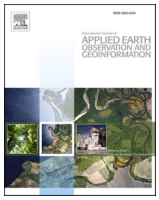
International Journal of Applied Earth Observations and Geoinformation 105 (2021) 102634

Contents lists available at ScienceDirect

International Journal of Applied Earth Observations and Geoinformation

journal homepage: www.elsevier.com/locate/jag



Airborne multispectral LiDAR point cloud classification with a feature Reasoning-based graph convolution network

Peiran Zhao [a], Haiyan Guan [a,*], Dilong Li [b], Yongtao Yu [c], Hanyun Wang [d], Kyle Gao [e], José Marcato Junior [f], Jonathan Li [e]

[a] School of Remote Sensing and Geomatics Engineering, Nanjing University of Information Science and Technology, Nanjing 210044, China
[b] Department of Computer Science and Technology, Fujian Key Laboratory of Big Data Intelligence and Security, Xiamen Key Laboratory of Computer Vision and Pattern Recognition, Huaqiao University, China
[c] Faculty of Computer and Software Engineering, Huaiyin Institute of Technology, Huaian, JS 223003, China
[d] School of Surveying and Mapping, Information Engineering University, Zhengzhou, HN 45000, China
[e] Department of Geography and Environmental Management and Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada
[f] Faculty of Engineering, Architecture and Urbanism and Geography, Federal University of Mato Grosso do Sul, Campo Grande 79070900, Brazil

ARTICLE INFO

Keywords:
Multispectral LiDAR
Point cloud classification
Deep learning
Graph convolution network
Feature reasoning

ABSTRACT

This paper presents a feature reasoning-based graph convolution network (FR-GCNet) to improve the classification accuracy of airborne multispectral LiDAR (MS-LiDAR) point clouds. In the FR-GCNet, we directly assign semantic labels to all points by exploring representative features both globally and locally. Based on the graph convolution network (GCN), a global reasoning unit is embedded to obtain the global contextual feature by revealing spatial relationships of points, while a local reasoning unit is integrated to dynamically learn edge features with attention weights in each local graph. Extensive experiments on the Titan MS-LiDAR data showed that the proposed FR-GCNet achieved a promising classification performance with an overall accuracy of 93.55%, an average F_1-*score* of 78.61%, and a mean Intersection over Union (IoU) of 66.78%. Comparative experimental results demonstrated the superiority of the FR-GCNet against other state-of-the-art approaches.

1. Introduction

The multispectral LiDAR (MS-LiDAR) is demonstrably superior to its single-channel counterparts in land-cover/ land-use classification (Ekhtari et al., 2018; Morsy et al., 2017). For example, a Titan LiDAR system nowadays is able to simultaneously operate at three channels with different looking angles and acquire three point clouds, i.e., Channel 1 at mid-inferead (MIR) with a wavelength of 1550 nm at 3.5° forward looking, Channel 2 at near-infreared (NIR) with a wavelength of 1064 nm at 0° nadir looking, and Channel 3 at Green with a wavelength of 532 nm at 7° forward looking. MS-LiDAR point clouds collected at different channels allow for a higher reliability and accuracy compared with the monochromatic channel LiDAR data. Previous studies (Guo et al., 2011; Griffiths and Boehm, 2019) have shown that the combined use of LiDAR height and intensity data can improve the land cover classification accuracy considerably, in comparison with that obtained through the combination of height data acquired by the single-channel LiDAR with the high-resolution multispectral imagery acquired by airborne or satellite-bonre optical imaging sensors. The MS-LiDAR technology has attracted increasing attention in many applications, such as forest inventory (Lindberg et al., 2021), land-cover classification (Ali et al., 2021), building extraction (Li et al., 2020), and coast-line or water-body extraction (Shaker et al., 2019).

Currently, most MS-LiDAR data processing methods refer to those of conventional single-channel LiDAR. In terms of input data format, the state-of-the-art methods can be grouped into two-dimensional (2D) feature image-based and 3D point cloud-based methods. Because an MS-LiDAR system with several spectral channels provides the corresponding individual point cloud for each band, the volume of MS-LiDAR data is considerable. Rasterizing voluminous MS-LiDAR point clouds into 2D feature images (height-derived and spectral information) is an efficient and effective mean. However, this conversion leads to a certain loss of spatial information, degrading the data processing quality. With the development of computational hardware and techniques, direct processing of MS-LiDAR points of large-scale scenes has drawn significant attention. Derived from LiDAR point clouds, some features, such as

* Corresponding author.
E-mail address: guanhy.nj@nuist.edu.cn (H. Guan).

https://doi.org/10.1016/j.jag.2021.102634
Received 19 July 2021; Received in revised form 18 November 2021; Accepted 19 November 2021

height, echo, point density, and spectral band, have been investigated by various feature descriptors and representation methods. These features are then selected according to specific objects of interest and analyzed by various classfiers, such as Random Forests (Matikainen et al., 2020), Support Vector Machine (Zhang et al., 2013), and Conditional Random Field (Niemeyer et al., 2011). Such feature engineering might be successful in some circumstances, for example, in small or unsophisticated scenes. However, for large-scale, complicated scenes, it is a tedious task to select the optimal features. Furthermore, the object recognition and classification accuracies might be unreliable due to the subjectivity of feature calculation and selection.

Since 2008, a variety of deep learning models gained much attention in 2D image classification and recognition tasks (Chen et al., 2018). Recently, an increasing number of scholars have extended deep learning models from 2D images to 3D point clouds. Initially, 3D point cloud classification or object recognition tasks were performed on 2D feature images converted from 3D point clouds by the established deep learning methods, mainly developed for 2D image processing (Felix et al. 2017). As aforementioned, data conversion leads to the loss of spatial information. Based on the encoder-decoder framework in deep learning, PointNet, a pioneering method for directly processing unstructured point clouds, employed multilayer perceptron (MLP) to extract features individually for each point and gave predicted label in an end-to-end manner (Qi et al., 2017a). PointNet++, proposed in the same year, encoded multiscale features through multi-scale grouping and complemented local features by neighborhood information. Afterward, networks improving on PointNet, such as PointCNN (Winiwarter et al., 2019), have been increasingly developed and popularized for point-wise point cloud classification and object recognition. Although inefficient at handling massive data, PointNet addresses the permutation and rotation invariance of point clouds, which provides a basic framework for MS-LiDAR point cloud classification.

This study explores the feasibility of airborne MS-LiDAR data in scene classification. We propose a feature reasoning-based graph convolution network (FR-GCNet) framework, which considers the relative distribution of neighborhood points and enhances the feature expression. Before the implementation of the FR-GCNet, a preprocessing procedure is first applied to the raw MS-LiDAR data, including data fusion and sample generation. Next, to extract locally and globally representative features of the class of interest, a feature reasoning module is introduced into the graph convolution network. The FR-GCNet framework is trained in an end-to-end manner and directly predicts semantic labels of each point. The main contributions of this paper are as follows:

1) The FR-GCNet framework, proposed for MS-LiDAR point cloud classification, is an improved PointNet embedded with a feature reasoning module and implemented directly on the irregular, unstructured 3D points, resulting in no loss of any data or object information.
2) A sampling strategy for sample generation is implemented by a combination of the farthest point sampling (FPS) and the k-nearest neighbors (KNN), which quickly obtains training samples that meet the input requirements of the FR-GCNet framework and provide complete coverage of the input scenes.
3) A feature reasoning module is proposed for feature enhancement. This module consists of two units: a local reasoning unit for extracting local features by a KNN graph structure and a global reasoning unit for extracting global features, strengthening feature representation, and further improving classification results.

2. Related work

MS-LiDAR point clouds, which include both multi-channel spectral information and geometrical structure information of the objects of interest, have increasingly attracted attention in the field of photogrammetry and remote sensing. Current studies on MS-LiDAR point clouds have proved to be promising in many applications. Moreover, Ekhtari et al. (2018) demonstrated that the classification performance of MS-LiDAR data outperformed those of conventional single-channel LiDAR data. In terms of input data type, the methods are then grouped into two categories: rasterized 2D feature image-based and 3D point cloud-based methods.

The former applied classic image processing algorithms, such as maximum likelihood, support vector machine (Ekhtari et al., 2018), and random forest (Matikainen et al., 2020; Ghaseminik et al., 2021), to the 2D feature images converted from 3D point clouds with regard to elevation and reflectance information. Specifically, employing multi-channel reflectances of MS-LiDAR data, Wallace et al. (2012) derived the Normalized Differential Vegetation Index (NDVI) and Photochemical Reflectance Index (PRI) to recover forest structural and biochemical parameters. A variety of features, such as texture, elevation, spectral, vegetation indices, and water indices, were derived from MS-LiDAR data and selected for object-based (Bakula et al., 2016) or pixel-based classification and object recognition (Lindberg et al. 2021; Bakula et al., 2016; Chen et al., 2020). However, these conventional image processing methods or classifiers achieved unreliable classification or object recognition accuracies due to handcrafted feature selection. Deep learning methods, increasingly explored and used in point cloud classification, automatically learn advanced and representative features from many representative training data. Pan et al. (2019) compared a deep Boltzmann machine (DBM) with two machine learning methods (i.e., RF and SVM) and demonstrated that the DBM (for high-level feature representation) outperformed the RF and principal component analysis (PCA) (for low-level feature extraction and selection) in MS-LiDAR land-cover classification. Yu et al. (2020 and 2021) applied a capsule network integrated with feature enhancement modules and the capsule-based attention modules to the Titan MS-LiDAR data for land-cover classification, respectively.

