the training process are the size of convolution kernels $r \times r$ and learning rate lr [16], [29]–[31], [33], [36], [37]. In these competitive algorithms, these parameters are set by cross-validation on the available training set for an optimal and automatic system.

Apart from these basic parameters, there are two specific parameters in the proposed FGCN. One is the ratio of channels allocated to the low-frequency part α in Octave convolution, and the other is the fractional order p in fractional Gabor convolution layers. Similarly, these parameters can be set by cross-validation on the available training set in practical applications. Specifically, the optimal parameters for the three HSI and LiDAR data sets are set as $r \times r = 11 \times 11, lr = 1e-3, \alpha = 0.3, p = 0.25$.

Efficient feature extraction and joint classification largely depend on parameter selection and tuning. To validate the effectiveness and sensitivity of parameters involved in the proposed FGCN, the experimental analyses of different parameters are compared using the overall classification accuracy (OA). The quantitative results in Fig. 8 show the optimal parameter combinations for the three HSI and LiDAR data sets.

As shown in Fig. 8(a), the effect of different sizes of convolution kernels $r \times r$ are associated with classification results. The researched range of $r \times r$ is constrained from 1×1 to 17×17 with other parameters $lr = 1e-3, \alpha = 0.3, p = 0.25$. The experimental results indicate that features extracted with different convolutional kernels yield different classification performance. For scenes containing complex spatial texture information, relatively large convolutional kernel sizes can obtain better spatial features, as well as different directional features. For example, the Muufl Gulfport (MUUFL) data set obtains better classification results with 11×11 and 13×13 convolutional kernels. Whereas the Trento and Houston data sets require smaller convolutional kernels to extract features depending more on homogeneous features.

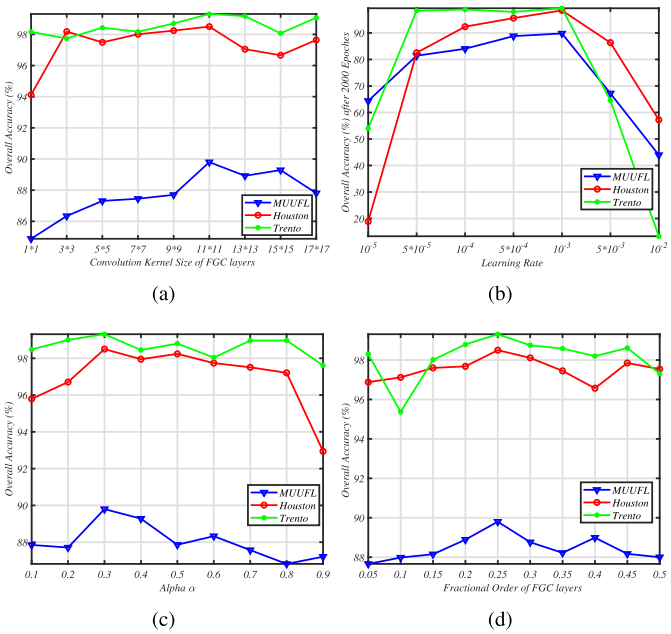

Fig. 8. Classification performance of the proposed FGCN with different parameters. (a) Convolution kernel size $r \times r$. (b) Learning rate lr. (c) Low-frequency ratio α of channels parameter in Octave convolution. (d) Fractional transform order p in fractional Gabor convolution layers.

As shown in Fig. 8(b), different learning rate lr affects the classification results as well. The search range of lr is constrained from 10^{-5} to 10^{-2} with other parameters $r \times r = 11 \times 11, \alpha = 0.3, p = 0.25$. The learning rate of the network is closely related to convergence, and the number of training epochs of the proposed FGCN is set to 2000. The algorithm with a small learning rate as $10^{-5}, 5*10^{-5}$ needs more epochs to reach convergence and the classification performance of complex classification scenes as MUUFL and Houston is poor. Large learning rates like 10^{-2} lead to large fluctuations in the objective function and may result in learning a suboptimal set of weights too fast or an unstable training process, especially for simple scenes like Trento.

For the specific ratio of channels parameter allocated to the low-frequency part α in Octave convolutions, the search range is constrained from 0.1 to 0.9, which indicates the classification performance varies with the ratio of low-frequency components. Other parameters are set to $r \times r = 11 \times 11, lr = 1e-3, p = 0.25$. A larger ratio of low-frequency components means a greater compression of homogeneous spatial information. As shown in Fig. 8(c), $\alpha = 0.3$ is optimal for the three cases, but the effect of α on the Houston and MUUFL data set is different from that of Trento. Due to the imbalance between HSI and LiDAR data, more useful homogeneous spatial information in LiDAR is compared with HSI. Therefore, for those HSI-dominated data sets such as Houston and MUUFL, a larger ratio of low-frequency components means more effectiveness for redundancy removal.

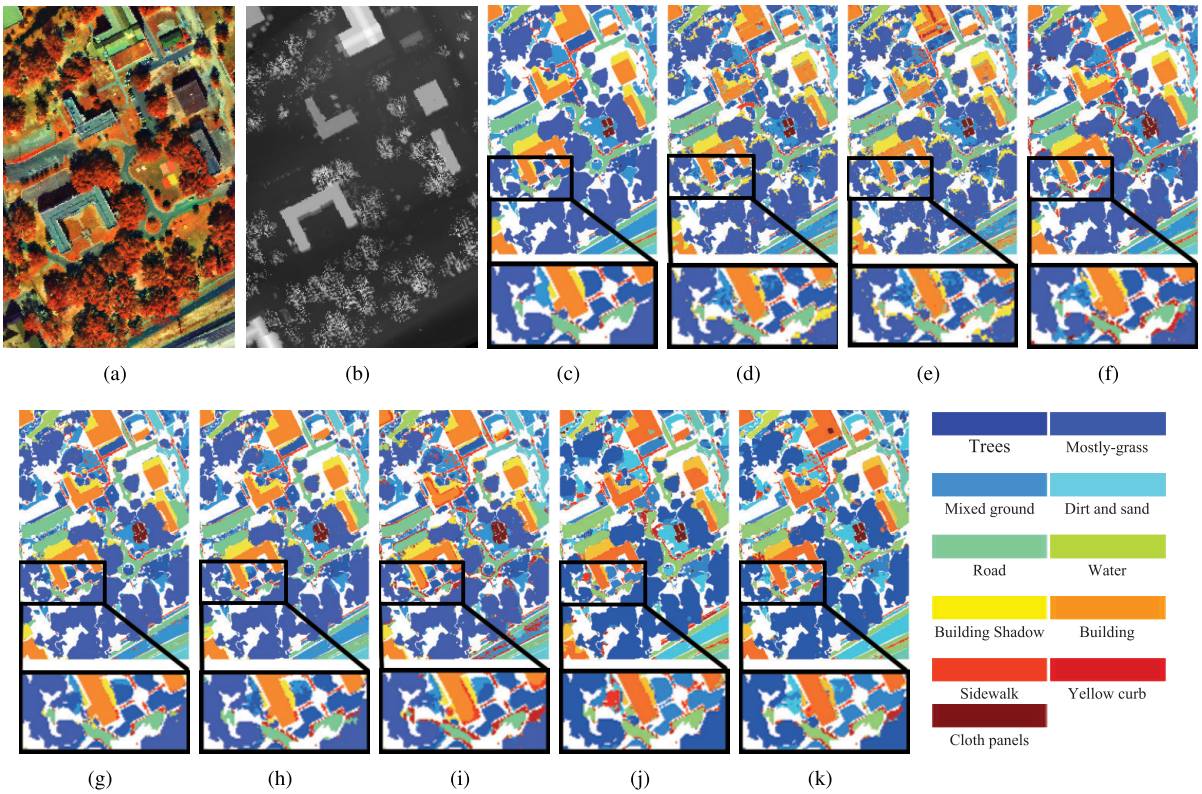

Fig. 9. Classification maps for the MUUFL Gulfport data obtained with different methods including: (a) Pseudo-color image for HSI, (b) LiDAR-based DSM, (c) Ground truth map, (d) SVM (82.63%), (e) RF (78.96%), (f) OTVCA (84.15%), (g) CRNN (85.01%), (h) TB-CNN(85.47%), (i) CCNN(86.01%), (j) CNNMRF (87.48%), and (k) FGCN (89.90%).

