Received April 19, 2020, accepted April 28, 2020, date of publication May 6, 2020, date of current version May 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2992612

CSPC-Dataset: New LiDAR Point Cloud Dataset and Benchmark for Large-Scale Scene Semantic Segmentation

GUOFENG TONG[1], YONG LI[1], DONG CHEN[2], (Member, IEEE), QI SUN[1], WEI CAO[2], AND GUIQIU XIANG[2]

[1]College of Information Science and Engineering, Northeastern University, Shenyang 110819, China
[2]College of Civil Engineering, Nanjing Forestry University, Nanjing 210037, China

Corresponding author: Yong Li (leoqiulin@126.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 41971415 and Grant 61175031, and in part by the National Key Technology Research and Development Program of the Ministry of Science and Technology of China under Grant 2015BAF13B00-5.

ABSTRACT Large-scale point clouds scanned by light detection and ranging (lidar) sensors provide detailed geometric characteristics of scenes due to the provision of 3D structural data. The semantic segmentation of large-scale point clouds is a crucial step for an in-depth understanding of complex scenes. Of late, although a large number of point cloud semantic segmentation algorithms have been proposed, semantic segmentation methods are still far from being satisfactory in terms of precision and accuracy of large-scale point clouds. For machine learning (ML) and deep learning (DL) methodologies, the semantic segmentation is largely influenced by the quality of training sets and methods themselves. Therefore, we construct a new point cloud dataset, namely CSPC-Dataset (Complex Scene Point Cloud Dataset) for large-scale scene semantic segmentation. CSPC-Dataset point clouds are acquired by a wearable laser mobile mapping robot. It covers five complex urban and rural scenes and mainly includes six types of objects, i.e., ground, car, building, vegetation, bridge, and pole. It provides large-scale outdoor scenes with color information, which has advantages such as the scene more complete, point density relatively uniform, diversity and complexity of objects and the high discrepancy between different scenes. Based on the CSPC-Dataset, we construct a new benchmark, which includes approximately 68 million points with explicit semantic labels. To extend the dataset into a wide range of applications, this paper provides the semantic segmentation results and comparative analysis of 7 baseline methods based on CSPC-Dataset. In the experiment part, three groups of experiments are conducted for benchmarking, which offers an effective way to make comparisons with different point-labeling algorithms. The labeling results have shown that the highest Intersection over Union (IoU) of pole, ground, building, car, vegetation, and bridge for all benchmarks is 36.0%, 97.8%, 93.7%, 65.6%, 92.0%, and 69.6%.

INDEX TERMS LiDAR, benchmark, point clouds, large-scale datasets, scene understanding.

I. INTRODUCTION

In recent years, machine learning (ML) and deep learning (DL) algorithms have achieved excellent performance in many fields. To the best of our knowledge, the applications of deep learning methods rely on a large number of data with labeled information for model learning. Therefore, it is of great significance to construct the labeled public datasets for the development of deep learning algorithms. For example, ImageNet [1], Pascal VOC [2] and COCO [3] play an important role in the development of deep learning algorithms in the field of 2D vision. Accompanying with recent advancements in point cloud acquisition sensors, such as LiDAR and RGB-D cameras, the number of point cloud data is increasing rapidly, and point clouds are widely used in unmanned driving [4], urban planning, and digital city [5], among others. Currently, machine learning and deep learning methods have been extensively used for point cloud labeling processing,

The associate editor coordinating the review of this manuscript and approving it for publication was Sudipta Roy.

which requires a certain degree of point clouds for training. However, there are few datasets containing large-scale scenes and varied objects for point cloud semantic labeling. In addition, most common datasets are indoor data or CAD models, such as NYU Depth v2 [6] and ModelNet10 [7]. Existing outdoor point cloud data, such as Sydney Urban Objects [8] Oakland [9], Paris-rue-Madame [10], Paris-Lille-3D [11] are relatively sparse, and lacking color information. From the perspective of scene understanding, there is a lack of large-scale point cloud datasets with object diversity, scene complexity, color information, dense points, and explicit semantic labels. Point cloud semantic segmentation is the key step for scene understanding tasks. Therefore, it is of vital significance to construct large-scale point cloud datasets with rich information. Based on a high-quality dataset, the corresponding benchmark dataset can be generated based on the state-of-the-art point cloud semantic segmentation algorithms. The generated benchmarks can undoubtedly promote the development of deep learning algorithms and their applications.

To promote large-scale point cloud scene understanding based on ML and DL methods, this paper improves the backpacked mobile laser scanning system [12] and uses a new backpacked mobile laser scanning mapping robot to obtain large-scale, complete and colored point clouds. After that, the point clouds are labeled manually. Thus, a new large-scale outdoor point cloud dataset, namely CSPC-Dataset (Complex Scene Point Cloud Dataset), is built for scene understanding, especially for point cloud scene semantic segmentation. The dataset contains approximately 68 million points, including the six classes of objects such as ground, building, car, bridge, vegetation, and pole. The scene of CSPC-Dataset covers a wide range of urban and rural scenes: streets, campuses, farmlands, residential regions, commercial buildings, etc. In contrast to other point cloud datasets, the CSPC-Dataset is the first dataset collected by a backpacked mapping robot. The dataset is dense, complete, with color information and diversity objects making it an appropriate dataset to evaluate semantic segmentation algorithms for large-scale point clouds. To easily access the dataset, we have released the CSPC-Dataset online using the Baidu cloud network disk.[1] Besides, seven state-of-the-art point cloud semantic segmentation algorithms, three of which are based on ML and four of which are based on DL, are used as the baselines of the constructed benchmark. To help readers use the constructed benchmarks easily, seven metrics are used to evaluate the performance of point cloud classification, and three groups of experiments are conducted to compare the effect of baselines.

To show the current research status of point cloud datasets and benchmarks more clearly, this section mainly analyzes the point cloud types, the existing point datasets and benchmarks, and the existing semantic labeling algorithms.

A. POINT CLOUD DATASETS AND BENCHMARKS

Different point cloud acquisition devices can acquire different types and characteristics of point clouds. As shown in Fig. 1, the point clouds obtained by different point cloud acquisition devices have obvious differences. These diverse 3D point clouds have their advantages and disadvantages. Refer to Appendix A for further details about the different types of point clouds.

According to the comparisons in Appendix A and Fig. 1, it is obvious to be found that colored 3D point clouds include rich information and hence are more beneficial to scene understanding of large-scale outdoor scenes.

Semantic segmentation [14], object classification [15], object detection [16], etc. are important technologies of scene understanding. To better evaluate the performance of different scene understanding algorithms, reliable benchmarks need to be established.

To show the differences of various datasets more clearly, this paper briefly describes some existing datasets. Refer the Appendix B for further details. For the RGB-D datasets including NYU Depth v2, SUN RGB-D [17], UW Object Dataset [18], SUN3D [19], S3DIS [20], Scannet [21] and so on, it is mostly used for scene understanding of indoor scenes. For example, the SUN RGB-D dataset acquired by four types of sensors contains 10,335 images with dense annotations in 2D and 3D for objects and rooms. The CAD model datasets mainly include ModelNet10 and ModelNet40, which are used to identify ten orientations and forty categories of datasets, respectively. The most recent ensemble method [22] reached performance over 97% on ModelNet10, which indicates a model overfit due to limited data. For vehicle mobile laser scanning datasets, such as Sydney Urban Objects [23], KAIST [24], Paris-rue-Madame database [10], Oakland dataset [25] and the dataset from the IQmulus & TerraMobilita Contest [26], they can only produce point clouds with linear road trajectories because of the platform constraints. The precision and density of the obtained point

[1] https://pan.baidu.com/s/1p4tG9asMrt6xPBteRpe-CQ

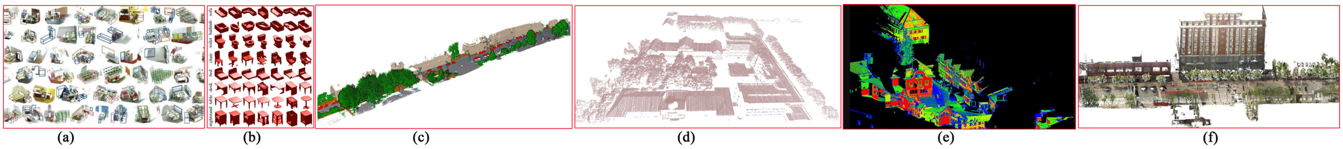

FIGURE 1. Different types of point clouds. (a) Point clouds scanned by Microsoft Kinect devices (SUN RGB-D). (b) Points generated from CAD models (ModelNet). (c) Point clouds scanned by mobile laser scanning system (Paris-Lille-3D). (d) Point clouds [13] scanned by airborne laser scanning system. (e) Point clouds scanned by terrestrial laser scanning system (Semantic3D). (f) Point clouds scanned by the proposed backpacked laser scanning system.

clouds are relatively low, and lacking color information. The Sydney Urban Objects dataset contains 631 individual scanned objects, including vehicles, pedestrians, and trees, which can be used to evaluate the matching or classification algorithms. The dataset from IQmulus & TerraMobilita Contest contains labels and classes; thus point-wise evaluation of detection, segmentation, and classification becomes possible. The Oakland dataset contains 1.6 million points with only x, y, and z coordinates and labels, separated in training, validation, and testing dataset. The scale and the point size of this dataset are small. Terrestrial laser scanning (TLS) datasets are usually more accurate and contain more details of objects. For example, Semantic3D [27], it is mostly used for semantic segmentation of large-scale outdoor scenes. The objects are divided into eight classes. Although point precision is relatively high, the point cloud scene suffers from object occlusion and data missing. Airborne laser scanning (ALS) datasets, such as the Tianjin ALS city point clouds provided by [13], are relatively sparse. The point density is about 20-30 points/m^2, and the point clouds are labeled into three classes (building, tree and vehicle). 3D Semantic Labeling [28], a large-scale ALS dataset was collected from Vaihingen, Germany, with different scenes, which defines nine classes for the 3D labeling challenge. In contrast to the above LiDAR systems, the emergence of wearable laser scanning (WLS) system with real-time registering has been extensively used in the indoor and outdoor mapping. WLS integrates a laser sensor and inertial measurement unit (IMU) in portable equipment, which can be handled by a single operator while walking during acquisition. WLS has a high degree of flexibility and penetrability, thereby maintaining the data completeness of the scanning scene. In this case, this paper aims to exploit the advantages of WLS point clouds and establishes a more challenging large-scale point cloud dataset for point cloud-based scene semantic segmentation. The comparisons between some existing point cloud based datasets are listed in Appendix B.

Generally, these datasets reviewed above have significantly boosted the research in point cloud classification and segmentation, while each dataset has its scope of applications and disadvantages. Although there are a variety of public datasets, to the best of our knowledge, only the well-known datasets, e.g., ModelNet, NYUv2, SUN RGB-D, S3DIS, ISPRS 3D Semantic Labeling, and Semantic3D have benchmarks. To bridge this gap, we present WLS CSPC-Dataset and its derivative benchmarks. CSPC-Dataset is a dense and high precision point cloud and covers varied typical scenes in real-world scenarios, which makes it more practical significance for point cloud semantic segmentation of large-scale complex scenes. To further boost the development of scene understanding, a semantic segmentation benchmark of CSPC-Dataset is built to evaluate scene semantic segmentation algorithms.

B. SEMANTIC SEGMENTATION ALGORITHMS

A large number of algorithms of segmentation and classification of 3D point clouds have been proposed in the past decade. As shown in Fig. 2, these algorithms can be roughly divided into two main categories based on the learning mechanisms: ML-based methods and DL-based methods. Among these methods, the features used for point cloud classification algorithms can be mainly divided into three categories: low-level features, mid-level features, and high-level features.

For the ML-based methods, existing methods can be divided into single-point based methods and point-set based methods according to the processing unit of point clouds. Generally, low-level or mid-level features are used as classification criteria in the ML-based methods.

Classification of single point based on ML [29]–[33] takes the feature vectors of the point clouds as input and the labeled

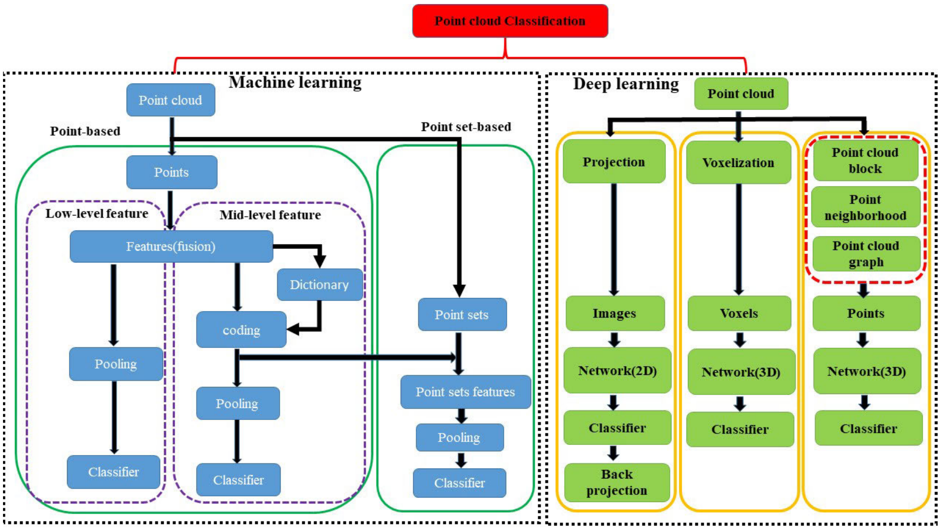

FIGURE 2. Classification and frameworks of large-scale point cloud classification algorithms.

point clouds as output. The features of point clouds can be low-level features [30], [31] and/or mid-level features that derived from a series of low-level features [32]. Each point can be assigned a specific label by the learned classifier model. For example, Hackel *et al.* [29] construct a scale pyramid based on density, and a total of 144-dimensional feature vectors for each point are calculated. Afterward, the classifier is trained and used to classify point clouds directly. Weinmann *et al.* [33] first select the nearest neighbors at different scales, then extract multiple features and use classifiers to classify large-scale point cloud scenes. This method can distinguish the boundary regions of different objects. However, due to the extremely large-scale point clouds, this method has high computational complexity, and some local regions have obvious under- and over-classifications using the trained classifier. Point-based methods are usually simple and efficient because of using less training data. However, these methods have limited accuracy and robustness. The semantic segmentation results always contain a high degree of misclassified noise and outliers.

The classification strategy based on the point set is to cluster the points with the same attribute together and uses these points as a unit to calculate features. The calculation of point set features does not depend directly on the selection of neighborhood size [13], [34]–[40] because the size of point set has been already defined in the process of point set generation. Compared with the single point-based classification, point set-based method can better express the topological relationships among points and point sets, facilitating to improve classification accuracy. For example, Xiang *et al.* [39] construct adjacency relationships of each point according to the normal information. Then large segmented blocks can be built, and support vector machine (SVM) is used to finalize point cloud classification of urban road scenes. Aijazi *et al.* [34] aggregate the super-voxels that are converted from the raw point clouds. The super-voxels are segmented by the pre-defined threshold, and the point cloud classification is achieved based on the super-voxel features. However, point set-based methods are sensitive to the results of point set construction, and the training points cannot be randomly selected. This kind of methods needs more training points and computational cost.

Of late, DL has been widely used in scene understanding. For example, many point cloud semantic segmentation networks based on DL have been proposed. The existing point cloud semantic segmentation methods based on DL mainly confront three challenges: (1) Unlike an image, which is represented by a regular grid, point clouds are discrete and unorganized. Because of this, the CNN filter cannot be directly used for processing point clouds. (2) In computational geometry, the sequence of point clouds does affect point cloud representation by a matrix. This property determines that point clouds can be represented by two completely different matrices. (3) For an image, it has constant pixels because a digital camera's CCD records an image using a fixed grid pattern. However, the number of point clouds is hardly to be estimated because it depends on scanning distance, scanning scene, scanning angle, the performance of the sensor, object reflectivity, etc.