Although 2D feature image-based methods effectively and efficiently processed large-scale MS-LiDAR data, data conversion from 3D point clouds brings quantization errors and spatial information loss. Thus, directly processing 3D multispectral point clouds becomes much attractive and also made great progress. Specifically, Mosry et al. (2017) first separated non-ground from ground points and then classified the non-ground points into building and tree points and the ground points into road and grass points. Sun et al. (2017) explored and classified MS-LiDAR data into different targets, including fresh and sere plants. Furthermore, to demonstrate the distinguishing capabilities of MS-LiDAR systems, Ekhtari et al. (2017) provided finer discrimination for ten classes (3 types of asphalt pavements, 2 types of soils, and 2 types of rooftop materials). Some studies have proved that 3D multispectral point cloud-based methods are superior to the 2D multispectral feature images-based methods with 10% higher accuracy in (Miller et al., 2016) and 3.8% in (Morsy et al. 2017). As mentioned in the review of 2D feature image-based methods, it is tedious to design handcrafted features to obtain satisfactory classification results. Some studies have applied state-of-the-art deep learning methods to MS-LiDAR data for object recognition and point cloud classification. Jing et al. (2021) proposed an SE-PointNet++ architecture, where a squeeze-and-excitation block was embedded in the PointNet++ architecture, for MS-LiDAR point classification with an overall accuracy, mean Intersection over Union (mIoU), and F_1-score of 91.16%, 60.15%, and 73.14%, respectively. Li et al. (2020) proposed a hierarchical architecture equipped with graph geometric moments convolution to extract buildings from Titan MS-LiDAR data with a correctness of 95.1%, a completeness of 93.7%, an F_1-score of 94.4%, and an intersection over union (IoU) of 89.5%, respectively. However, there are still few deep learning studies on MS-LiDAR point cloud classification.

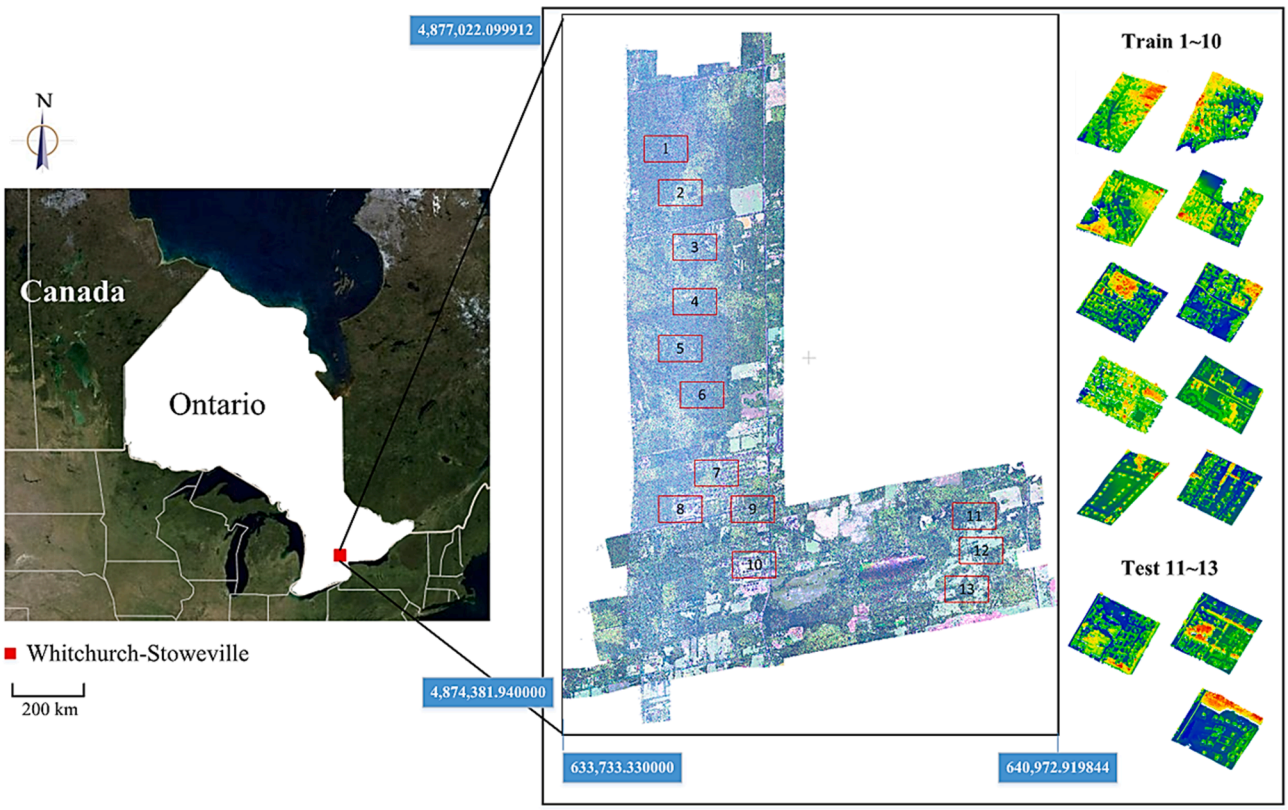


Fig. 1. Study area and MS-LiDAR data.

Table 1
Number of points of each class in the training and test datasets (# represents the number of points).

Class	Training dataset/ (Train 1-10) (#)	Test dataset/ (Test 11-13) (#)
Road	596,987	258,942
Building	415,763	116,085
Grass	2,577,194	983,724
Tree	2,623,317	863,519
Soil	124,566	33,214
Powerline	12,155	7,545

3. Study area and data

The study area is within a typical Canadian town (longitude 79°15′00″, latitude 43°58′00″ at the center), located in Whitchurch-Stowell, Ontario, Canada, which covers dense vegetation (trees and grass), roads, soil, and residential houses. In this area, several powerlines are overhung through the field. The study area also contains two large water regions, Musselman Lake and Windsor Lake, as seen from the high-resolution satellite image in Fig. 1. The elevation of this study area ranges from 250 m to 325 m.

The experimental data were acquired on July 2, 2015, by a Teledyne Optech Titan MS-LiDAR system, containing three channels with wavelengths of 1,550 nm (MIR), 1,064 nm (NIR), and 532 nm (Green), each of which produced point cloud with a laser pulse repetition rate of 300 kHz. The average flying height was about 1000 m − 1100 m. The three channels were arranged with an interval of 3.5° in deflection angle and the average point density for each channel was about 3.6 points/m^2. This data collection covered an area of 2,052 m × 1,566 m with 19 stripes perpendicular to each other.

In this study, we selected a total of 13 sample scenes from the obtained MS-LiDAR data, as shown in Fig. 1. Scenes 1 to 10 (Train 1 to 10) were selected as the training dataset and scenes 11 to 13 (Test11 to 13) as the test dataset to evaluate our proposed framework. Table 1 shows the number of points for each land cover. According to the integrated investigation of the collected LiDAR data associated with high-resolution satellite images, we mainly considered six classes in this classification task, i.e., road, building, grass, tree, soil, and powerline. According to our previous study (Pan et al., 2020), this study area contains only two large water bodies, which degraded the overall classification accuracies due to imbalanced class distributions. Therefore, we did not include water class in this classification task. Correspondingly, no water points were included in the selected thirteen scenes. As shown in Table 1, the road, building, grass, and tree classes contain a sufficient number of points in both training and test datasets, while the powerline class contains only a tenth of the number of soil points.

4. Method

4.1. Data preprocessing

The data preprocessing consists of data fusion and sample generation. Data fusion aims to merge the three independent point clouds from the three channels of Green, NIR, and MIR into a single point cloud, with each point containing three-channel reflectances. As presented in previous studies (Morsy et al.,2017; Pan et al., 2020), data fusion is realized in the following three steps: (1) Each channel point cloud is in turn taken as the reference data, where each point is processed to find its neighboring points in the other two channel point clouds using a nearest neighbor searching algorithm. The maximum search distance is set to 1.0 m according to the average point density of 3.6 points/m^2; (2) for each point in the reference data, the intensities of the other two channels are obtained by an inverse-distance-weighted interpolation method. If there were no neighboring points in the one of two channels, the intensity value is set to zero; (3) finally, the processed points of all channels are merged, and the duplicate points are deleted to obtain a single point cloud containing the coordinates and three-wavelength intensity

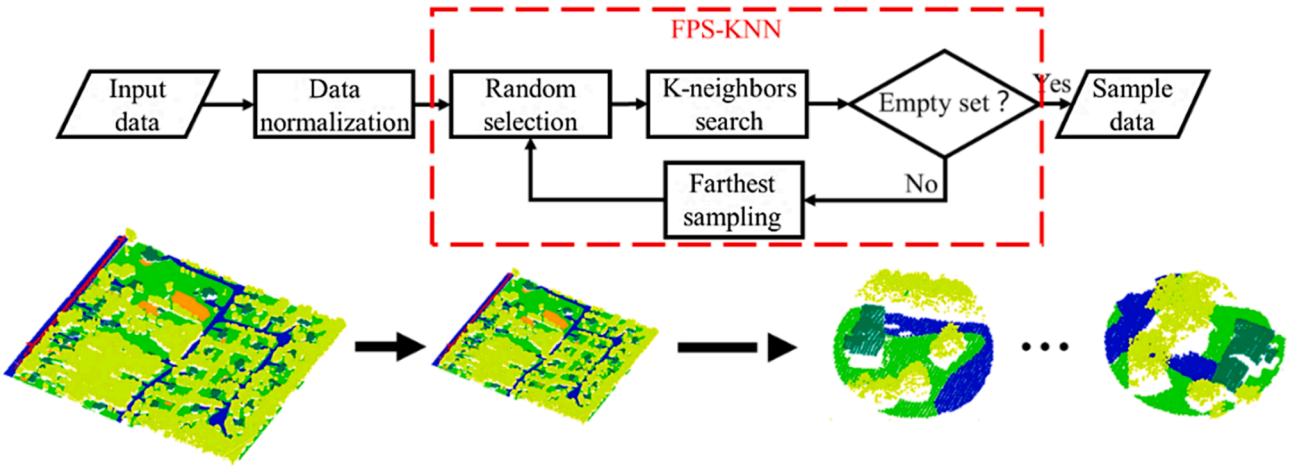


Fig. 2. Schematic diagram of the FPS-KNN sampling strategy.