TABLE I

COMPARISON OF THE CLASSIFICATION ACCURACY (%) USING THE MUUFL GULFPORT DATA

No.	Class(Train/Test)	Performance							
		SVM	RF	OTVCA	CRNN	TB-CNN	CCNN	CNNMRF	FGCN
1	Trees (100/23246)	84.03±0.21	97.27±0.97	84.74±7.28	88.64±0.75	89.97±1.12	91.27±4.25	88.47±0.87	90.92±0.58
2	Mostly grass (100/4270)	85.97±0.12	55.61±1.13	82.47±0.71	79.46±2.31	80.61±4.98	81.22±7.79	87.03±1.58	89.06±0.65
3	Mixed ground surface (100/6882)	69.02±0.11	77.08±3.87	69.93±1.40	73.37±1.65	73.25±8.15	72.07±7.41	77.03±5.74	82.72±0.96
4	Dirt and sand (100/1826)	84.94±0.18	68.13±0.76	85.93±0.62	84.99±1.01	83.46±1.48	84.01±0.62	91.51±0.86	93.70±0.14
5	Road (100/6687)	87.71±0.31	87.74±0.60	83.76±2.84	88.04±2.11	88.04±0.48	86.71±1.97	91.86±4.22	90.88±0.47
6	Water (100/466)	93.99±0.06	71.91±0.92	99.45±1.14	67.60±0.59	67.60±0.13	67.17±1.54	97.21±0.13	100.00±0.00
7	Building shadow (100/2233)	87.06±0.15	45.21±2.81	91.19±0.61	84.55±1.37	84.15±3.01	85.49±6.07	93.91±0.66	88.76±0.96
8	Building (100/6240)	82.32±0.39	88.63±3.46	96.66±5.85	94.25±0.94	92.92±0.81	94.42±1.05	88.89±0.87	93.80±0.69
9	Sidewalk (100/1385)	74.08±0.54	42.25±2.54	75.80±0.39	66.57±2.85	67.73±4.46	68.01±2.60	74.73±5.91	79.35±7.67
10	Yellow curb (100/183)	97.81±0.72	20.62±0.39	89.16±0.33	15.85±1.48	17.49±2.71	16.94±1.26	92.90±3.82	96.72±2.07
11	Cloth panels (100/269)	98.88±0.78	60.41±0.12	98.22±0.31	42.75±0.63	43.12±0.41	42.75±0.20	97.77±1.53	100.00±0.00
	OA	82.63±0.10	78.96±0.65	84.15±0.56	85.01±0.37	85.47±1.05	86.01±2.45	87.48±0.77	**89.90±0.28**
	Kappa (×100)	77.79±0.11	73.27±1.17	79.57±2.00	83.91±0.29	84.47±1.27	85.18±1.45	83.86±2.03	**86.90±0.61**
	AA	85.98±0.11	64.99±0.87	87.03±0.64	71.46±0.45	71.67±1.76	71.82±2.02	89.21±0.99	**91.45±0.35**

For the Trento data set, both the HSI and LiDAR data contain redundant homogeneous information to be compressed, resulting in insensitivity to α.

For the specific fractional order p in fractional Gabor convolution layers, the search range is constrained from 0.05 to 0.5 at an 0.05 interval. Other parameters are set to $r \times r = 11 \times 11, lr = 1e-3, \alpha = 0.3$. The fractional order means a transform parameter between two convolutional layers, which indicate the output features of three fractional Gabor convolution (FGC) layers with order $p, 2p, 3p$ transformed from the input feature (for the additivity of fractional transform order [49]). A larger p imply features close to the feature map in the frequency domain, and small orders retain more spatial information. As shown in Fig. 8(d), the effect of p on the MUUFL, Houston, and Trento data set varies in a similar trend and $p = 0.25$ is optimal for the three data sets. As shown in Fig. 7, large orders extract more concentrated spatial information and a larger p means a larger output scale difference between layers. For the Houston, Trento, and MUUFL data set, distinct fractional features with $p = 0.25$ are effective for complex spatial information. The corresponding fractional order of three FGC layers $p_1 = 0.25, p_2 = 0.5, p_3 = 0.75$ extract features between the spatial and frequency domains uniformly.

C. Experimental Results and Comparison

To validate the effectiveness of the proposed FGCN, experimental results of the FGCN on the three HSI and LiDAR data

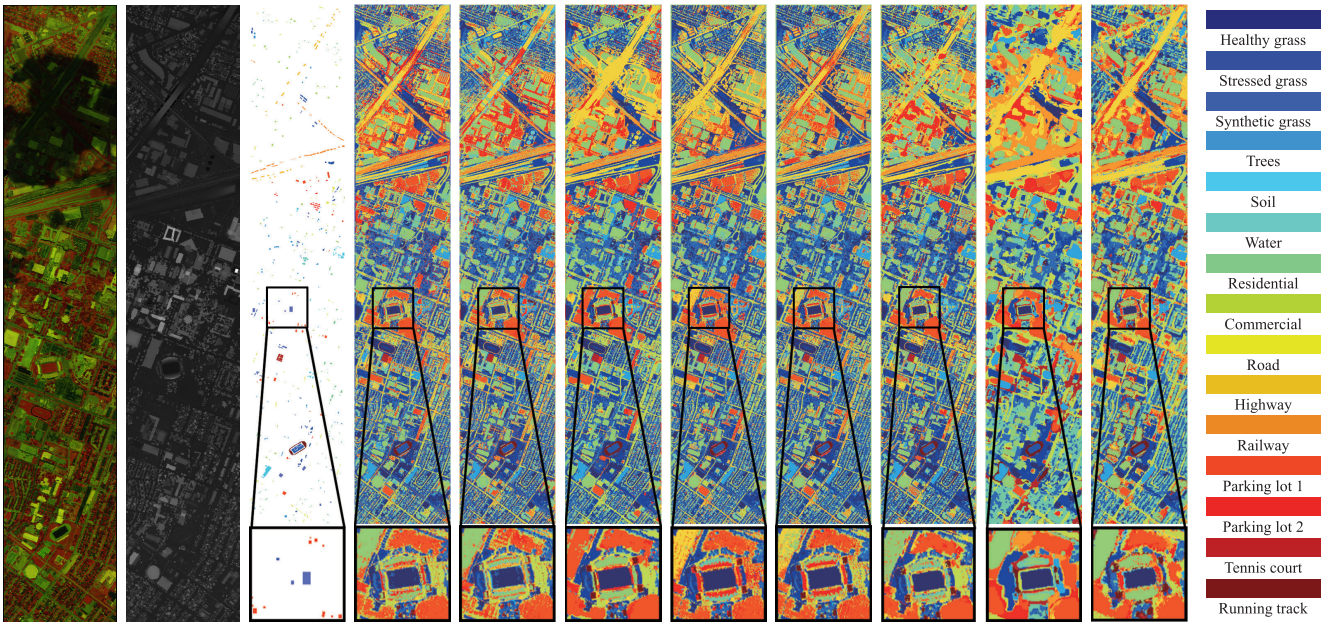

Fig. 10. Classification maps for the Houston data obtained with different methods. (a) Pseudo-color image for HSI (b) LiDAR-based DSM. (c) Ground truth map. (d) SVM (94.34%). (e) RF (88.63%). (f) OTVCA (97.51%). (g) CRNN (93.01%). (h) TB-CNN(91.60%). (i) CCNN(91.67%). (j) CNNMRF (95.15%). (k) FGCN (98.50%).

TABLE II

COMPARISON OF THE CLASSIFICATION ACCURACY (%) USING THE HOUSTON DATA

No.	Class(Train/Test)	Performance							
		SVM	RF	OTVCA	CRNN	TB-CNN	CCNN	CNNMRF	FGCN
1	Health grass (100/1251)	95.44±0.57	95.80±0.51	98.94±0.51	92.50±6.10	86.51±3.26	95.38±1.88	92.15±3.74	98.72±0.07
2	Stressed grass (100/1254)	97.85±2.16	94.53±2.82	96.89±1.46	99.15±7.50	97.27±8.13	94.82±2.24	91.78±1.33	98.30±0.39
3	Synthetic grass (100/697)	100.00±0.00	99.15±3.82	100.00±0.00	100.00±0.00	100.00±0.00	100.00±0.00	99.71±0.17	98.87±0.49
4	Tress (100/1244)	94.61±2.01	96.46±0.97	99.59±1.21	97.44±1.51	94.22±2.82	97.70±1.47	98.23±1.59	98.54±0.51
5	Soil (100/1242)	99.11±0.19	96.53±0.68	99.76±0.55	99.81±0.98	99.72±0.50	99.76±1.19	99.27±2.04	99.91±0.30
6	Water (100/325)	96.92±1.08	93.52±0.81	99.39±0.18	99.30±2.30	98.60±5.51	100.00±0.00	99.38±1.07	100.00±0.00
7	Residential (100/1268)	92.11±8.19	86.07±4.47	96.88±1.72	88.81±1.46	87.69±2.03	88.91±7.08	95.45±2.06	98.15±0.62
8	Commercial (100/1244)	93.81±4.45	89.31±8.73	99.58±3.82	88.13±9.00	91.07±3.87	95.32±9.71	98.19±7.98	97.68±0.90
9	Road (100/1252)	88.34±1.12	78.71±7.69	97.73±0.43	85.84±1.51	88.20±7.12	81.92±4.93	91.42±3.45	97.27±0.38
10	Highway (100/1227)	94.13±1.77	85.36±5.83	96.31±1.69	96.91±1.22	80.41±1.36	88.18±3.21	94.87±1.75	98.87±1.14
11	Railway (100/1235)	95.79±9.10	79.67±6.63	94.56±1.14	94.40±6.42	89.85±4.95	85.04±8.80	92.03±2.70	97.58±1.70
12	Parking lot 1 (100/1233)	88.48±5.76	84.78±4.35	97.53±1.19	76.66±5.31	88.28±4.37	82.96±3.79	99.65±7.19	98.18±0.58
13	Parking lot 2 (100/469)	81.66±4.64	57.24±6.82	84.97±6.63	97.89±3.50	98.25±6.73	78.21±7.15	86.35±5.82	98.97±4.88
14	Tennis court (100/428)	99.53±0.13	95.28±4.68	97.94±5.65	100.00±0.00	100.00±0.00	98.38±0.99	90.85±4.37	100.00±0.00
15	Running track (100/660)	99.70±0.17	99.10±0.15	99.40±0.81	100.00±0.00	100.00±0.00	98.21±0.78	98.06±2.06	100.00±0.00
	OA	94.34±5.30	88.63±6.19	97.51±0.49	93.01±4.52	91.60±3.65	91.67±3.52	95.15±2.49	**98.50±0.29**
	Kappa (×100)	93.88±5.72	87.71±6.68	97.31±0.53	92.42±3.39	90.88±2.74	91.00±2.00	94.76±1.56	**98.38±0.32**
	AA	94.50±4.52	88.77±5.77	97.30±0.33	94.46±5.73	93.34±4.79	92.32±4.42	95.16±2.91	**98.74±0.41**

