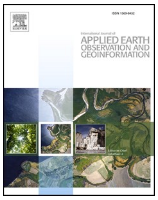
Contents lists available at ScienceDirect

International Journal of Applied Earth Observation and Geoinformation

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



A feature perturbation weakly supervised learning network for airborne multispectral LiDAR pointcloud classification

Ke Chen [a], Haiyan Guan [a,*], Lanying Wang [b], Yongtao Yu [c], Yufu Zang [a], Nannan Qin [a], Jiacheng Liu [a], Jonathan Li [b,d]

[a] *School of Remote Sensing and Geomatics Engineering, Nanjing University of Information Science and Technology, Nanjing 210044, China*
[b] *Department of Geography and Environmental Management, University of Waterloo, Waterloo ON N2L 3G1, Canada*
[c] *Faculty of Computer and Software Engineering, Huaiyin Institute of Technology, Huaian 223003, China*
[d] *Department of Systems Design Engineering, University of Waterloo, Waterloo ON N2L 3G1, Canada*

ARTICLE INFO

Keywords:
Multispectral LiDAR
Weakly supervised learning
Pointcloud classification
Dual semantic inference structure
Consistency constraint
Mutual pseudo-labeling loss

ABSTRACT

Currently, most pointcloud classification methods heavily rely on huge numbers of labeled samples. Notably, labeling a large-scale multispectral LiDAR (MS-LiDAR) pointcloud is time-consuming and costly. To address this issue, we propose a feature perturbation weakly supervised network for classifying MS-LiDAR pointclouds using a few labeled samples, termed as FPWS-Net. In the FPWS-Net, we innovatively design a dual semantic inference structure, including a primary semantic inference module and an auxiliary semantic inference module. To provide the network with rich, accurate supervised signals, we embed kernel point convolution (KPConv) into the network for modelling the contextual information of MS-LiDAR pointclouds and propagating the signals between labeled and unlabeled points. Additionally, to constrain feature perturbations resulting from the dual semantic inference structure, fully leverage unlabeled points, and fit the architecture of the FPWS-Net, we combine consistency constraint and mutual pseudo-labeling loss. The proposed FPWS-Net is tested on two datasets, and achieves at least an average F_1-score of 83.69 %, an mIoU of 78.81 %, and an OA of 95.97 % using only 0.1 % labeled points. The comparative experimental results demonstrate that the FPWS-Net not only outperforms the state-of-the-art (SOTA) weakly supervised networks, but also achieves the comparable classification performance to the fully supervised methods in the airborne MS-LiDAR pointcloud classification tasks.

1. Introduction

An airborne multispectral LiDAR (MS-LiDAR) system, which simultaneously captures multiple-band pointcloud datasets, provides the integrated spatial-spectral information of the areas to be surveyed, contributing to the description of the surveyed areas in detail. Thus, MS-LiDAR technology has gained increasing attention, and more and more institutions and companies dedicated themselves to the development of MS-LiDAR prototypes. For instance, the first commercial MS-LiDAR system, released by Teledyne Optech (Toronto, ON, Canada), contains three wavelengths of 1550 nm, 1064 nm, and 532 nm (Bakuła, 2015). Wuhan University has developed a MS-LiDAR system prototype (Gong et al., 2015) with four wavelength channels of 556 nm, 670 nm, 700 nm, and 780 nm. Compared with conventional single-band LiDAR data, data acquired by MS-LiDAR systems significantly improve object identification and pointcloud classification accuracies (Ekhtari et al., 2018; Wang and Gu, 2020). Currently, MS-LiDAR pointclouds have been increasingly used in a range of applications, e.g., urban observation (Yokoya et al., 2018), coastal management (Shaker et al., 2019), forest monitoring (Yan et al., 2020), and land cover classification (Ghaseminik et al., 2021).

In terms of input data format, MS-LiDAR pointcloud processing methods can be broadly categorized into two groups: two-dimensional (2D) rasterized multispectral images-based and three-dimensional (3D) pointcloud-based. The former first transforms a 3D MS-LiDAR pointcloud into 2D multispectral images, to which machine learning (Matikainen et al., 2020, 2017) or deep learning (Yu et al., 2020, 2022) image processing algorithms are applied for object recognition and pointcloud classification. However, this data conversion damages the geometric completeness of objects, such as trees and buildings, resulting

[*] Corresponding author.
E-mail addresses: c_ke@nuist.edu.cn (K. Chen), guanhy.nj@nuist.edu.cn (H. Guan), junli@uwaterloo.ca (L. Wang), allennessy@hyit.edu.cn (Y. Yu), 3dmapzangyufu@nuist.edu.cn (Y. Zang), nnqin@nuist.edu.cn (N. Qin), ilneuac@nuist.edu.cn (J. Liu).

https://doi.org/10.1016/j.jag.2024.103683
Received 12 September 2023; Received in revised form 12 January 2024; Accepted 23 January 2024

in a reduction in point cloud processing accuracy. Comparatively, the latter directly processes 3D MS-LiDAR pointclouds via machine learning or deep learning algorithms (Chen et al., 2020; Morsy et al., 2016). Although machine learning-based MS-LiDAR pointcloud classification methods have achieved a large success, feature design and selection is tedious and subjective, resulting in the unreliability of data processing accuracy. Deep learning has quickly attracted much attention to MS-LiDAR pointclouds due to its advantages in automatic feature extraction and its abilities of handling complex high-dimensional data.

Inspired by various classic deep learning methods for processing single-band pointclouds, such as PointNet++ (Qi et al., 2017), DGCNN (Wang et al., 2019), and RSCNN (Liu et al., 2019), a variety of deep learning methods have been specifically designed for MS-LiDAR point clouds. For example, Jing et al. (2021) enhanced the classification accuracies of MS-LiDAR pointclouds by embedding Squeeze-and-Excitation blocks into PointNet++. This integration aimed to accentuate crucial channels while suppressing channels that were less informative for accurate predictions. To further utilize contextual features, Zhao et al. (2021), based on a graph convolution network (GCN), developed a global reasoning unit for capturing global contextual features to reveal spatial relationships among points and a local reasoning unit for dynamically adjusting edge features with attention weights, finally improving MS-LiDAR pointcloud classification performance. To learn better geometry feature representations of MS-LiDAR pointclouds, Li et al. (2022a) designed a attention graph geometric moment convolution (AGGM Convolution) and constructed a feature pyramid to extract and fuse multi-scale features. Although the aforementioned methods or networks have demonstrated their superiorities in airborne MS-LiDAR pointcloud classification tasks, they heavily relied on a significant number of labeled samples. Notably, labeling large-scale pointclouds is costly and complex. Consequently, weakly supervised learning methods have become increasingly appealing to 3D pointcloud classification since they require only partially labeled samples.

Currently, in terms of labeling means, weakly supervised pointcloud classification methods can be grouped into indirect labeling-based and direct labeling-based methods. The former converts an input pointcloud into 2D images (Wang et al., 2020), sub-pointclouds (Wei et al., 2020), or segments (Tao et al., 2022) before pointcloud labeling operations. Although the indirect labeling-based methods can effectively reduce the workload of manually labeling, they require data pre-processing, which is still time-consuming and labor-intensive, especially for large-scene LiDAR pointclouds.

Direct labeling-based methods involve sparsely labeling a subset of a pointcloud to classify the entire pointcloud data. For example, Xu and Lee (2020) developed a muti-branch weakly supervised framework for pointcloud classification using 10 % sampling points. However, 10 % labeled points is still a considerable workload for large-scene, high-density pointclouds. To further reduce the labeling workload, a semantic query network (SQN), proposed by Hu et al. (2021), exploited the semantic similarity between sparsely labeled points and their neighbors to enhance the underlying supervised signals through a trilinear interpolation. This approach enabled pointcloud classification using only 0.1 % labeled points. However, SQN inadequately explored the features of unlabeled points to accurately describe the characteristics of objects, thereby lowering the pointcloud classification accuracy. Zhang et al. (2021) proposed a two-step knowledge transfer method, in which a self-supervised network was designed to extract the prior knowledge between color information and classes of interest, and a weakly supervised network was used to integrate the knowledge for improving feature representation. However, this method required pre-training, therefore lowering the training efficiency. Thus, to increase the utilization of supervised signals, sample perturbation was employed to augment input samples. Specifically, Jiang et al. (2021), based on contrastive learning between input samples and augmented samples, improved feature representation and model generalization via pseudo-label guidance and confidence guidance. Zhang et al. (2021b) proposed perturbed self-distillation (PSD), in which an attribute attention module was used to construct a perturbed branch conducive to the model, and then incorporated GCN into a decoder to capture relationships among local points based on high-dimensional feature distances, generating affinity loss, and providing more supervised signals to the model. HybridCR, proposed by Li et al. (2022b), improved the diversity of perturbed samples via a dynamic point augmentor, and employed local and global guidance contrastive losses to enhance model performance. Although sample perturbations could effectively increase supervised signals and achieved promising pointcloud classification results, they normally increased memory consumption because of inputting double samples to the networks. Moreover, only using neighboring points to encode feature vectors poorly modeled high-dimensional feature relationships among points, and the rigid information encoded in the high-dimensional feature space severely hampered network generalization (Qiu et al., 2021).

Therefore, we propose a novel <u>f</u>eature <u>p</u>erturbation <u>w</u>eakly <u>s</u>upervised pointcloud classification framework, termed as FPWS-Net, for airborne MS-LiDAR pointclouds. Firstly, to construct simple and effective feature perturbations, the FPWS-Net is designed as a dual semantic inference structure including a primary semantic inference module and an auxiliary semantic inference module. Subsequently, we embed kernel point convolution (KPConv) (Thomas et al., 2019) as a context-aware module into the primary semantic inference module to provide accurate underlying supervised signals for the FPWS-Net. Furthermore, we employ a consistency constraint and a mutual pseudo-labeling loss to minimize prediction discrepancy between the two semantic inference modules and fully leverage abundant unlabeled data. The main contributions of this paper are as follows:

1. We propose a novel weakly supervised deep learning framework, termed as FPWS-Net, for MS-LiDAR pointcloud classification, which achieves pointcloud classification performance comparable to those of the fully supervised methods by utilizing 0.1 % labeled data.
2. The FPWS-Net innovatively employs two semantic inference modules to construct feature perturbations within the model. To enhance contextual information and provide accurate underlying supervised signals for the weakly supervised network, we embed KPConv into the auxiliary semantic inference module.
3. We combine the consistency constraint loss and the mutual pseudo-labeling loss to constrain feature perturbations from the two semantic inference modules, fully leverage abundant unlabeled data, and provide more supervised signals to the model.

The remainder of the paper is organized as follows. Section 2 details the two large-scale airborne MS-LiDAR pointcloud datasets. Section 3 briefly presents data preprocessing and gives a comprehensive description of the proposed FPWS-Net. Section 4 discusses the experimental results. Finally, Section 5 gives concluding remarks.