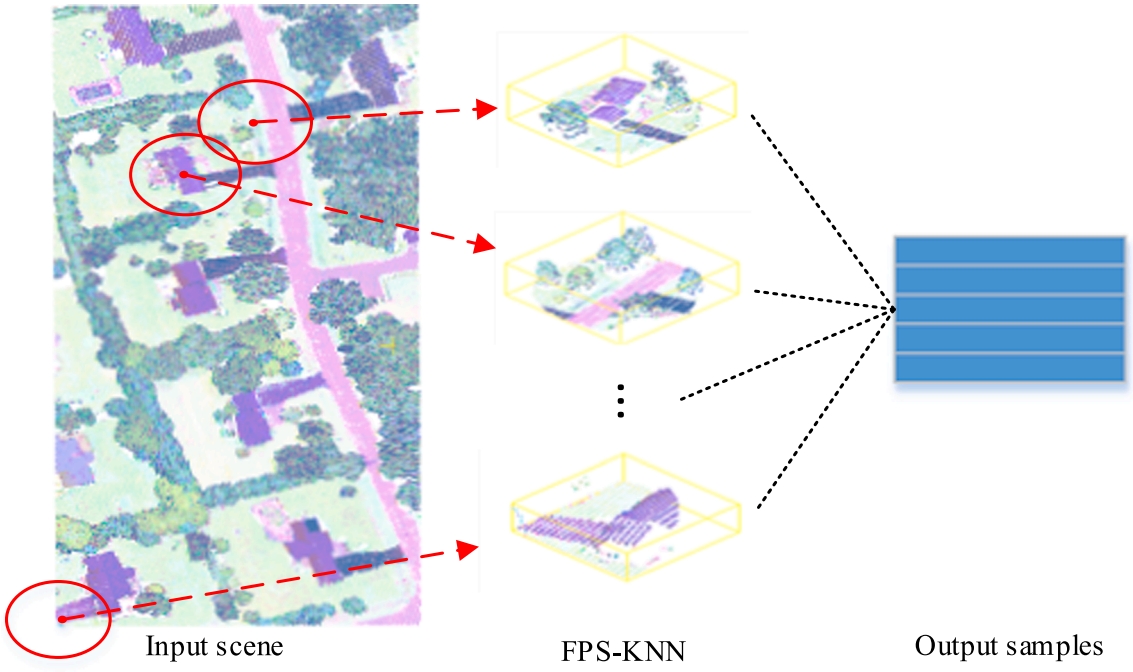


Fig. 3. Illustration of the proposed FPS-KNN sampling strategy.

values. Note that the Titan MS-LiDAR data contains no system metadata and trajectory data. Thus, absolute radiometric and geometric corrections were not implemented in our study. Moreover, to collect ground truth, we manually labeled each point in the selected scenes and assigned corresponding digital label according to the specific class. Thus, each point contains seven attributes, including 3D coordinates, three wavelengths, and labelled class number.

Sample generation aims to obtain the required sample data that are directly input to the FR-GCNet framework. Due to GPU memory limitations, it is impossible to directly input all points of a large-scale scene into the framework. Moreover, deep learning-based point cloud methods require a fixed number of points as the input data. Therefore, we proposed a sampling strategy, a combination of farthest point sampling (FPS) and *k*-nearest neighbors (KNN) methods, termed as FPS-KNN, to quickly obtain training samples with both the correct number of points and a full coverage of the scene. Fig. 2 illustrates the diagram of the FPS-KNN sampling strategy.

Data normalization. The input data are firstly normalized to improve the convergence speed of network optimization. In this study, to reduce structural distortions of the MS-LiDAR data and ensure the FR-GCNet learns accurate geometric features, an isotropic normalization method is adopted to normalize the coordinates of all points into a range of [-1,1]. The data normalization is implemented as follows.

$$(X_m, Y_m, Z_m) = \left(\frac{\max(X_{in})+\min(X_{in})}{2}, \frac{\max(Y_{in})+\min(Y_{in})}{2}, \frac{\max(Z_{in})+\min(Z_{in})}{2}\right)$$

$$\delta = \max\{X_{in} - X_m, Y_{in} - Y_m, Z_{in} - Z_m\}$$

$$(X_{out}, Y_{out}, Z_{out}) = \frac{(X_{in} - X_m, Y_{in} - Y_m, Z_{in} - Z_m)}{\delta} \quad (1)$$

where (X_{in}, Y_{in}, Z_{in}) denote the coordinates of a raw point cloud. (X_m, Y_m, Z_m) represents the mean values of max and min coordinates. δ refers to a scaling factor. (X_{out}, Y_{out}, Z_{out}) are the coordinates of the normalized point cloud. The intensities normalization is also performed on the three spectral channels of the MS-LiDAR data, which is also realized by

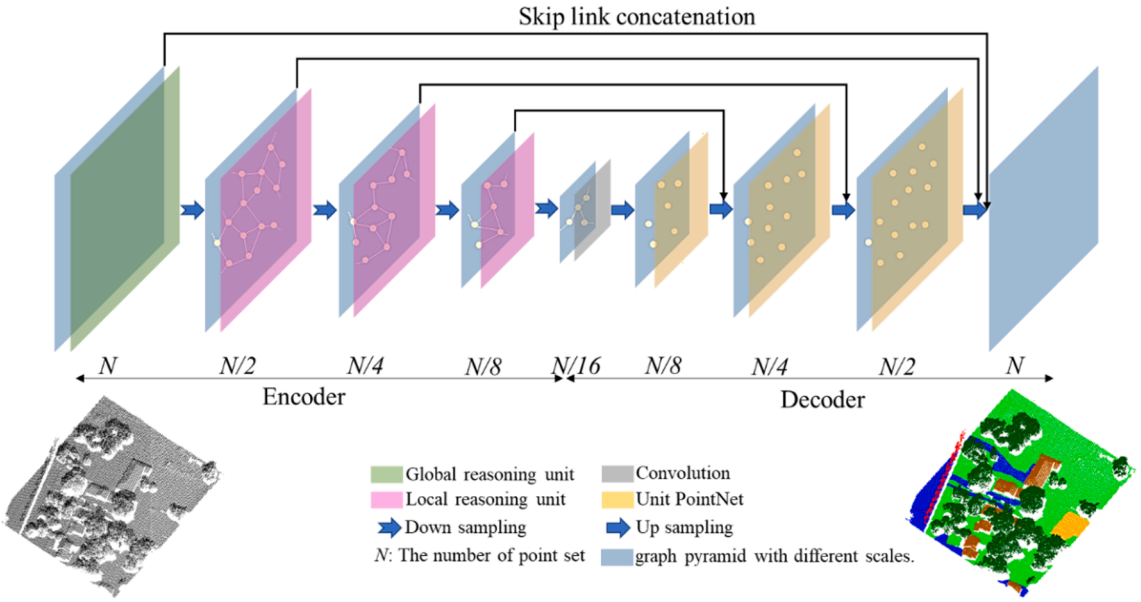


Fig. 4. Illustration of the proposed FR-GCNet network.

equation (1).

FPS-KNN sampling. Fig. 3 shows the proposed FPS-KNN sampling strategy. FPS has been actively used for point sampling selection due to its robustness to point clouds with different point densities and its full scene coverage (Moenning and Dodgson, 2003; Qi et al., 2017b). However, the FPS loses some point-wise spatial relations during down-sampling. KNN is adopted to obtain a fixed number of neighboring points of the FPS points to address this issue. The specific steps of the FPS-KNN sampling strategy are detailed as follows:

1) For an input scene, one point is randomly selected as the seed point, and its k nearest neighboring points are then searched from the point cloud to form one sample. The vaule of k is determined by the size of samples. In this paper, we set the k vaule to 4095. Correspondingly, each sample contains 4096 points, and then are removed from the input.
2) The distances from the seed point to the remaining points in the reference point cloud are calculated. The farthest point is specified as the next seed point. This seed point and its k nearest neighboring

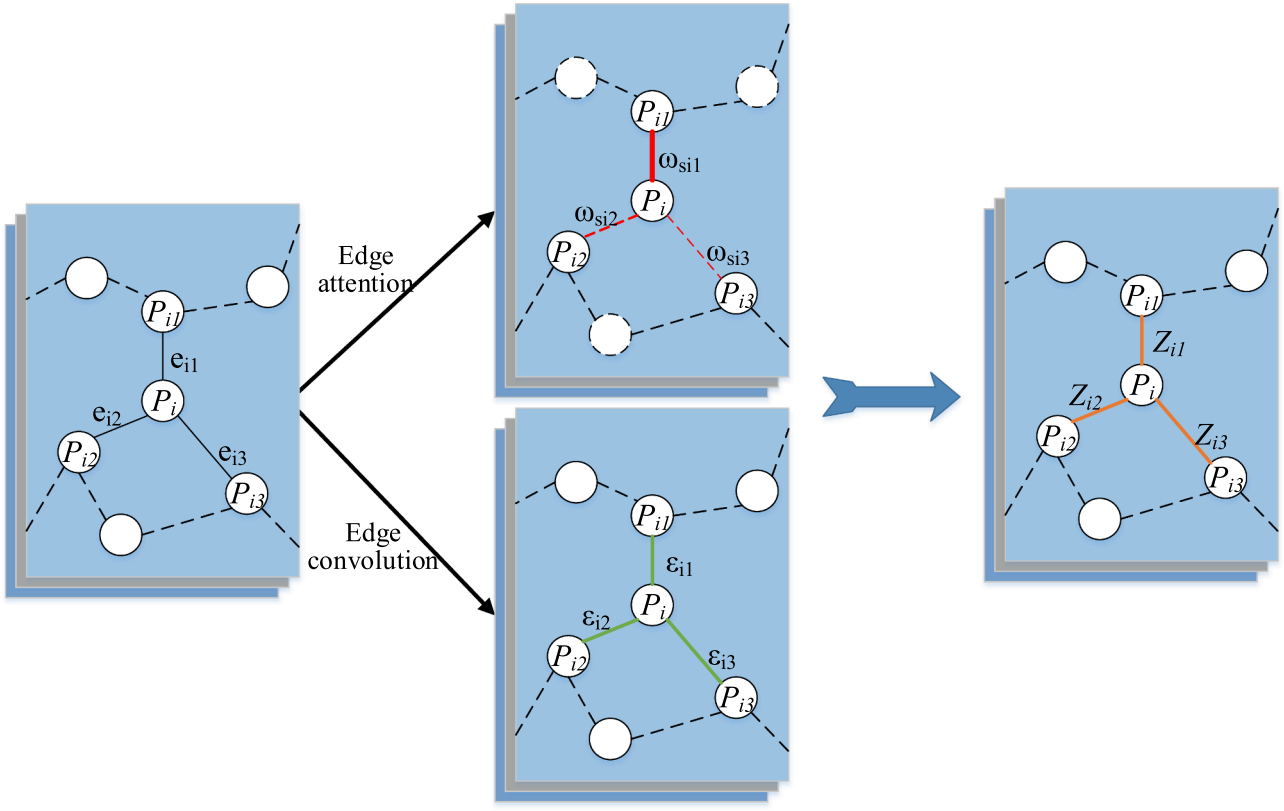


Fig. 5. Illustration of local reasoning unit, including edge attention and edge convolution.