sets are compared with other competitive classifiers including classical machine learning methods, morphological method, and recently convolutional networks:

1) Support Vector Machines (SVMs) [55] classifier implemented using the LIBSVM toolbox.[2] The regularization parameters range for the fivefold cross-validation is from 2^{-8} to 2^{10}, while a Gaussian radial basis function with $\gamma = 0.5$ is used for training.
2) Random Forest (RF) [56]. The number of trees is 200, while the number of the prediction variable is set to the square root of the number of input bands.
3) OTVCA [23]. The smoothness level λ is set to 1% of the maximum intensity.
4) CRNN [31]. There are two recurrent layers with Long Short-Term Memory (LSTM) units and 3×3 convolutional kernels.
5) Two-branch Convolutional Neural Network (TBCNN) [33]. The original input of the HSI branch is updated to the fused cube by GS-merge.
6) Contextual CNN (CCNN) [30]. The spatial sizes of extracted patches are set to $1 \times 1, 3 \times 3, 5 \times 5$.
7) Convolutional Neural Network with Markov Random Fields (CNNMRF) [36]. The prior distribution is smoothed by Markov Random Fields using three bands of the merged cube with the smoothness parameter $\mu = 20$.

CNN-based methods including CRNN, TBCNN, CCNN, CNNMRF are implemented with Tensorflow, and the rest of the experiments are implemented by MATLAB R2019a [57].

[2]http://www.csie.ntu.edu.tw/ cjlin/libsvm/

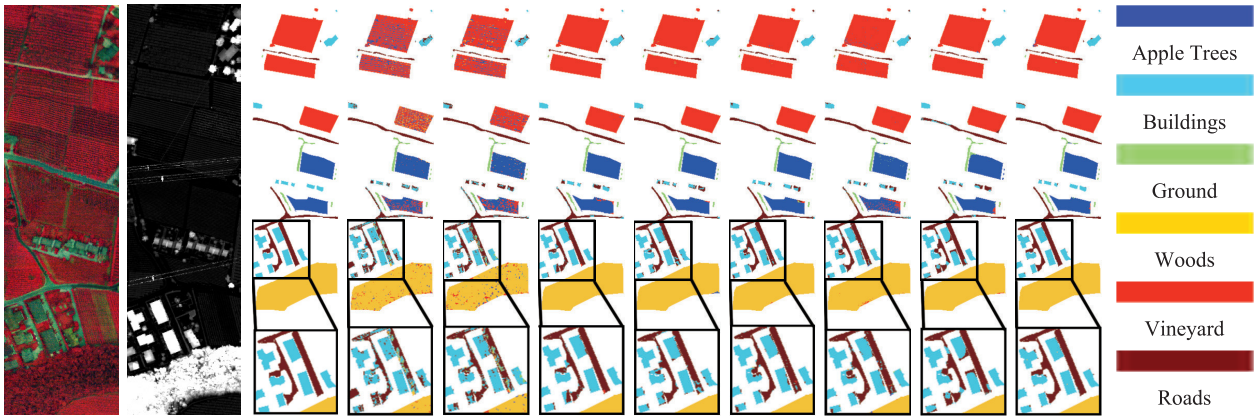

Fig. 11. Classification maps for the Trento data obtained with different methods including: (a) Pseudo-color image for HSI, (b) LiDAR-based DSM, (c) Ground truth map, (d) SVM (82.44%), (e) RF (86.44%), (f) OTVCA (98.12%), (g) CRNN (97.98%), (h) TB-CNN(98.46%), (i) CCNN(96.37%), (j) CNNMRF (96.60%), and (k) FGCN (99.32%).

TABLE III

COMPARISON OF THE CLASSIFICATION ACCURACY (%) USING THE TRENTO DATA

No.	Class(Train/Test)	Performance							
		SVM	RF	OTVCA	CRNN	TB-CNN	CCNN	CNNMRF	FGCN
1	Apple trees (50/4034)	89.49±0.30	67.42±4.28	99.35±2.70	99.23±0.16	98.74±0.25	92.09±3.69	99.21±0.65	99.32±0.30
2	Buildings (50/2903)	76.47±0.16	81.47±0.47	93.50±0.88	90.25±2.07	91.87±1.97	90.08±6.04	88.82±2.96	96.88±2.76
3	Ground (50/479)	97.70±0.91	67.72±0.48	98.54±3.30	100.00±0.00	99.37±0.12	94.78±3.83	76.13±5.97	96.04±5.03
4	Woods (50/9123)	93.63±0.70	97.46±0.15	99.95±3.24	99.68±1.67	99.93±0.22	99.38±1.20	100.00±0.00	100.00±0.00
5	Vineyard (50/10501)	73.53±0.36	90.66±0.27	99.14±2.69	99.81±1.18	99.88±2.14	98.77±0.97	99.34±0.06	99.68±0.07
6	Roads (50/3174)	73.91±0.60	81.40±1.11	92.09±4.10	92.19±5.13	95.09±1.47	91.21±8.94	85.48±3.50	98.78±2.38
OA		82.44±0.13	86.44±0.78	98.12±0.72	97.98±1.02	98.46±0.82	96.37±0.39	96.60±1.19	**99.32±0.30**
Kappa (×100)		77.03±0.14	82.09±0.92	97.49±1.23	97.30±1.49	97.95±1.40	95.15±2.95	95.47±0.63	**99.09±1.05**
AA		84.12±0.17	81.02±1.07	97.09±0.97	96.86±1.35	97.48±1.08	94.38±1.50	91.50±1.74	**98.45±0.41**

During the training process, the number of randomly selected training samples are shown in Tables I–III from the ground truth map shown in Figs. 9–11(c), and then all the rest samples are the test set. Parameters of the competitive algorithms are optimized and the same training and testing samples are used for a fair comparison.

As the qualitative classification results of different data sets shown in Figs. 9–11, the traditional method is susceptible to noise due to the loss of spatial information. Specifically in the Trento data set in Fig. 11 (d) and (e), it can be seen that lack of spatial information makes it difficult for SVM and RF to maintain spatial continuity, which leads to 10% lower accuracy as listed in Table III. Conversely, some methods that only use spatial neighborhood information such as OTVCA, CRNN, and CCNN can get smoother results, resulting in better classification performance in homogeneous areas. However, these methods perform differently on specific data sets. For example, CCNN performs well in smoother Trento, but the performance is limited in dealing with small-size categories such as yellow curb in the MUUFL data set and parking lot 2 in the Houston data set. The use of spatial information alone may lead to excessively smooth misclassification. Considering the respective advantages and disadvantages of spatial information and spectral information, spatial-spectral methods such as TBCNN and CNNMRF integrate their advantages in the classification process. As shown in Figs. 9–11(h) and (j), by performing convolution operations on both spatial and spectral dimensions, or a trade-off between spatial smoothness and boundary information, they not only reduce the impact of noise but also obtain better object boundaries. Unifying the spatial-spectral leads to better quantitative results as 98.46% OA of TBCNN for the Trento data set and 87.48% OA of CNNMRF for the MUUFL data set. Although the above methods perform well in pixel-level classification tasks using spatial information and spectral information, they cannot address various spectral variabilities and semantic changes of local scenes or objects. Due to some of their inherent limitations, such as the extracted spatial features cannot adaptively establish effective connections between objects of different proportions, which limits their ability to describe semantic structural information. Thus, in scenes with more challenging categories and more complex semantic variations as MUUFL and Houston data sets, the improvement in the performance of these comparison methods is limited such as the courtyard in Houston and trees in MUUFL.