2. Study areas and datasets

The two MS-LiDAR pointcloud datasets were collected by a Teledyne Optech Titan MS-LiDAR system, which equipped with laser scanners operating at 1550 nm (MIR), 1064 nm (NIR), and, 532 nm (Green), respectively. Each channel of the system provided 300 kHZ repetition per pulses (RPP). The three channels in the system were arranged at an interval of 3.5°, and simultaneously provided three individual pointcloud datasets. Table1 lists the detailed specifications of the system.

The first study area is situated in Whitchurch-Stouffville, a small town in Ontario, Canada, with the center coordinates at longitude $79^{\circ}15'00''$ and latitude $43^{\circ}58'00''$. The Titan MS-LiDAR system collected MS-LiDAR data at a flight height of around 1000 m, and correspondingly, the point density was estimated 3.6 points/m^2 for each channel. The system scanned 19 strips with overlapping, covering an area of 2,052 m × 1,566 m. Thirteen representative and feature-rich scenes (see

Table 1
Specifications of the Teledyne Optech Titan MS-LiDAR system (Jing et al., 2021; Zhao et al., 2021).

Parameter	Specification
Wavelength (nm)	Channel 1: 1550 (shortwave infrared SWIR) Channel 2: 1064 (near infrared NIR) Channel 3: 532 (green GREEN)
Look angle (degree)	Channel 1: 3.5° Channel 2: 0° Channel 3: 7°
Scanning field of view angle (degree)	+/−20°
Pulse repletion frequency (kHZ)	300
Flying height (m)	~1000
Point density (points/m^2)	~3.6
Point spacing (points/m)	~0.5

Fig. 1 (b)) were selected to construct the Whitchurch-Stouffville (WS) dataset. For the WS dataset, Areas 11–13 were selected as testing data, and the remaining scenes were used as training data.

The second study area is situated in Tobermory, at the northern end of the Bruce Peninsula in southeastern Ontario, Canada, centered at coordinates of longitude 81°39′57″ and latitude 45°15′24″. The Titan MS-LiDAR system collected 10 vertically intersecting strips. Twelve representative scenes were selected to build the Tobermory dataset (see Fig. 1 (c)). For the Tobermory dataset, Areas 9 and 12 were selected as testing data, and the remaining scenes were used as training data. In this study, due to the absence of system parameters and flight trajectories during data acquisition, absolute intensity calibration was not performed.

3. Methodology

3.1. Data pre-processing

To create the input data to the proposed FPWS-Net, a pre-processing is required for the WS and Tobermory datasets, which mainly involves two steps: data fusion and labeling, and sample generation.

3.1.1. Data fusion and labeling

The Titan MS-LiDAR system captures three independent pointclouds for the three laser channels (i.e., 1550 nm, 1064 nm, and 532 nm). To create a single high-density multispectral LiDAR pointcloud dataset, we fuse the three independent pointclouds via the method mentioned in Jing et al. (2021). After data fusion, each point in the two datasets contains not only its own geometric information (X, Y, Z) but also the spectral information of three channels (1550 nm, 1064 nm, and 532 nm). We then manually labeled the fused MS-LiDAR datasets point-by-point. For the WS dataset, we labeled it into six classes, including road, building, grass, tree, soil, and powerline. For the Tobermory dataset, we also labeled it into six classes, including road, building, land (soil and grass), tree, water, and powerline. Table 2 shows the number of points and the proportion of each class in the two MS-LiDAR datasets. It is worth noting that these two datasets suffer from severe data imbalance, for instance, the number of powerline points is only about 0.2 % of the total points.

Table 2
The number of points and proportion of each class in the WS and Tobermory datasets, respectively.

WS dataset			Tobermory dataset		
Class	Nums (#)	Proportion (%)	Class	Nums (#)	Proportion (%)
Road	855,929	10.03	Road	894,911	3.40
Building	531,848	6.23	Building	712,916	2.70
Grass	3,480,918	40.80	Land	4,798,345	18.21
Tree	3,486,836	40.87	Tree	16,425,016	62.32
Soil	157,180	1.84	Water	3,460,372	13.13
Powerline	19,700	0.23	Powerline	64,738	0.25

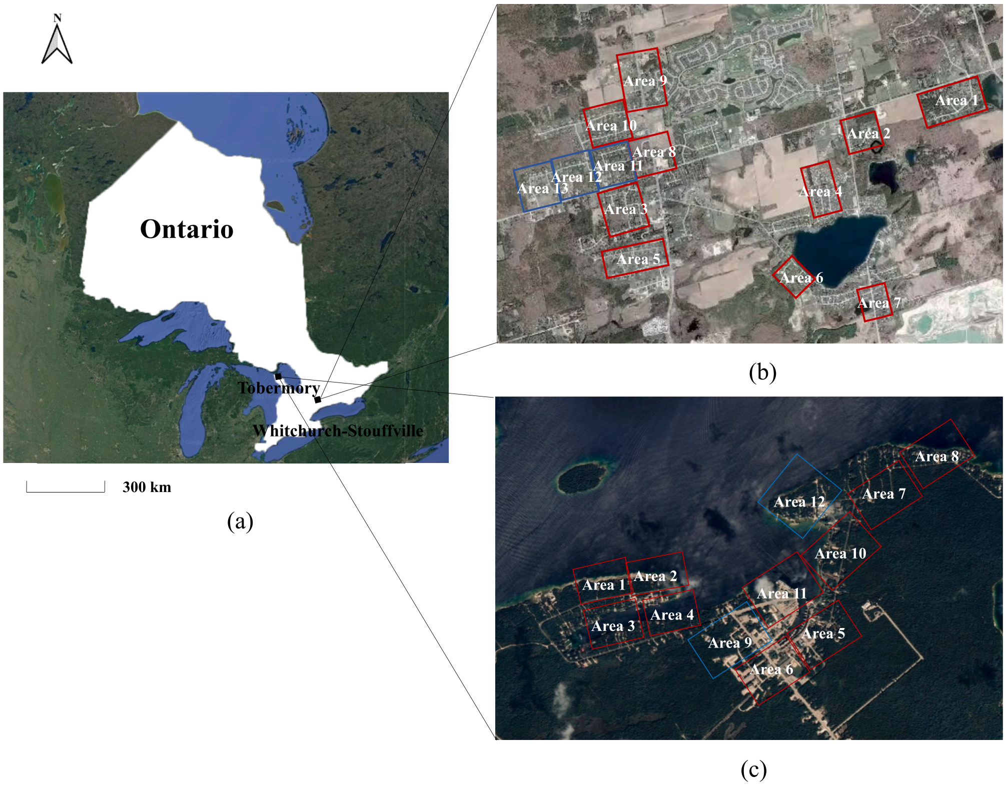


Fig. 1. Illustration of two study areas, (a) the locations, (b) Whitchurch-Stouffville study area, and (c) Tobermory study area.

3.1.2. *Sample generation*

To generate a certain number of training samples, we design a sample generation method via grid sampling and KD-Tree (K-dimensional tree). The specific sample generation method is detailed as follows.

1) A pointcloud scene is divided into a series of 3D sub-grids with a side length of 0.5 m, with regards to the average point density of the MS-LiDAR dataset (around 3.6 points/m^2 for each channel in this study).
2) For each sub-grid, we first record the geometric and spectral statistics of all points, respectively, and calculate their mean values as a sampling point for the sub-grid. Moreover, we count the number of points for each class, and the class with the largest proportion is selected as the class for the sampling point.
3) Finally, each sampling point is used to establish a KD-Tree, by which we obtain its $N-1$ neighbors, where N is the number of input points at a single batch.

By using the grid sampling and KD-Tree, we acquire a fixed number of sampling points for network training, which helps maintain the completeness of geometric structure of a pointcloud. Thus, the samples input into the model contain three geometric coordinates (X, Y, Z), three spectral bands (1550 nm, 1064 nm, and 532 nm), and a class label.

3.2. *Framework of the FPWS-Net*

Fig. 2 illustrates the framework of our FPWS-Net. The FPWS-Net is a dual semantic inference structure, which includes the primary and auxiliary semantic inference modules. To provide accurate, semantically rich, and underlying supervised signals to the network, we embed KPConv into the primary semantic inference module. Additionally, to constrain the feature perturbations generated by the two semantic inference modules, consistency constraint is used to enhance the stability of the model. Finally, a mutual pseudo-labeling loss is employed to convert a large number of unlabeled points into auxiliary supervised signals.

As shown in Fig. 2, the FPWS-Net is an end-to-end structure, which takes N training samples as input and outputs the point-by-point classification predictions by the primary semantic inference module. To begin with, the training samples, each of which includes three coordinates (X, Y, Z), three spectral bands (i.e., 1550 nm, 1064 nm, and 532 nm), and a labeled class, are fed into an encoder to extract features. The encoder is built by a fully connected layer, five layers of combination of random sampling and local feature aggregation (Hu et al., 2020). The fully connected layer uniformly converts the training samples to 8-dimensional features, enabling the FPWS-Net to accommodate various types of pointcloud data. The subsequent five layers are constructed to gradually down-sample points at a certain scale (i.e., 1, 1/4, 1/16, 1/64, 1/256, 1/512) and increase the feature dimension at scales of 8, 16, 64, 128, 256, and 512. Notably, at each sampling operation, the coordinates of the input MS-LiDAR pointcloud are also used for the auxiliary semantic inference modules.

Next, the output features of the encoder are input to the primary semantic inference module and the auxiliary semantic inference module, respectively. These two semantic inference modules generate predictions A and B, respectively, which result in feature perturbations. The feature perturbations are then supervised by two classification losses L_{clA} and L_{clB} of labeled points, and a consistency constraint loss L_{cc} of all points. The mutual pseudo-labeling loss L_{pl} is applied to unlabeled points of predictions A and B, generating soft pseudo-labels for mutual supervision, respectively. Finally, the predictions obtained by the primary semantic inference module are considered as the final predictions of the FPWS-Net, i.e., assigning a class to each point.