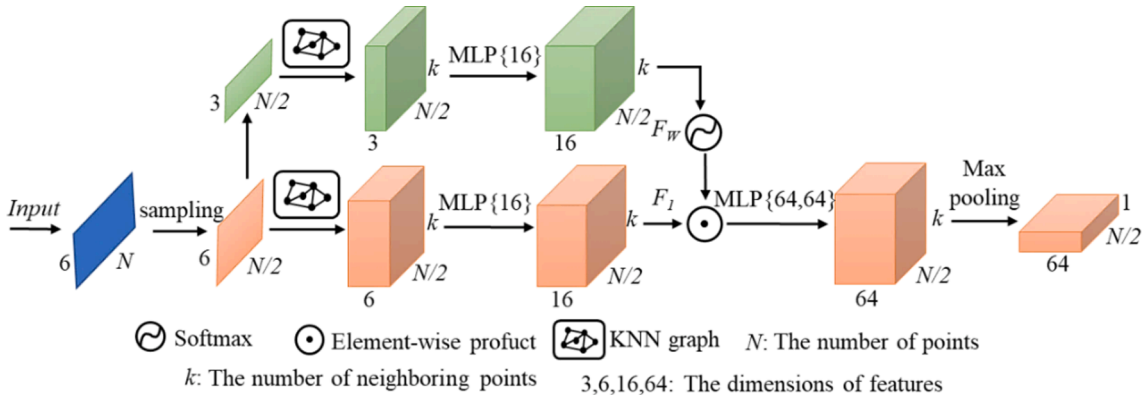


Fig. 6. Illustration of local reasoning.

points then form another sample, and also are removed from the point cloud.

3) Step 2 is iteratively performed until all samples contain the input scene. These ouput samples with fixed number of points are directly input into our FR-GCNet framework. Note that, for the points within the overlapped parts, we choose the most predicted label as its final classified label.

4.2. FR-GCNet framework

The architecture of the proposed FR-GCNet is illustrated in Fig. 4. It consists of one encoder, one decoder, and a set of skip link concatenations. The FR-GCNet directly takes the training samples as the input and outputs point-wise classification results. The encoder, implemented by four down-sampling layers, one global reasoning unit, and three local reasoning unit, aims to extract different contextual features at different levels and learn multi-dimensional statistical features. The decoder, composed by four up-sampling layers and three "PointNet unit" operations, concentrates on the representative features for specific objects and gradually up-sample the points.

The training sample set generated by the FPS-KNN sampling strategy is input to the RF-GCNet network. In the encoder, the contextual features of all input points are encoded by a global reasoning unit. Then, four successive down-sampling layers gradually reduce the size of the point set to *N/2*, *N/4*, *N/8*, and *N/16*. The new subsampled points generate a different graph pyramids and feature dimension in each layer. The local reasoning unit is then used three times to extract multiscale local features dynamically. In the decoder, four up-sampling layers and three "Unit PointNet" operations (Qi et al., 2017b) propagate the learned features from the compressed points of the encoder to the input point set. The up-sampling operations are achieved by an inverse distance weighting (IDW) interpolation, and the "Unit PointNet" operations are performed by a 1 × 1 CNN convolution. The number of the points corresponding to the interpolated features are gradually up-sampled to *N/8*, *N/4*, *N/2*, and *N*. To enhance the feature integrity and avoid the vanishing gradient, three skip link concatenations are used to integrate the encoded features from the down-sampling layers with the high-level features from the up-sampling layers. Moreover, the interpolated features are then connected to the global feature by an additional skip link concatenation, and all features are fed into a 1 × 1 fully connected layer to predict the categorical label of each point.

4.3. Feature reasoning module

4.3.1. Local reasoning unit

1) Graph convolution with edge attention weights

Let denote $\boldsymbol{P} = \{p_1, p_2, \ldots, p_N\} \in \mathrm{R}^{N\times(3+C)}$ a point set, where N is the number of points in the point set $\boldsymbol{P}$, 3 and C represent the dimensions of the spatial coordinates and multiple spectral information, respectively. A neighborhood graph structure, $G(\boldsymbol{V}, \boldsymbol{E})$, is built on the input point set $\boldsymbol{P}$, where $\boldsymbol{V} = \{v_1, v_2, \ldots, v_N\}$ indicates the set of vertices and $\boldsymbol{E}$ indicates the set of edges.

As shown in Fig. 5, to explicitly exploit the local structural information of the MS-LiDAR point clouds, the local reasoning unit, including edge attention and edge convolution, is added to the graph convolution neural network. According to the spatial dependence of point pairs, edge attention strengthens important features to obtain more feature details of the objects and suppresses irrelevant information. Edge convolution aims to dynamically learn high-dimensional local features from the weighted edges of the neighborhood graph, G. The specific implementation of these two processes is as follows.

Edge attention. Specifically, in the neighborhood graph, as shown in Fig. 5, vertex p_i is the central node, p_{ij} (j = 1,2,3) are its nearest neighboring points, and e_{ij} (j = 1,2,3) are the edges between the central node and its nearest neighbors. Let denote s_{ij} as the edge weight of e_{ij}, which is defined by $s_{ij} = (p_i - p_{ij})$. To calculate the attention weights of all edges, a linear transformation function, F_w, is defined by

$$F_w(s_{ij}) = \sum_{j=1}^{3} W_{sij} * s_{ij} + b_i \tag{2}$$

where, W_{sij} is the connection coefficient, and b_i is the bias parameter.

To unify the neighborhood attention coefficients across the nodes and highlight the weights of important elements in the whole graph structure, a softmax function is used to normalize the edge attention weights from all neighbors connected to the reference node, which can be defined by

$$\omega_{sij} = \frac{\exp(F_w(s_{ij}))}{\sum_{j=1}^{3}\exp(F_w(s_{ij}))} \tag{3}$$

where ω_{sij} refers to the normalized edge attention weight. As shown in Fig. 5, the edge attention weight of a point pair in the graph structure is represented by the boldness of the red line, and the lighter relations may be shielded.

Edge convolution. To compute the edge features corresponding to the defined edges, we denote the edge function as $\varepsilon_{ij} = f_\sigma(p_i, p_{ij})$, where f_σ: $\mathrm{R}^{3+C} \times \mathrm{R}^{3+C} \rightarrow \mathrm{R}^{C'}$. The features R^{3+C} of center node p_i concatenated with the features R^{3+C} of its neighbors p_{ij} are taken as the function input to map the edge features, which can be defined by:

$$\varepsilon_{ij} = f_\sigma(p_i, p_{ij}) = f_\sigma'(p_i, p_i - p_{ij}) \tag{4}$$

The obtained edge features combine the global structure composed of all the vertexes captured by the center p_i, and the local shape information in each graph, captured by $p_i - p_{ij}$. In detail, the operation is defined as follows:

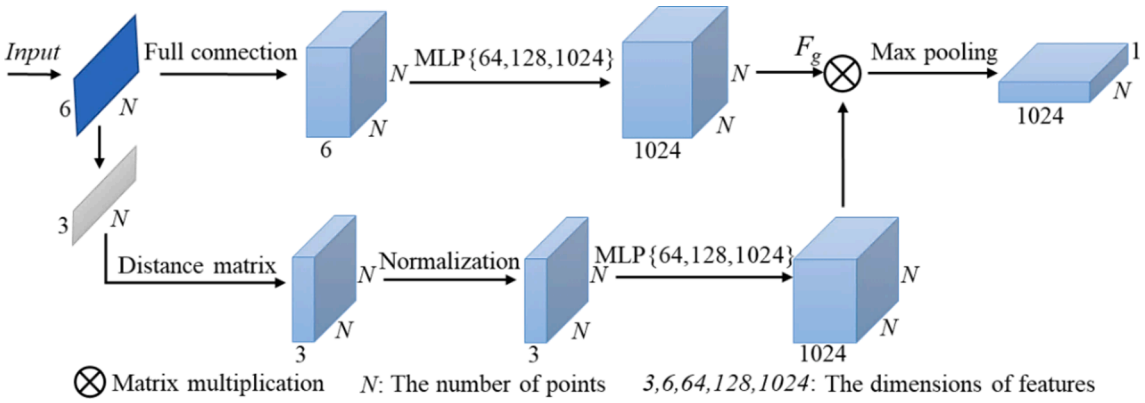


Fig. 7. Illustration of global reasoning.

$$\varepsilon_{ijm} = \mathrm{ReLU}\left(\theta_m \bullet \left(\mathrm{p_i} - \mathrm{p_{ij}}\right) + \varphi_m \bullet \mathrm{p_i}\right) \tag{5}$$

where, $\sigma = (\theta_m, \varphi_m)$ is a set of learnable parameters, which encode the weight coefficients, and θ_m and φ_m have the same dimensionality as p_i. ReLU is the rectified linear unit activation function.