On the contrary, the proposed FGCN method can extract the intrinsic feature representation of data sets in different transform domains to obtain more comprehensive data modeling and feature extraction. As shown in Fig. 9(k) of the MUUFL data set, there are fewer noise points in the classification map of the proposed FGCN. At the bottom right corner, the edges of the tree and the path are classified better with texture changes in different directions, the result is closer to the manually labeled ground truth compared to other methods. From the figures, the features extracted by the FGCN have better continuity in the area formed by the same substance and a more prominent boundary between the areas formed by different materials. Furthermore, for the object deformations

TABLE IV

ABLATION ANALYSIS OF THE PROPOSED FGCN IN TERMS OF OA (%) AND KAPPA (×100) ON THE THREE DATA SETS

Spectral	Spatial	Spatial-Spectral	LiDAR	Fractional Gabor	Methods	MUUFL		Houston		Trento	
						OA	Kappa	OA	Kappa	OA	Kappa
√	×	×	×	×	SpecFCN	67.49	58.99	54.25	51.14	61.42	50.91
×	√	×	×	×	SpatFCN	76.49	69.88	90.58	89.80	97.05	96.05
√	√	√	×	×	SSFCN	80.87	75.34	92.11	91.46	96.88	95.82
√	√	√	√	×	Octave-FFSCN	85.46	81.11	96.73	96.46	98.55	98.06
√	√	√	√	√	FGCN	89.21	85.91	98.50	98.38	99.18	98.91

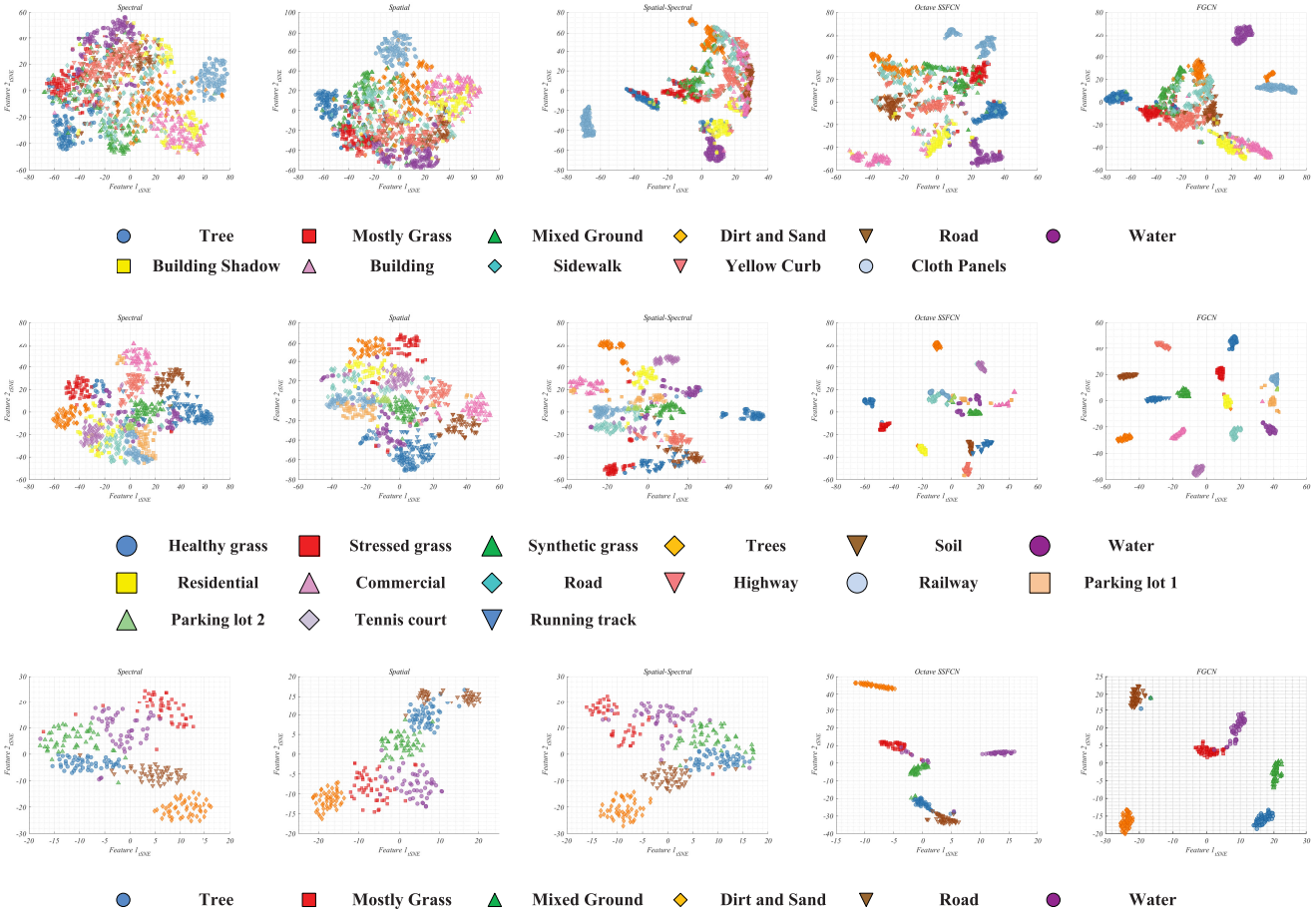

Fig. 12. Feature visualization of three multisource remote sensing data sets. (Top) MUUFL. (Middle) Houston. (Bottom) Trento. Stepwise addition of feature extraction strategy are listed.

in the central stadium area of the image, the deformation is of different scales throughout the figure, and the frequency of change is also different. The proposed FGCN can acquire good classification results in these areas, which is because of the sensitivity to the local semantic information change. The fractional Gabor filters in the FGCN architecture can extract local change information at different scales and in different directions. Unlike the above spatial-spectral unified method, our proposed FGCN can extract features from the transform domains between the spatial and frequency domains, which theoretically proved robust to changes in semantic information including shift, rotation, sensor noises, or distortions. Furthermore, for the use of multiscale filters in the convolutional layers, these semantic changes can be recognized even if at different scales. For example, courtyards of different sizes in the Houston data set and tree crowns in MUUFL, etc.

In general, the proposed FGCN can obtain a smoother classification result map, which means gathering targets of similar materials and elevations, and separating categories with clear boundaries. Specifically, the nonresidential buildings that constitute the large-scale coverage of the entire scene of Houston and the tree categories that constitute the MUUFL data set. For example, 82.72% mixed ground accuracy in the MUUFL data set, various road surfaces in Houston and Trento data sets, they also have regular spatial shapes. The FGCN uses the fractional Gabor filter to eliminate and compress noises while retaining meaningful semantic information and targets, which leads to improved classification accuracy.

D. Ablation Analysis

To investigate the performance improvement and demonstrate the motivations of our proposed FGCN, different

TABLE V

RATIOS OF KL DIVERGENCE FOR CLASSES FOR OCTAVE-SSFCN USING MUUFL

Class	2	3	4	5	6	7	8	9	10	11
1	2.47	3.02	4.40	3.36	2.10	2.43	3.56	3.23	3.14	2.76
2	-	2.59	3.76	3.08	2.66	3.10	3.24	2.84	2.67	2.70
3	-	-	3.97	2.96	3.91	3.77	2.91	2.91	3.01	3.76
4	-	-	-	4.47	9.30	7.78	4.12	3.70	4.36	4.00
5	-	-	-	-	5.65	6.44	3.20	3.44	3.45	3.63
6	-	-	-	-	-	1.62	5.36	6.71	6.45	4.65
7	-	-	-	-	-	-	5.99	7.67	7.89	5.17
8	-	-	-	-	-	-	-	4.09	4.15	3.68
9	-	-	-	-	-	-	-	-	3.08	4.91
10	-	-	-	-	-	-	-	-	-	4.70