As shown in Fig. 2, point based DL networks are divided into three categories: the network based on 2D projection, the convolutional neural network based on 3D voxelization, and the network model based on discrete point clouds. Projection-based network could cause loss of shape information due to self-occlusions. It tends to need a considerable number of views for obtaining decent performance. Voxelization-based network is memory intensive, and the fined details of objects are hard to be captured. A discrete point based network is restricted to a relatively small region, making the process of large-scale point cloud impossible.

- **2D projection based CNN:** Inspired by the promising results of deep learning on 2D images, a series of methods such as MVCNN [41], VMVCNN [42], Snapnet [43] and DeePr3SS [44] project 3D point clouds onto 2D images as an input of convolutional neural network (CNN). Using the network models of object detection or semantic segmentation, trained by a large number of images in 2D images, as the pre-trained models, this method can obtain better detection and classification results of 2D images. However, these kinds of methods will easily cause the loss of three-dimensional structure/shape information due to self-occlusions. The best way to select the optimal projection angles is another tough problem. Also, it will have the different representative abilities of the object even though we have constant projection angles. That means we assume a series of fixed virtual cameras surround an object. Once the object has a certain degree of rotations, the acquired projected 2D images are quite different. The above problems both affect the generalization ability of 2D projection based CNN.
- **3D voxelization based CNN:** To highly explore the 3D structural information of point clouds, the 2D CNN models have been extended into 3D CNN models based on point cloud voxelization, and other relevant preprocessing techniques. A series of publications such as VoxNet [45], OctNet [46], label-3D CNN [47], Semantic3D.Net (DeepNet) [27], MVF-CNN [48], MVSNet [49] along this line demonstrate the effectiveness of 3D voxelization based CNN. This kind of method retains 3D structure/shape information of objects, which makes the process of feature extraction easier. In addition, this method provides the data structure of point clouds, which solves the problem of point arrangement. However, the computational complexity of the 3D CNN convolution is very high. To solve this problem, the reduction of voxels' resolution is usually adopted, but this increases the quantization errors of voxels. It should be aware that in existing voxelization CNN networks, only the structure/shape information of point clouds is used, and the other relevant information such as color and intensity are usually ignored.

- **Discrete point based CNN:** To make full use of the multi-mode information of point clouds and achieve an end-to-end point cloud processing network, the network models based on discrete are proposed. In this method, the point clouds of a large scene are generally clustered to obtain the appropriate size of point sets, which are fed to the network to learn the features through convolution operations. The basic networks such as PointNet [50], PointNet++ [51], PointCNN [52], PointSIFT [53] and RS-CNN [54] are commonly used. Although these frameworks obtain a promising result in semantic labeling based on discrete point CNN models, they have limitations for processing large-scale point clouds. To solve this problem, some works such as Kpconv [55], SO-net [56], PointFlowNet [57], SPGraph [58], AGC [59], LDGCNN [60], RGCNN [61] and RandLA-Net [62] are proposed to process large-scale point scenes.

Through the above analysis of the characteristics of various point clouds, some representative point cloud datasets and benchmarks, and reviews of semantic labeling algorithms for point clouds, we state our original contributions as follows:

- **Backpack Mobile Mapping Robot:** A relatively advanced backpack mobile mapping robot is presented to collect outdoor large-scale point clouds. By comparing different types of point clouds (see Appendix A), it can be seen that the improved backpack mobile mapping robot has better penetrability, and can obtain the colored LiDAR point clouds and panoramic images with position information. The acquired point clouds are complete, relatively uniform, and have high precision.
- **CSPC-Dataset:** A new point cloud dataset, namely CSPC-Dataset (complex scene point cloud dataset), is constructed for large-scale scene semantic segmentation. This dataset contains complex and diverse scenes, covering streets, schools, grasslands, residential regions, commercial buildings, etc. After a comprehensive summary and comparison with the existing point cloud datasets, it can be concluded CSPC-Dataset is more challenging for large-scale point cloud semantic segmentation methods.
- **Benchmark:** A representative benchmark is built on CSPC-Dataset. Each point in the dataset has a corresponding label (6 categories in total). Based on CSPC-Dataset, we select seven state-of-the-art deep learning algorithms to conduct point labeling experiments. The choice of algorithms considers the factors, including basic processing unit, i.e., points or point sets, feature types and learning mechanisms. It is worth mentioning that the original PointNet++ algorithm has been enhanced by ourselves to obtain a better semantic segmentation for large-scale point clouds. We provide a detailed comparison of seven representative methods to establish a benchmark and reference for investigating large-scale point cloud semantic segmentation.

In the following parts of this paper, we introduce the backpack-based colored LiDAR point cloud acquisition system and its characteristics in Section II. Then the data acquisition, labeling and characteristics of CSPC-Dataset is discussed in Section III. Section IV introduces the ML-based and DL-based baselines. The implementation, evaluation metrics, three benchmarks and corresponding discussions are shown in Section V. Finally, we conclude this paper in Section VI.

II. BACKPACK-BASED COLORED LiDAR POINT CLOUD ACQUISITION SYSTEM

In this section, we first introduce the improved backpack-based mobile laser scanning mapping robot and the method of generating colored point clouds. After that, we compare the differences of large-scale point cloud datasets collected by different acquisition devices.

A. BACKPACK MOBILE LASER SCANNING SYSTEM

According to the existing laser scanning systems [63], [64] (Figs. 3(a)-3(e)), an improved backpack 3D laser scanning mapping system is designed based on our previous published patent [65], as shown in Fig. 3(f). The characteristics of hardware composition is shown in Table 1.

TABLE 1. Hardware composition.

Components	Specifications
Laser scanner:	Two 16-beam laser scanners (Velodyne VLP-16)
Image acquisition:	Panoramic camera (Ladybug 5)
IMU:	Xsens MTi-3
Controller:	CPU: Intel Core I7, RAM: 8G, SSD 512G, USB3.0, WiFi
Trestle:	Carbon fiber, 3.5 kg
Battery:	Lithium battery, 12V 40Ah, Working Time: 4 hours
Tablet handheld terminal:	Surface Pro3

As shown in Fig. 3(f), the backpack mobile laser scanning mapping system is mainly composed of two 16-line 3D laser scanners, a panoramic camera, a controller, a handheld terminal, a mobile power supply, a support bar, and a backpack support. The total weight of this system is 12 kg. When collecting data, a pedestrian carries the system to walk in the outdoor environment to obtain the LiDAR point clouds and the panoramic images. The workflow of the colored point generation is shown in Fig. 4.

A laser of the backpack robot is placed horizontally for acquiring the LiDAR point clouds in the horizontal direction P_h. Another laser is tilted 45 degrees behind the horizontal laser for collecting the LiDAR point clouds in tilted direction P_t. The calibration matrix of the positional and orientational relationships of the tilted laser relative to the horizontal laser is T_c, which is calculated by the algorithm in [66]. Point clouds from two lasers are fused using Eq. 1. 3D simultaneous localization and mapping (SLAM)

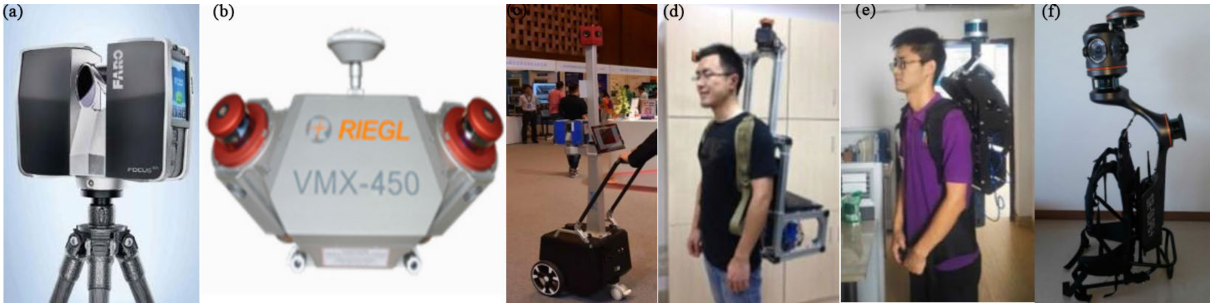

FIGURE 3. Laser scanning systems. (a) TLS scanner. (b) MLS scanner. (c) Cart-based mapping system. (d) Single laser mobile backpack scanning system [63]. (e) Dual backpack mobile laser scanning system [64]. (f) Backpack mobile laser scanning mapping robot.

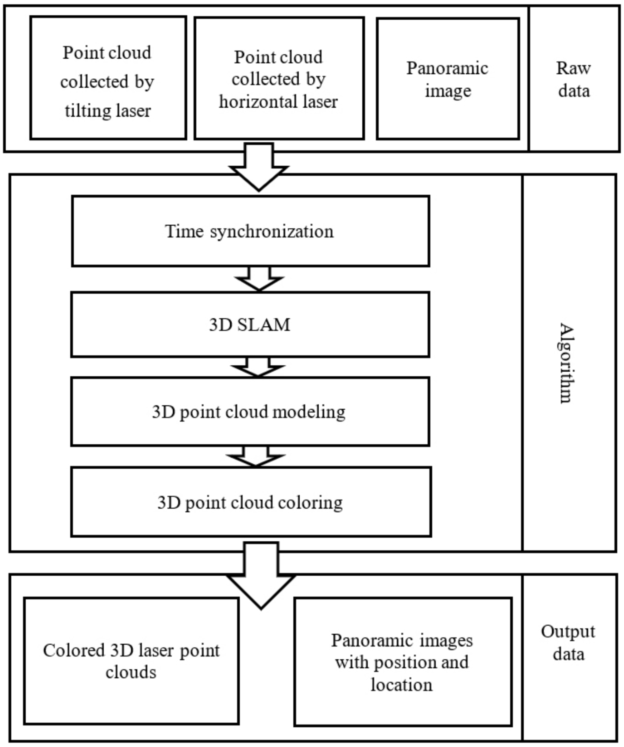
Point cloud collected by tilting laser
Point cloud collected by horizontal laser
Panoramic image
Raw data
Time synchronization
3D SLAM
3D point cloud modeling
3D point cloud coloring
Algorithm
Colored 3D laser point clouds
Panoramic images with position and location
Output data

FIGURE 4. The flowchart of colored point cloud generation.

algorithm [67] is adopted to register/stitch 3D point clouds into a common coordinate system. The point clouds ($P_{pointcloud}$) of the scene can be built [67] according to Eq. 1.

$$P_{pointcloud} = P_h + P_t \times T_c \tag{1}$$

The panoramic camera is placed directly above the horizontal laser, and it is used to capture panoramic images of the surrounding environment and to color the point clouds. According to the relative spatial position and orientation relationship between the panoramic camera and horizontal laser, and the spatial position and orientation of corresponding generated LiDAR point clouds, the position, and orientation of each frame of panoramic images are calculated. Then one-to-one correspondence between point clouds and pixels of panoramic images are created, and the RGB color values are assigned to LiDAR point clouds. The accuracy of the colored point clouds collected by the backpack robot is shown in Table 2.

TABLE 2. Accuracy of backpack mobile laser scanning mapping robot.

Metrics	Values
Absolute Accuracy (σ):	$\sigma \leq 0.03$ m
Relative Accuracy (ρ):	0.01 m - 0.05 m. **It is noted that ρ is determined by the density of point clouds. For example, if the density is 0.005 m, $\rho = 0.01$ m. When the density is 0.02 m, $\rho = 0.04$ m.**

B. CHARACTERISTICS OF THE LARGE-SCALE POINT CLOUDS

As shown in Fig. 3(a), terrestrial laser scanning (TLS) can be used to acquire indoor and outdoor environments. The laser scanner needs to be fixed in a certain position to acquire point clouds. The colored point clouds obtained by the TLS system with an externally mounted camera have a higher density, but its density is easily affected by the scanning distance. The density of point cloud is extremely high when the scanned objects are close to the scanner, as demonstrated in Fig. 5(a). This leads to greatly varying point density. Meanwhile, many objects cannot be scanned due to occlusions, self-occlusions and constraint of scanning distance, resulting in incomplete scenes. Besides, obtaining the whole point clouds acquired by TLS is inefficient because multiple scans need to be registered to capture the complete targets.

To enhance flexibility and efficiency of data acquisition, increasingly more attention has been paid to mobile laser scanning (MLS) techniques. As shown in Fig. 3(b), the vehicle mobile mapping system can quickly collect point clouds of large-scale outdoor scenes as it uses a mobile platform that is equipped with a laser and the GPS/IMU system. Due to the constraint of vehicle paths, the application environment of the MLS system has been significantly constrained, and its accuracy is influenced by positioning signals. As shown in Fig. 5(b), the point clouds acquired by a vehicle-based platform along the road are incomplete, only scanning structures in the front, but lack of 3D structures of back due to occlusions.

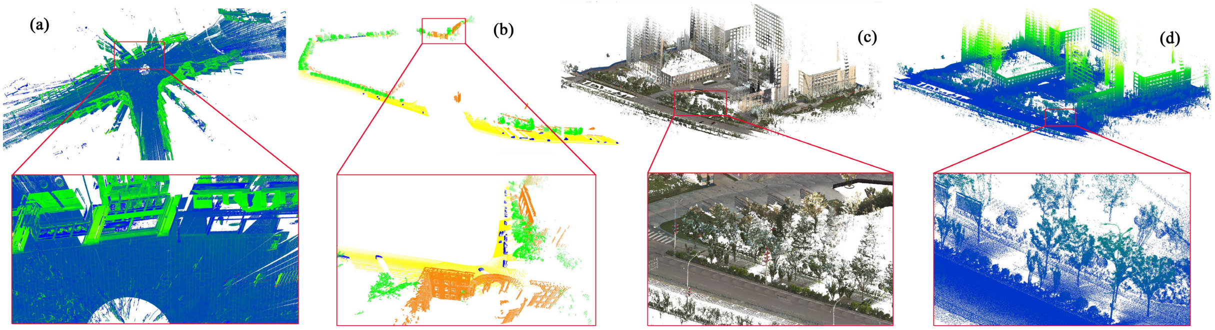

FIGURE 5. The generated point clouds by various laser scanning system. (a) Point clouds are collected by the TLS systems. The point clouds are rendered by their elevation. (b) Point clouds are collected by mobile laser scanning vehicle, and the point colors are rendered by the different classes. Subfigures (c) and (d) are point clouds collected by a backpack mobile laser scanning mapping robot. The point colors are rendered in these two subfigures based on RGB color and elevation information.

Of late, with the advancement of SLAM techniques, SLAM has a hot application in mobile point cloud acquisition. The point cloud acquisition system based on the SLAM cart is shown in Fig. 3(c), it can work without GNSS signal. However, this system can only work in a horizontal plane. Compared with the above acquisition systems, the backpack mobile laser scanning mapping robot has strong mobility, minimal space constraints, and can obtain more complete point clouds. The single laser backpack laser scanning system, as shown in Fig. 3(d), has higher data acquisition stability than the system based on a cart, although its accuracy requires to be further improved. After the improvement of a single laser system, a dual laser mobile backpack scanning system in Fig. 3(e) can be used in the indoor scenes without GNSS signal and non-horizontal scene constraints. The collected point clouds can meet the requirements of indoor environments in high-definition mapping and autonomous vehicles driving. However, this system is mainly used in indoor environment and the acquired point clouds without color information. For the system in Fig. 3(e), there are limited public outdoor point clouds, and the accuracy of data acquisition is not provided.