3.2.1. *Primary semantic inference module*

Usually, the pointcloud classification performance of the network significantly degrades with a decrease of the number of labeled samples because the supervised signals obtained from the sparsely labeled points are poorly propagated to unlabeled points. To address this issue and obtain rich underlying supervised signals, we design a primary semantic inference module, which consists of a group of five up-sampling layers and five Multi-Layer Perceptions (MLPs), a KPConv, a group of three

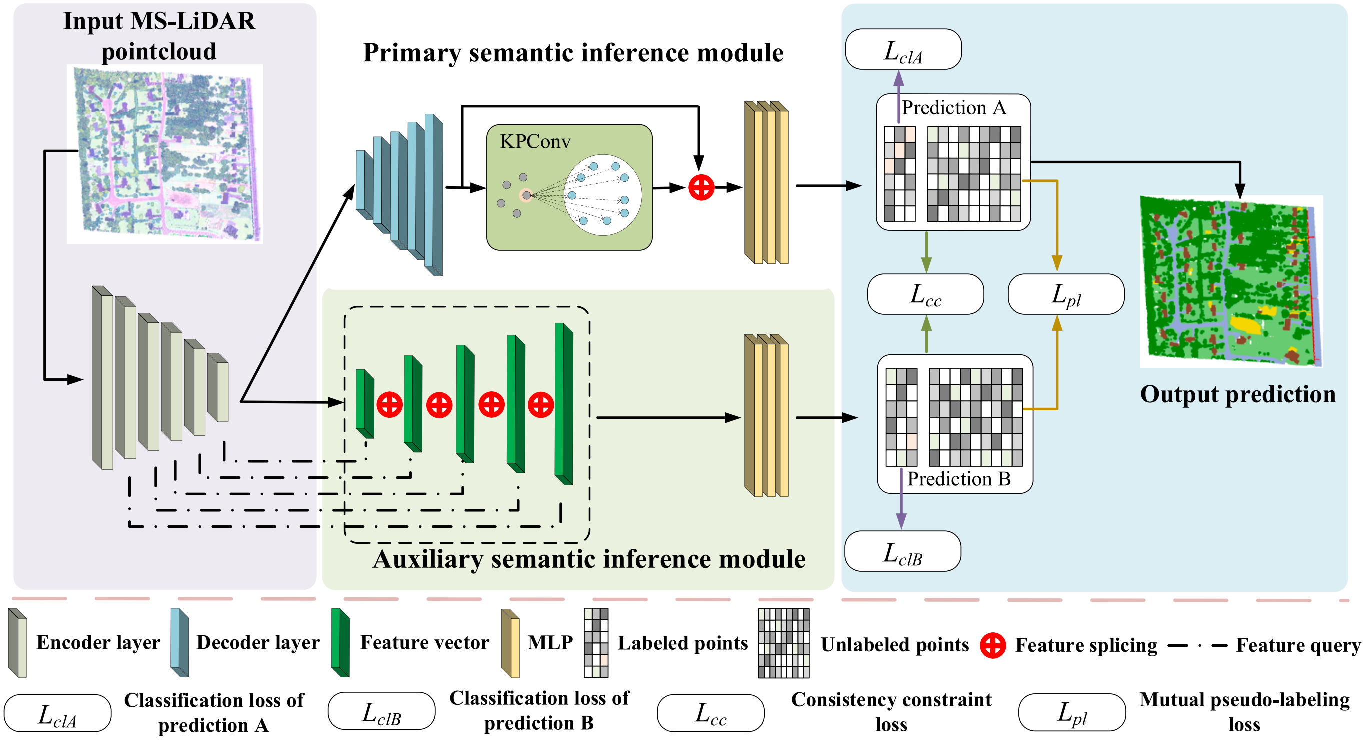

Input MS-LiDAR pointcloud
Primary semantic inference module
KPConv
L_{clA}
Prediction A
L_{cc}
L_{pl}
Prediction B
L_{clB}
Output prediction
Auxiliary semantic inference module
Encoder layer
Decoder layer
Feature vector
MLP
Labeled points
Unlabeled points
Feature splicing
Feature query
L_{clA} Classification loss of prediction A
L_{clB} Classification loss of prediction B
L_{cc} Consistency constraint loss
L_{pl} Mutual pseudo-labeling loss

Fig. 2. The framework of the FPWS-Net.

fully connected layers and one dropout layer. Specifically, the features obtained from the encoder first go through the group of five up-sampling layers and five MLPs, which gradually up-sample to obtain the same number of as its original number of points while increasing the feature dimensions of {512, 256, 128, 64, 16, 8}. Subsequently, the outputs are fed to the KPConv for establishing the relationships between local points, thereby generating corresponding local features. The features output from KPConv are then concatenated with its input features and fed into the group of three fully connected layers and one dropout layer, resulting in the predictions of points.

In the primary semantic inference module, the KPConv, a context-aware module, establishes the relationships among local points, facilitating the propagation of signals between labeled and unlabeled points. Fig. 3 details the diagram of the KPConv. As seen in Fig. 3, KPConv first assigns neighboring features with different weights to the kernel points, based on the linear correlation function between the kernel points and the neighboring points of the input points. Then the kernel points are multiplied by the learnable weight matrix to obtain the contextual feature and output them. This process provides the network with robust, effective underlying supervised signals. In addition, to improve the efficiency of training model, simple rigid convolutional kernel points are adopted.

Concretely, for any one point p_i in the high-dimensional feature space with X, Y, Z coordinates, we assume it as a center point and query its neighboring points ($p_i^k, k = 1, 2, \cdots, K_a, Ka$ is the number of neighboring points) via a K-nearest neighboring (KNN). The geometric coordinates of point p_i^k and its high-dimensional features f_i^k are arranged into two arrays, i.e., (K_a, 3) and (K_a, 8), respectively. Denote the kernel points as x_j, ($j = 1, 2, \ldots, N_j$, N_j is the number of the kernel points) in a unit ball in the Euclidean space. The contextual feature vector $\overrightarrow{F}_i$ of the central point p_i is calculated by

$$\overrightarrow{F}_i = \sum_{k=1}^{K_a} \sum_{j=1}^{N_j} \mathbf{W_j} h\left(p_i^k, x_j\right) f_i^k \tag{1}$$

where $\mathbf{W_j}$ denotes the weight matrix of the kernel points learned from network training. h denotes the linear correlation function between the neighboring points and the kernel points, which is written by

$$h\left(p_i^k, x_j\right) = \max\left(0, 1 - \frac{\|p_i^k - x_j\|}{d}\right) \tag{2}$$

where d denotes the defined radius of the kernel points. It can be observed that the closer the distance between a neighboring point and a kernel point, the greater the weight.

Afterwards, the contextual feature vector $\overrightarrow{F}_i$ of the central point p_i, is generated and then concatenated with the features output from the group of five up-sampling layers and five MLPs, enhancing feature representation. The enhanced feature representation provides the network with a semantically rich, underlying supervised signal, ultimately improving the probability of the predictions.

3.2.2. Auxiliary semantic inference module

The primary semantic inference module uses both sparsely labeled and unlabeled points as supervised signals to update the network parameters. Because the unlabeled points may include many unimportant training signals, an auxiliary semantic inference module is designed to fully leverage the sparse yet valuable supervised signals. The auxiliary semantic inference module gathers features from the sparsely labeled points and their neighbors across the encoding layers for model training. This process backpropagates these valuable, accurate supervised signals into an expanded spatial context, thus implicitly enhancing the network supervision.

Fig. 4 details the diagram of the auxiliary semantic inference module. As shown in Fig. 4, in the auxiliary semantic inference module, each sparsely labeled point is used as a query point to obtain its K_b nearest neighbors in the high-dimensional feature space of each encoding layer via KNN. Then, the features of the K_b nearest neighbors are compressed into a feature vector by a trilinear interpolation. Finally, the feature vectors coming from the five encoding layers are concatenated together to obtain supervised signals of the query point, which are then fed to the MLPs to infer the point class information. Taking the first encoding layer as an example, the steps are specifically detailed as follows.

1) In the first encoding layer, a labeled point $p_l \in R_l^3$ (containing X, Y and Z coordinates) is given as a query point, its K_b neighboring points $p_l^k (k = 1, 2, \cdots, K_b$, K_b is the number of neighboring points) and its 16-dimensional feature f_l^k are obtained accordingly.
2) Then, the nearest neighbors p_l^k is compressed into a feature vector $\overrightarrow{F_l^1}$ of point p_l by the trilinear interpolation, according to the distance between points p_l and p_l^k. The feature vector $\overrightarrow{F_l^1}$ serves as the feature vector of point p_l in the first encoding layer. This operation is done in the five encoding layers respectively, and the feature vectors $\{\overrightarrow{F_l^1}, \overrightarrow{F_l^2}, \overrightarrow{F_l^3}, \overrightarrow{F_l^4}, \overrightarrow{F_l^5}\}$ coming from these five layers are concatenated to obtain the representative features of point p_l.
3) Finally, the predicted probability of point p_l is obtained through three 1 × 1 fully connected layers and one dropout layer.

Note that, to facilitate network operation and loss function calculation during training, we input the feature information of all points into the auxiliary semantic inference module. However, we only use the supervised signals of the labeled points to supervise the network.

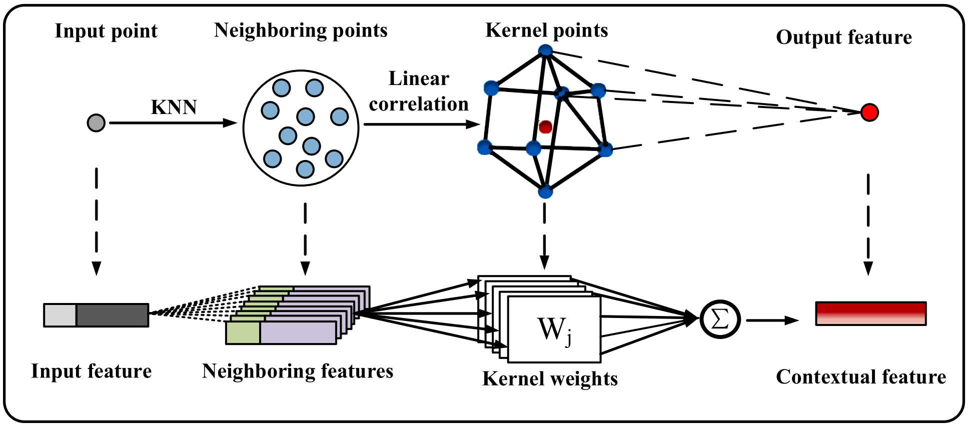


Fig. 3. The diagram of the KPConv.