TABLE VI

RATIOS OF KL DIVERGENCE FOR CLASSES FOR OCTAVE-SSFCN USING HOUSTON

Class	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	2.26	2.25	2.17	2.42	2.74	2.19	2.65	2.86	2.66	2.71	2.43	2.88	2.72	2.39
2	-	2.58	2.20	2.38	2.49	2.47	2.62	2.18	2.16	2.14	2.21	2.47	2.01	2.09
3	-	-	2.51	6.54	3.80	2.30	4.38	4.78	5.18	3.34	4.04	4.59	3.90	5.58
4	-	-	-	2.65	2.51	2.42	2.94	2.79	2.56	2.77	2.64	3.20	2.79	2.61
5	-	-	-	-	3.27	3.92	2.71	2.75	2.50	3.11	2.95	2.95	2.78	2.16
6	-	-	-	-	-	3.73	4.44	3.52	3.99	3.51	4.03	4.88	3.14	4.05
7	-	-	-	-	-	-	3.37	5.88	6.06	4.19	4.23	6.05	4.56	5.29
8	-	-	-	-	-	-	-	2.52	2.36	2.43	2.40	2.73	2.40	1.95
9	-	-	-	-	-	-	-	-	2.51	2.39	3.14	3.05	2.36	2.42
10	-	-	-	-	-	-	-	-	-	2.59	2.71	2.86	2.32	2.20
11	-	-	-	-	-	-	-	-	-	-	2.98	3.15	2.28	2.72
12	-	-	-	-	-	-	-	-	-	-	-	3.26	2.56	2.65
13	-	-	-	-	-	-	-	-	-	-	-	-	2.32	2.31
14	-	-	-	-	-	-	-	-	-	-	-	-	-	2.83

features extracted by the proposed FGCN are step-wise added. The proposed FGCN in Section II includes components such as Octave convolution layers for frequency components separation and synthesis, spatial feature extraction, spectral feature extraction, and feature extraction in multiple fractional transform domains. As listed in Table IV, the feature extraction and classification performance of the FGCN are gradually improved with the utilization of different modules. In Table IV, SpecFCN means convolutional networks using only spectral information, SpatFCN means using only spatial information, spectral-spatial fully convolutional networks (SSFCN) is a spectral-spatial unified network, Octave-SSFCN is an SSFCN with Octave convolution layers. As the quantitative analysis showed, the successful utilization of different steps can effectively achieve different extents of enhancement in the performance, effectively increasing the separability between different classes. As the effect of different steps on the three data shown in Fig. 12, different categories are more discriminative, which further yields better joint classification accuracy. These feature maps show the advantages of the proposed FGCN and demonstrate the motivations in Section II-D.

Detail Ablation Study for Octave Convolutional Layers: To quantitatively investigate the benefits of the proposed strategies (i.e., the Octave convolutional layers), the Kullback–Leibler (KL) divergence [58] is employed to measure the dissimilarity between distributions for classes after feature extraction and fusion strategies. For instance, the ratios of KL divergence for any two classes using Octave convolutional layers are listed in Tables V–VII. The values larger than one mean the discriminant pattern between classes increases as compared to the original HSI and LiDAR data. As listed in the tables, the Octave convolutional layers introduced in Section II-A are useful for information fusion and improved the feature discrimination, which can improve the classification accuracy and demonstrates the significance to combine the effective information in HSI and LiDAR data. It can also explain why directly connecting the data of different sensors as input reduce the accuracy.

TABLE VII

RATIOS OF KL DIVERGENCE FOR CLASSES FOR OCTAVE-SSFCN USING TRENTO

Class	2	3	4	5	6
1	2.75	2.67	2.52	2.65	2.06
2	-	2.97	4.05	5.45	2.50
3	-	-	3.38	3.79	2.01
4	-	-	-	3.11	2.82
5	-	-	-	-	2.46

While the redundancy in low-frequency components of the Octave convolutional output is reduced, the compression of low-frequency components can further reduce the volume of parameters and computational complexity compared with the normal convolutional networks [48]. To further evaluate the efficiency of Octave convolutional layers, giga-multiply and accumulation (GMACs) and accuracy trade-off comparisons are listed in Table VIII. We begin by using the normal 3-D convolutional layers as the baseline CNN and compared with our proposed Octave layers in FGCN to examine the GMACs-accuracy trade-off. In particular, we vary the low-frequency ratio $\alpha = 0.2, 0.4, 0.6, 0.8$ to compare the classification accuracy (i.e., OA) versus computational cost (i.e., GMACs) with the baseline.

From Table VIII, following observations are made. 1) The GMACs-Accuracy trade-off first rises up and then slowly goes down. 2) The network gets similar or better results even when the GMACs are reduced by about $1/3$ when $\alpha = 0.6$. 3) The network reaches its better accuracy when $\alpha = 0.2$ or 0.4, 2% higher than the normal convolutions. The increase in accuracy is because of the effective utilization of multifrequency processing of Octave convolutional layers and the corresponding fused HSI and LiDAR information which provides more comprehensive information to the network. The accuracy does not suddenly drop but decreases slowly after reaching the accuracy peak, which indicates compressing that the low-frequency part does not lead to significant information loss.

From the ablation analysis, the following key points can be summarized:

1) The frequency components separation and synthesis, low-frequency components compression through Octave convolution layers can improve the classification accuracy, which demonstrates the significance to combine the effective information in HSI and LiDAR data. It can also explain why directly connecting the data of different sensors as input reduce the accuracy. Further, the Octave convolutional layers can reduce the volume of parameters and computational complexity compared with the normal convolutional networks. Octave convolutional layers can improve accuracy while decreasing the GMACs.
2) Using spectral fully convolutional networks (FCN) alone yields poor results, which shows that using spectral information alone is not enough to distinguish targets with different characteristics. Although the effect of spatial FCN is better than that of spectral FCN, the performance of the classifier is limited without spectral information.

TABLE VIII
ABLATION ANALYSIS FOR OCTAVE CONVOLUTIONAL LAYERS

	MUUFL		Houston		Trento	
Octave Ratio (α)	OA (%)	GMACs	OA (%)	GMACs	OA (%)	GMACs
Normal Convolution	86.22	4.32	92.94	7.47	97.61	3.78
0.2	87.71	4.02	96.71	7.25	99.01	3.27
0.4	89.28	3.52	97.95	7.05	98.45	2.73
0.6	88.33	3.26	97.74	6.56	98.04	2.72
0.8	86.82	3.30	97.21	6.31	98.97	2.65

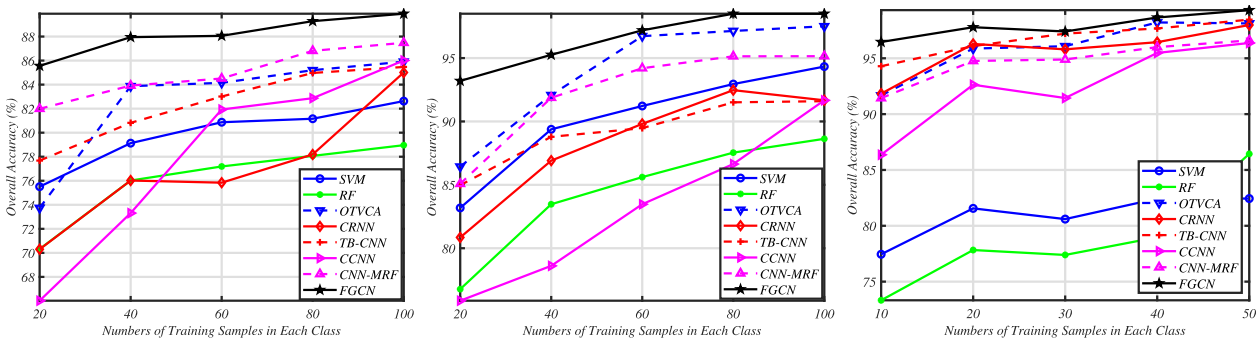

Fig. 13. Classification performance with different sizes of training samples using: (a) MUUFL, (b) Houston, and (c) Trento.

3) The difference between FGCN and Octave-SSFCN is whether the multidirection, multiscale features in fractional domains are utilized. Although the combination of spatial, spectral, and elevation information can achieve good classification performance, it is still limited without transform (frequency) features. The difference shows that FGC layers add diversity, discrimination, and robustness to all these spatial-spectral-frequency features.

E. Robustness Analysis

A robust algorithm can adapt to various conditions, which include the lack of labeled training samples, data containing noise, and data with low spatial resolution. To validate the robustness of the proposed FGCN, the three experiments are designed as follows.

1) Robustness to Number of Training Samples: In the actual multisource classification task, the training samples with labels are usually difficult to obtain causing many tasks that need to be performed with a small number of training sample [59]. To verify the robustness of the FGCN and other competitive methods for different numbers of the training sample, the classification performance of various methods using different numbers of training sample are shown in Fig. 13. In the experiment, the number of training samples in each class is set to 20%–100% of the original number of samples. With different training sample numbers, the FGCN shows the best classification performance among methods. As the number of training samples changes, the classification performance of the FGCN on the MUUFL data set stays stable while other methods decrease a lot. In Fig. 13(b), even for a small training data size of 20 pixels for the Houston data set, the FGCN still provides excellent classification performance with 93% OA while those of other methods are all below 87%.