To obtain colored LiDAR point clouds more efficiently, the data acquisition backpack robot for simultaneously obtaining panoramic images and LiDAR point clouds is developed, as demonstrated in Fig. 3(f). Point clouds of large-scale scenes collected by the backpack robot are shown in Figs. 5(c) and 5(d). The backpack robot is a new generation of data acquisition platform with high penetrability and precision. The biggest advantage of the proposed robot has great flexibility, allowing it to be used in multiple environments and situations. The acquired point clouds and panoramic images can achieve the full scene coverage, overcoming the occlusions in MLS and TLS systems.

In this paper, the backpack robot can acquire LiDAR point clouds of buildings, residential areas, blocks, and other scenes in real-time. It can automatically perform stitching, coloring, and other operations for the dataset. The backpack robot can work in both walking and riding modes and has the capability to acquire indoor and outdoor scenes seamlessly. It is no requirements of initialization, interruption, residence, working time constraints, and multi-scan registration. Compared with other point clouds in Fig. 5, the point cloud scene collected by the backpack robot is complete, dense, uniform, accurate, and being rich color information, making it an appropriate data source to express the whole scene.

III. COLORED POINT CLOUD DATASET FOR OUTDOOR SCENE SEMANTIC SEGMENTATION

Although many public point cloud datasets are used for semantic segmentation, there are few large-scale LiDAR datasets, including complex outdoor urban scenes and having colored information. For the advancement of 3D SLAM and the mobile mapping robot, the benefits of using 3D point clouds to model projects have been widely recognized. The wearable laser scanning system enjoys a high reputation due to its flexible data acquisition and high qualified collected dataset. Because of this, the portable scanning technique is extensively used in indoor and outdoor modeling, high-definition mapping, automatic driving, etc. In this case, we construct the CSPC-Dataset: a large-scale complex outdoor scene LiDAR dataset for point cloud classification based on the backpacked mobile mapping robot. We first briefly introduce the collection of the large-scale datasets. After that, we describe the details of the point cloud labeling method. At last, we provide a statistical analysis on CSPC-Dataset.

A. DATASET ACQUISITION

In this paper, five large-scale scenes are acquired by a backpack mobile mapping robot (see Section II). The collected five large-scale point datasets contain nearly 68 million 3D point clouds, covering urban streets, rural areas, university campuses, and rural residential areas. Various point cloud scenes are collected at different times. All the point clouds are collected in China, including a wide variety of styles of buildings. As shown in Fig. 6, the pedestrian is carrying a mobile mapping robot in the scene to collect data. The robot collects the data of these scenes by laser sensors and panoramic cameras. After the fine modeling and coloring processing of point clouds, the complete scanning scene represented by massive colored points can be recorded. The scanning frequency of lasers can reach 600,000 points/sec,

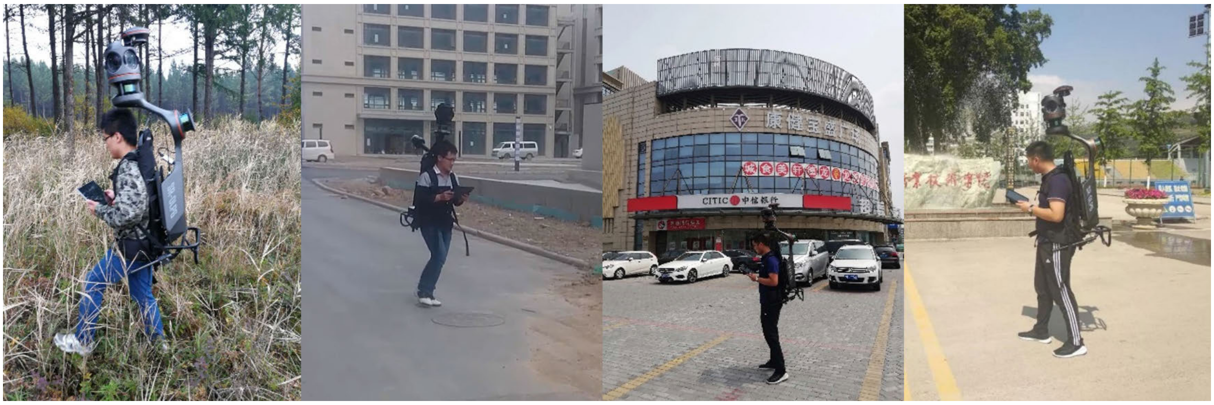

FIGURE 6. Data collection for various scenes using proposed backpack mobile mapping robot.

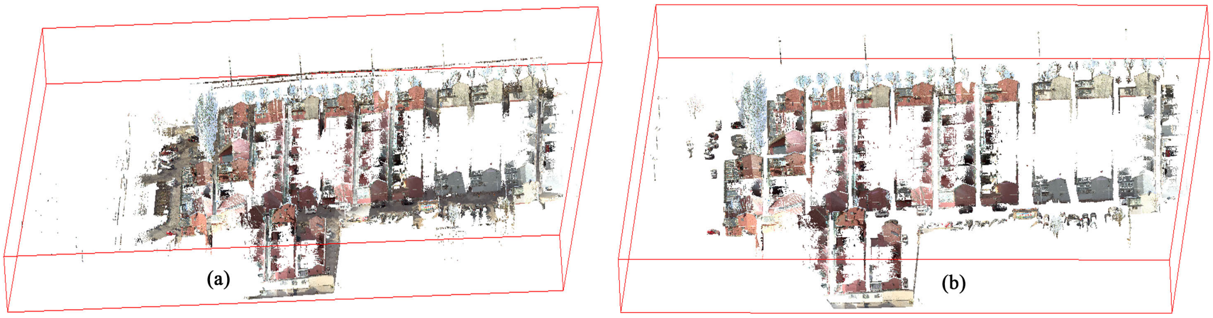

FIGURE 7. Ground filtering using CSF algorithm embedded in CloudCompare open source tool. (a) The raw data with RGB color information. (b) Non-ground measurements after implementing filtering.

and the maximum scanning ranges can reach up to 100 m. The relative and absolute precisions are less than 0.03 m and 0.05 m, respectively.

B. DATASET LABELING

To make CSPC-Dataset more applicable in point cloud scene semantic segmentation, this paper mainly classifies the collected point clouds into six categories:

- Ground: including unnatural ground points, i.e., mainly sidewalks and roadways and natural ground points, i.e., grassland and forest ground.
- Buildings: including high-rise commercial buildings, residential buildings, low-rise factories, and rural residential houses.
- Vehicles: including ordinary cars and trucks.
- Bridges: including the common overpasses in urban roads.
- Vegetation: including trees and low vegetation such as understories.
- Poles: containing power poles and street lamps.

Note that other scanning artifacts labeled as symbol “0”, in most applications, should be filtered with some heuristic rules. However, considering the completeness of the dataset and comparisons with other relevant algorithms, we do not perform any heuristic preprocessing. To manually label the collected point cloud of the large-scale scene more accurate, we use an open-source tool Cloudcompare[2] to assist point labeling by the following three steps:

1) GROUND EXTRACTION

To accurately extract ground points, we use a cloth simulation filtering (CSF) algorithm [68] to coarsely separate ground and non-ground measurements. If the terrain is complex, some errors will occur. To reduce these errors, we manually check these two parts of point clouds from multiple views by using Cloudcompare tool. In this way, the ground and non-ground measurements are significantly refined. It should be noted that the extraction of ground points plays a solid foundation for labeling other non-ground measurements because once the ground points are recognized and eliminated, the remaining non-ground objects can be divided into some extents. The raw point clouds, shown in Fig. 7(a), are filtered to obtain ground and non-ground measurements roughly. Once the ground points are removed, the separability of off-terrain points can be enhanced due to the data gaps produced by the CSF algorithm, as demonstrated in Fig. 7(b).

2) MANUAL OBJECT SEGMENTATION

We use human-machine interaction to further manually label non-ground measurements. More specifically, we use

[2] http://www.cloudcompare.org/

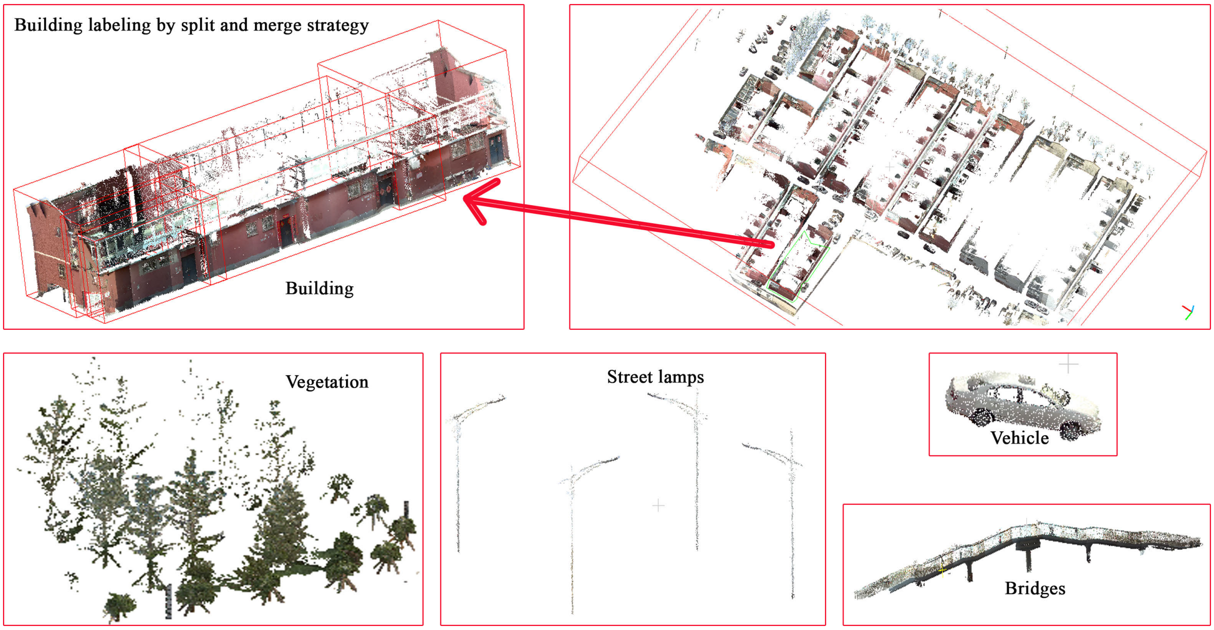

FIGURE 8. Manual object labeling. Note that for large objects such as large-sized buildings, we manually use "split and merge" strategy to guarantee labeling accuracy as much as possible.

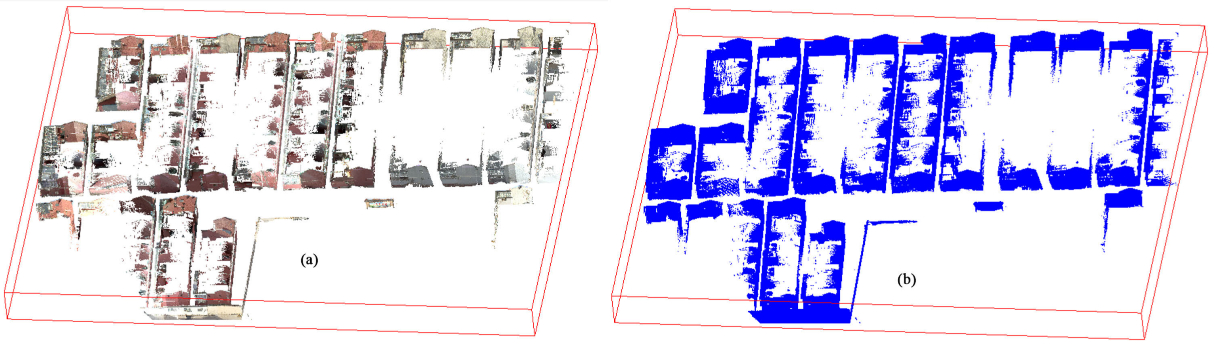

FIGURE 9. Point labeling refinement. Subfigures (a) and (b) represent the refined buildings in RGB and Scalar rending modes.

"split and merge" strategy to label large objects. For example, for a specific target, a good view is selected to split the large objects manually. In a selected view, only parts of an object with high probability of belonging to a specific category are split. Then, the remaining of the same object is split from other views. This process is repeated from other possible point of views until the object are totally split. Afterward, the components of the same object belonging to the same category are merged together. The large complex, shown in Fig. 8 is accurately split into multiple patches and then merged into complete buildings. Other large objects, such as trees and bridges, can be treated in the same way.

3) POINT LABELING REFINEMENT

To guarantee the point labeling accuracy, the point refinement process is implemented. To this end, we first eliminate some duplicated points that are produced in the split and merge operations. After that, each point is unique and associated with one specific class. In addition, a double check is implemented to remove some noises and outliers. The noises and outliers are excluded from the RGB and Scalar (rendering the point clouds by their associated labels) modes (see Fig. 9). Finally, the extracted point clouds are assigned to the labels of corresponding objects. The labeled point clouds of CSPC-Dataset are completed by two operators, and another two operators are implemented to check and re-label the initial labeled data to obtain the complete point clouds with high-accurate labels.

C. DATASET CHARACTERISTICS

The constructed dataset for large outdoor scenes is shown in Fig. 10. The point cloud statistics for each class in five scenes are depicted in Table 3. The percentage of each object in the corresponding scene is given in brackets. For CSPC-Dataset, the point distribution of six objects is shown in Fig. 11. The CSPC-Dataset can be used to the ground

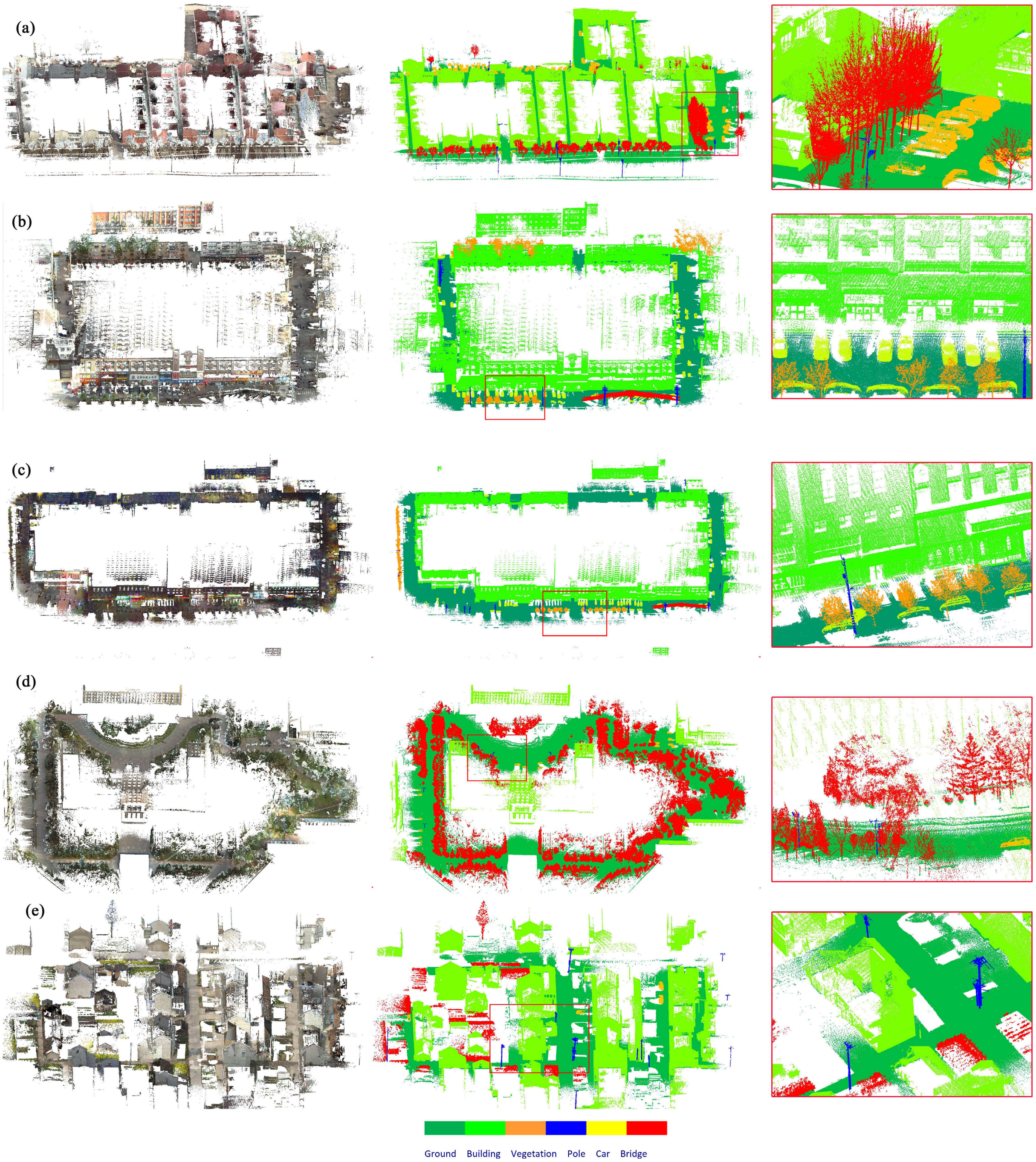

FIGURE 10. CSPC-Dataset. Five scenes from (a) to (e) represent Scene 1 to Scene 5. The leftmost column represents the different scenes rendered by the RGB color. The middle column are shown by the object class, and the rightmost column represents the enlarged views from the red rectangle areas in the middle column.

filtering of point clouds, 3D vehicle detection, point cloud segmentation, classification, and recognition, as well as evaluation of the representation ability of point cloud's features.