The weighted edge features labeled by orange in Fig. 5, Z_{ij} ($j = 1,2,3$), are obtained by assigning the edge attention weights and aggregating all the weighted edge features Z_{ij} to capture local features in the current neighborhood graph using a channel-wise symmetric function. The edge feature calculation can be expressed by:

$$F^E_{local} = \max_{j:(i,j)\in E} \varepsilon_{ijm} \bullet \omega_{\mathrm{sij}} \tag{6}$$

2) Local reasoning

Fig. 6 shows the local feature reasoning unit embedded into our FR-GCNet network. Firstly, a sampling layer is applied for down-sampling the $N \times 6$ input data to $N/2 \times 6$ subsamples via the FPS. Then, for the edge attention (rendered by green), while only considering the spatial information, a $N/2 \times k \times 3$ graph is built from the $N/2 \times 3$ subsamples by searching for the k neighboring points around each central node. The edge attention is implemented by a shared MLP (16). For the edge convolution (rendered by orange), a $N/2 \times k \times 6$ graph is built considering the spatial and spectral information of the neighboring points. A shared MLP layer (16) is employed to obtain $N/2 \times k \times 16$ edge features. An element-wise product between the edge attention weights and the corresponding $N/2 \times k \times 16$ edge features is acquired to produce a feature map F1. Finally, to capture deeper feature information, the feature map F1 is then input to another two MLP layers (64 and 64). A max-pooling layer is used to obtain the $N/2 \times 64$ local features by aggregating the $N/2 \times k \times 64$ features.

4.3.2. Global reasoning unit

Most previous deep learning studies paid much attention to the extraction of local features. Although PointNet++, the baseline of our framework, obtains global information by accumulating all local features via a symmetric function, the spatial relationships of all points in a point cloud are still neglected. To further enhance contextual information, a global reasoning module is introduced to obtain global attention weights from the input point clouds.

As shown in Fig. 7, based on the spatial coordinates of the training dataset, we first build an $N \times N \times 3$ distance tensor, D, each of whose elements is the Euclidean separation between every two individual points in each coordinate direction. $D_{ij} = [\, p_{xi} - p_{xj}, p_{yi} - p_{yj}, p_{zi} - p_{zj} \,]$, the point-wise separation between points p_i and p_j, is normalized by a softmax function as follows:

Table 2
The network hyper-parameters.

Hyper-parameters	
Epoch	200
Batch size	4
Optimizer	Adam
Learning rate	initial rate 0.001 divided by 3 every 2,000 steps
Loss function	Cross-entropy
Dropout rate	0.55

$$d_{ij} = \frac{\exp(D_{ij})}{\sum_{j=1}^{N} \exp(D_{ij})} \tag{7}$$

where d_{ij} denotes the normalized distance.

The final global attention weights, g_{ij}, can be obtained by applying three shared MLP layers (64,128, and 1024) as nonlinear function to the normalized distance. Then, an $N \times N \times 6$ graph, built by the full connections among the input point pairs, are input to three shared MLP layers (64, 128, and 1024) to obtain an $N \times N \times 1024$ high-dimensional feature, F_{ij}.

The global feature map, $Fg \in \mathrm{R}^{N\times N\times 1024}$, is generated according to the matrix multiplication between the high-dimensional features and the global attention weights. Finally, a max-pooling function is employed to obtain the global contextual feature, F^E_{global}, which is given by

$$F^E_{global} = \max_{j:(i,j)\in E} F_{ij} \bullet \mathrm{g_{ij}} \tag{8}$$

4.4. Implementation details

The proposed FR-GCNet was implemented based on the Pytorch1.2 framework (Mccaffrey, 2019). All experiments in our study were performed on a workstation with a 6-GB NVIDIA GeForce RTX 2060 GPU, an Intel Core i7-9700 CPU, and a 16-GB RAM. According to the density of the Titan MS-LiDAR point set, when constructing graphs, the number of neighbors, k, was set to 32. During the training, the hyper-parameters are presented in Table 2. To quantitatively evaluate the performance of the proposed FR-GCNet, we adopted the five commonly used metrics, including *overall accuracy* (*OA*), *precision*, *recall*, *mean Intersection over Union* (*mIOU*), and F_1-*score*.

5. Results and discussion

5.1. Classification results

To demonstrate the MS-LiDAR point classification performance of the FR-GCNet, we applied it to the three test datasets (i.e., Test11 to 13).

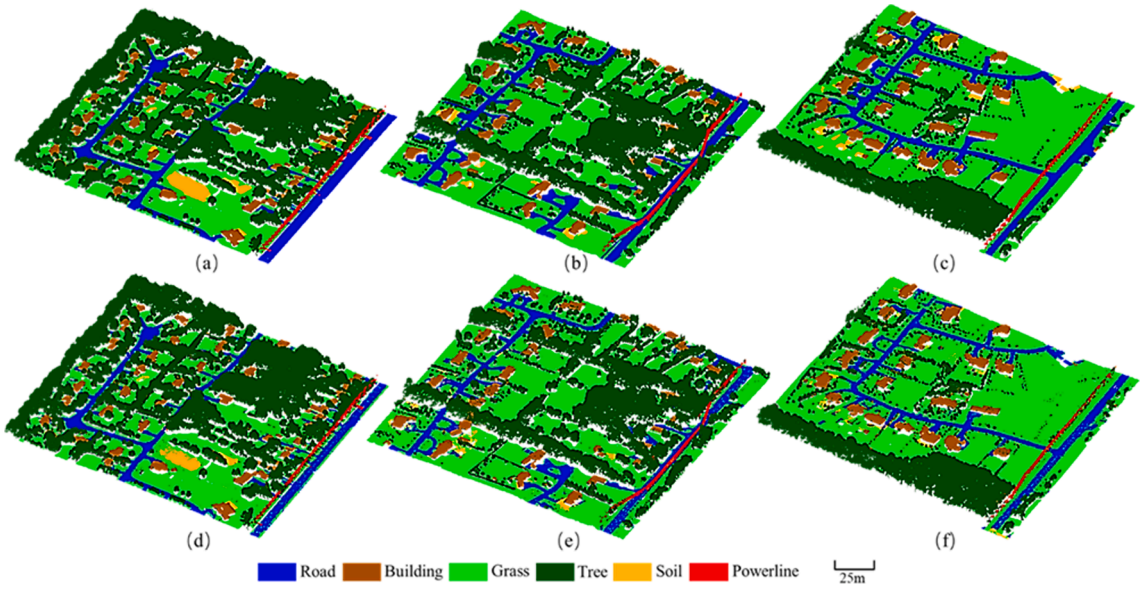


Fig. 8. Visual results of the FR-GCNet on the three test scenes. (a) ground truth of Test11, (b) ground truth of Test12, (c) ground truth of Test13, (d) predicted result of Test11, (e) predicted result of Test12, and (f) predicted result of Test13.

Table 3
Confusion matrix and assessment metrics of the the proposed FR-GCNet. The rows and columns represent the predicted points and reference points of each class, respectively.

Classes	road (#)	building (#)	grass (#)	tree(#)	soil (#)	powerline (#)
Road	165,819	21	27,522	57	5,634	0
Building	192	96,954	195	6,024	117	6
Grass	21,345	198	738,111	2,562	9,771	0
Tree	237	12,717	4,614	671,769	174	228
Soil	6,114	36	14,709	39	7,737	0
powerline	0	114	0	3,966	0	6174
Precision (%)	85.60	88.11	94.01	98.15	33.02	96.35
Recall (%)	83.30	93.69	95.61	97.39	27.02	60.21
F_1-*score* (%)	84.43	90.81	94.80	97.77	29.72	74.11

Fig. 8 shows the FR-GCNet classification results and the ground truth (GT) on the three test scenes. As shown in Fig. 8, compared with the GT, we intuitively observed that the FR-GCNet achieved satisfactory classification results for most classes, such as road, building, grass, tree, and powerline. However, some points belonging to classes soil and road with long and narrow shapes were misclassified as grass This was especially noticeable for the Test13 scene. Some powerline points sandwiched in the middle of trees were misclassified as tree, especially for the Test12 scene.

To quantitatively analyze the MS-LiDAR classification performance of the FR-GCNet, Table 3 reports the confusion matrix, *precision*, *recall*, and F_1-*score* of each class. Our FR-GCNet achieved the F_1-*score* of over 70% for the road, building, grass, tree, and powerline classes, the highest F_1-*score* of 97.77% for tree, and the lowest F_1-*score* of 29.72% for soil. Although the powerline class, a special corridor-like structure, is underrepresented in the dataset, the FR-GCNet achieved the F_1-*score* of 74.11%. As shown in Table 3, the use of LiDAR elevation information easily distinguishes classes with different elevations. For instance, only 21 and 57 road points were misclassified as building and tree,

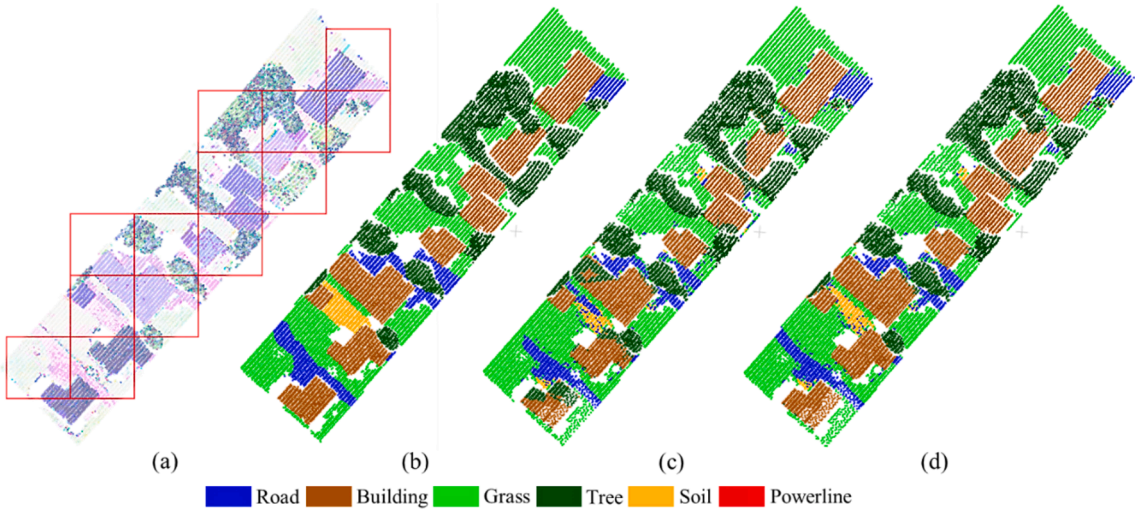


Fig. 9. Visual comparison of the FR-GCNet training with the sampling methods: (a) Illustration of the block sampling method, (b) GT, (c) classification results using the block sampling strategy, and (d) classification results using the FPS-KNN sampling strategy.