2) Robustness to Noise: Actual multisensor remote sensing data sets are often affected by data noise and label noise, which reduces the classification accuracy [60]–[63]. The presence of different noise sources in multisource remote sensing data makes its modeling and the classification task challenging. The real HSI and LiDAR data may exist both signal independent noise like thermal noise and quantization noise and also more challenging signal-dependent noise like shot (photon) noise. To verify the robustness of FGCN to the noise that may exist in practice and compare it with other comparison methods, we studied the impact of different degrees of data noise on the classification results as shown in Fig. 14. Usually, corrupted and noisy spectral bands are removed in benchmark data sets. Thus, in our experiment, signal-dependent Gaussian white noise with different degrees of zero means is added to the density map satisfies the following model:

$$E' = E(1 + \sigma G) \tag{14}$$

where E and E' are, respectively, the original and noisy data sets, G denotes Gaussian white noise, with the power of noise is 0 dBW. σ denoting noisy level that varies from 0.1 to 0.7. In Fig. 14, the FGCN shows the best classification performance under the influence of different noise intensities. Even if affected by a strong noise level of 0.7, the FGCN still obtains an OA of more than 95% on the Houston data set. But the proposed method is also susceptible to noise in complex scenes such as MUUFL, and the classification performance decreases as noise intensity increases.

3) Robustness to Data Resolution: In practice, multisource data are usually with medium or low resolution. To verify the stability of the algorithm for medium and low-resolution data, we studied the performance of all comparison methods under different resolution data conditions in Fig. 15. Gaussian downsampling operations are used to resample the experimental

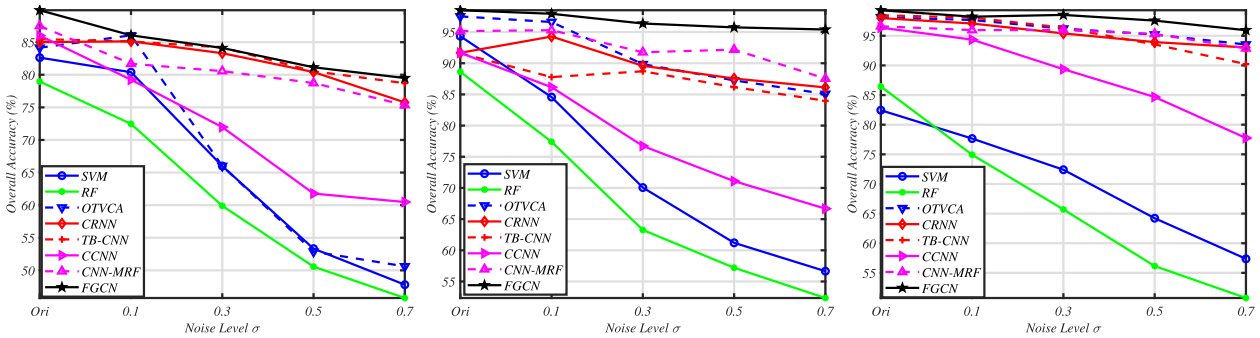

Fig. 14. Classification performance under different noise level using: (a) MUUFL, (b) Houston, and (c) Trento.

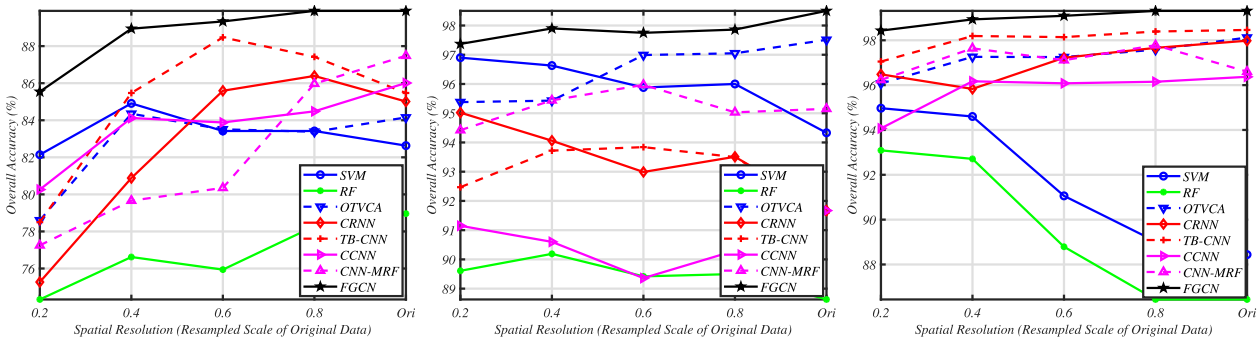

Fig. 15. Classification performance with different resolutions of data sets using: (a) MUUFL, (b) Houston, and (c) Trento.

TABLE IX
COMPUTATIONAL COMPLEXITY ANALYSIS USING GMACS AND NUMBER OF PARAMETERS OF DEEP NETWORKS

Method (Input Size)	MACs (G)	Total Params (M)	Trainable Params (M)	Non-trainable Params (M)
CRNN (1*1*64)	0.0268	2.6446	2.6362	0.0084
TB-CNN (1*1*64)	0.0170	5.5212	5.4627	0.0585
CCNN (1*1*64)	0.0064	2.0068	2.0058	0.0010
CNNMRF (1*1*64)	0.0054	1.2593	1.2586	0.0006
FGCN (1*1*64)	0.0038	1.0043	0.3346	0.6697
FGCN: MUUFL (325*220*64)	3.5236	1.0048	0.3347	0.6701
FGCN: Houston (349*1905*144)	6.7940	3.1471	1.0486	2.0985
FGCN: Trento (600*166*63)	2.8426	0.9859	0.3284	0.6575

data sets to lower resolution. Then simple interpolation is used to resize the data as the original data for classification, and the effect of different levels $0.2 - 1$ of downsampling on OA is shown in Fig. 15. Except that in MUUFL, a complex scene containing small targets, the performance is reduced due to the loss of small category information, the FGCN is stable with the best classification performance among classifiers. Another interesting phenomenon is that in simple scenarios like Trento, some methods such as SVM and RF classification accuracy are higher in the case with low spatial resolution.

F. Computational Cost Analysis

To evaluate the computational complexity of the proposed FGCN, Table IX lists the computational cost including the number of parameters and GMACs of the proposed FGCN and other competitive networks. As listed in Table IX, the practical GMACs vary in real scenarios. Because the Octave convolutional layers and the FGC layers are applied on the 3-D data cube, the nontrainable parameter including initial parameters of convolutional weights and biases take $2/3$ of all the parameters. From Table IX, the design of FGCN leads to three aspects: 1) In each training epoch, GMACs cost grows with the size of data sets. 2) The FGCN model is more stable under different data and label conditions and less training time costs for one training epoch. 3) The proposed FGCN can deliver important practical benefits, rather than only saving GMACs in theory. Through Octave convolutional layers, the model costs less actual training and inference time in practice. In Table X, we demonstrate the GMACs and parameter saving of the whole data set is reflected in the actual training time. In Table IX, the computation cost of FGCN is computed using the average of different ratio α, with the corresponding analysis shown in Section III-D.

The training and testing computational cost of the competitive methods is listed in Table X. The implementation of

TABLE X
ELAPSED TIME (*s*: SECOND) OF TRAINING AND TESTING TIME FOR THE PROPOSED METHOD USING THE EXPERIMENTAL DATA SETS

		MUUFL	*Houston*	*Trento*
SVM	Training	20.05	20.74	3.42
	Testing	4.05	9.06	1.13
RF	Training	0.77	10.13	0.49
	Testing	0.29	4.62	0.27
OTVCA	Training	5.98	94.32	4.06
	Testing	0.28	4.57	0.26
CRNN	Training	576.21	612.37	504.14
	Testing	11.05	11.34	12.11
TB-CNN	Training	432.42	540.18	396.61
	Testing	20.15	13.14	12.27
CCNN	Training	1548.77	1642.33	1368.75
	Testing	18.12	7.61	7.89
CNNMRF	Training	1944.54	2304.21	1835.91
	Testing	16.92	18.20	16.74
FGCN	Training	216.34	2376.41	252.27
	Testing	0.33	0.34	0.33

computational time experiments uses the same configuration of hardware and software. The training process is more time-consuming than the test of the whole scene. Because of the utilization of fully convolutional layers, the proposed FGCN can avoid the patch extraction step and train more efficiently. But at the same time, the utilization of fully convolutional layers caused a nonlinear increment of the running time as the size of the data set increases.

IV. CONCLUSION

In this article, an FGCN was proposed to extract comprehensive deep features for the joint classification of HSI and LiDAR data. The proposed FGCN first used Octave convolution layers to perform multisource information fusion and improve discrimination. Second, the FGC layers were proposed to extract multiscale, multidirectional features, and local semantic changes. The completeness and discrimination of the multisource features using different FGC kernels were improved, yielding robust feature extraction against semantic changes. Finally, the fractional Gabor feature and spectral feature were combined with two weighting factors which can be learned during the network training. Experimental results and comparisons with state-of-the-art multisource classification methods indicated the effectiveness of the proposed FGCN. However, the proposed method still has some specific limitations. For example, extending the proposed FGCN for large-scene data sets requires more research. With the use of fully convolutional layers in comprehensive feature extraction, a nonlinear increment of the running time was caused as the size of the data set increases. Furthermore, the performance declines with extremely few training samples or a completely disjoint training set. More care needs to be taken to minimize overlap between training and test frames, which is the focus of further work. Our code and all the results are available at https://github.com/xudongzhao461/FGCN for the sake of reproducibility.