In contrast to other datasets depicted in Appendix B, CSPC-Dataset has the following characteristics:

(1) Large-scale outdoor scenes: Most frequently used point cloud datasets for benchmarking are indoor scenes or CAD models. Other outdoor benchmark datasets such as Paris-rue-Madame [10], Paris-Lille-3D [11] and MLS1-TUM city campus dataset [69] are acquired by MLS system. The data acquisition is strongly restricted by the vehicle path. The incomplete characteristics of MLS point clouds are prominent because of occlusions and self-occlusions of objects. Although the large-scale TLS benchmark such as Semantic3D [27] includes multiple scans to register the relatively complete data. However, the density is varied according

TABLE 3. Statistics of CSPC-Dataset. Note that the number in (·) represents the percentage of the points in each scene.

Scenes	Ground	Building	Car	Bridge	Vegetation	Pole	Total	Type
Scene 1	6,082,987 (37%)	9,032,520 (54%)	651,442 (4%)	0 (0%)	641,970 (4%)	24,034 (0.15%)	16,433,953	Simple street
Scene 2	4,358,082 (47%)	3,992,075 (43%)	525,815 (5.7%)	90,637 (0.1%)	257,708 (2.8%)	43,930 (4.7%)	9,268,247	Busy city street
Scene 3	8,736,662 (5.6%)	5,996,45 (8.7%)	469,271 (3%)	97,712 (0.6%)	163,830 (1.1%)	46,579 (0.3%)	15,510,510	Busy city street (night)
Scene 4	10,282,388 (63%)	835,169 (5.1%)	71,577 (0.44%)	0 (0%)	5,116,352 (31.3%)	8,285 (0.5%)	16,323,771	Campus
Scene 5	5,332,925 (53.7%)	4,197,404 (42.2%)	34,960 (0.35%)	0 (0%)	322,488 (3.2%)	49,397 (0.5%)	9,937,174	Rural street
Total	34,793,044	24,053,624	1,753,065	188,349	6,502,348	172,225	67,473,655	

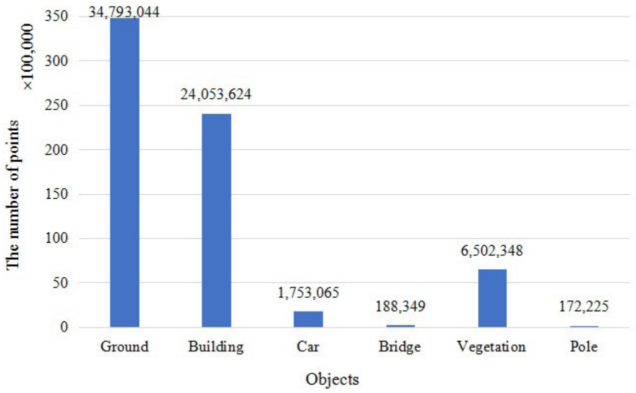

FIGURE 11. The number of points in each category of CSPC-Dataset.

to the distance from the object to the scanner. To the best of our knowledge, it lacks wearable laser scanning (WLS) point clouds as a benchmark. We bridge these gaps by producing a WLS benchmark (CSPC-Dataset) using the backpack mobile scanning robot. In contrast to other types of benchmarks, the proposed WLS benchmark is large-scale, totally including nearly 68 million point clouds. The acquired objects are relatively complete due to the flexible ways of acquisition. The scanned scenes are diverse, although the precision of point clouds based on the CSPC-Dataset is slightly low.

(2) The complete scene and relatively uniform point density: Compared with Semantic3D scanned by the static laser (see Fig. 1(d)) and KAIST and Oakland datasets acquired by the vehicle mobile laser scanning (see Fig. 1(c)), CSPC-Dataset is collected by a backpack mobile scanning robot, which has the capability to acquire points in indoor and very narrow space. That is, the scanning path is less restricted, although the point accuracy could be strongly affected by the characteristics of the trajectory, such as the traveling speed and the path followed. As shown in Fig. 10(a), Fig. 10(b) and Fig. 10(d), the collected point clouds of objects are complete, allowing the whole scene more complete. Therefore, it guarantees the completeness and comprehensiveness of the information expressions of the scanning scenes. In the Semantic3D dataset, the occlusion is inevitable due to the horizontal and vertical field-of-view restrictions. In addition, compared with the Semantic3D dataset, the density of CSPC-Dataset is more uniform because the scanning distance varies gently during a manner of walking scan mode.

(3) Diversity and complexity of objects: CSPC-Dataset provides complex objects in diverse types and shapes in different regions. The selected objects of CSPC-Dataset are more important for scene understanding, digital city, and urban planning applications. Compared with the ModelNet series, the objects of CSPC-Dataset are captured from the real world, and the shapes of objects are more complex and diverse. The buildings of Semantic3D are relatively homogeneous in architectural styles. In contrast, the contained buildings of CSPC-Dataset have a wide variety of geometric shapes and architectural styles.

(4) The high discrepancy between different scenes: CSPC-Dataset chooses different types of scenarios in different regions. Although Semantic3D's point clouds have 15 different scenarios, they have high similarity in scene composition. However, the discrepancy between CSPC-Dataset scenes is prominent, which helps data demanding methods like deep learning based algorithms to unleash their full potential power and learn high richer 3D representations.

IV. BASELINE METHODS

For large-scale point cloud scene, semantic segmentation is to assign a separate category label to each point in the point cloud. This paper provides seven methods of point cloud semantic segmentation for the benchmark generation. In this paper, these methods are divided into machine learning and deep learning separately. Machine learning methods are further selected from single point-based and point set-based methods. Methods based on deep learning mainly include SnapNet, improved PointNet++, Label 3D CNN, and DeepNet.

A. METHODS BASED ON MACHINE LEARNING

Machine learning has received increasing more attention in terms of point cloud labeling and semantic segmentation because it requires less training data. In this paper, three machine learning-based methods are selected as baselines. Two of them are single point based methods: point labeling based on covariance and spin images [38], and multi-scale features fusion [37]. Another method is a point set based method, which uses multi-layer clustering for the generation of multi-scale point sets and adopts high-order conditional random fields (CRF) for point labeling point sets. The detailed explanations of each method are as follows:

Method 1: Point labeling based on features derived from covariance eigenvalues and spin images is proposed in [38]. We define this method as SVM-based method, which extracts these two kinds of features of each point defined at a particular support domain. To obtain more discriminative multi-scale features, the support domains need to be constructed at different scales by changing the number of neighbors or the neighboring radius. According to the recommendations in [38], we select the spherical region with the support domain radius of 0.08 m, 0.32 m, and 1.28 m to create multi-scale features. Then two multi-scale features are fused and fed to SVM classifier for classification of points. This method is also used as a comparison algorithm in [13], [40]. In this paper, this method was run on MATLAB 2017b. The platform of the experiments is a personal computer, equipped with a 4.20 GHz Intel Core i7-7700k CPU, 24 GB of main memory.

Method 2: As shown in Fig. 12, it shows the flowchart of point cloud classification based on single point multi-scale feature fusion and pyramid neighborhood optimization [30]. The point cloud classification algorithm first determines the neighborhood region of each point and then extracts the features of a single point, including elevation feature (Elevation), normal angle distribution histogram (NAD), latitude direction sampling histogram (LSH), covariance eigenvalue feature (CF), and plane point ratio feature (PPR). After that, a multi-scale feature of a single point is constructed by using multiple resolutions of point clouds and multi-scale neighborhoods. The fusion features which are normalized and reduced are fed to SVM for point labeling. Finally, the final results of point cloud classification are obtained by neighborhood optimization based on the multi-scale pyramid.

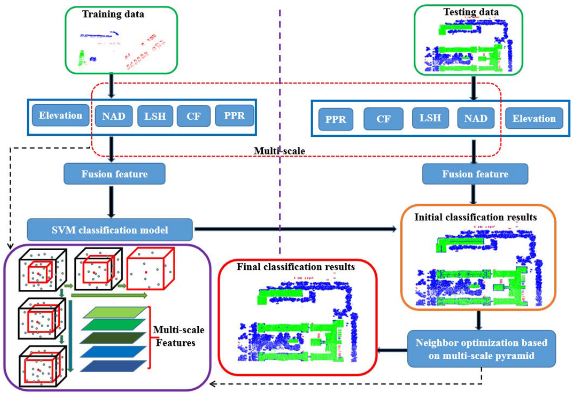

FIGURE 12. The flowchart of Method 2.

Method 3: We implement a point set based machine learning method [70] for point clouds labeling. The entire framework mainly includes two key steps: hierarchical clustering and high-order CRF optimization. The flowchart of the algorithm is shown in Fig. 13. In hierarchical clustering, the original point cloud is over-segmented into fine-grained point sets by combining DBSCAN (Density-Based Spatial Clustering of Applications with Noise) clustering and K-means clustering. Then, covariance features (CF), elevation features (EF), and latitude direction sampling histogram (LSH) are extracted and concatenated for each fine-grained point set. Using these generated point set's features, the fine-grained point sets are initially classified by the SVM classifier. Next, the neighborhood topological relationships of the fine-grained point sets are built by transforming the problem of the topological relationship construction into a clustering problem. Using the topological context and fine-grained point sets, the CRF model, including the first-order term, second-order term, and high-order term, is constructed to refine the initial labeling results of fine-grained point sets.

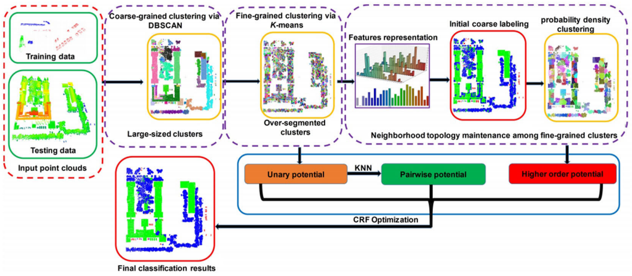

FIGURE 13. The flowchart of Method 3.

B. METHODS BASED ON DEEP LEARNING

For large-scale scenarios, it probably contains hundreds of millions of point clouds. Therefore, the large-scale dataset makes the deep learning methods better learn 3D representation and understand the point cloud scenes. This paper chooses the following four typical networks as the baselines for point cloud semantic segmentation. SnapNet [43] is a point cloud projection-based method; PointNet++ [51] is a network based on point set; 3D CNN [47] is a single-scale voxel-based method and DeepNet [27] is a network based on multi-scale voxel. To implement these four methods, we choose the default parameter settings from their original papers.

Method 4: SnapNet [43] projects point clouds onto 2D images and uses image semantic segmentation networks based on deep learning to classify large-scale point clouds. The method firstly meshes the point clouds and then sets a virtual camera to take photos of the point scene at different scales with different views to obtain a series of projected RGB images, depth images, and unique face color images, etc. Next, these three kinds of images at each view are trained based on the U-net [71] and the residual correction network [72]. Once the pre-trained model is created, it can be used to segment the projected images with different views. Finally, the results of the images are back-projected onto the 3D point clouds to finalized the raw point labeling.

Method 5: Method 5 uses an enhanced version of PointNet++ [51] to achieve point cloud classification. PointNet++ is composed of sampling layer, grouping layer and PointNet layer (see Fig. 14). In the sampling layer, iterative farther-most point sampling method (FPS) is used to select the center of the local region. That is, a point is randomly selected first, then the farthest-most point of the selected point is regarded as the starting point, and the iteration is continued until selecting the required number

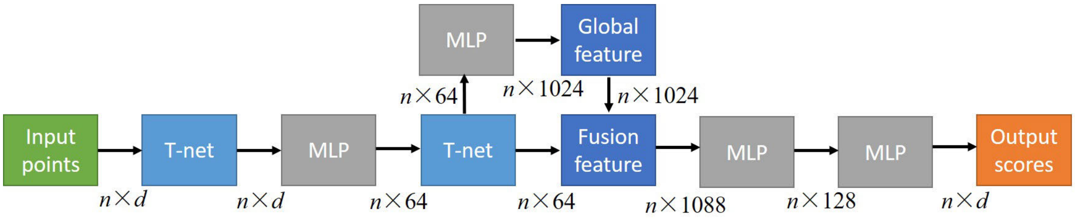

FIGURE 14. The network of PointNet.

of points. Next, the neighbor points of the center within a given radius r are selected to construct the point set. The constructed point sets have the same number of points by downsampling method, and the point sets feed into the PointNet [50] to obtain high-dimensional expression of the point set. The above process is called set abstraction (SA). The SA process, i.e., sampling, grouping, and PointNet, is then repeated for high dimensional feature extraction. In the abstract setting, the original point set is downsampled. For the semantic segmentation tasks, all the point features in the original point set need to be acquired. Therefore, the interpolation method is used to connect with the corresponding point set and features to achieve the purpose of feature propagation. Finally, the classification results of all points in the point cloud are obtained. For the last classification layer of the network, the Batch Normalization and ReLu are added to all the full-connected networks.

However, the PointNet++ cannot be directly used for processing large-scale point clouds. To overcome this deficiency, we improve the PointNet++ as follows: First, the input large-scale point clouds are divided into multiple subsets, each of which contains 100,000 to 300,000 points. To ensure the precision of floating point calculations, coordinate transformation is performed on each segmented subscene. We downsample points within each subscene using voxel-based data structure. More specifically, dynamically sampling with 1.5 m × 1.5 m on horizontal plane of each subscene is employed to obtain a series of point cloud blocks, each block having 8,192 point clouds. In each block, we transform the coordinate system of contained points and normalized point's elevation. We make the origin (X = 0 and Y = 0) of the coordinate system at the center of each block and we normalize the minimum elevation of points with zero. Data enhancement is successively performed by randomly rotating each point cloud block. Finally, the network is trained according to the following steps: SA(1024, 0.5, [32, 32, 64]) → SA(256, 1.0, [64, 64, 128]) → SA(64, 2.0, [128, 128, 256]) → SA(16, 4.0, [256, 256, 512]) → FP(256, 256) → FP(256, 256) → FP(256, 128) → FP(128, 128, 128, 128, K). SA(K, r, [l_1, l_2, l_3]) indicates that a PointNet in K local regions with radius r contains three fully connected layers of l_1, l_2 and l_3, respectively. FP($l_1, \cdots, l_f$) indicates that feature propagation (FP) has f fully connected layers, and the dimensions of each fully connected layer is l_f.