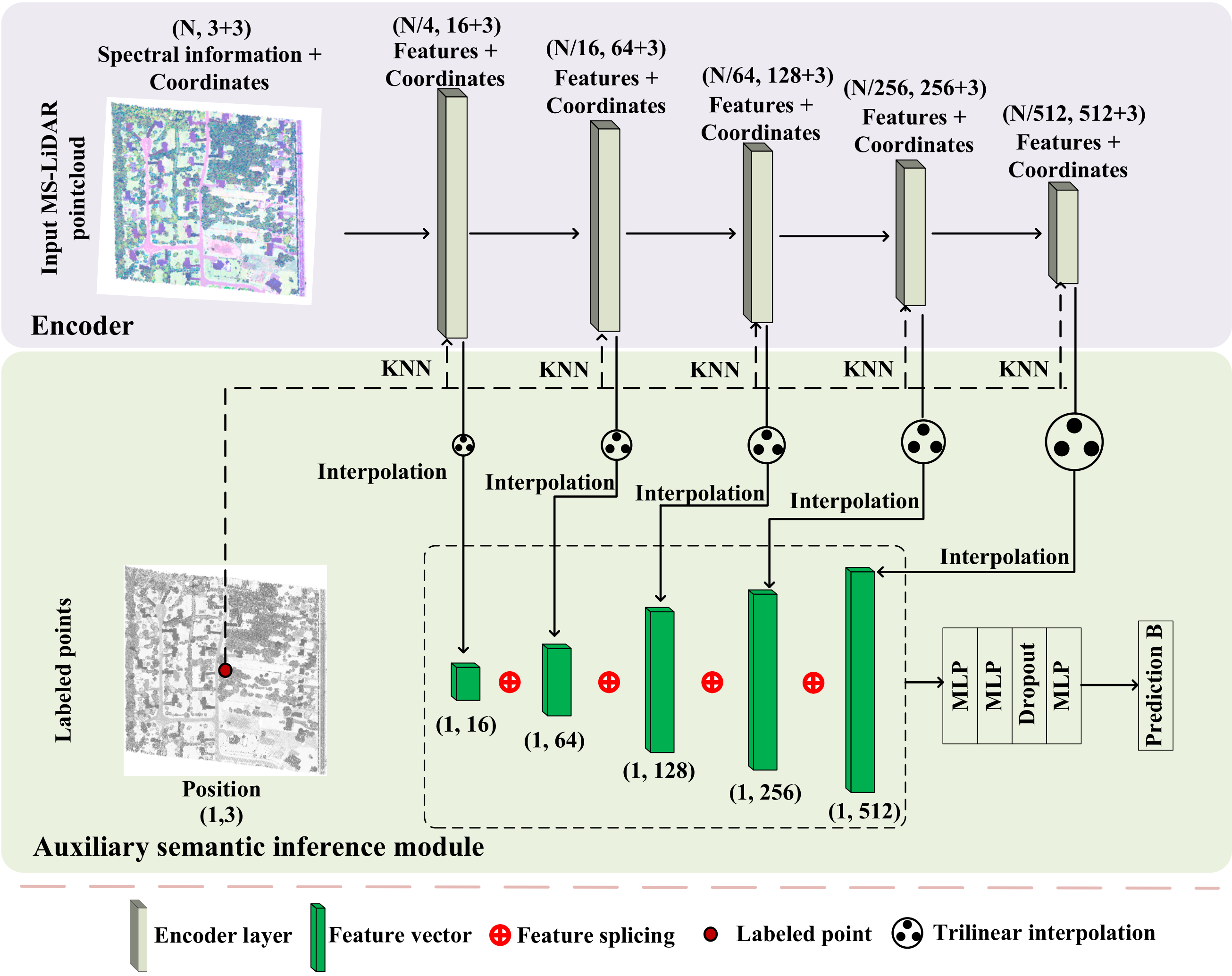


Fig. 4. The diagram of auxiliary semantic inference module.

3.2.3. Losses

In the training process of the FPWS-Net, the total loss L_{total} contains three components: the consistency constraint loss function L_{cc} for supervising all the points, the mutual pseudo-labeling loss function L_{pl} for supervising the unlabeled points, and the classification loss functions L_{cl} for supervising the labeled points.

1) Consistency constraint loss

To measure the differences between the predicted probabilities of the two semantic inference modules, a Jensen-Shannon (*JS*) divergence is adopted to calculate the consistency constraint loss L_{cc}, which is written by:

$$L_{cc} = \frac{1}{2N}\sum_{i}^{N} JS(P_A \| P_B)$$

$$= \frac{1}{4N}\sum_{i}^{N}\sum_{j}^{C} KL(P_{Aij}\|\frac{P_{Aij}+P_{Bij}}{2}) + KL(P_{Bij}\|\frac{P_{Aij}+P_{Bij}}{2}) \tag{3}$$

$$= \frac{1}{2N}\sum_{i}^{N}\sum_{j}^{C} P_{Aij}log\frac{P_{Aij}}{P_{Aij}+P_{Bij}} + P_{Bij}log\frac{P_{Bij}}{P_{Aij}+P_{Bij}}$$

Where *KL* denotes the Kullback-Leibler divergence. P_{Aij} and P_{Bij}, respectively, are the probabilities that the primary semantic inference module and the auxiliary semantic inference module predict point p_i belonging to class *j*. *N* is the number of input points at a single batch, and *C* is the number of classes.

By the Jensen-Shannon divergence, consistency constraint is performed on the model to realize its self-supervision and provide additional, auxiliary supervised signals for the model, finally improving the stability of the model prediction. This loss applies to both labeled and unlabeled data.

2) Mutual pseudo-labeling loss

To fully utilize the large numbers of unlabeled points, provide the network with more auxiliary supervised signals, and align with the FPWS-Net framework, we propose a mutual soft pseudo-labeling loss. Specifically, first, we convert the predicted probabilities P_{UA} and P_{UB} of the unlabeled points output from the primary semantic inference module and the auxiliary semantic inference module, respectively, into soft pseudo-labels sPL_A and sPL_B by a sharpening function, which is written by

$$sPL = \frac{P^T}{P^T + (1-P)^T} \tag{4}$$

where *T* is a constant that controls the degree of sharpening. Compared with the hard pseudo-labels generated by a fixed threshold, the soft pseudo-labels effectively eliminate the negative influence of training data caused by incorrect prediction tags, which reduces uncertainty and provides the predictions with high confidence.

Then, we employ entropy regularization to supervise the predictions (P_{UB}) of the auxiliary semantic inference module using the soft pseudo-labels (sPL_A) produced by the primary semantic inference module. The entropy regularization L_{Aer} is defined by

$$L_{Aer} = -\frac{1}{M}\sum_{i}^{M}\sum_{j}^{C} sPL_{Aij} log P_{UBij} \tag{5}$$

where M is the number of unlabeled points. P_{UBij} is the probability in the predictions P_{UB} of point p_i predicted to be class j. sPL_{Aij} is the probability of class j in the soft pseudo-labels of point p_i. C is the total number of classes.

Similarly, we use entropy regularization to supervise the predictions (P_{UA}) of the primary semantic inference module using the soft pseudo-labels (sPL_B) produced by the auxiliary semantic inference module. The entropy regularization L_{Ber} is defined by

$$L_{Ber} = -\frac{1}{M}\sum_{i}^{M}\sum_{j}^{C} sPL_{Bij} log P_{UAij} \tag{6}$$

Finally, the mutual pseudo-labeling loss consists of the above two entropy regularizations, as shown in the following equation:

$$L_{pl} = L_{Aer} + L_{Ber} \tag{7}$$

The mutual pseudo-labeling loss encourages the generated soft pseudo-labels to maintain low entropy, promoting highly confident predictions and finally enhancing the model performance and generalization ability.

3) Classification loss

To address the issue of class imbalance in weak supervised methods, we use a weighted cross-entropy loss function for labeled points. The weighted cross-entropy loss function assigns different weights according to the proportion of each class. Thus, the supervised signals are biased towards the classes with a small number of points, improving their classification accuracy.

Moreover, because the cross-entropy loss function calculates the loss regarding the model prediction probability, a lovasz-softmax loss function is optimized directly for intersection-over-union measure to balance the classification loss performance of the model (Berman et al., 2018). The final classification loss function L_{cl} is given by

$$L_{cl} = -\frac{1}{N-M}\sum_{i=1}^{N-M} w_i \sum_{j=1}^{C} y_{ic} log \frac{\exp\left(P_{ij}^{l}\right)}{\sum_{j=1}^{C}\exp\left(P_{ij}^{l}\right)} + \frac{1}{C}\sum_{j\in C}\overline{\Delta J_j}(M_j) \tag{8}$$

where P_{ij}^{l} denotes the probability that the i-th sparsely labeled point belongs to class j. w_i represents the class weight of sample i. $\overline{\Delta J_j}(M_j)$ denotes the Jaccard coefficient constructed by the error distribution probability vector M_j for the samples belonging to class j.

Accordingly, the primary semantic inference module and the auxiliary semantic inference module separately compute their respective classification loss functions, denoted as L_{clA} and L_{clB}.

4) Total loss

The final network total loss L_{total} can be written by

$$L_{total} = L_{clA} + L_{clB} + L_{cc} + \lambda L_{pl} \tag{9}$$

At the early stage of model training, the soft pseudo-labels generated by the FPWS-Net are unreliable, hence the mutual pseudo-label loss should not be used when optimizing the network parameters. However, when the network training proceeds, the soft pseudo-label is gradually stabilized and the weight of its loss function is gradually increased, and hence we introduce a nonlinear parameter λ to control the weight of the mutual pseudo-labeling loss in model supervision, and λ is expressed by

$$\lambda = \begin{cases} 0, & epoch < 30 \\ e^{\frac{epoch}{\max\_epoch}-1}, & epoch \geqslant 30 \end{cases} \tag{10}$$

Where $epoch$ and max_epoch are the current number of iterations and the total number of iterations, respectively, when model training.

4. Results and discussion

4.1. Implementation details

All experiments were conducted on a workstation with hardware consisting of an Intel Core i7-9700 [CPU], 16 GB RAM and 8 GB NVIDIA GeForce RTX 2070 [GPU], under the Ubuntu 20.04 operating system. For model training, the batch size, the initial learning rate, the max_epoch, and the decay rate per cycle were set to be 2, 0.001, 100, 5 %, respectively, and the Adam optimizer was used. The number of convolution kernel points was set as 9, the number of the nearest neighbors in the KNN was set as 16, and the number of input points at a single batch was set as 40,960. The temperature constant T in the pseudo-label sharpening function was set as 10.

Based on the point density (3.6 point/m^2) of the WS and Tobermory datasets, the grid sampling size used the configuration of 0.5 m in this study. We adopted the sparse labeling strategy of SQN, which randomly selected points from data sampling with a certain percentage of labeled points. The number of total points, grid sampling point, training points, and sparsely labeled points for the two datasets are shown in Table 3.

In this study, three evaluation metrics, including Overall Accuracy (OA), Mean Intersection over Union (mIoU), and F$_1$-score, were used to quantitatively analyze the model performance.

4.2. Overall classification performances

4.2.1. Classification results of WS dataset

To assess the performance of the proposed FPWS-Net on MS-LiDAR pointcloud classification, we trained the model with 10 scenes of the WS dataset using only 0.1 % labeled points, and tested it over the remaining three scenes (Areas 11–13). Fig. 5 shows the classification results of the three scenes. As seen in Fig. 5, our network achieved a promising performance for classes building, tree, and powerline with clear and obvious spatial structures. However, it encountered challenges in accurately and completely classifying soil points.

We presented two of the close-up view regions (black oval-shaped regions A and B in Fig. 5) in Fig. 6. Visual inspection reveals that the majority of the MS-LiDAR points were correctly classified. However, there were some misclassifications observed for classes soil and road along the edges of the scenes. The misclassification of class road was primarily caused by a lack of sufficient road neighboring features provided to the network in the edge regions of the scene. For the poor classification performance of soil points, this may be because the MS-LiDAR data were collected in July, when the majority of soil regions were covered by dense vegetation. Additionally, soil regions shared spectral similarities with its adjacent roads, thereby significantly increasing the misclassifications of soil points as road points.