REFERENCES

[1] M. Zhang, W. Li, and Q. Du, "Collaborative classification of hyperspectral and visible images with convolutional neural network," *J. Appl. Remote Sens.*, vol. 11, no. 4, 2017, Art. no. 042607.

[2] M. Schmitt and X. X. Zhu, "Data fusion and remote sensing: An ever-growing relationship," *IEEE Geosci. Remote Sens. Mag.*, vol. 4, no. 4, pp. 6–23, Dec. 2016.

[3] M. Khodadadzadeh, J. Li, S. Prasad, and A. Plaza, "Fusion of hyperspectral and LiDAR remote sensing data using multiple feature learning," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2971–2983, Jun. 2015.

[4] R. Hansch and O. Hellwich, "Fusion of multispectral LiDAR, hyperspectral, and RGB data for urban land cover classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 2, pp. 366–370, Feb. 2021.

[5] T. Matsuki, N. Yokoya, and A. Iwasaki, "Hyperspectral tree species classification of Japanese complex mixed forest with the aid of lidar data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 5, pp. 2177–2187, May 2015.

[6] M. Khodadadzadeh, A. Cuartero, J. Li, A. Felicisimo, and A. Plaza, "Fusion of hyperspectral and Lidar data using generalized composite kernels: A case study in Extremadura, Spain," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2015, pp. 61–64.

[7] C. Weitkamp, *Lidar: Range-Resolved Optical Remote Sensing of the Atmosphere*, vol. 102. Cham, Switzerland: Springer, 2006.

[8] C. Mallet and F. Bretar, "Full-waveform topographic Lidar: State-of-the-art," *ISPRS J. Photogramm. Remote Sens.*, vol. 64, no. 1, pp. 1–16, Jan. 2009.

[9] J. Jung, E. Pasolli, S. Prasad, J. C. Tilton, and M. M. Crawford, "A framework for land cover classification using discrete return LiDAR data: Adopting pseudo-waveform and hierarchical segmentation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 2, pp. 491–502, Feb. 2014.

[10] M. Brell, K. Segl, L. Guanter, and B. Bookhagen, "Hyperspectral and lidar intensity data fusion: A framework for the rigorous correction of illumination, anisotropic effects, and cross calibration," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2799–2810, May 2017.

[11] P. Ghamisi *et al.*, "Multisource and multitemporal data fusion in remote sensing: A comprehensive review of the state of the art," *IEEE Geosci. Remote Sens. Mag.*, vol. 7, no. 1, pp. 6–39, Mar. 2019.

[12] X. Zhu, F. Cai, J. Tian, and T. Williams, "Spatiotemporal fusion of multisource remote sensing data: Literature survey, taxonomy, principles, applications, and future directions," *Remote Sens.*, vol. 10, no. 4, p. 527, Mar. 2018.

[13] X. Zhao, R. Tao, and W. Li, "Multisource remote sensing data classification using deep hierarchical random walk networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 2187–2191.

[14] B. Rasti, P. Ghamisi, J. Plaza, and A. Plaza, "Fusion of hyperspectral and LiDAR data using sparse and low-rank component analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 11, pp. 6354–6365, Nov. 2017.

[15] X. Kang, S. Li, L. Fang, M. Li, and J. A. Benediktsson, "Extended random walker-based classification of hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 1, pp. 144–153, Jan. 2015.

[16] X. Zhao *et al.*, "Joint classification of hyperspectral and LiDAR data using hierarchical random walk and deep CNN architecture," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 10, pp. 7355–7370, Oct. 2020.

[17] J. Xia, W. Liao, and P. Du, "Hyperspectral and LiDAR classification with semisupervised graph fusion," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 4, pp. 666–670, Apr. 2020.

[18] P. Duan, X. Kang, S. Li, P. Ghamisi, and J. A. Benediktsson, "Fusion of multiple edge-preserving operations for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 12, pp. 10336–10349, Dec. 2019.

[19] R. Luo *et al.*, "Fusion of hyperspectral and LiDAR data for classification of cloud-shadow mixed remote sensed scene," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 8, pp. 3768–3781, Aug. 2017.

[20] Y. Zhang and S. Prasad, "Multisource geospatial data fusion via local joint sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 6, pp. 3265–3276, Jun. 2016.

[21] P. Ghamisi, J. A. Benediktsson, and S. Phinn, "Land-cover classification using both hyperspectral and LiDAR data," *Int. J. Image Data Fusion*, vol. 6, no. 3, pp. 189–215, Jul. 2015.

[22] P. Duan, J. Lai, J. Kang, X. Kang, P. Ghamisi, and S. Li, "Texture-aware total variation-based removal of sun glint in hyperspectral images," *ISPRS J. Photogramm. Remote Sens.*, vol. 166, pp. 359–372, Aug. 2020.

[23] B. Rasti, P. Ghamisi, and R. Gloaguen, "Hyperspectral and LiDAR fusion using extinction profiles and total variation component analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3997–4007, Jul. 2017.