Compared to the original PointNet++, the enhanced version of PointNet++ has the following advantages:

(1) When dynamically acquiring point cloud blocks, the size of each block is set to 1.5 m × 1.5 m × Δz. Parameter Δz represents the high difference between point clouds in each block. We do not impose any restrictions on the height of point clouds, which helps this dynamical sampling more adaptable to the characteristics of the outdoor scenes.

(2) As the coordinate transformation of each point cloud block is implemented before training, the local features of the point cloud block can be learned sufficiently.

(3) In order to adapt to the point cloud distribution characteristics of large-scale outdoor scenes, the parameter r in SA(K, r, [l_1, l_2, l_3]) is set to 0.5, 1.0, 2.0, and 4.0, respectively. The increase of the neighborhood radius helps to better extract the local features of large-scale outdoor scenes.

Method 6: Label-3D CNN is a neural network model using 3D CNN for large-scale complex point cloud labeling [47]. The input point clouds are implemented a sparse voxelization by point clouds' extents and the pre-defined voxel size. In our CSPC-Dataset, we set the value of voxel size to 0.3 m. In each voxel, it is divided into multiple grids (20 × 20 × 20). The point clouds within each grid are organized by Binary Grid Occupied [45]. Then, the constructed occupied voxels are put into the 3D CNN network (see Fig. 15). The network contains two convolution layers, two max-pooling layers, and one full connection layer. At the end of the network, Softmax outputs the category label of each voxel. In the process of our implementation, the kernel size of the convolutional and max-pooling layers are set to 5 × 5 × 5 and 2 × 2 × 2, respectively. Finally, the label of each voxel is assigned to all points included in the corresponding voxel. This method can effectively reduce the amount of computational time in the process of point cloud learning and provide a solution to process large-scale point clouds during training and testing effectively. In this paper, the values of the maximum number of epochs, learning rate, and batchsize of this method are set to 300, 0.0002, and 240, respectively.

Method 7: DeepNet [27] is a baseline method provided in Semantic3D, which is based on voxelized point cloud as input to the network. Considering the advantages of CNN for feature extraction in the image, the voxel-based point cloud classification network is constructed by a 3D CNN network, which is designed based on the VGG network framework.

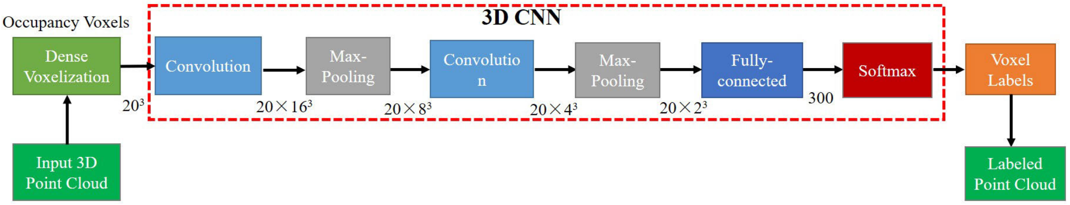

FIGURE 15. The network of Label-3D CNN.

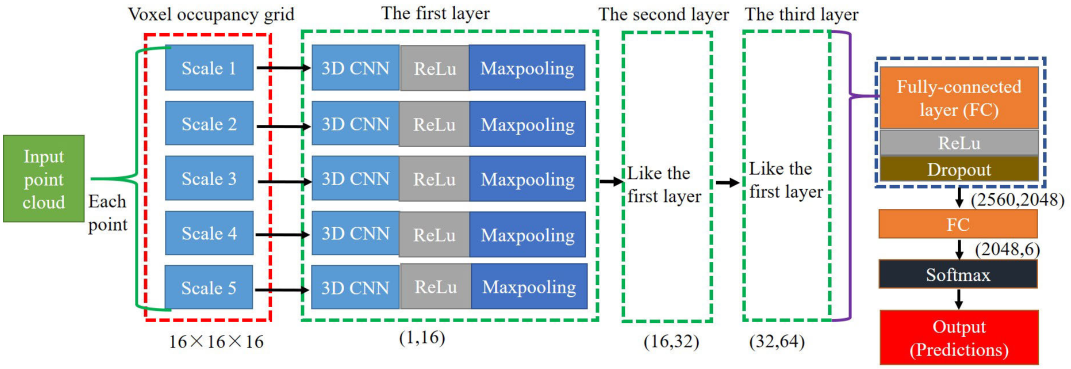

FIGURE 16. The network of DeepNet.

The network structure is shown in Fig. 16, from which we can see that for each point, the number of established voxels is 16 × 16 × 16. The symbol "(m, n)" indicates that we give the number m as input channels and n as output channels. The 3 × 3 × 3 convolution kernel is used in three 3D CNNs, and the 2 × 2 × 2 receptive field is used for max-pooling. The five-scale voxel radii of DeepNet are: 0.0125 m, 0.025 m, 0.05 m, 0.1 m and 0.2 m. The maximum number of epochs of DeepNet, the learning rate, and the batchsize are set to 300, 0.0002, and 240. Because the point clouds in CSPC-Dataset are large-scale, to improve the computational efficiency, we first use Octree to downsample the raw point clouds and label the reduced/downsampled point clouds using the DeepNet. Once the reduced point clouds have been labeled, other unlabeled points in the raw data are assigned the labels according to the principle of proximity to the labeled points.

V. EVALUATION METHODS AND BENCHMARK SYSTEMS

In this section, we firstly introduce the experimental platform and the selected evaluation metrics for the evaluation of point labeling accuracy. Then we use the seven baseline methods (see IV) to conduct experiments on the CSPC-Dataset. After three groups of experiments, the result evaluations are used to build a benchmark based on CSPC-Dataset.

A. IMPLEMENTATION

The baseline methods are all running on an Intel Core i7-7700K CPU, 4.20 GHz, 24-GB RAM computer. The implementation of algorithms is based on PCL 1.8.0 (C++) and Tensorflow 1.4.0 (Python 3.6). Seven baseline methods are tested on the CSPC-Dataset, and the important parameters of the seven baseline methods are given in Section IV.

B. EVALUATION METRICS

We use comprehensive evaluation metrics including Precision/Recall, F_1-score, Intersection over Union (IoU), Overall Accuracy (OA), and Kappa to evaluate the accuracy of point cloud labeling. The specific calculation method is as follows. Our multi-class labeling problem can be transformed into binary labeling sub-problems. The evaluation of binary labeling sub-problems generally uses a confusion matrix, as shown in Table 4, from which T_p, F_n, F_p and T_n represent the number of true positives, false negatives, false positives, and true negatives, respectively. Once we have obtained a clear understanding of the above four metrics, we can confidently evaluate Precision/Recall, F_1-score, mIoU, OA, and Kappa.

TABLE 4. Definition relationships between predicted and true values.

Ground Truth	Predicted	
	Positive	Negative
Positive	True Positive (T_p)	False Negative (F_n)
Negative	False Positive (F_p)	True Negative (T_n)

Precision represents the success probability of making a correct positive labeling points, which is calculated as numbers of F_p divided by the sum of F_p and T_p.

$$Precision = \frac{T_p}{T_p + F_p} \tag{2}$$

Recall explains how sensitive the model is towards identifying the positive class. More specifically, recall is the ratio of correctly predicted positive points to all points in the positive class.

$$Recall = \frac{T_p}{T_p + F_n} \tag{3}$$

To comprehensively evaluate the labeling ability of the classifier for each category, F_1-score is usually used to measure the overall classifier ability. $F_1 - score$ is the weighted average of the precision and recall and is defined as:

$$F_1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4}$$

This score takes both F_p and F_n into account. If the scene has an uneven class distribution, this score can more reflect the overall performance of the labeling algorithm.

In addition, IoU is also a commonly used evaluation metric. For i-th class, IoU_i of this specific category is calculated as below:

$$IoU_i = \frac{C_{ii}}{C_{ii} + \sum_{j \neq i} C_{ij} + \sum_{k \neq i} C_{ki}} \tag{5}$$

where C is a $L \times L$ classification confusion matrix. L is the number of an object category. C_{ij} is the true label of i-th class classified to the j-th class. IoU_i is the comprehensive evaluation of the classification effect on the i-th category.

Since CSPC-Dataset includes multiple categories, labeling based on this data is a multi-class labeling problem. Therefore, comprehensive evaluation metrics that reflect the performance of labeling algorithms on all classes are needed. We use OA, $mIoU$, and $Kappa$ to evaluate the performance of different point cloud semantic labeling algorithms.

$$OA = \frac{\sum_{i=1}^{L} C_{ii}}{\sum_{j=1}^{L} \sum_{k=1}^{L} C_{jk}} \tag{6}$$

$$mIoU = \frac{\sum_{i=1}^{L} IoU_i}{L} \tag{7}$$

$$\begin{aligned} Kappa &= \frac{OA - p_e}{1 - p_e} \\ s.t.\ p_e &= \frac{\sum_{j=1}^{L} \sum_{i=1}^{L} (C_{ij} \times C_{ji})}{N \times N} \end{aligned} \tag{8}$$

where, N is the number of all points.

C. BENCHMARK SYSTEMS

To fully evaluate the performance of different algorithms on different data sizes and scenarios, three groups of experiments are implemented to evaluate the baseline algorithms comprehensively. Group 1 Benchmark is mainly used to evaluate the algorithms, including machine learning-based and deep learning-based methods, with few training samples on the CSPC-Dataset. Due to the machine learning-based algorithms are generally inapplicable for model training with large training data, Group 2 Benchmark is mainly used to evaluate the deep learning-based algorithms on CSPC-Dataset. Group 3 Benchmark can be used to evaluate the generalization performance of different algorithms, and it is also a point cloud classification benchmark for the overall CSPC-Dataset. These benchmarks conducted by the three groups of experiments are as follows:

1) GROUP 1 BENCHMARK

To verify the effect of different algorithms with relatively fewer training samples, we select three different scenes, i.e., Scene 2, Scene, 4 and Scene 5, as shown in Fig. 17 for experiments. We select relatively few points from these three scenes as the training set and the rest as the testing set. The ground point clouds are filtered in advance. The detailed statistics of the training set and testing set of this group experiment are shown in Table 5. Seven baseline methods,

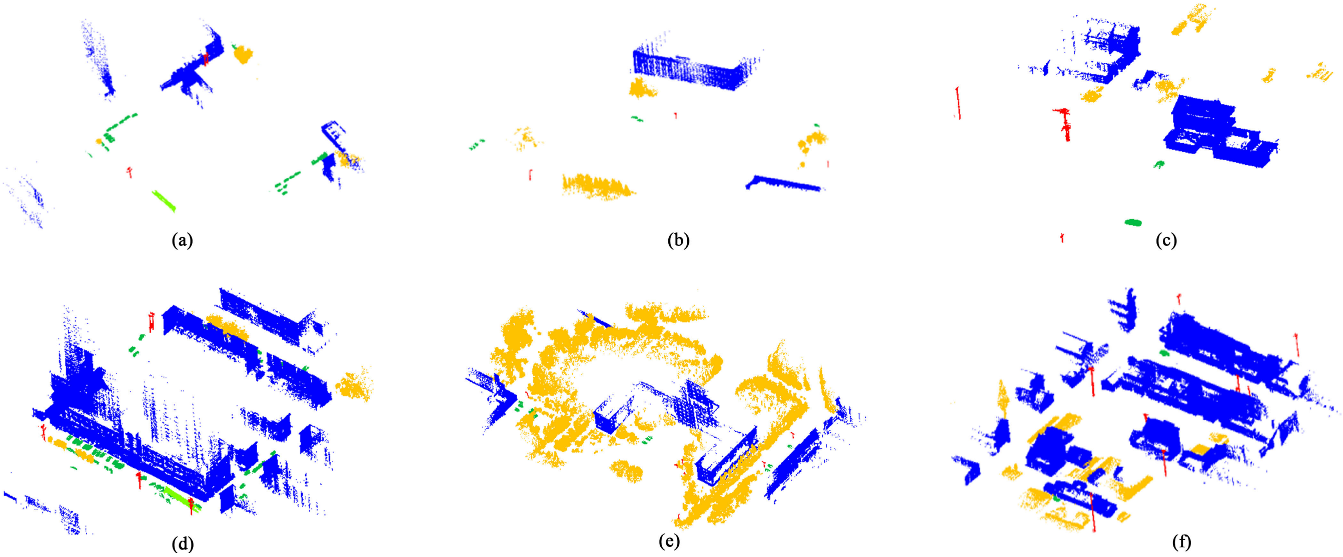

FIGURE 17. The ground truth of the experimental CSPC-Dataset. Subfigures (a)-(c) are training sets. Subfigures (d)-(f) are testing sets. Note that the left-most column, the middle column and the right-most column are selected from Scene 2, Scene 4 and Scene 5. Color legend: blue=building, dark green=vehicle, red=pole, yellow=vegetation and light green=bridge.

TABLE 5. Point clouds of Scene 2, Scene 4 and Scene 5 without including ground points. The symbol "-" represents the class does not exist in the corresponding scene.

Scenes	Data type	Building	Car	Bridge	Vegetation	Pole	Total	Ratio	Type
Scene 2	Training	656,850	188,455	39,459	72,859	18,290	975,913	19.6%	Urban
	Testing	3,407,169	337,360	51,178	184,849	25,640	3,934,252	80.4%	
Scene 4	Training	190,581	18,498	-	335,462	2,433	546,974	9.1%	Campus
	Testing	644,588	53,079	-	4,780,890	5,852	5,848,409	90.9%	
Scene 5	Training	558,032	16,470	-	30,367	22,423	627,292	13.6%	Town
	Testing	3,639,372	18,490	-	292,121	26,974	3,976,957	86.4%	

TABLE 6. Classification results of precision/recall, IoU/F_1-score, OA and Kappa (%). The symbol "-" represents the class does not exist in the corresponding scene. The highest metric values are highlighted in bold.