Table 4 lists the quantitative classification results obtained by the

Table 3
Overview of the two MS-LiDAR pointcloud datasets.

Dataset	Total points (#)	Grid sampling points (#)	Training points (#)	Labeled points (0.1 %) (#)
WS	8,532,411	6,629,894	4,906,189	4,906
Tobermory	26,356,298	11,484,255	22,828,786	22,828

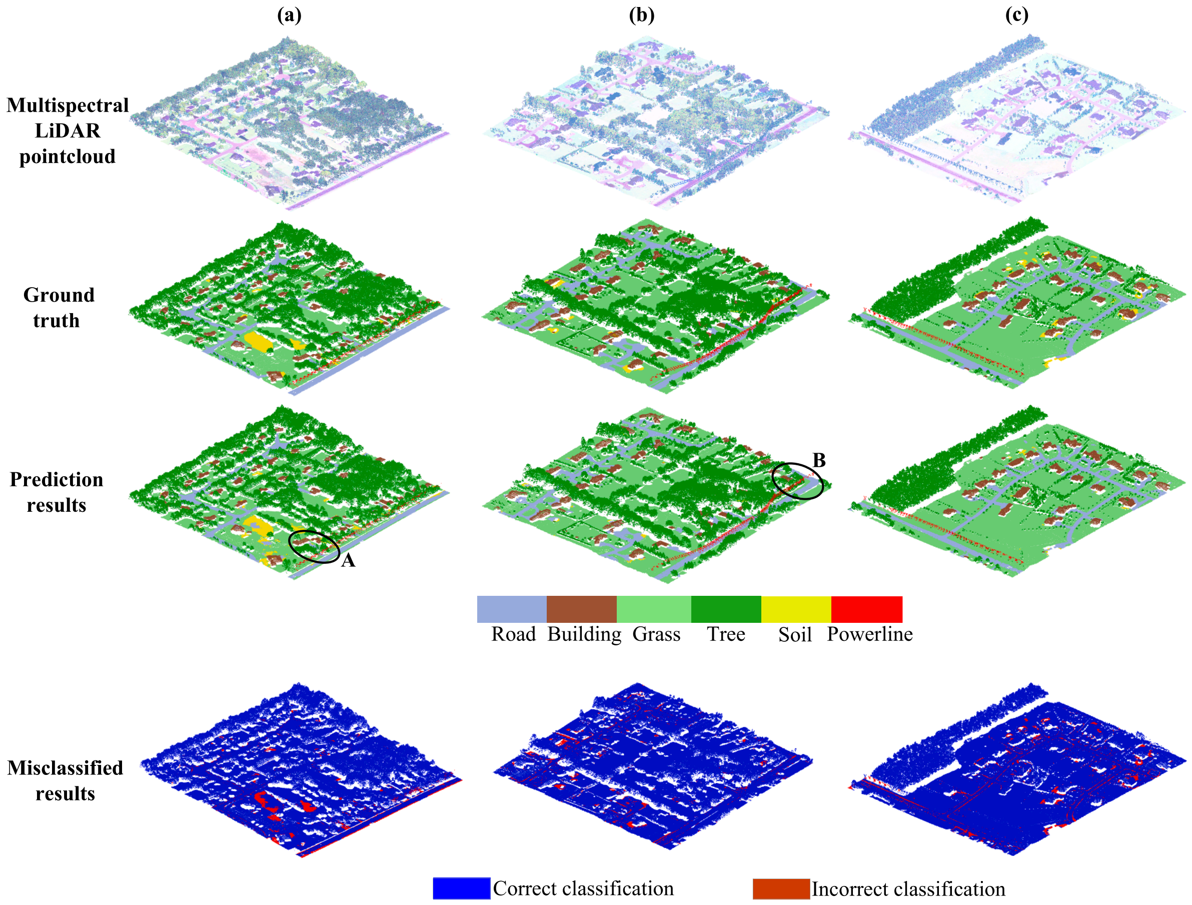


Fig. 5. Classification results obtained by the FPWS-Net on the WS dataset, (a) Area 11, (b) Area 12, and (c) Area 13.

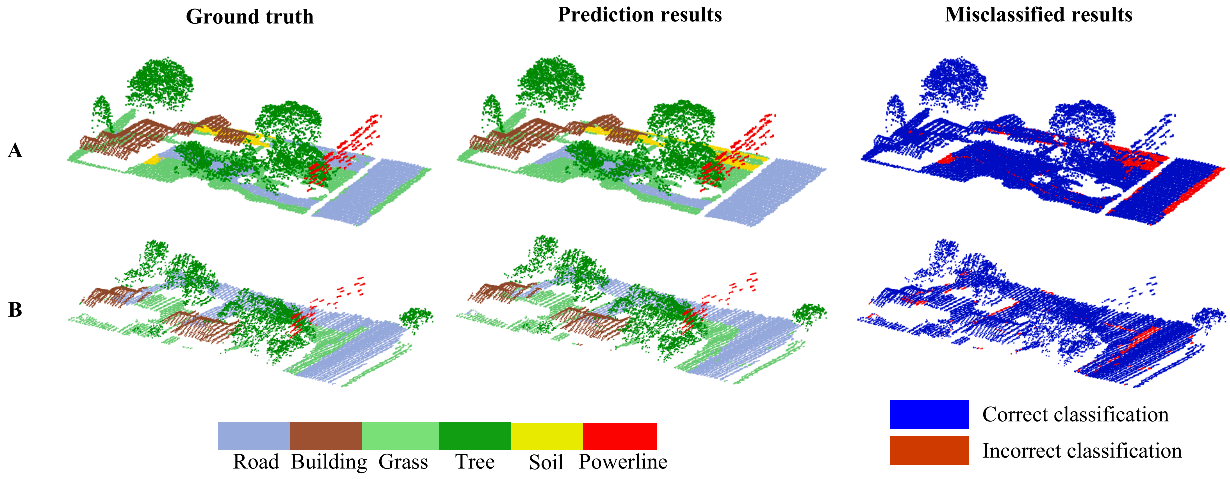


Fig. 6. Close-up view of the MS-LiDAR pointcloud classification results, where A and B represent detailed views of the two areas marked in Fig. 5.

Table 4
Quantitative results obtained by the FPWS-Net on the WS dataset.

	F_1-score (%)						Average F_1-score (%)	mIoU (%)	OA (%)
	Road	Building	Grass	Tree	Soil	Powerline			
Area 11	86.67	98.77	94.81	99.77	40.53	86.45	84.50	77.54	95.86
Area 12	88.15	98.66	97.14	99.40	23.42	98.48	84.21	79.95	96.44
Area 13	85.02	99.15	96.91	99.22	22.16	91.73	82.36	76.98	95.55

FPWS-Net on the WS dataset. As listed in Table 4, our method achieved the pointcloud classification accuracies of an average F_1-score of over 82.36 %, an mIoU of over 76.98 %, and an OA of over 95.55 % on the three test scenes. Among them, five classes, including road, building, grass, tree, and powerline, achieved an F_1-score of over 85.02 %. Particularly, the FPWS-Net obtained the highest F_1-scores of 99.15 %, 97.14 %, and 99.77 %, respectively, for classes building, grass, and tree. This is mainly because the WS dataset, which was collected in July, provided rich vegetation information, improving the classification accuracies of classes grass and tree. Additionally, for buildings with distinct geometric structures, the FPWS-Net was capable of learning their rich neighboring features from the data. The KPConv also provided more supervised signals from the neighborhoods of labeled points, which improves MS-LiDAR pointcloud classification accuracy.

However, for class soil, we obtained poor classification performance with the lowest F_1-score reaching only 22.16 %. Learning the feature differences between class soil and other classes from a small fraction of labeled soil points poses significant challenges for the model, resulting in poor MS-LiDAR pointcloud classification performance for class soil.

4.2.2. Classification results of Tobermory dataset

To further demonstrate the applicability of the FPWS-Net on MS-LiDAR pointcloud data, we conducted classification experiments on the Tobermory dataset. Fig. 7 presents the pointcloud classification results on the two test scenes of the Tobermory dataset. Visual inspection indicates that the FPWS-Net obtained promising classification results for Areas 9 and 12. Specifically, the FPWS-Net performed well when identifying classes building, land, tree, water, and powerline. Notably, even without clear boundaries between classes water and land, the FPWS-Net can effectively distinguish between them due to their distinct spectral differences. Comparatively, due to similarities in spectral and geometric features, some road points have been misclassified as land points.

Table 5 reports the quantitative evaluation results obtained by the proposed FPWS-Net on the two test scenes of the Tobermory dataset. As shown in Table 5, the FPWS-Net obtained an average F_1-score of 93.90 %, an mIoU of 88.83 % and an OA of 97.52 % for Area 9, and an average F_1-score of 91.06 %, an mIoU of 84.55 % and an OA of 96.88 % for Area 12. Specifically, the F_1-scores for classes tree and water are both above 98.65 % because, even using randomly sparse labeling, the two classes