[24] M. Zhang, P. Ghamisi, and W. Li, "Classification of hyperspectral and LIDAR data using extinction profiles with feature fusion," *Remote Sens. Lett.*, vol. 8, no. 10, pp. 957–966, Oct. 2017.
[25] D. Lemp and U. Weidner, "Improvements of roof surface classification using hyperspectral and laser scanning data," in *Proc. ISPRS Joint Conf., 3rd Int. Symp. Remote Sens. Data Fusion Over Urban Areas (URBAN), 5th Int. Symp. Remote Sens. Urban Areas (URS)*, 2005, pp. 14–16.
[26] W. Liao, R. Bellens, A. Pizurica, S. Gautama, and W. Philips, "Graph-based feature fusion of hyperspectral and Lidar remote sensing data using morphological features," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, vol. 7, Jul. 2013, pp. 4942–4945.
[27] W. Liao, R. Bellens, A. Pizurica, S. Gautama, and W. Philips, "Combining feature fusion and decision fusion for classification of hyperspectral and LiDAR data," in *Proc. IEEE Geosci. Remote Sens. Symp.*, Jul. 2014, pp. 1241–1244.
[28] M. E. Paoletti, J. M. Haut, R. Fernandez-Beltran, J. Plaza, A. J. Plaza, and F. Pla, "Deep pyramidal residual networks for spectral–spatial hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 740–754, Feb. 2019.
[29] Y. Xu, B. Du, and L. Zhang, "Beyond the patchwise classification: Spectral-spatial fully convolutional networks for hyperspectral image classification," *IEEE Trans. Big Data*, vol. 6, no. 3, pp. 492–506, Sep. 2020.
[30] H. Lee and H. Kwon, "Going deeper with contextual CNN for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 26, no. 10, pp. 4843–4855, Oct. 2017.
[31] L. Mou, P. Ghamisi, and X. X. Zhu, "Deep recurrent neural networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3639–3655, Jul. 2017.
[32] A. Merentitis, C. Debes, R. Heremans, and N. Frangiadakis, "Automatic fusion and classification of hyperspectral and LiDAR data using random forests," in *Proc. IEEE Geosci. Remote Sens. Symp.*, Jul. 2014, pp. 1245–1248.
[33] X. Xu, W. Li, Q. Ran, Q. Du, L. Gao, and B. Zhang, "Multisource remote sensing data classification based on convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 937–949, Feb. 2018.
[34] Q. Feng, D. Zhu, J. Yang, and B. Li, "Multisource hyperspectral and LiDAR data fusion for urban land-use mapping based on a modified two-branch convolutional neural network," *ISPRS Int. J. Geo-Inf.*, vol. 8, no. 1, p. 28, Jan. 2019.
[35] R. Hang, Z. Li, P. Ghamisi, D. Hong, G. Xia, and Q. Liu, "Classification of hyperspectral and LiDAR data using coupled CNNs," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 7, pp. 4939–4950, Jul. 2020.
[36] X. Cao, F. Zhou, L. Xu, D. Meng, Z. Xu, and J. Paisley, "Hyperspectral image classification with Markov random fields and a convolutional neural network," *IEEE Trans. Image Process.*, vol. 27, no. 5, pp. 2354–2367, Jan. 2018.
[37] M. Zhang, W. Li, Q. Du, L. Gao, and B. Zhang, "Feature extraction for classification of hyperspectral and LiDAR data using patch-to-patch CNN," *IEEE Trans. Cybern.*, vol. 50, no. 1, pp. 100–111, Jan. 2020.
[38] Y. Zhong, Q. Cao, J. Zhao, A. Ma, B. Zhao, and L. Zhang, "Optimal decision fusion for urban land-use/land-cover classification based on adaptive differential evolution using hyperspectral and LiDAR data," *Remote Sens.*, vol. 9, no. 8, p. 868, Aug. 2017.
[39] K. Liu *et al.*, "Rotation-invariant HOG descriptors using Fourier analysis in polar and spherical coordinates," *Int. J. Comput. Vis.*, vol. 106, no. 3, pp. 342–364, Feb. 2014.
[40] S. Luan, C. Chen, B. Zhang, J. Han, and J. Liu, "Gabor convolutional networks," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4357–4366, Sep. 2018.
[41] Y. Chen, L. Zhu, P. Ghamisi, X. Jia, G. Li, and L. Tang, "Hyperspectral images classification with Gabor filtering and convolutional neural network," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 12, pp. 2355–2359, Dec. 2017.
[42] X. Wu, D. Hong, J. Chanussot, Y. Xu, R. Tao, and Y. Wang, "Fourier-based rotation-invariant feature boosting: An efficient framework for geospatial object detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 2, pp. 302–306, Feb. 2020.
[43] D. Hong, X. Wu, P. Ghamisi, J. Chanussot, N. Yokoya, and X. X. Zhu, "Invariant attribute profiles: A spatial-frequency joint feature extractor for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 6, pp. 3791–3808, Jun. 2020.
[44] R. Tao, X. Zhao, W. Li, H.-C. Li, and Q. Du, "Hyperspectral anomaly detection by fractional Fourier entropy," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 12, pp. 4920–4929, Dec. 2019.
[45] Q. Xu, D. Wang, and B. Luo, "Faster multiscale capsule network with octave convolution for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 2, pp. 361–365, Feb. 2021.
[46] X. Tang *et al.*, "Hyperspectral image classification based on 3-D octave convolution with spatial–spectral attention network," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 3, pp. 2430–2447, Mar. 2021.
[47] C. A. Laben and B. V. Brower, "Process for enhancing the spatial resolution of multispectral imagery using pan-sharpening," U.S. Patent 6 011 875, Jan. 4, 2000.
[48] Y. Chen *et al.*, "Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3435–3444.
[49] H. M. Ozaktas and M. A. Kutay, "The fractional Fourier transform," in *Proc. Eur. Control Conf. (ECC)*, 2001, pp. 1477–1483.
[50] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. ICML*, 2013, p. 3.
[51] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.
[52] Q. Huang, W. Li, B. Zhang, Q. Li, R. Tao, and N. H. Lovell, "Blood cell classification based on hyperspectral imaging with modulated Gabor and CNN," *IEEE J. Biomed. Health Informat.*, vol. 24, no. 1, pp. 160–170, Jan. 2020.
[53] G. Paul, Z. Alina, C. Ryan, A. Jen, and T. Grady, "MUUFL gulfport hyperspectral and LiDAR airborne data set," Univ. Florida, Gainesville, FL, USA, Tech. Rep. REP-2013-570, 2013.
[54] W. Y. W. Hu Huang Li and F. H. Zhang Li, "Deep convolutional neural networks for hyperspectral image classification," *J. Sensors*, vol. 2015, Jan. 2015, Art. no. 258619.
[55] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intell. Syst. Appl.*, vol. 13, no. 4, pp. 18–28, Jul. 1998.
[56] J. Ham, Y. Chen, M. M. Crawford, and J. Ghosh, "Investigation of the random forest framework for classification of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 492–501, Mar. 2005.
[57] *The Mathworks*, MATLAB MathWorks, Natick, MA, USA, 1992.
[58] M. N. Do, "Fast approximation of Kullback-Leibler distance for dependence trees and hidden Markov models," *IEEE Signal Process. Lett.*, vol. 10, no. 4, pp. 115–118, Apr. 2003.
[59] G. Camps-Valls and L. Bruzzone, "Kernel-based methods for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 6, pp. 1351–1362, Jun. 2005.
[60] B. Frenay and M. Verleysen, "Classification in the presence of label noise: A survey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 5, pp. 845–869, May 2014.
[61] J. Jiang, J. Ma, Z. Wang, C. Chen, and X. Liu, "Hyperspectral image classification in the presence of noisy labels," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 851–865, Feb. 2019.
[62] C. Pelletier, S. Valero, J. Inglada, N. Champion, C. M. Sicre, and G. Dedieu, "Effect of training class label noise on classification performances for land cover mapping with satellite image time series," *Remote Sens.*, vol. 9, no. 2, p. 173, Feb. 2017.
[63] P. Duan, X. Kang, S. Li, and P. Ghamisi, "Noise-robust hyperspectral image classification via multi-scale total variation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 6, pp. 1948–1962, Jun. 2019.

Xudong Zhao (Student Member, IEEE) received the B.S. degree from the Department of Science and Technology of Electronic Information, Beijing Institute of Technology, Beijing, China, in 2016. He is pursuing the Ph.D. degree in information and communication engineering with the Beijing Institute of Technology (BIT), Beijing, and the Ph.D. degree in computer science engineering with Ghent University, Ghent, Belgium.

His research interests include multisource remote sensing and fractional signal processing.

Ran Tao (Senior Member, IEEE) received the B.S. degree from the Electronics Engineering Institute of PLA, Hefei, China, in 1985, and the M.S. and Ph.D. degrees from the Harbin Institute of Technology, Harbin, China, in 1990 and 1993, respectively.

He has been a Senior Visiting Scholar with the University of Michigan, Ann Arbor, MI, USA, and the University of Delaware, Newark, DE, USA, in 2001 and 2016, respectively. He is a Professor with the School of Information and Electronics, Beijing Institute of Technology, Beijing, China. His research interests include fractional Fourier transform and its applications, theory, and technology for radar and communication systems.

Dr. Tao is a fellow of the Institute of Engineering and Technology and the Chinese Institute of Electronics. He was a recipient of the National Science Foundation (NSF) of China for Distinguished Young Scholars in 2006 and a Distinguished Professor of Changjiang Scholars Program in 2009. He has been a Chief-Professor of the Creative Research Groups with the National Natural Science Foundation of China since 2014, and he was a Chief-Professor of the Program for Changjiang Scholars and Innovative Research Team in University from 2010 to 2012. He is the Vice-Chair of the IEEE China Council. He is also the Vice-Chair of the International Union of Radio Science (URSI) China Council and a member of Wireless Communication and Signal Processing Commission, URSI. He was a recipient of the First Prize of Science and Technology Progress in 2006 and 2007, and the First Prize of Natural Science in 2013, both awarded by the Ministry of Education.

Wei Li (Senior Member, IEEE) received the B.S. degree in telecommunications engineering from Xidian University, Xi'an, China, in 2007, the M.S. degree in information science and technology from Sun Yat-sen University, Guangzhou, China, in 2009, and the Ph.D. degree in electrical and computer engineering from Mississippi State University, Starkville, MS, USA, in 2012.

Subsequently, he spent one year as a Postdoctoral Researcher with the University of California, Davis, CA, USA. He is a Professor with the School of Information and Electronics, Beijing Institute of Technology.

Dr. Li received the 2015 Best Reviewer Award from the IEEE Geoscience and Remote Sensing Society (GRSS) for his service for the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATION AND REMOTE SENSING (JSTARS). He is serving as an Associate Editor for the IEEE SIGNAL PROCESSING LETTERS, and the IEEE JSTARS. He is also the Topical Associate Editor for the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING. He has served as the Guest Editor for the Special Issue of *Journal of Real-Time Image Processing, Remote Sensing*, and the IEEE JSTARS.

Wilfried Philips (Senior Member, IEEE) was born in Aalst, Belgium, in 1966. He received the Diploma degree in electrical engineering and the Ph.D. degree in applied sciences from Ghent University, Ghent, Belgium, in 1989 and 1993, respectively.

He is a Senior Full Professor with the Department of Telecommunications and Information Processing, Ghent University, where he heads the Image Processing and Interpretation Research Group. He also leads the activities in image processing and sensor fusion within the research institute IMEC. His main research interests include image and video quality improvement and estimation, real-time computer vision, and sensor data processing. He is also a Co-Founder of the Senso2Me company, which provides Internet of Things solutions for elderly care.

Wenzhi Liao (Senior Member, IEEE) received the B.S. degree in mathematics from Hainan Normal University, Haikou, China, in 2006, the Ph.D. degree in engineering from the South China University of Technology, Guangzhou, China, in 2012, and the Ph.D. degree in computer science engineering from Ghent University, Ghent, Belgium, in 2012.

Since 2012, he has been working as a Post-Doctoral Research Fellow first with Ghent University and then with the Research Foundation Flanders (FWO), Vlaanderen, Belgium. Since 2020, he has been with Sustainable Materials Management, Flemish Institute for Technological Research (VITO), Mol, Belgium. His research interests include pattern recognition, remote sensing, and image processing. In particular, his interests include mathematical morphology, multitask feature learning, multisensor data fusion, and hyperspectral image restoration.

Dr. Liao was a recipient of the Best Paper Challenge Awards in both the 2013 IEEE GRSS Data Fusion Contest and the 2014 IEEE GRSS Data Fusion Contest. He is serving as an Associate Editor for the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATION AND REMOTE SENSING (JSTARS), and the *IET Image Processing*.