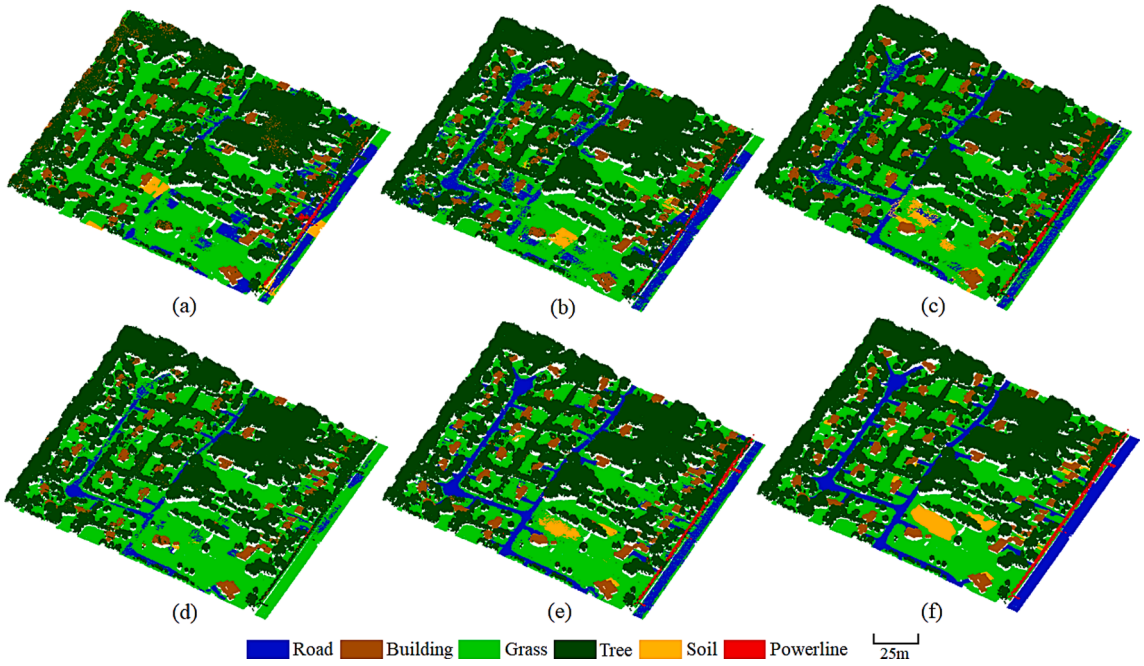


Fig. 10. Visual results of the FR-GCNet with different input data. (a) Case-1, (b) Case-2, (c) Case-3, (d) Case-4, (e) Case-5, and (f) GT.

respectively. Similarly, 192 building points were misclassified as road, 195 as grass, and 117 as tree. However, the soil class obtained a poor identification performance since one-half of the soil points were incorrectly classified as grass, and one fourth as road.

5.2. Sampling performance

To verify the performance of the FPS-KNN sampling strategy in data preprocessing, we compared it with the commonly-used block sampling method (Sheshappanavar and Kambhamettu, 2020) on the Test11 dataset. The training samples generated from the two sampling methods and their FR-GCNet classification results were compared. As shown in Fig. 9 (a), with a given grid size, the block sampling method split a point cloud into a series of blocks without overlapping, which ensured a full coverage of the scene. For a fair comparison, the grids with less than 4096 points were randomly up-sampled to 4096, and those with more than 4096 points were randomly down-sampled to 4096. In this study, the grid size was set to 0.12 × 0.12, which was determined by the ranges in the X- and Y- directions. Fig. 9 (b) shows a subset of the Test11, which mainly includes buildings and trees. Visual inspection demonstrates that the FR-GCNet trained with the samples generated by the block sampling method achieved unsatisfactory classification results. For example, many building roof points were misclassified as tree. The FR-GCNet trained with samples generated by the FPS-KNN method achieved better performance of visual classification. More specifically, as shown in Fig. 9 (c), most building points near the grid boundaries were incorrectly classified as tree, whereas, as shown in Fig. 9 (d), most building roof points were correctly classified.

To further analyze the two sampling strategies, the *OA* and average *F_1-score* were calculated as 92.04% and 73.90% for the block sampling strategy, and 93.55% and 78.61% for the FPS-KNN sampling strategy. Specifically, the FR-GCNet using the FPS-KNN sampling strategy achieved an improvement of 1.5 % on the *OA* and 4.5% on the average *F_1-score*, which is consistent with the visual inspection. This phenomenon is because the block sampling strategy damages the integrity of the input scenes, such as the geometrical structure completeness of a building; thereby the network trained by this sampling strategy struggles to accurately label all the points. Conversely, the FPS-KNN sampling strategy can maintain the objects' completeness and effectively generate the samples covering all the scenes.

Table 4
The quantitative classification results of the FR-GCNet with different input data.

Experiment No.	*OA* (%)	average *Precision* (%)	average *Recall* (%)	average *F_1-score* (%)	*mIOU* (%)
Case-1	76.81	57.86	53.83	53.75	49.65
Case-2	84.74	69.53	63.22	65.53	59.80
Case-3	89.97	78.21	73.54	75.59	64.52
Case-4	84.38	65.10	57.47	61.52	55.50
Case-5	**93.55**	**82.54**	**76.20**	**78.61**	**65.78**

5.3. Input features

To clearly demonstrate the superiority of the MS-LiDAR data in point cloud classification, sample data with different input spectral features were input to the FR-GCNet. We conducted a series of experiments on the Test11 scene with different input data: (1) only geometrical elevation data (Case-1), (2) geometrical elevation and MIR-channel spectral data (Case-2), (3) geometrical elevation and NIR-channel spectral data (Case-3), (4) geometrical elevation and Green-channel spectral data (Case-4), and (5) geometrical elevation and three-channel spectral data (Case-5). Fig. 10 shows the comparative point cloud classification results of the trained FR-GCNet on the Test11 scene. As shown in Fig. 10, Case-5 demonstrates the best classification performance against other input data types. Specifically, Case-1 with only the geometrical input data achieved poor classification accuracies of the objects with similar elevation information. For example, there are many misclassification results that occurred between the soil and grass classes, as well as between the building and tree classes. Compared with Case-1, the remaining cases, with the input data integrating elevation with one- or multi-channel spectral data, achieved a significant point cloud classification performance improvement. Specifically, the aforementioned misclassification rates between the building and tree classes and the grass and soil classes were greatly reduced due to the spectral differences of these similarly elevated objects. Among the experiments of Case-2 to Case-4, we found that Case-4, the input data with the geometrical and Green-channel information, obtained the worst classification accuracies. Vegetation is poorly reflected in the green and

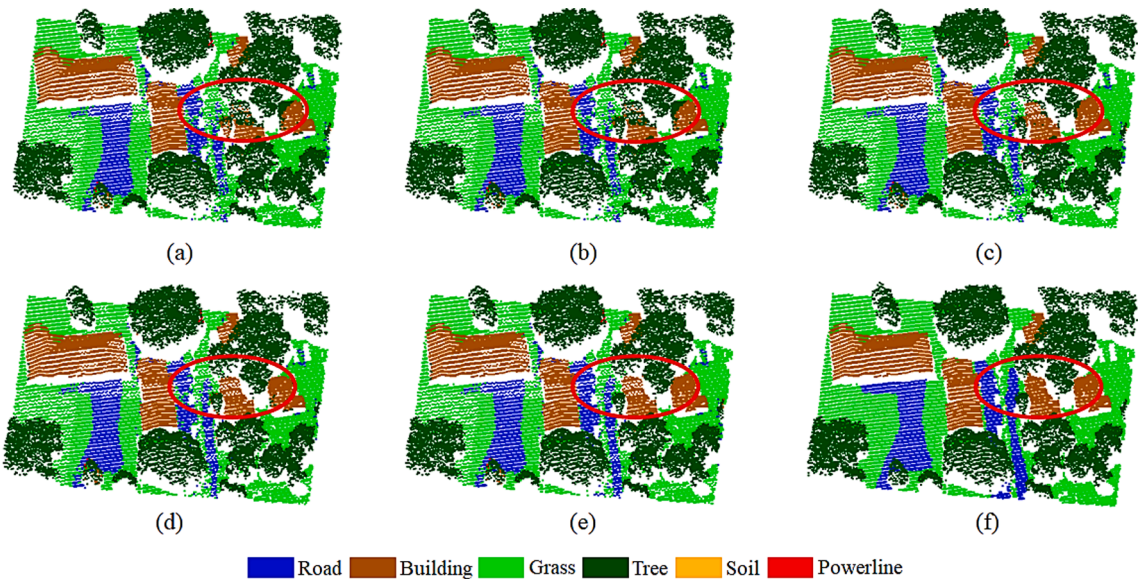


Fig. 11. A close view of the classification results obtained by (a) baseline, (b) FR-GCNet-LR, (c) FR-GCNet-EA, (d) FR-GCNet-GR, (e) FR-GCNet, and (f) GT.

appears darker, while asphalt roads also appear dark, which increases the misclassification rates between the road/soil and grass classes. Although Case-5 obtained the best classification accuracies by using all three-channel spectral information, the classification of the soil class was still problematic.