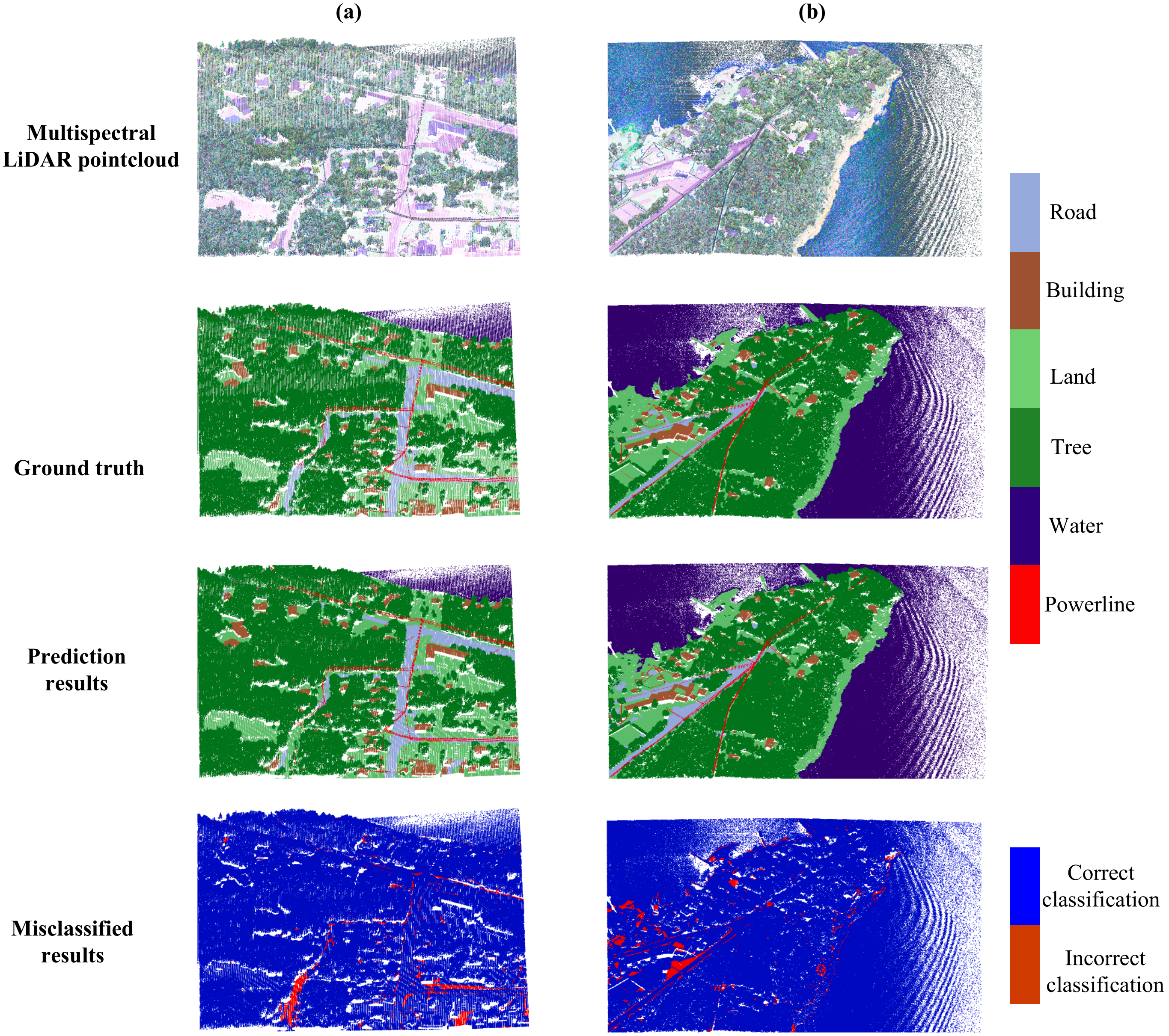


Fig. 7. Qualitative results obtained by the FPWS-Net on the Tobermory dataset, (a) Area 9 and (b) Aera 12.

Table 5
Quantitative results obtained by our FPWS-Net on the Tobermory dataset.

	F_1-score (%)						Average F_1-score (%)	mIoU (%)	OA (%)
	Road	Building	Land	Tree	Water	Powerline			
Area 9	86.76	92.21	94.78	99.62	98.71	91.32	93.90	88.83	97.52
Area 12	74.62	88.56	92.08	98.65	98.85	93.59	91.06	84.55	96.88

contained a large number of points and distinct spectral differences. However, for class road, the FPWS-Net obtained the lowest F_1-score, reaching only 86.76 % and 74.62 %, respectively, for the two test scenes. This can be attributed to the road points having geometric features that are similar to both land and water, coupled with the absence of distinctive spectral information. Overall, the FPWS-Net effectively utilizes both the geometric and spectral information of airborne MS-LiDAR pointclouds, resulting in good classification using only 0.1 % labeled points.

4.3. Comparative studies

To further analyze the feasibility and superiority of the proposed FPWS-Net, a series of comparative experiments were performed with some SOTA networks on the WS dataset. The selected comparative methods included two categories: fully supervised and weakly supervised. In the fully supervised category, we selected seven SOTA networks, i.e., PointNet++, DGCNN, RSCNN, KPFCNN (Thomas et al., 2019), RandLA-Net, SE-PointNet++, and FR-GCNet. Among them, the first five methods are the commonly used 3D pointcloud processing models, and the pointcloud classification results were obtained by their official source codes. The last two networks, i.e., SE-PointNet++ and FR-GCNet, are recently developed SOTA airborne MS-LiDAR pointcloud classification networks, and their classification results were obtained in the literature (Jing et al., 2021; Zhao et al., 2021).

Table 6 lists the MS-LiDAR pointcloud classification results obtained by the comparative methods on the WS dataset. As seen in Table 6, for the fully supervised methods, KPFCNN and RandLA-Net performed well on the WS dataset and demonstrated obviously more advantageous pointcloud classification accuracies than the other networks, achieving more than 79.17 % average F_1-score, 73.42 % mIoU, and 95.45 % OA, while PointNet++, DGCNN, and RSCNN behaved similarly and obtained relatively lower pointcloud classification performance with an average F_1-score ranging from 71.47 % to 73.90 %. This is primarily because the advanced model design and learning strategies of KPFCNN and RandLA-Net enabled them to better capture the features of pointclouds, improving the MS-LiDAR pointcloud classification accuracy. The two fully supervised classification networks specifically developed for airborne MS-LiDAR pointclouds, i.e., SE-PointNet++ and FR-GCNet, got the middle pointcloud classification accuracies with an average F_1-score of 75.84 % and 78.61 %, respectively. Because SE-PointNet and FR-GNet were built upon PointNet++ and DGCNN, respectively, they accommodated the characteristics of MS-LiDAR data into the networks. However, they were unable to effectively handle large-scale pointclouds due to the limited receptive fields of the models, thereby preventing them from achieving classification accuracy comparable to those of KPFCNN and RandLA-Net.

Although our FPWS-Net performed MS-LiDAR pointcloud classification using only 0.1 % labeled data, it obtained an average F_1-score of 83.69 %, an mIoU of 78.81 %, and an OA of 95.97 %. Compared with SE-PointNet++ and FR-GCNet, our FPWS-Net yielded an improvement of 7.85 %, 14.49 %, and 2.96 % for the average F_1-score, mIoU, and OA, respectively, and a gain of 5.08 %, 13.03 %, 2.42 %, respectively. Compared with the fully supervised method, i.e., RandLA-Net, the FPWS-Net achieved improvements of 2.34 %, 5.39 %, and 0.4 % for the average F_1-score, mIoU, and OA. Moreover, the FPWS-Net achieved the best classification performance on three classes, i.e., road, grass, and powerline among all fully supervised methods. It is noteworthy that the FPWS-Net achieved overall classification accuracy surpassing that of fully supervised methods, primarily attributable to the incorporation of KPConv. KPConv furnished the network with additional supplementary signals, particularly excelling in processing features with unique geometrical characteristics, such as buildings and powerlines, which effectively compensated for the deficiencies of baseline in these respects.

Because few weakly supervised methods have been developed for airborne MS-LiDAR pointclouds, we selected two recently proposed weakly supervised networks, i.e., SQN and PSD, for MLS or single-band ALS pointclouds for comparison. Additionally, we also selected two mainstream backbones, i.e., RandLA-Net and KPFCNN, to conduct weakly supervised experiments using 0.1 % labeled points. Notably, for a fair comparison, all experiments were configured with the same training parameters and the same 0.1 % sparse labels.

As shown in Table 6, our FPWS-Net achieved the best average of F_1-score, mIoU, and OA in all the weakly supervised methods. The two baselines, i.e., RandLA-Net and KPFCNN, behaved similarly and obtained lower pointcloud classification accuracies than the other weakly supervised methods. To be specific, compared with our FPWS-Net, the RandLA-Net obtained a performance degradation of 18.09 %, 18.67 %, and 23.41 % for the average F_1-score, mIoU, and OA, respectively, when using 0.1 % labels. SQN, which was built upon the RandLA-Net baseline,

Table 6
Quantitative results of different fully supervised methods and weakly supervised methods on the WS dataset. The underlined represents the best scores among all methods, and the bold represents the best scores of FPWS-Net compared to all other methods.

Method		F_1-score (%)						Average F_1-score (%)	mIoU (%)	OA (%)
		Road	Building	Tree	Grass	Soil	Powerline			
Fully supervised	PointNet++	73.91	83.98	86.64	96.74	30.24	57.28	71.47	55.84	90.19
	DGCNN	70.43	90.25	93.62	97.93	21.97	55.24	71.57	52.04	91.36
	RSCNN	71.18	89.00	91.42	95.63	26.43	70.03	73.90	56.10	92.44
	KPFCNN	83.41	99.39	95.53	99.35	5.75	91.59	79.17	76.09	95.45
	RandLA-Net	84.72	93.47	96.34	99.12	31.68	82.75	81.35	73.42	95.57
	SE-PointNet++	70.32	85.64	94.70	97.05	37.02	70.35	75.84	64.32	93.01
	FR-GCNet	82.63	90.81	95.33	98.77	28.72	74.11	78.61	65.78	93.55
Weakly supervised (0.1 %)	KPFCNN	44.74	98.33	60.85	99.38	10.92	92.69	67.83	60.19	73.22
	RandLA-Net	58.70	97.57	65.79	99.06	2.24	70.21	65.60	60.14	72.56
	SQN	75.21	98.93	93.05	99.34	3.70	71.73	70.66	67.20	93.15
	PSD	69,74	97.98	90.24	99.23	1.45	75.42	72.34	65.26	90.84
	Ours	**86.61**	98.86	96.29	**99.46**	28.70	92.22	**83.69**	**78.81**	**95.97**

outperformed the RandLA-Net in MS-LiDAR pointcloud classification, with a gain of 5.06 %, 7.06 %, and 20.59 % for the average F_1-score, mIoU, and OA, respectively. This is because the point feature querying network of the SQN strongly leveraged the semantic similarity of local points, implicitly enhancing sparse supervised signals. However, SQN lacked of sufficient unlabeled data, which hindered its performance on MS-LiDAR LiDAR pointcloud classification tasks. Comparatively, the FPWS-Net not only provided rich bottom-up supervised signals to the model through KPConv and the auxiliary semantic inference module but also effectively converted a large number of unlabeled points into auxiliary supervised signals using the mutual pseudo-labeling loss, thereby obtaining a gain of 13.03 %, 11.61 %, and 2.82 % for the average F_1-score, mIoU, and OA.

PSD obtained the commendable MS-LiDAR pointcloud classification performance using 0.1 % labels. This is because PSD constrained the data perturbations generated twice from input pointclouds, providing additional auxiliary supervised signals to the model. In addition, GCN used in the PSD helped explore more underlying supervised signals. Compared with PSD, the FPWS-Net obtained a gain of 11.35 %, 13.55 %, and 5.13 % for the average F_1-score, mIoU, and OA. The reason might be that GCN only modelled the feature distances of neighboring points and poorly aggregated their location information, while the KPConv in our FPWS-Net assigned different weights to the kernel points based on their distances to neighboring points, aggregating positional distances and feature distances of neighboring points, providing more accurate classification cues for the model, and finally improving pointcloud classification accuracies.