Scene 2	Pole	Building	Car	Vegetation	Bridge	OA	mIoU	Kappa
Method 1	**46.9**/16.5 13.9/24.4	94.8/**87.9** **83.8/91.2**	43.6/71.3 37.1/54.1	51.4/77.5 44.7/61.8	**31.4**/1.2 1.2/2.3	**84.4**	36.2	50.92
Method 2	46.5/48.6 **31.2/47.5**	99.2/72.6 72.2/83.8	38.3/93.1 37.3/54.3	**64.9/94.5** **62.5/77.0**	6.1/**44.9** 5.7/10.7	74.9	41.8	44.38
Method 3	11.3/716.6 7.2/22.2	96.9/76.1 74.3/85.2	47.9/88.4 45.1/62.2	39.3/86.1 37.0/54.0	29.9/28.1 **16.9/29.0**	76.9	36.1	47.15
Method 4	1.1/27.7 1.0/2.1	90.6/52.9 50.2/66.8	8.0/13.3 5.3/10.0	18.0/73.4 16.9/28.9	0.0/0.0 0.0/0.0	49.7	14.7	9.70
Method 5	33.8/**71.9** 29.9/46.0	**99.3**/81.2 71.0/83.0	**53.6/94.4** **51.9/68.4**	43.4/81.7 39.5/56.7	9.2/37.6 8.0/14.8	81.7	**42**	**53.67**
Method 6	7.1/26.7 6.0/11.2	95.6/60.7 59.1/74.3	21.9/65.0 19.6/32.8	25.4/63.6 22.2/36.3	0.0/0.0 6.5/12.3	60.7	22.7	23.69
Method 7	6.4/27.7 5.5/10.4	95.5/74.0 71.5/83.4	27.9/63.5 24.1/38.8	34.7/70.8 30.4/46.6	18.6/34.5 13.7/24.2	72.2	29.1	33.9
Scene 4	**Pole**	**Building**	**Car**	**Vegetation**	**Bridge**	**OA**	**mIoU**	**Kappa**
Method 1	**17.6/55.7** **15.4/26.8**	66.1/**79.2** 56.3/72.1	7.7/8.5 4.2/8.1	**96.2**/92.7 89.5/94.4	-	89.9	41.4	64.02
Method 2	0.0/0.0 0.0/0.0	0.0/0.0 0.0/0.0	0.0/0.0 0.0/0.0	87.2/**100** 87.2/93.2	-	87.2	21.8	0.25
Method 3	0.5/2.0 0.4/0.8	21.7/71.8 20.0/33.3	0.0/0.0 0.0/0.0	90.2/84.6 77.4/87.3	-	88.3	24.5	30.13
Method 4	0.0/0.0 0.0/0.0	21.1/41.5 16.3/28.0	0.0/0.0 0.0/0.0	90.7/80.0 73.9/85.0	-	74.6	22.6	16.13
Method 5	0.0/0.0 0.0/0.0	**88.3**/73.0 **66.6/79.9**	**15.9/50.1** **13.7/24.1**	95.8/95.8 **92.0/95.8**	-	**92.6**	**43.1**	**67.58**
Method 6	0.6/3.6 0.6/1.0	32.1/47.6 23.7/38.3	4.3/25.3 3.8/7.4	92.0/80.6 75.3/85.9	-	76.1	25.9	23.70
Method 7	2.4/15.5 2.2/4.2	38.4/53.5 28.8/44.7	4.7/25.5 4.1/7.9	93.8/83.6 79.2/88.4	-	79.4	28.6	32.10
Scene 5	**Pole**	**Building**	**Car**	**Vegetation**	**Bridge**	**OA**	**mIoU**	**Kappa**
Method 1	14.1/63.3 13.0/23.1	95.5/98.0 **93.7/96.7**	20.1/5.3 4.3/8.4	66.9/26.1 23.1/37.6	-	92.1	33.5	42.48
Method 2	67.0/4.2 4.2/7.9	92.2/**99.8** 92.1/95.8	0.0/0.0 0.0/0.0	**82.5**/9.8 9.6/17.5	-	92.1	26.5	14.81
Method 3	**71.5**/7.9 7.6/14.2	92.2/99.7 91.9/95.8	**50.2**/0.9 0.9/1.8	81.4/9.8 9.6/17.5	-	90.8	27.6	15.35
Method 4	0.0/0.0 0.0/0.0	92.2/97.6 90.2/94.8	0.0/0.0 0.0/0.0	30.8/12.2 9.5/17.5	-	90.2	25.0	12.46
Method 5	36.5/**96.2** **36.0/52.9**	**98.9**/94.1 93.1/96.4	10.0/**61.5** **9.4/17.2**	72.7/**81.2** **62.2/76.7**	-	**93.1**	**50.2**	**64.63**
Method 6	2.8/25.2 2.6/5.0	94.8/82.9 79.3/88.5	2.4/17.6 2.2/4.2	26.8/38.7 18.8/31.7	-	78.9	25.7	18.82
Method 7	4.2/24.2 3.7/7.2	94.6/87.9 83.7/91.1	2.7/20.7 2.4/4.8	38.2/38.6 23.8/38.4	-	83.5	28.4	23.61

including three machine learning methods and four deep learning methods described in Section IV are used to conduct experiments in the selected three scenes. The benchmark of this group experiment is constructed based on the evaluation metrics in Section V-A The experimental results are shown in Table 6.

From this table, the results show that the overall evaluation metrics of the machine learning method are mostly superior to the deep learning methods. However, for all algorithms, deep learning-based Method 5 has the best classification performance in three scenes. This is because Method 5 processes large-scale point clouds by the strategy of segmentation and resampling to enhance the training data. The more discriminative classification model can be trained with the enhanced training point clouds. Method 2 and Method 3 perform worse in Scene 4 because the features constructed by statistics the neighborhood of points are relatively simple. The discrepancy between various types of point clouds in a complex scene cannot be well represented. In Method 1, which is also a machine learning method, the features of spin image and covariance eigenvalue features are more discriminant than those in Method 2 and Method 3 in the complex scene. In addition, although Method 2 and Method 3 use sample features, Method 3 uses point sets as the classification units and constructs a high-order CRF model for optimization, allowing relatively high classification accuracy than Method 2. It can be seen that the adopted classification units, i.e., single point or point set, the ability of feature expression, and the used optimization process are both dominant factors for semantic labeling of point clouds.

For deep learning methods, Method 5 achieved the highest values of 81.7%/42.0%/53.7%, 92.6%/43.1%67.6% and 93.1%/50.2%/64.6% with regard to *OA*, *mIoU* and *Kappa* in Scene2, Scene4 and Scene5, respectively. It outperforms the other three deep learning methods. For Method 4, the number of each type of point cloud in the training set has larger differences. For classes of poles and bridges, they account for only a relatively small proportion in the projected images derived from point clouds. The model has poor discrimination ability for these two categories. For Method 5, as the dataset is preprocessed by segmentation and coordinate transformation, and the training samples are downsampled or resampled, the trained model has a relatively good performance. But we should note that although Methods 5 outperforms other deep learning based methods, the labeling accuracy for poles and cars is not ideal due to the limited number of training samples. For voxel-based methods such as Method 6 and Method 7, due to the small number of samples in the training set and the insufficient sample types, the trained model can only obtain promising results regarding samples enriched categories on precision/recall and IoU/F_1-score. The performance degeneration for overall classification regarding *OA*, *mIoU*, and *Kappa* is significantly reduced. Through experiments, we also find that for the deep learning methods based on voxel, the more voxel scales there are, the better of classification performance of the network model can be achieved.

2) GROUP 2 BENCHMARK

Deep learning methods are affected by the number of training samples and the enriched sample types. To make the dataset better adapt to deep learning methods and represent the advantages of the deep learning methods, we conduct group 2 benchmarking experiments. More precisely, Scene 2, Scene 4 and Scene 5 are still used. In each scene, 70% of the points are randomly selected as the training set, and the rest points are used as the testing set. The K-fold cross-validation method is used to carry out the experiment. In this section, the parameter K is set to 3. Here, we assume that Scene 2 contains six categories of objects, and Scene 4, and Scene 5 only include five types of objects.

As shown in Table 7, four state-of-the-art deep learning networks, i.e., Snapnet (Method 4), PointNet++ (Method 5), Label-3D CNN (Method 6) and DeepNet (Method 7), are used for semantic labeling of three different large-scale point cloud scenes. From Table 7, we can see that Method 4 performs worse than the other three deep-learning based methods in three scenes, especially for labeling pole, bridge, and car. These classes have a relatively small size of geometry or have only a few samples for training, making it hard to be classified well. Although the absolute numbers of these three categories training samples are increased significantly, the relatively small proportion of these three objects in projection images results in performance degeneration of Method 4 using the imbalance sample for training. In the overall evaluation of the three scenes, Method 5 achieved the highest values of 93.3%/56.7%/88.2%, 96.6%/56.2%/93.1% and 93.1%/50.2%/64.6% with regards to *OA*, *mIoU* and *Kappa* in the three scenes, respectively. Compared with the results in Table 6, it can be seen that the classification performance of Method 5 is greatly improved which the *OA*, *mIoU* and *Kappa* at least increase 4%, 13.1% and 25.4% for Scene2, and Scene4, when the number of training data is increased accordingly. Similarly, compared with Table 6, the classification performance of voxel-based methods has greatly improved in this group experiment. However, the overall evaluation metrics of Method 7 are superior to Method 6, i.e., the *OA*, *mIoU* and *Kappa* values of Method 7 are at least 1.6%, 2.2% and 2.2% more than Method 6, which explains the increase of voxel scales can effectively enhance the ability of feature expression and classification model.

By making comparisons between Tables 6 and 7, although the *OA* of Scene 5 in group 2 is lower than the group 1, other metrics *mIoU* and *Kappa* of group 2 show that the classification performance of Methods 5-7 on the three scenes is improved. Thus, increasing the number of training samples can indeed obviously improve the performance of Methods 5-7.

3) GROUP 3 BENCHMARK

In the above two group experiments, the training and testing sets come from the same scene. The above two group experiments cannot evaluate the generalization ability of different labeling algorithms. To test algorithms' generalization ability, in this group, we choose the training points and testing points from different scenes. More specifically, for machine learning algorithms, we select the training points from Scene 2 and Scene 4 to train models separately and test labeling

TABLE 7. Classification results of precision/recall, IoU/F_1-score, OA and Kappa (%). The symbol "-" represents the class does not exist in the corresponding scene. The highest metric values are highlighted in bold.

Scene 2	Pole	Ground	Building	Car	Vegetation	Bridge	OA	mIoU	Kappa
Method 4	0.0/0.0 0.0/0.0	87.9/71.6 65.2/78.9	64.0/90.8 60.1/75.1	6.4/3.7 2.4/4.7	71.7/12.8 12.2/21.7	0.0/0.0 0.0/0.0	74.0	23.3	53.04
Method 5	0.0/0.0 0.0/0.0	**98.6/94.6** **93.4/96.6**	**93.8/93.9** **88.4/93.9**	**44.3/97.7** **43.9/61.0**	**88.4/96.0** **85.2/92.0**	**45.3/45.9** **29.5/45.6**	**93.3**	**56.7**	**88.22**
Method 6	9.7/5.3 3.6/6.9	93.8/88.1 83.3/90.9	86.7/78.2 69.8/82.2	14.0/64.3 13.0/23.0	47.8/33.1 24.3/39.1	16.6/28.8 11.8/21.1	80.6	34.3	67.03
Method 7	**17.0/9.0** **6.2/11.8**	93.3/90.3 84.8/91.2	86.3/79.0 70.2/82.5	16.0/68.3 14.9/25.9	59.0/34.7 27.9/43.7	37.2/26.3 18.2/30.8	82.2	37.1	69.26
Scene 4	**Pole**	**Ground**	**Building**	**Car**	**Vegetation**	**Bridge**	**OA**	**mIoU**	**Kappa**
Method 4	0.0/0.0 0.0/0.0	73.7/57.4 47.7/64.5	1.1/1.7 0.7/1.3	0.0/0.0 0.0/0.0	45.0/65.5 36.4/53.3	-	55.8	17.0	19.13
Method 5	**18.2/76.6** **17.3/29.4**	**99.4/98.4** **97.8/98.9**	**81.2/86.0** **71.7/83.5**	**44.9**/2.4 **2.3/4.6**	**94.2/97.4** **91.9/95.8**	-	**96.6**	**56.2**	**93.05**
Method 6	0.7/1.8 0.5/1.0	96.6/95.7 92.6/96.1	32.5/24.5 16.2/27.9	4.7/**4.3** **2.3**/4.5	76.9/82.9 66.4/79.8	-	87.3	35.6	73.68
Method 7	6.3/3.5 2.3/4.5	98.4/96.0 94.5/97.2	46.7/24.4 19.1/32.1	1.4/0.5 0.4/0.7	77.6/91.8 72.5/84.1	-	89.8	37.8	78.95
Scene 5	**Pole**	**Ground**	**Building**	**Car**	**Vegetation**	**Bridge**	**OA**	**mIoU**	**Kappa**
Method 4	0.0/0.0 0.0/0.0	76.5/45.6 40.0/57.1	60.5/85.5 54.9/70.9	0.0/0.0 0.0/0.0	18.4/37.1 14.1/24.6	-	64.4	21.8	31.81
Method 5	**35.1/74.5** **31.4/47.7**	**98.3/93.1** **91.7/95.6**	**93.2/97.6** **91.1/95.3**	**17.2**/3.6 3.1/6.0	**48.6/68.5** **39.7/56.9**	-	**94.5**	**51.4**	**89.37**
Method 6	4.5/29.2 4.0/7.8	87.4/82.4 73.6/84.8	86.8/70.1 63.3/77.6	6.1/**18.4** 4.8/9.2	6.1/60.8 5.9/11.1	-	75.7	30.4	57.84
Method 7	8.6/26.3 7.2/13.0	87.6/85.0 75.9/86.3	86.5/78.9 70.2/82.5	8.4/15.0 **5.7/10.8**	9.3/51.0 8.5/15.7	-	81.1	33.5	65.40

TABLE 8. Generalization ability test data for deep learning algorithms.

	Scene	Ground	Building	Car	Bridge	Vegetation	Poles	Total
Training	Scene 1	6,082,987	9,032,520	651,442	0	641,970	24,034	40,394,238
	Scene 3	873,666	5,996,456	469,271	97,712	163,830	46,579	
	Scene 4	10,282,388	835,169	71,577	0	5,116,352	8,285	
	Total	17,239,041	15,864,145	1,192,290	97,712	5,922,152	78,898	
Testing	Scene 2	4,358,082	3,992,075	525,815	90,637	257,708	43,930	9,268,247
	Scene 5	5,332,925	4,197,404	34,960	0	322,488	49,397	9,937,174

TABLE 9. Classification of Scene 5 results by the evaluation metrics of precision/recall, IoU/F_1-score, OA and Kappa (%). Note that the upper Scene 5 results are trained using the point clouds in Scene 2. In contrast, the lower Scene 5 results are trained using the point clouds in Scene 4.

Scene 5	Pole	Building	Car	Vegetation	OA	mIoU	Kappa
Method 1	**37.1/38.9** **23.5/38.0**	**95.3/69.2** **66.9/80.2**	1.6/94.2 **1.6/3.1**	15.3/**12.4** 7.4/13.4	**64.9**	**19.9**	**9.27**
Method 2	2.7/5.7 1.9/3.7	75.1/2.4 2.4/4.7	0.5/**95.7** 0.5/1.0	5.8/3.7 2.3/4.5	3.0	1.8	-0.53
Method 3	0.1/95.9 0.1/0.2	76.5/10.4 10.1/18.3	**7.4**/1.2 1.0/2.1	**96.4**/8.4 **8.4/15.5**	8.7	4.9	1.88
Scene 5	**Pole**	**Building**	**Car**	**Vegetation**	**OA**	**mIoU**	**Kappa**
Method 1	**49.5/29.1** **22.4/36.7**	**98.7/69.5** **68.9/81.6**	**7.4/11.6** **4.7/9.0**	**19.6/91.7** **19.2/32.3**	**70.6**	**28.8**	**23.6**
Method 2	0.7/87.8 0.7/1.4	96.3/3.3 3.3/6.4	0.0/0.0 0.0/0.0	15.7/11.7 7.2/13.4	4.5	2.8	1.0
Method 3	0.0/0.0 0.0/0.0	78.0/15.7 71.1/26.1	1.3/1.5 1.0/1.4	4.0/41.5 0.6/7.3	17.4	18.2	0.71

performance of these two models in Scene 5. For deep learning algorithms, we choose the training and testing sets according to the configurations depicted in Table 8.

The generalization capability of machine learning algorithms is shown in Table 9. The statistics show that the models of machine learning algorithms trained by Scene 2

TABLE 10. Classification results of precision/recall, IoU/F_1-score, OA and Kappa (%). The symbol "-" represents the class does not exist in the corresponding scene. The highest metric values are highlighted in bold.