To further demonstrate the classification performance of the FR-GCNet with different input data, Table 4 lists the quantitative results. As shown in Table 4, Case-2 and Case-4 achieved point cloud classification improvement by increased *OA* of about 8%, average F_1-*score* of 12%, and *mIOU* of 10%. Case-3 achieved the better classification results with an *OA* of 89.97%, an average F_1-*score* of 75.59%, and a *mIOU* of 64.52% due to the 1064 nm-channel spectral data, which provides a good reflectance of vegetation. Case-5 achieved the best classification accuracies with an *OA* of 93.55%, an average F_1-*score* of 78.61%, and a *mIOU* of 65.78%, demonstrating that using three-channel spectral information contributes to the improvement of the MS-LiDAR point cloud classification.

5.4. Ablation study

A group of ablation experiments were designed to analyze the effectiveness of the reasoning module, including the global reasoning and local reasoning units, on the upgradation of the point cloud classification accuracy. To this end, we modified the proposed FR-GCNet to construct four networks. First, we removed the feature reasoning module from the FR-GCNet, and termed the resultant network as the baseline. Then, we removed the global reasoning unit and the local reasoning unit, respectively, from the FR-GCNet, and named the resultant networks as FR-GCNet-GR and FR-GCNet-LR, respectively. Finally, we removed edge attention from the local reasoning unit, and named the resultant network as FR-GCNet-EA. These modified networks were constructed by the same training and testing datasets, as well as the same training strategy.

As shown in Fig. 11, a close view of the Test11 scene demonstrated that the proposed FR-GCNet completely recognized the six classes of interest and classified them clearly. The other networks failed to classify some points of the six classes correctly. The baseline network failed to separate the building points from the tree points and completely distinguished the road points from the grass points. The FR-GCNet-GR and FR-GCNet networks correctly classified the points in the tree-and-building adjoining areas. This is because the use of the local reasoning unit improved the local neighboring spatial feature representation and learned the geometric deep features from spatial relations. The FR-GCNet-GR obtained a good identification of the tree and road points. However, without the global feature reasoning unit, some classes with long structures, such as road, were often misclassified as the others. Moreover, compared with the FR-GCNet-EA, the FR-GCNet-GR and FR-GCNet more accurately identified the tree points from the building points. This is because of the weight difference for different classes by the edge attention.

Table 5
Ablation study results.

Networks	*OA* (%)	average *Precision* (%)	average *Recall* (%)	average *F1-score* (%)	*mIOU* (%)
Baseline	89.49	75.68	67.63	71.72	60.14
FR-GCNet-LR	90.07	77.29	68.90	72.74	62.08
FR-GCNet-EA	91.89	78.11	74.27	75.60	64.32
FR-GCNet-GR	92.94	80.60	73.14	76.82	65.21
Our FR-GCNet	**93.55**	**82.54**	**76.20**	**78.61**	**65.78**

Table 5 quantitatively records the point cloud classification results obtained by the five networks. The proposed FR-GCNet obtained the best point cloud classification results with an *OA* of 93.55%, an average F_1-*score* of 78.61%, and an *mIOU* of 65.78%. The baseline network, without integrated with any units or modules, obtained an accuracy degradation of about 4%, 7%, and 5% with respect to the *OA*, average F_1-*score*, and *mIOU*, respectively, compared with the FR-GCNet. This performance degradation was mainly caused by the presence of similarly-elevated objects. To learn local features and enhance local expression ability, the FR-GCNet-GR, integrated with the local reasoning unit, obtained a performance improvement by about 3% on the OA, 5% on the average F_1-score, and 5% on the mIOU. Compared with the FR-GCNet-LR, the improvement of the FR-GCNet-EA indicates edge convolution collects local features by max-pooling edge features in local neighborhoods. Similarly, we also found that edge attention in the FR-GCNet took account for a large proportion of the performance improvement. This is because edge attention provides edge weights for edge convolution and distinguishes critical points in the overlapping areas such as trees and buildings. We verified that, without the reasoning feature module for comprehensively exploring local neighboring and global spatial

Table 6
The MLP size of the comparative networks.

Networks	Encoder	Decoder
PointNet	[64, 64] [64, 128, 1024]	[512,256,128]
PointNet++	[16,16,32] [32,32,64] [64,64,128] [64,96,128] [128,196,256] [128,196,256] [256,256,512] [256,384,512]	[512,512] [512,512] [256,256] [128,128]
SE-PointNet++	[32,32,64] [64,96,128] [128,196,256] [256,384,512]	[512,256] [256,256] [256,128] [128,64]
DGCNN	[64,64] [64, 64] [192,1024]	[512,256]
RSCNN	[64,16,128] [128,131,512]	[512,512,256,128,128]
GACNet	[32,32,64] [64,64,128] [128,128,256] [256,256,512]	[256,256] [256,128] [128,128,128]

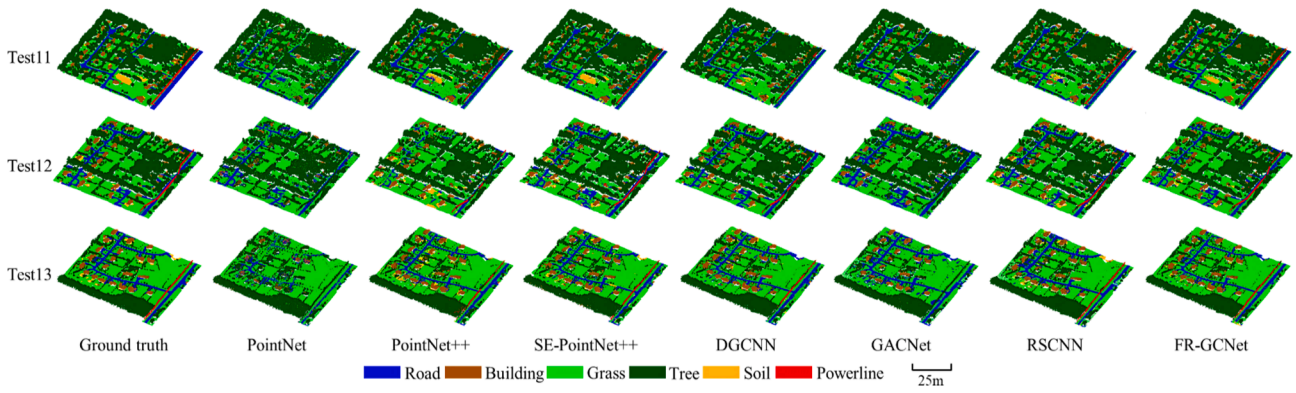


Fig. 12. Visual comparison of the classification results on test scenes.

structures associated with the six classes of interest, the quality of the output features used for classifying point clouds was weakened. Consequently, a degradation occurred in the MS-LiDAR point cloud classification performance of large-scale, complicated scenes.

5.5. Comparisons with other methods

A group of comparative experiments were conducted on the Titan MS-LiDAR point clouds to demonstrate the advantages of the FR-GCNet. So far, there are few deep learning-based methods developed for directly classifying MS-LiDAR point clouds. Therefore, to better estimate our network, five widely-used networks were selected for MS-LiDAR point cloud classification performance comparisons, i.e., PointNet (Qi et al., 2017a), PointNet++ (Qi et al., 2017b), DGCNN (Wang et al., 2019a), RSCNN (Liu et al., 2019), and GACNet (Wang et al., 2019b). Our previous study, SE-PointNet++ (Jing et al., 2021) based on PointNet++, was also selected for this comparison. In the above netwroks, encoder and decoder are specifically implemented by MLP which raises and lowers the feature dimensions. Table 6 shows the MLP sizes.

For a fair comparison, the same samples and hyper-paramters were used to construct these models and evaluate their performances. Fig. 12 shows the MS-LiDAR point classification results obtained by the different networks on all the test scenes. Taking Test11 scene as an example, visusal inspection indicates that the proposed FR-GCNet achieved superior overall classification performance by reducing misclassification and noise. Prominently, the FR-GCNet, improved by the PointNet, correctly predicted the soil and powerline points and clearly classified the similarly-elevated classes, such as tree and building, as well as grass and soil. As shown in Fig. 12, the performances of the DGCNN and GACNet were both poor in powerline point classification. Although the RSCNN achieved the classification accuracy comparable to that of the FR-GCNet, the former still has many misclassifications of the soil points. Notably, the PointNet++ misclassified some powerline points as tree but classified soil points more accurately than the FR-GCNet. Note that, compared with other methods, the SE-PointNet++ correctly recognized most soil points consistent to those of the ground truth.

The seven deep neural networks were used on the Test11 scene for MS-LiDAR point cloud classification and quantitative results are reported in Table 7. Among them, the DGCNN, RSCNN, PointNet++, and SE-PointNet++ outperformed the other methods with an OA of over 90.00 %. The GACNet performed modestly and achieved an OA of 87.59%. The PointNet obtained the worst performance with an OA of 83.36%. Comparatively, our FR-GCNet achieved promising results on the test scenes. A high point cloud classification performance with an *OA* of 93.55% and an average F_1-*score* of 78.61% were obtained. For the PointNet, the spatial relations among points were ignored, thereby degrading the quality of the output features used for predicting the classes of interest. Although the GACNet introduced attention mechanism into the graph convolution network for avoiding features pollution

Table 7
Classification results of the Test11 scene and statistical tests obtained by the comparative networks. The first six columns are F_1-*score* of each class.