As illustrated in Fig. 8, our FPWS-Net, exploring its exceptional capability for extracting underlying semantic features and efficiently utilizing unlabeled data, achieved classification performance significantly surpassing those of the SQN and PSD weakly supervised methods. Notably, there are two examples marked by dashed and solid oval-shaped areas in Fig. 8. In the dashed oval-shaped area, our FPWS-Net correctly distinguished linearly-structured powerline points from their surrounding tree points, whereas SQN and PSD incorrectly identified powerline points as tree points. In the solid oval-shaped area, our FPWS-Net accurately recognized road points with distinct boundary lines. In contrast, PSD misclassified some road points as tree points, while SQN obtained the worst classification performance by misclassifying the majority of road points as tree and soil points.

Moreover, the FPWS-Net obtained the best F_1-score values of 86.61 %, 96.29 %, 99.46 %, and 28.70 %, respectively, for class road, tree, grass, and soil, among all weakly supervised methods. Thus, it demonstrates that the proposed FPWS-Net can achieve the MS-LiDAR pointcloud classification accuracy comparable to the fully supervised methods when using 0.1 % labeled points and outperformed the weakly supervised methods.

4.4. Ablation studies

In our FPWS-Net, the modules, i.e., the mutual pseudo-labeling loss, the consistency constraint and the KPConv, contributed positively and significantly to the enhancement of the quality of both the feature representation and the pointcloud classification. Thereby, we intently tested the advanced superiorities of these three modules to the improvement of the airborne MS-LiDAR pointcloud classification accuracies. To this end, we performed a set of ablation experiments on the WS dataset to analyze these important components. To better analyze the contribution of each component, 0.1 % labeled points were used in all the experiments. First, we removed the mutual pseudo-labeling loss from the FPWS-Net, and the resultant network was termed as Model A. Second, we removed the consistency constraint from the FPWS-Net, and the modified network was termed as Model B. Finally, we replaced KPConv with GCN, and the modified network was termed as Model C. These models were trained using the same data, strategies, and hyperparameter settings. Table 7 details the MS-LiDAR pointcloud classification results obtained by these modified models on the WS dataset. The metrics, i.e., average F_1-score and OA, were also used for quantitative comparisons and analyses.

As seen in Table 7, Model A, by removing the mutual pseudo-labeling loss, behaved less promisingly with a slight decrease of 2.24 % and 0.97 % for the average F_1-score and OA, respectively. The reason is that

Table 7
Quantitative results obtained by the modified networks on the WS dataset.

Model	Mutual pseudo-labeling loss	Consistency constraint	KPConv	GCN	Average F_1-score (%)	OA (%)
Ours	√	√	√		83.69	95.97
A		√	√		81.45	95.00
B	√		√		82.21	95.44
C	√	√		√	79.60	94.99

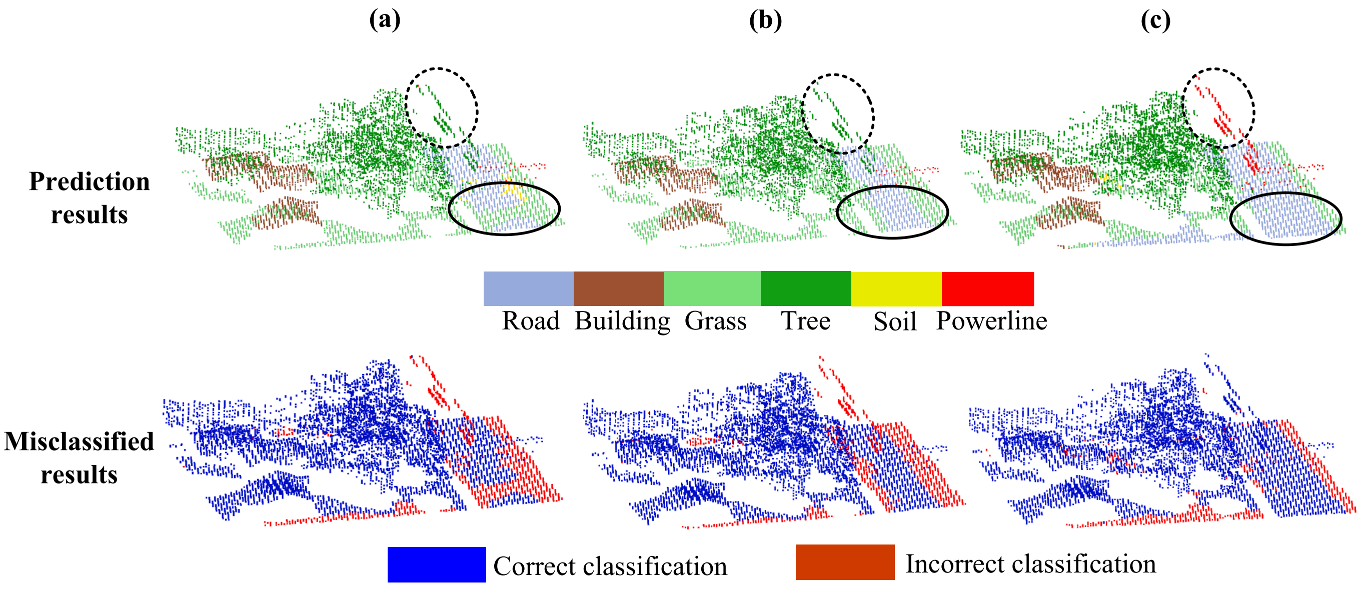


Fig. 8. Detailed visualization of the comparison methods, (a) SQN, (b) PSD, and (c) the proposed FPWS-Net.

without the mutual pseudo-labeling loss, the large numbers of unlabeled points were not fully exploited to provide additional supervised signals for better learning data features and class distributions. Only relying on the training data randomly selected from the spares labeled points limited the model performance. Model B, by abandoning the consistency constraint, obtained a degradation of 1.48 % and 0.53 % for the average F_1-score and OA, respectively. This is because that model B without the consistency constraint, lacked of unified supervised signals on the feature perturbations of the two semantic inference modules during the learning process. This led to the two semantic inference modules learning inconsistent feature representations, thereby resulting in a decrease in classification accuracy. The mutual pseudo-labeling loss and the consistency constraint expanded the extent and distribution of training data and constrained the prediction differences between the two semantic inference modules, respectively, providing the model with additional supervised signals and increasing the MS-LiDAR pointcloud classification performance.

Model C, by replacing KPConv with GCN, performed less poorly with an average F_1-score decrease of 4.09 % and an OA decrease of 0.98 % in MS-LiDAR pointcloud classification tasks. This can be attributed to the advantages of KPConv over GCN. KPConv assigned weights to the kernel points based on their distances to points, ensuring the kernel points to capture more useful features and providing more accurate supervised signals for the model. Moreover, the learned kernel point weight matrix properly allocated weights to each kernel point during model training. On contrary, GCN only modeled the local relationships between the high-dimensional point features regarding feature distance, thereby neglecting the crucial geometric distances between points.

To further validate the effectiveness of the KPConv in the FPWS-Net, we embedded it into RandLA-Net and SQN, respectively, named as RandLA-Net-KPC and SQN-KPC. In the RandLA-Net, the KPConv was used to construct context-aware information for all points. In the SQN, only the features of labeled points were utilized for semantic inference. Table 8 lists the MS-LiDAR pointcloud classification results obtained by these modified methods on the WS dataset.

As listed in Table 8, the classification performance was improved when embedding the KPConv into the networks. Specifically, compared with RandLA-Net and SQN, the RandLA-Net-KPC and SQN-KPC obtained an average F_1-score gain of 2.91 % and an OA gain of 19.9 %, as well as an average F_1-score gain of 3.61 % and an OA gain of 2.27 %, respectively. This is because, by embedding the KPConv into the RandLA-Net and the SQN, the RandLA-Net-KPC and the SQN-KPC promoted the feature semantic quality, thereby leading to the improvement of MS-LiDAR pointcloud classification. Moreover, our FPWS-Net achieved a higher average F_1-score of 15.18 % and 9.42 %, and a higher OA of 3.51 % and 0.55 % than those of the RandLA-Net-KPC and SQN-KPC, respectively. This demonstrates that constraining the feature perturbations generated by the two semantic inference modules effectively enhances the stability and classification performance of the model.

To further discuss the influence of the number of labeled points on the MS-LiDAR pointcloud classification performance, we varied the number of labeled points with four labeling ratios, i.e., 0.05 %, 0.1 %, 1 %, and 10 %. Fig. 9(a) illustrates the MS-LiDAR pointcloud classification accuracies at the four labeling ratios. The average F_1-score and OA metrics were also leveraged for quantitative analyses. As seen in Fig. 9 (a), the pointcloud classification accuracies increased with the increase of the labeling ratio. To be specific, a rapid increase occurred when the ratio changed from 0.05 % to 0.1 %, indicating that less than 0.1 % labeled points may be insufficient for the network to learn adequately features. Although the pointcloud classification accuracies improved when the ratio changed from 0.1 % to 10 %, this improvement became insignificant. This demonstrates that more labeled points provide the network with more supervised signals, allowing it to better learn the underlying feature patterns. Therefore, to trade-off the annotation efforts and model performance, 0.1 % labeling ratio was suitable for the weakly supervised networks in MS-LiDAR pointcloud classification tasks.

Table 8
Quantitative results obtained by different methods on the WS dataset.

Model	Average F_1-score (%)	OA (%)
RandLA-Net	65.60	72.56
RandLA-Net-KPC	68.51	92.46
SQN	70.66	93.15
SQN-KPC	74.27	95.42
Ours	83.69	95.97

To validate the accuracy variability of the FPWS-Net using different labeled points, we repetitively trained the FPWS-Net six times, which means the randomly selected 0.1 % labeled points at each time were different for training. The experimental results are presented in Fig. 9 (b). It is observable that there are slight but not significant differences among the six classification results, which indicates that the proposed FPWS-Net exhibits the robustness to the randomly selected labeled points.

5. Conclusion

This paper presented a novel weakly-supervised deep learning framework, termed as FPWS-Net, for MS-LiDAR pointcloud classification tasks. The FPWS-Net innovatively employed a dual semantic inference structure including a primary semantic inference module and an auxiliary semantic inference module. To be specific, the feature perturbance generated by the primary and auxiliary semantics inference modules were minimized to enhance model stability by a consistency constraint loss and a mutual pseudo-labeling loss. Moreover, a KPConv was embedded into the network to fully explore the geometric and spectral feature relationships between local points in a high-dimensional feature space, which provided more features and supervisory signals for training the network.

The proposed FPWS-Net was tested on two MS-LiDAR pointcloud datasets, and obtained at least an average F_1-score of 83.69 %, an mIoU of 78.81 %, and an OA of 95.97 %, using only 0.1 % labeled points. All experiments also indicated the generalization and effectiveness of the FPWS-Net in MS-LiDAR pointcloud classification tasks. However, the FPWS-Net failed to deal with some classes, such as soil, due to the lack of abundant supervised signals. In the future, we will focus on obtaining more effective supervised signals and more contextual information to achieve higher classification performance using less data labeling.