Scene 2	Pole	Ground	Building	Car	Vegetation	Bridge	OA	mIoU	Kappa
Method 4	0.0/0.0 0.0/0.0	70.0/52.4 42.8/59.9	57.2/65.3 43.9/61.0	21.2/7.7 6.0/11.3	11.8/57.3 10.8/19.6	0.0/0.0 0.0/0.0	54.8	17.3	26.8
Method 5	**56.2/33.3** **26.4/41.8**	**95.0/97.6** **92.8/96.3**	**96.5/93.6** **90.5/95.0**	**84.9/74.2** **65.6/79.2**	**72.9/99.3** **72.5/84.1**	**90.2/75.2** **69.6/82.0**	**94.1**	**69.6**	**90.0**
Method 6	0.6/4.3 0.5/1.1	84.9/90.8 78.2/87.8	78.6/32.5 29.9/46.0	11.0/1.4 1.3/2.5	5.6/56.1 5.4/10.2	1.5/0.1 0.1/0.2	58.4	19.2	38.64
Method 7	0.4/0.7 0.3/0.5	89.5/88.2 79.9/88.8	83.3/37.9 35.3/52.1	13.8/19.0 8.7/16.0	8.8/80.3 8.6/15.9	0.0/0.0 0.0/0.0	61.2	22.2	43.35
Scene 5	**Pole**	**Ground**	**Building**	**Car**	**Vegetation**	**Bridge**	**OA**	**mIoU**	**Kappa**
Method 4	0.0/0.0 0.0/0.0	55.7/59.1 40.2/57.3	64.4/48.7 38.4/55.5	0.3/0.9 0.2/0.5	10.0/34.4 8.4/15.5	-	52.3	17.5	19.3
Method 5	**61.6/26.6** **22.8/37.2**	**95.2/94.0** **89.8/94.6**	**94.0/92.3** **87.2/93.1**	**25.9/88.5** **25.1/40.1**	**47.7/55.6** **34.5/51.3**	-	**91.2**	**51.9**	**84.09**
Method 6	1.9/7.3 1.5/3.1	84.7/81.4 71.0/83.0	84.2/63.2 56.5/72.2	1.6/6.1 1.3/2.5	10.4/41.2 9.1/16.6	-	69.9	27.9	50.71
Method 7	0.7/2.4 0.5/1.1	84.2/82.3 71.3/83.2	91.2/46.9 44.9/61.9	1.1/4.2 0.9/1.7	10.9/75.5 10.6/19.0	-	63.3	25.6	44.74

TABLE 11. Classification results of mIoU, OA and Kappa (%) for three groups of benchmarks. The symbol "-" represents the class does not exist in the corresponding scene. The highest metric values are highlighted in bold.

Scene 2	Metrics	Method 1	Method 2	Method 3	Method 4	Method 5	Method 6	Method 7
Group 1	OA	**84.4**	74.9	76.9	49.7	81.7	60.7	72.2
	mIoU	36.2	41.8	36.1	14.7	**42.0**	22.7	29.1
	Kappa	50.9	44.4	47.2	9.7	**53.7**	23.7	33.9
Group 2	OA	-	-	-	74	**93.3**	80.6	82.2
	mIoU	-	-	-	23.3	**56.7**	34.3	37.1
	Kappa	-	-	-	53	**88.2**	67.0	69.3
Group 3	OA	-	-	-	54.8	**94.1**	58.4	61.2
	mIoU	-	-	-	17.3	**69.6**	19.2	22.2
	Kappa	-	-	-	26.8	**90.0**	22.2	43.4
Scene 4	**Metrics**	**Method 1**	**Method 2**	**Method 3**	**Method 4**	**Method 5**	**Method 6**	**Method 7**
Group 1	OA	89.9	87.2	88.3	74.6	**92.6**	76.1	79.4
	mIoU	41.4	21.8	24.5	22.6	**43.1**	25.9	28.6
	Kappa	64.0	0.3	30.1	16.1	**67.6**	23.7	32.1
Group 2	OA	-	-	-	55.8	**96.6**	87.3	89.8
	mIoU	-	-	-	17.0	**56.2**	35.6	37.8
	Kappa	-	-	-	19.1	**93.1**	73.7	79.0
Group 3	OA	-	-	-	-	-	-	-
	mIoU	-	-	-	-	-	-	-
	Kappa	-	-	-	-	-	-	-
Scene 5	**Metrics**	**Method 1**	**Method 2**	**Method 3**	**Method 4**	**Method 5**	**Method 6**	**Method 7**
Group 1	OA	92.1	92.1	90.8	90.2	**93.1**	78.9	83.5
	mIoU	33.5	26.5	27.6	25.0	**50.2**	25.7	28.4
	Kappa	42.5	27.6	15.4	12.5	**64.6**	18.8	23.6
Group 2	OA	-	-	-	64.4	**94.5**	75.7	81.1
	mIoU	-	-	-	21.8	**51.4**	30.4	33.5
	Kappa	-	-	-	31.8	**89.4**	57.8	65.4
Group 3	OA	-	-	-	52.3	**91.2**	69.9	63.3
	mIoU	-	-	-	17.5	**51.9**	27.9	25.6
	Kappa	-	-	-	19.3	**84.1**	50.7	44.8

and Scene 4 cannot accurately predict point labels in Scene 5. The performance of Method 1 is superior to the other two methods, although its generalization capability is still unsatisfied. For example, the metrics *mIoU* and *Kappa* for labeling Scene 5 are only 19.9%/9.27% and 28.8%/23.6% using training point clouds from Scene 2 and Scene 4 due to a pretty huge discrepancy in terms of point's elevation between different scenes, causing a big discrepancy between coordinate-based features derived from different scenes. Method 1 is moderately affected by these coordinate-based

TABLE 12. The comparisons of different baselines for CSPC-dataset.

Types	Methods	Unit and pre-processing	Complexity	Advantages	Disadvantages
ML	Methods 1 and 2	Single point; no preprocessing	Low	Simple and high efficiency; Less training data	Limited accuracy and robustness, with more misclassified noise
	Method 3	Point set; Point cloud segmentation	Relatively low	Strong ability of feature expression, good classification robustness; with less classification noise; strong engineering applicability	Sensitive to the effect of point set construction; with certain constraints on the training set, training data cannot be randomly selected
	Method 4	Image; mesh and projection	Relatively high	Making full use of deep learning methods in the field of images, especially data sets and pre-training networks; considering a wider range of context information; multiple types of texture information can be used	With heavily dependent on the projection results and image semantic segmentation network; the classification accuracy and generalization performance of the model are susceptible to color, density, scene size and scene complexity; the selection of training set is more constrained
DL	Method 5	Point set; point cloud cutting	General	Less loss of 3D information, more directly reflecting the characteristics of point clouds; more feature extraction schemes; wide application range	The limited application of large outdoor scenes; need pre-processing; more complicated of the network settings; affected by the characteristics of disorder and scale invariance
	Methods 6 and 7	Voxel; voxelization	High	Simple method; can make full use of the advantages of CNN; relatively few constraints on the selection of training data	Affected by the voxel construction method; large network parameters; losing some 3D information on preprocessing; with classification noise

features; however, Method 2 and Method 3 are greatly affected by coordinate-based features and the statistical features, which have a huge discrepancy caused by the change of scene. To sum up, machine learning based algorithms have poor generalization performance due to the limited representation ability of features.

Four deep learning networks are trained by using the training data (Scene 1, Scene 3, and Scene 4) which contains 40,394,238 points, as shown in Table 8. Then, Scene 2 and Scene 5 are classified by the trained model, and the comparison results of generalization ability for deep learning networks are obtained, as shown in Table 10. In this table, the classification performance of Method 4 is the worst in both scenes. As this method requires a higher diversity of training samples, it cannot effectively classify the point cloud scenes that are significantly different from the training scene. Method 5 achieves the best classification performance among the four deep learning methods and has distinct advantages in the overall classification evaluation metrics, achieving the maximum of 96.6%, 56.7% and 93.05% regarding *OA*, *mIoU* and *Kappa*. However, for categories with very few samples, such as the poles, Method 4 is almost impossible to classify them, but voxel-based methods, such as Method 6 and Method 7, can correctly classify some categories with few samples. The performance of Method 6 in Scene 5 is superior to Method 7, because Method 7 having more voxel scales is more susceptible to the classification of point clouds with varied density.

Through making a comparison between Table 9 and Table 10, we can make a safe conclusion that machine learning algorithms are more suitable for labeling point clouds with fewer training samples. The generalization ability of the deep learning methods outperforms the machine learning methods, which is because the features used in machine learning algorithms are low-level features, weakening their expression ability. In contrast, the deep learning algorithms construct high-level features of point clouds, thereby enhancing the feature expression and semantic labeling accuracy.

D. DISCUSSIONS

The overall benchmark datasets for three groups of experiments with three comprehensive evaluation metrics are shown in Table 11. In this table, the overall performance of different methods on each group and the overall performance of the same method on different groups with different training samples can be observed clearly. According to Table 11, we can observe that with the increasing number of training samples, the performance of Methods 4-7 has been improved significantly. However, the projection-based Method 4 has the worst performance, and the point set-based deep learning Method 5 achieves the highest labeling accuracy. Therefore, unlike projection-based methods and voxel-based methods, which lose part of the original point cloud information, direct processing of the original point cloud can extract more expressive features of point clouds. In addition, Method 5 (improved PointNet++) preprocesses the input point clouds and uses coordinate transformation and segmentation of large-scale point cloud into many point cloud blocks to make the training data fully explored. Method 7 has many voxel scales, while Method 6 has only one voxel scale. The multi-scale voxels of Method 7 have more abundant representation for various objects. Therefore, the performance of Method 7 based on voxelization is better than Method 6 based on a single voxel scale. Because DeepNet and Pointnet++ adopt the multi-scale method, the discriminating ability of features extracted by the network is enhanced. More comparisons and summaries of different baselines for CSPC-dataset are shown in Table 12.

In addition, as shown in Table 8, the training set used in the experiments of Group 3 contains more samples than Group 2, and the training scenes and test scenes are different. Therefore, Table 10 can be further used to evaluate the effectiveness of various deep learning algorithms for large-scale point cloud semantic segmentation. Similar to Semantic3D, the benchmark constructed from Table 10 can be directly used to evaluate the performance of deep learning algorithms.

According to the comparison results of Table 11, we can conclude that most of the current labeling algorithms do not obtain a promising result for labeling CSPC-Dataset. Therefore, the point cloud semantic segmentation based on CSPC-Dataset is challenging. It should be noted that CSPC-Dataset contains five different scene types. Users can also carry out different types of experiments on the datasets according to specific user requirements.

VI. CONCLUSION

In this paper, we firstly analyze ubiquitous point clouds and compare the existing point cloud datasets and benchmarks. Besides, we also comprehensively review the point cloud semantic segmentation algorithms for scene understanding in Section I. After that, we introduce an improved backpack mobile mapping robot for large-scale point cloud acquisition and summary the characteristics of point clouds collected by that robot through the comparisons of different laser scanning systems in Section II. In Section III, we briefly describe the dataset acquisition method and environment. Then, we propose an improved backpack mobile mapping robot for large-scale point cloud acquisition. Based on Cloudcompare tools, we further propose a three-step point cloud labeling method, which progressively labels point clouds from coarse to fine labels in manual mode. Based on these labeled point clouds, a high-quality outdoor point cloud dataset, i,e., CSPC-Dataset is constructed, which contains nearly 68 million manually labeled points in 5 complex large-scale scenarios. Next, four characteristics of CSPC-Dataset are summarized. To show the point cloud semantic segmentation performance of different kinds of algorithms on CSPC-Dataset, we select seven state-of-the-art algorithms, including ML-based and DL-based algorithms, as the baselines for benchmarking in Section IV. To provide a comprehensive evaluation of the semantic labeling algorithms, we design three groups of experiments for benchmarking using seven representative evaluation metrics. In addiction, more comparisons, analysis and discussions of the benchmarks are provided in Section V. We hope that in the future, more semantic labeling algorithms and comparisons will be built on our datasets as the proposed dataset can help the in-depth study of the algorithm and promote significant progress in the scene understanding.

APPENDIX A
UBIQUITOUS POINT CLOUDS

See Table 13.

APPENDIX B
COMPARISONS OF EXISTING POINT CLOUD DATASETS

See Table 14.

TABLE 13. Ubiquitous Point Clouds.

Data Type	Data Samples	Characteristics	Disadvantages
RGB-D (Kinect) Fig. 1(a)	SUN RGB-D	Indoor small scene, dense point cloud, close-ranging scanning; including depth and color texture.	Easily affected by light, only suitable for indoor scenes, small field-of-view and measurement range.
CAD Model Fig. 1(b)	Modelnet40	Dense point cloud model, marking orientation.	Only small objects, without large outdoor objects.
Mobile laser scanning 3D point clouds Fig. 1(c)	Oakland dataset	Outdoor scene, relatively sparse, with XYZ coordinates and intensity, less affected by environmental factors.	Limited by the platform, incomplete point clouds.
Terrestrial laser scanning 3D point clouds. Fig. 1(d)	Semantic3D	Outdoor scene, high-precision information, dense point cloud, with XYZ coordinates and intensity.	Fixed position, incomplete and seriously affected by additional self-occlusions, occlusions from objects and background clusters.
Airborne laser scanning 3D point clouds Fig. 1(e)	Dataset provided in [13]	Outdoor scene, relatively low accuracy, sparse point cloud, suitable for large scale rough modeling.	Without ground details and color, relatively sparse.
Backpacked laser scanning 3D point clouds Fig. 1(f)	CSPC-Dataset	Outdoor scene, high-precision information, dense point cloud, mobile modeling, high passability, with with XYZ coordinates, color and intensity, less affected by environmental factors.	Color information affected by lighting and other factors.

TABLE 14. Comparisons of existing point cloud datasets.

Dataset	Acquisition equipment	Scene Type	Acquisition method	Point number	Class number	Data type
NYUv2 [6]	Kinect v1	indoor	mobile acquisition, video sequences of various indoor scenes	1,449 RGB-D labeled images	26 scene types, 1,000+ Classes	R, G, B, D, label
SUN RGB-D [17]	Kinect v1, Kinect v2, Intel RealSense and Asus Xtion Live Pro	indoor	recording: 30Hz synchronization and alignment: 640,480 RGB image and depth image	146,617 2D polygons, 58,657 3D bounding boxes, 10,000 RGB-D images	37 object categories	R, G, B, D, label
ModelNet10/40 [7]	CAD models	indoor	3DShapeNets generating	4,900/12,311 models	10/40	.off, X, Y, Z
IQmulus&TerraMobilita Contest [26]	mobile laser scans	outdoor	mobile mapping car	300 million	8	.ply, X, Y, Z, I, label
V4RL Aerial Inspection [73]	VI-Sensor [74] and Leica TS15 Total Station	outdoor	small rotorcraft UAV	<107	No class	R, G, B, X, Y, Z, I, label
Dataset released in [13]	High-definition laser scanner (Leica ALS50 system)	outdoor	ALS, Flying height: 500m, Field of view: 45°	879,746	3	X, Y, Z, I, label
Semantic3D [27]	Static laser scan	outdoor	static scanning	4 billion	8	R,G,B,X,Y,Z,I, label
Oakland [9]	Navlab11, equipped with side view SICK LMS laser scanner	outdoor	push-broom	1.6 million	5 (total 44 labels)	X, Y, Z, label, confidence
Sydney Urban Objects [8]	Velodyne HDL-64E LiDAR	outdoor	mobile mapping car	631 individual scans of objects	26	t, intensity, id, X, Y, Z, azimuth, range, pid, label
KAIST [24]	Laser scanner (SICK LMS 291), camera (PGR Firefly MV), DGPS (HUACE B20), and IMU (MTi).	outdoor	mobile mapping car	∼2,88 million	5	R, G, B, X, Y, Z, I, label
CSPC-Dataset	**Omni SLAM**^{TM}	**outdoor**	**mobile mapping robots**	**<70 million**	**6**	**R, G, B, X, Y, Z, I**

ACKNOWLEDGEMENT

The authors would like to thank Jianjun Zhang, Wenbo Zhao, Hao Wang, and Guangxu Ji at Northeastern University for data collecting and labeling. The authors would like to thank Xiance Du, Lin Yang, and Tingting Zhu for providing the experiment results of Methods 3-5 for comparisons. *(Guofeng Tong, Yong Li, and Dong Chen are co-first authors.)*

REFERENCES

[1] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[2] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[3] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2014, pp. 740–755.