Classes	road	building	grass	tree	soil	powerline	*OA* (%)	Average F_1-*score* (%)
PointNet	0.5081	0.7920	0.6861	0.7521	0.1273	0.2256	83.36	51.52
PointNet++	0.7108	0.8398	0.9324	0.9645	0.3024	0.5728	90.43	72.05
SE-PointNet++	0.7032	0.8564	0.9470	0.9705	**0.3702**	0.7035	93.01	75.84
DGCNN	0.7042	0.9025	0.9362	0.9793	0.2197	0.5524	91.19	71.57
RSCNN	0.7118	0.8900	0.9142	0.9563	0.2643	0.7003	92.44	73.90
GACNet	0.6451	0.8421	0.9341	0.9666	0.2277	0.3383	87.59	67.65
FR-GCNet	**0.8263**	**0.9081**	**0.9533**	**0.9877**	0.2872	**0.7411**	**93.55**	**78.61**

of marginal areas, the representative features of abundant points were not fully explored. Comparatively, the DGCNN obtained an increase of the point cloud classification performance due to the local and global shape features considered in the local neighborhood graphs. The RSCNN learned contextual shape-aware representation by the geometric priors among points, which upgraded the MS-LiDAR point classification performance.

6. Conclusion

MS-LiDAR systems can quickly and accurately obtain both geometrical and spectral information of ground objects simultanously. In this paper, we presented a feature reasoning-based graph convolution network (FR-GCNet) for MS-LiDAR point cloud classification. Our training sample was generated by the FPS-KNN sampling method during data preprocessing. The proposed FR-GCNet is an end-to-end encoder-decoder network consisting of the feature reasoning module with a local reasoning unit and a global reasoning unit. These two units extract local structural features with attention weights and global contextual features based on the input data,respectively. Aided by our FR-GCNet, we conducted the classification experiments on the Titan MS-LiDAR data. We achieved the *OA*, average F_1*-score* and *mIOU* of 93.55%, 78.61%, and 65.78%, respectively, which shows the excellent performance of our network on MS-LiDAR point cloud classification.

However, we also found that the proposed FR-GCNet performed less effectively on soil class. This is because the proposed FR-GCNet based on the graph convolution still cannot avoid the characteristic of heavily relying on geometric differences. As a result, there were still false detections and misidentifications in the MS-LiDAR point cloud results. In our future work, we hope to design a more powerful network architecture to further upgrade the robustness, representativeness, and distinctiveness of the output features, and exploit favorable prior knowledge of the scenes to further improve the point cloud classification correctness and accuracy.

Funding

This research was supported by the National Natural Science Foundation of China [grant numbers 41971414, 62076107] and Natural Science Foundation of Fujian Province [No. 2021J05059].

CRediT authorship contribution statement

Peiran Zhao: Conceptualization, Software, Methodology, Writing – original draft, Writing – review & editing. **Haiyan Guan:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Project administration, Funding acquisition. **Dilong Li:** Validation, Formal analysis, Investigation, Supervision. **Yongtao Yu:** Data curation, Writing – review & editing, Funding acquisition. **Hanyun Wang:** Investigation, Writing – review & editing. **Kyle Gao:** Writing – review & editing. **José Marcato Junior:** Writing – review & editing. **Jonathan Li:** Resources, Supervision, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Ali, M.E.N.O., Taha, L.-D., Mohamed, M.H.A., Mandouh, A.A., 2021. Generation of digital terrain model from multispectral LiDar using different ground filtering techniques Egypt. J. Remote Sens. Space Sci. 24 (2), 181–189.

Bakuła, K., Kupidura, P., Jełowicki, Ł., 2016. Testing of land cover classification from multispectral airborne laser scanning data. ISPRS Archives XLI-B7, 161–169.

Chen, B., Shi, S., Gong, W., Sun, J., Chen, B., Guo, K., Du, L., Yang, J., Xu, Q., Song, S., 2020. A spectrally improved point cloud classification method for multispectral LiDAR. ISPRS Archives, XLIII-B3-2020, 501–505.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L., 2018. DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Trans. Pattern Anal. Mach. Intell. 40 (4), 834–848.

Ekhtari, N., Glennie, C., Fernandez-Diaz, J.C., 2018. Classification of airborne multispectral Lidar point clouds for land cover mapping. IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens. 11 (6), 2068–2078.

Ekhtari, N., Glennie, C., Fernandez-Diaz, J.C., 2017. Classification of multispectral lidar point clouds. In Proc. IGARSS 23–28.

Felix, J. L., Martin, D., Patrik, T., Goutam, B., fahad, S. K., Michael, F., 2017. Deep projective 3D semantic segmentation. In: Compu. Anal. Images Pattern, doi: 10.1007/978-3-319-64689-3_8.

Ghaseminik, F., Aghamohammadi, H., Azadbakht, M., 2021. Land cover mapping of urban environments using multispectral LiDAR data under data imbalance. Remote Sens Appl. Soc. Environ. 21, 100449. https://doi.org/10.1016/j.rsase.2020.100449.

Griffiths, D., Boehm, J., 2019. Improving public data for building segmentation from Convolutional Neural Networks (CNNs) for fused airborne lidar and image data using active contours. ISPRS J. Photogramm. Remote Sens. 154, 70–83.

Guo, L., Chehata, N., Mallet, C., Boukir, S., 2011. Relevance of airborne lidar and multispectral image data for urban scene classification using Random Forests. ISPRS J. Photogramm. Remote Sens. 66 (1), 56–66.

Jing, Z., Guan, H., Zhao, P., Li, D., Yu, Y., Zang, Y., Wang, H., Li, J., 2021. Multispectral LiDAR point cloud classification using SE-PointNet++. Remote Sens. 13 (13), 2516.

Li, D., Shen, X., Yu, Y., Guan, H., Li, J., Zhang, G., Li, D., 2020. Building extraction from airborne multispectral LiDAR point clouds based on graph geometric moments convolutional neural networks. Remote Sens. 12 (19), 3186.

Lindberg, E., Holmgren, J., Olsson, Håkan, 2021. Classification of tree species classes in a hemi-boreal forest from multispectral airborne laser scanning data using a mini raster cell method. Int. J. Appl. Earth Obs. Geoinfo. 100, 102334. https://doi.org/10.1016/j.jag.2021.102334.

Liu, Y., Fan, B., Xiang, S., Pan, C., 2019. Relation-Shape convolutional neural network for point cloud analysis. In Proc. CVPR 8887–8896.

Matikainen, L., Karila, K., Litkey, P., Ahokas, E., Hyyppä, J., 2020. Combining single photon and multispectral airborne laser scanning for land cover classification. ISPRS J. Photogramm. Remote Sens. 164, 200–216.

Mccaffrey, J., 2019. Introduction to PyTorch on Windows. MSDN Magazine 34 (1), 16–23.

Miller, C. I., Thomas, J. J., Kim, A. M., Metcalf, J. P., Olsen, R. C., 2016. Application of image classification techniques to multispectral lidar point cloud data. Proc. SPIE 9832, Laser Radar Technology and Applications doi.org/10.1117/12.2223257.

Moenning, C., Dodgson, N.A., 2003. Fast marching farthest point sampling for implicit surfaces and point clouds. Comput. Lab. Technical Report 565, 1–12.

Morsy, S., Shaker, A., El-Rabbany, A., 2017. Multispectral LiDAR data for land cover classification of urban areas. Sensors 17 (5), 958.

Niemeyer, J., Wegner, J.D., Mallet, C., Rottensteiner, F., Soergel, U., 2011. Conditional random fields for urban scene classification with full waveform LiDAR data. In: Photogramm. Image Anal. 233–244.

Pan, S., Guan, H., Chen, Y., Yu, Y., Nunes Gonçalves, W., Marcato Junior, J., Li, J., 2020. Land-cover classification of multispectral LiDAR data using CNN with optimized hyper-parameters. ISPRS J. Photogramm. Remote Sens. 166, 241–254.

Pan, S., Guan, H., Yu, Y., Li, J., Peng, D., 2019. A comparative land-cover classification feature study of learning algorithms: DBM, PCA, and RF using multispectral LiDAR data. IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens. 12 (4), 1314–1326.

Qi, C.R., Su, H., Kaichun, M., Guibas, L.J., 2017a. PointNet: deep learning on point sets for 3D classification and segmentation. In Proc. CVPR 77–85.

Qi, C.R., Yi, L., Su, H., Guibas, L.J., 2017b. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In Proc. NeurIPS 5099–5108.

Shaker, A., Yan, W.Y., LaRocque, P.E., 2019. Automatic land-water classification using multispectral airborne LiDAR data for near-shore and river environments. ISPRS J. Photogramm. Remote Sens. 152, 94–108.

Sheshappanavar, S.V., Kambhamettu, C., 2020. A novel local geometry capture in Pointnet++ for 3D classification. In Proc. CVPR 1059–1068.

Sun, J., Shi, S., Chen, B., Du, L., Yang, J., Gong, W., 2017. Combined application of 3D spectral features from multispectral LiDAR for classification. In Proc. IGARSS 5264–5267.

Wallace, A., Nichol, C., Woodhouse, I., 2012. Recovery of forest canopy parameters by inversion of multispectral LiDAR data. Remote Sens. 4 (2), 509–531.

Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M., 2019a. Dynamic graph CNN for learning on point clouds. ACM Trans. Graphics 38 (5), 1–12.

Wang, L., Huang, Y., Hou, Y., Zhang, S., Shan, J., 2019b. Graph attention convolution for point cloud semantic segmentation. In Proc. CVPR 10288–10297.

Winiwarter, L., Mandlburger, G., Schmohl, S., Pfeifer, N., 2019. Classification of ALS point clouds using end-to-end deep learning. Photogramm.-Fernerkund.-Geoinfo. 87 (3), 75–90.

Yu, Y., Guan, H., Li, D., Gu, T., Wang, L., Ma, L., Li, J., 2020. A hybrid capsule network for land cover classification using multispectral LiDAR data. IEEE Geosci. Remote Sens. Lett. 17 (7), 1263–1267.

Yu, Y., Liu, C., Guan, H., Wang, L., Gao, S., Zhang, H., Zhang, Y., Li, J., 2021. Land cover classification of multispectral LiDAR data with an efficient Self-Attention capsule network. IEEE Geosci. Remote Sens. Lett. https://doi.org/10.1109/LGRS.2021.3071252.

Zhang, J., Lin, X., Ning, X., 2013. SVM-based classification of segmented airborne LiDAR point clouds in urban areas. Remote Sens. 5 (8), 3749–3775.