Funding sources

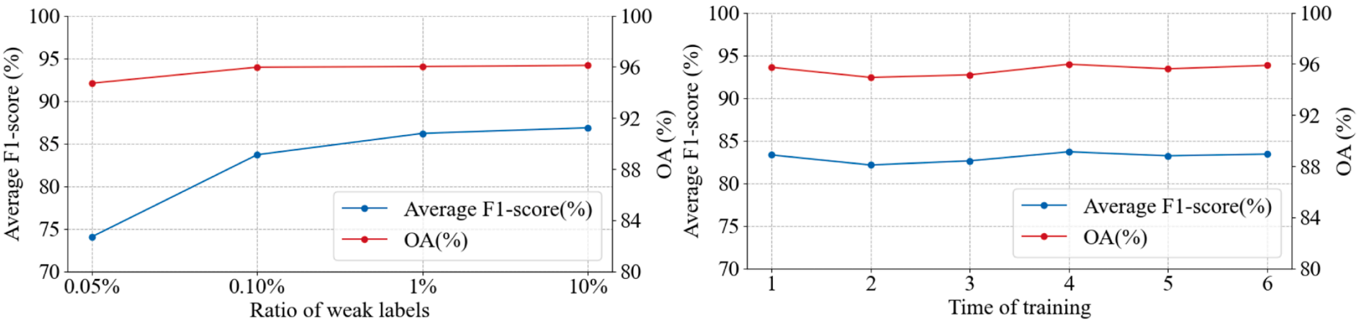
This research was partially funded by the National Natural Science Foundation of China under Grants No. 42371447, 62076107, and 4200140, and partially funded by Key Laboratory of Land Satellite Remote Sensing Application, Ministry of Natural Resources of the People’s Republic of China (Grant No. KLSMNR-G202305)”.

CRediT authorship contribution statement

Ke Chen: Writing – review & editing, Writing – original draft, Project administration, Methodology, Conceptualization. **Haiyan Guan:** Writing – review & editing, Writing – original draft, Software, Methodology, Funding acquisition, Conceptualization. **Lanying Wang:** Writing – review & editing, Data curation. **Yongtao Yu:** Writing – review & editing, Data curation. **Yufu Zang:** Writing – review & editing, Data curation. **Nannan Qin:** Writing – review & editing. **Jiacheng Liu:** Writing – review & editing. **Jonathan Li:** Supervision, Resources.

Declaration of competing interest

The authors declare that they have no known competing financial



(a)

(b)

Fig. 9. Sensitivity analysis of FPWS-Net with different labeled points, (a) classification accuracies using different labeled ratios, and (b) classification accuracies using different labeled points.

interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

References

Bakuła, K., 2015. Multispectral airborne laser scanning-a new trend in the development of LiDAR technology. Archiwum Fotogrametrii, Kartografii i Teledetekcji 27. 10.14681/afkit.2015.002.

Berman, M, Triki, A.R., Blaschko, M.B., 2018. The lovasz-softmax loss: a tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In: Proc. CVPR, pp. 4413–4421. doi: 10.1109/CVPR.2018.00464.

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. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLIII-B3-2020, 501–505. doi: 10.5194/isprs-archives-XLIII-B3-2020-501-2020.

Ekhtari, N., Glennie, C., Fernandez-Diaz, J.C., 2018. Classification of airborne multispectral LiDAR point clouds for land cover mapping. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 11, 2068–2078. https://doi.org/10.1109/JSTARS.2018.2835483.

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.

Gong, W., Sun, J., Shi, S., Yang, J., Du, L., Zhu, B., Song, S., 2015. Investigating the potential of using the spatial and spectral information of multispectral LiDAR for object classification. Sensors 15 (9), 21989–22002. https://doi.org/10.3390/s150921989.

Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., Markham, A., 2020. RandLA-Net: efficient semantic segmentation of large-scale point clouds. In: Proc. CVPR, pp. 11105–11114. doi: 10.1109/CVPR42600.2020.01112.

Hu, Q., Yang, B., Fang, G., Guo, Y., Leonardis, A., Trigoni, N., Markham, A., 2021. SQN: Weakly-supervised semantic segmentation of large-scale 3D point clouds with 1000x fewer labels. In arXiv preprint arXiv: 2104.04891. https://arxiv.org/abs/2104.04891.

Jiang, L., Shi, S., Tian, Z., Lai, X., Liu, S., Fu, C.-W., Jia, J., 2021. Guided point contrastive learning for semi-supervised point cloud semantic segmentation. In: Proc. ICCV, pp. 6403–6412. doi: 10.1109/ICCV48922.2021.00636.

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. (Basel) 13, 2516. https://doi.org/10.3390/rs13132516.

Li, M., Xie, Y., Shen, Y., Ke, B., Qiao, R., Ren, B., Lin, S., Ma, L., 2022b. HybridCR: Weakly-supervised 3D point cloud semantic segmentation via hybrid contrastive regularization. In: Proc, CVPR, pp. 14910–14919. doi: 10.1109/CVPR52688.2022.01451.

Li, D., Shen, X., Guan, H., Yu, Y., Wang, H., Zhang, G., Li, J., Li, D., 2022a. AGFP-Net: attentive geometric feature pyramid network for land cover classification using airborne multispectral LiDAR data. Int. J. Appl. Earth Obs. Geoinf. 108, 102723 https://doi.org/10.1016/j.jag.2022.102723.

Liu, Y., Fan, B., Xiang, S., Pan, C., 2019. Relation-shape convolutional neural network for point cloud analysis. In: Proc. CVPR, pp. 8887–8896. doi: 10.1109/CVPR.2019.00910.

Matikainen, L., Karila, K., Hyyppä, J., Puttonen, E., Ahokas, E., 2017. Feasibility of multispectral airborne laser scanning for land cover classification, road mapping and map updating. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLII-3/W3, 119–122. https://doi.org/10.5194/isprs-archives-XLII-3-W3-119-2017.

Matikainen, L., Karila, K., Litkey, P., Ahokas, E., Hyypp, J., 2020. Combining single photon and multispectral airborne laser scanning for land cover classification. ISPRS J. Photogramm. Remote Sens. 164, 200–216. https://doi.org/10.1016/j.isprsjprs.2020.04.021.

Morsy, S., Shaker, A., El-Rabbany, A., LaRocque, P.E., 2016. Airborne multispectral LiDAR data for land-cover classification and land/water mapping using different spectral indexes. ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. III–3, 217–224. https://doi.org/10.5194/isprsannals-III-3-217-2016.

Qi, C.R., Yi, L., Su, H., Guibas, L.J., 2017. PointNet++: deep hierarchical feature learning on point sets in a metric space. In: Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 2017 December, pp. 5099–5108. http://arxiv.org/abs/1706.02413.

Qiu, S., Anwar, S., Barnes, N., 2021. Semantic segmentation for real point cloud scenes via bilateral augmentation and adaptive fusion. In: Proc. CVPR, pp. 1757–1767. doi: 10.48550/arXiv.2103.07074.

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. https://doi.org/10.1016/j.isprsjprs.2019.04.005.

Tao, A., Duan, Y., Wei, Y., Lu, J., Zhou, J., 2022. SegGroup: Seg-level supervision for 3D instance and semantic segmentation. IEEE Trans. Image Process. 31, 4952–4965. https://doi.org/10.1109/TIP.2022.3190709.

Thomas, H., Qi, C.R., Deschaud, J.-E., Marcotegui, B., Goulette, F., Guibas, L.J., 2019. KPConv: flexible and deformable convolution for point clouds. In: Proc. ICCV, pp. 6411–6420. doi: 10.1109/ICCV.2019.00651.

Wang, H., Rong, X., Yang, L., Feng, J., Xiao, J., Tian, Y., 2020. Weakly supervised semantic segmentation in 3D graph-structured point clouds of wild scenes. In arXiv preprint arXiv:2004.12498. https://arxiv.org/abs/2004.12498.

Wang, Q., Gu, Y., 2020. A discriminative tensor representation model for feature extraction and classification of multispectral LiDAR data. IEEE Trans. Geosci. Remote Sensing 58 (3), 1568–1586. https://doi.org/10.1109/TGRS.2019.2947081.

Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M., 2019. Dynamic graph CNN for learning on point clouds. ACM Trans. Graph. 38 (5), 146. https://doi.org/10.1145/3326362.

Wei, J., Lin, G., Yap, K.-H., Hung, T.-Y., Xie, L., 2020. Multi-path region mining for weakly supervised 3D semantic segmentation on point clouds. In: Proc. CVPR, pp. 4383–4392. doi: 10.1109/CVPR42600.2020.00444.

Xu, X., Lee, G.H., 2020. Weakly supervised semantic point cloud segmentation: Towards 10× fewer labels. In: Proc. CVPR, pp. 13703–13712. doi: 10.1109/CVPR42600.2020.01372.

Yan, W.Y., van Ewijk, K., Treitz, P., Shaker, A., 2020. Effects of radiometric correction on cover type and spatial resolution for modeling plot level forest attributes using multispectral airborne LiDAR data. ISPRS J. Photogramm. Remote Sens. 169, 152–165. https://doi.org/10.1016/j.isprsjprs.2020.09.001.

Yokoya, N., Ghamisi, P., Xia, J., Sukhanov, S., Heremans, R., Tankoyeu, I., Bechtel, B., Le Saux, B., Moser, G., Tuia, D., 2018. Open data for global multimodal land use classification: outcome of the 2017 IEEE GRSS Data Fusion Contest. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 11, 1363–1377. https://doi.org/10.1109/JSTARS.2018.2799698.

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. https://doi.org/10.1109/LGRS.2019.2940505.

Yu, Y., Liu, C., Guan, H., Wang, L., Gao, S., Zhang, H., Zhang, Y., Li, J., 2022. Land cover classification of multispectral LiDAR data with an efficient self-attention capsule network. IEEE Geosci. Remote Sens. Lett. 19, 1–5. https://doi.org/10.1109/LGRS.2021.3071252.

Zhang, Y., Li, Z., Xie, Y., Qu, Y., Li, C., Mei, T., 2021a. Weakly supervised semantic segmentation for large-scale point cloud. In: Proc. AAAI 35(4), 3421–3429. doi: 10.1609/aaai.v35i4.16455.

Zhang, Y., Qu, Y., Xie, Y., Li, Z., Zheng, S., Li, C., 2021b. Perturbed self-distillation: Weakly supervised large-scale point cloud semantic segmentation. In: Proc. ICCV, pp. 15500–15508. doi: 10.1109/ICCV48922.2021.01523.

Zhao, P., Guan, H., Li, D., Yu, Y., Wang, H., Gao, K., Marcato Junior, J., Li, J., 2021. Airborne multispectral LiDAR point cloud classification with a feature reasoning-based graph convolution network. Int. J. Appl. Earth Obs. Geoinf. 105, 102634 https://doi.org/10.1016/j.jag.2021.102634.