[4] Z. Yang, Y. Sun, S. Liu, X. Shen, and J. Jia, "IPOD: Intensive point-based object detector for point cloud," 2018, *arXiv:1812.05276*. [Online]. Available: http://arxiv.org/abs/1812.05276

[5] B. Yang, Z. Dong, Y. Liu, F. Liang, and Y. Wang, "Computing multiple aggregation levels and contextual features for road facilities recognition using mobile laser scanning data," *ISPRS J. Photogramm. Remote Sens.*, vol. 126, pp. 180–194, Apr. 2017.

[6] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2012, pp. 746–760.

[7] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3D ShapeNets: A deep representation for volumetric shapes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1912–1920.

[8] M. De Deuge, A. Quadros, C. Hung, and B. Douillard, "Unsupervised feature learning for classification of outdoor 3D scans," in *Proc. Australas. Conf. Robitics Autom.*, vol. 2, 2013, p. 1.

[9] D. Munoz, J. A. Bagnell, N. Vandapel, and M. Hebert, "Contextual classification with functional max-margin Markov networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 975–982.

[10] A. Serna, B. Marcotegui, F. Goulette, and J.-E. Deschaud, "Paris-rue-madame database: A 3D mobile laser scanner dataset for benchmarking urban detection, segmentation and classification methods," in *Proc. INSTICC 3rd Int. Conf. Pattern Recognit., Appl. Methods (ICPRAM)*, Angers, France. Cham, Switzerland: Springer, Mar. 2014.

[11] X. Roynard, J.-E. Deschaud, and F. Goulette, "Paris-Lille-3D: A large and high-quality ground-truth urban point cloud dataset for automatic segmentation and classification," *Int. J. Robot. Res.*, vol. 37, no. 6, pp. 545–557, May 2018.

[12] H. Huang, L. Wang, B. Jiang, and D. Luo, "Precision verification of 3D slam backpacked mobile mapping robot," *Bull. Surv. Mapp*, vol. 12, pp. 68–73, Dec. 2016.

[13] Z. Zhang, L. Zhang, X. Tong, P. T. Mathiopoulos, B. Guo, X. Huang, Z. Wang, and Y. Wang, "A multilevel point-cluster-based discriminative feature for ALS point cloud classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 6, pp. 3309–3321, Jun. 2016.

[14] X. Ren, L. Bo, and D. Fox, "RGB-(D) scene labeling: Features and algorithms," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2759–2766.

[15] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, "Learning rich features from RGB-D images for object detection and segmentation," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2014, pp. 345–360.

[16] S. Song and J. Xiao, "Sliding shapes for 3D object detection in depth images," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2014, pp. 634–651.

[17] S. Song, S. P. Lichtenberg, and J. Xiao, "SUN RGB-D: A RGB-D scene understanding benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 567–576.

[18] K. Lai, L. Bo, X. Ren, and D. Fox, "A scalable tree-based approach for joint object and pose recognition," in *Proc. 25th AAAI Conf. Artif. Intell.*, 2011, pp. 1–7.

[19] J. Xiao, A. Owens, and A. Torralba, "SUN3D: A database of big spaces reconstructed using SFM and object labels," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1625–1632.

[20] I. Armeni, O. Sener, A. R. Zamir, H. Jiang, I. Brilakis, M. Fischer, and S. Savarese, "3D semantic parsing of large-scale indoor spaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1534–1543.

[21] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Niessner, "ScanNet: Richly-annotated 3D reconstructions of indoor scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5828–5839.

[22] A. Brock, T. Lim, J. M. Ritchie, and N. Weston, "Generative and discriminative voxel modeling with convolutional neural networks," 2016, *arXiv:1608.04236*. [Online]. Available: http://arxiv.org/abs/1608.04236

[23] M. De Deuge, A. Quadros, C. Hung, and B. Douillard, "Unsupervised feature learning for classification of outdoor 3D scans," in *Proc. Australas. Conf. Robitics Autom.*, vol. 2, 2013, p. 1.

[24] Y. Choe, I. Shim, and M. J. Chung, "Urban structure classification using the 3D normal distribution transform for practical robot applications," *Adv. Robot.*, vol. 27, no. 5, pp. 351–371, Apr. 2013.

[25] D. Munoz, J. A. Bagnell, N. Vandapel, and M. Hebert, "Contextual classification with functional max-margin Markov networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 975–982.

[26] B. Vallet, M. Brédif, A. Serna, B. Marcotegui, and N. Paparoditis, "TerraMobilita/iQmulus urban point cloud analysis benchmark," *Comput. Graph.*, vol. 49, pp. 126–133, Jun. 2015.

[27] T. Hackel, N. Savinov, L. Ladicky, J. D. Wegner, K. Schindler, and M. Pollefeys, "Semantic3D.Net: A new large-scale point cloud classification benchmark," 2017, *arXiv:1704.03847*. [Online]. Available: http://arxiv.org/abs/1704.03847

[28] J. Niemeyer, F. Rottensteiner, and U. Soergel, "Contextual classification of lidar data and building object detection in urban areas," *ISPRS J. Photogramm. Remote Sens.*, vol. 87, pp. 152–165, Jan. 2014.

[29] T. Hackel, J. D. Wegner, and K. Schindler, "Fast semantic segmentation of 3D point clouds with strongly varying density," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 3, pp. 177–184, Jun. 2016.

[30] Y. Li, G. Tong, X. Du, X. Yang, J. Zhang, and L. Yang, "A single point-based multilevel features fusion and pyramid neighborhood optimization method for ALS point cloud classification," *Appl. Sci.*, vol. 9, no. 5, p. 951, 2019.

[31] S. K. Lodha, E. J. Kreps, D. P. Helmbold, and D. Fitzpatrick, "Aerial LiDAR data classification using support vector machines (SVM)," in *Proc. 3rd Int. Symp. 3D Data Process., Visualizat., Transmiss. (3DPVT)*, Jun. 2006, pp. 567–574.

[32] J. Mei, L. Zhang, Y. Wang, Z. Zhu, and H. Ding, "Joint margin, cograph, and label constraints for semisupervised scene parsing from point clouds," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 7, pp. 3800–3813, Jul. 2018.

[33] M. Weinmann, B. Jutzi, S. Hinz, and C. Mallet, "Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers," *ISPRS J. Photogramm. Remote Sens.*, vol. 105, pp. 286–304, Jul. 2015.

[34] A. Aijazi, P. Checchin, and L. Trassoudaine, "Segmentation based classification of 3D urban point clouds: A super-voxel based approach with evaluation," *Remote Sens.*, vol. 5, no. 4, pp. 1624–1650, 2013.

[35] B. Guo, X. Huang, F. Zhang, and G. Sohn, "Classification of airborne laser scanning data using jointboost," *ISPRS J. Photogramm. Remote Sens.*, vol. 100, pp. 71–83, Feb. 2015.

[36] M. Li and C. Sun, "Refinement of LiDAR point clouds using a super voxel based approach," *ISPRS J. Photogramm. Remote Sens.*, vol. 143, pp. 213–221, Sep. 2018.

[37] H. Ni, X. Lin, and J. Zhang, "Classification of ALS point cloud with improved point cloud segmentation and random forests," *Remote Sens.*, vol. 9, no. 3, p. 288, 2017.

[38] Z. Wang, L. Zhang, T. Fang, P. T. Mathiopoulos, X. Tong, H. Qu, Z. Xiao, F. Li, and D. Chen, "A multiscale and hierarchical feature extraction method for terrestrial laser scanning point cloud classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2409–2425, May 2015.

[39] B. Xiang, J. Yao, X. Lu, L. Li, and R. Xie, "Segmentation-based classification for 3D urban point clouds," in *Proc. IEEE Int. Conf. Inf. Autom. (ICIA)*, Aug. 2016, pp. 172–177.

[40] Z. Zhang, L. Zhang, X. Tong, B. Guo, L. Zhang, and X. Xing, "Discriminative-dictionary-learning-based multilevel point-cluster features for ALS point-cloud classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7309–7322, Dec. 2016.

[41] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3D shape recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 945–953.

[42] C. R. Qi, H. Su, M. NieBner, A. Dai, M. Yan, and L. J. Guibas, "Volumetric and multi-view CNNs for object classification on 3D data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5648–5656.

[43] A. Boulch, J. Guerry, B. Le Saux, and N. Audebert, "SnapNet: 3D point cloud semantic labeling with 2D deep segmentation networks," *Comput. Graph.*, vol. 71, pp. 189–198, Apr. 2018.

[44] F. J. Lawin, M. Danelljan, P. Tosteberg, G. Bhat, F. S. Khan, and M. Felsberg, "Deep projective 3d semantic segmentation," in *Proc. Int. Conf. Comput. Anal. Images Patterns*. Cham, Switzerland: Springer, 2017, pp. 95–107.

[45] D. Maturana and S. Scherer, "VoxNet: A 3D convolutional neural network for real-time object recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 922–928.

[46] G. Riegler, A. O. Ulusoy, and A. Geiger, "OctNet: Learning deep 3D representations at high resolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3577–3586.

[47] J. Huang and S. You, "Point cloud labeling using 3D convolutional neural network," in *Proc. 23rd Int. Conf. Pattern Recognit. (ICPR)*, Dec. 2016, pp. 2670–2675.

[48] Y. Li, G. Tong, X. Li, L. Zhang, and H. Peng, "MVF-CNN: Fusion of multilevel features for large-scale point cloud classification," *IEEE Access*, vol. 7, pp. 46522–46537, 2019.

[49] L. Wang, Y. Huang, J. Shan, and L. He, "MSNet: Multi-scale convolutional network for point cloud classification," *Remote Sens.*, vol. 10, no. 4, p. 612, 2018.

[50] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 652–660.

[51] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5099–5108.

[52] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "Pointcnn: Convolution on X-transformed points," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 820–830.

[53] M. Jiang, Y. Wu, T. Zhao, Z. Zhao, and C. Lu, "PointSIFT: A SIFT-like network module for 3D point cloud semantic segmentation," 2018, *arXiv:1807.00652*. [Online]. Available: http://arxiv.org/abs/1807.00652

[54] Y. Liu, B. Fan, S. Xiang, and C. Pan, "Relation-shape convolutional neural network for point cloud analysis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8895–8904.

[55] H. Thomas, C. R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette, and L. Guibas, "KPConv: Flexible and deformable convolution for point clouds," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6411–6420.

[56] J. Li, B. M. Chen, and G. H. Lee, "SO-net: Self-organizing network for point cloud analysis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9397–9406.

[57] A. Behl, D. Paschalidou, S. Donné, and A. Geiger, "PointFlowNet: Learning representations for rigid motion estimation from point clouds," 2018, *arXiv:1806.02170*. [Online]. Available: https://arxiv.org/abs/1806.02170

[58] L. Landrieu and M. Simonovsky, "Large-scale point cloud semantic segmentation with superpoint graphs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4558–4567.

[59] Z. Xie, J. Chen, and B. Peng, "Point clouds learning with attention-based graph convolution networks," *Neurocomputing*, early access, Apr. 8, 2020, doi: 10.1016/j.neucom.2020.03.086.

[60] K. Zhang, M. Hao, J. Wang, C. W. de Silva, and C. Fu, "Linked dynamic graph CNN: Learning on point cloud via linking hierarchical features," 2019, *arXiv:1904.10014*. [Online]. Available: http://arxiv.org/abs/1904.10014

[61] G. Te, W. Hu, A. Zheng, and Z. Guo, "RGCNN: Regularized graph CNN for point cloud segmentation," in *Proc. ACM Multimedia Conf. Multimedia Conf. (MM)*, 2018, pp. 746–754.

[62] Q. Hu, B. Yang, L. Xie, S. Rosa, Y. Guo, Z. Wang, N. Trigoni, and A. Markham, "RandLA-net: Efficient semantic segmentation of large-scale point clouds," 2019, *arXiv:1911.11236*. [Online]. Available: http://arxiv.org/abs/1911.11236

[63] C. Wen, S. Pan, C. Wang, and J. Li, "An indoor backpack system for 2-D and 3-D mapping of building interiors," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 7, pp. 992–996, Jul. 2016.

[64] C. Wang, S. Hou, C. Wen, Z. Gong, Q. Li, X. Sun, and J. Li, "Semantic line framework-based indoor building modeling using backpacked laser scanning point cloud," *ISPRS J. Photogramm. Remote Sens.*, vol. 143, pp. 150–166, Sep. 2018.

[65] G. Tong, B. Jiang, D. Liang, and Y. Li, "A backpack mobile mapping system based on laser and panoramic camera," China Patent 106 443 687 B, Aug. 31, 2016.

[66] A. Harrison and P. Newman, "TICSync: Knowing when things happened," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2011, pp. 356–363.

[67] J. Zhang and S. Singh, "LOAM: Lidar odometry and mapping in real-time," in *Proc. Robot., Sci. Syst.*, vol. 2, Jul. 2014, p. 9.

[68] W. Zhang, J. Qi, P. Wan, H. Wang, D. Xie, X. Wang, and G. Yan, "An easy-to-use airborne LiDAR data filtering method based on cloth simulation," *Remote Sens.*, vol. 8, no. 6, p. 501, 2016.

[69] J. Gehrung, M. Hebel, M. Arens, and U. Stilla, "An approach to extract moving objects from MLS data using a volumetric background representation," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 4, pp. 107–114, May 2017.

[70] Y. Li, D. Chen, X. Du, S. Xia, Y. Wang, S. Xu, and Q. Yang, "Higher-order conditional random fields-based 3D semantic labeling of airborne laser-scanning point clouds," *Remote Sens.*, vol. 11, no. 10, p. 1248, 2019.

[71] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241.

[72] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[73] L. Teixeira and M. Chli, "Real-time mesh-based scene estimation for aerial inspection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 4863–4869.

[74] J. Nikolic, J. Rehder, M. Burri, P. Gohl, S. Leutenegger, P. T. Furgale, and R. Siegwart, "A synchronized visual-inertial sensor system with FPGA pre-processing for accurate real-time SLAM," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 431–437.

GUOFENG TONG received the Ph.D. degree in control theory and control engineering from the College of Information Science and Engineering, Northeastern University, Shenyang, China, in 2001. He is currently a Professor with the College of Information Science and Engineering, Northeastern University. His research interests include computer vision, 3-D urban reconstruction, and deep learning.

YONG LI received the M.S. degree in pattern recognition and intelligent systems from Yanshan University, China, in 2016. He is currently pursuing the Ph.D. degree with the College of Information Science and Engineering, Northeastern University, Shenyang, China. His research interests include point cloud processing, computer vision, and pattern recognition.

DONG CHEN (Member, IEEE) received the Ph.D. degree in geographical information sciences from Beijing Normal University, Beijing, China, in 2013. He is an Associate Professor with Nanjing Forestry University, Nanjing, China. He is also a Postdoctoral Fellow with the Department of Geomatics Engineering, University of Calgary, Calgary, AB, Canada. His research interests include image-and LiDAR-based segmentation and reconstruction, full-waveform LiDAR data processing, and related remote sensing applications in the field of forest ecosystems.

QI SUN is currently pursuing the master's degree with the College of Information Science and Engineering, Northeastern University, Shenyang, China. His research interests include laser point cloud classification, LiDAR-based segmentation, and reconstruction.

WEI CAO is currently pursuing the master's degree with the College of Civil Engineering, Nanjing Forestry University, Nanjing, China. His research interests include tree modeling and geometry processing.

GUIQIU XIANG is currently pursuing the master's degree with the College of Civil Engineering, Nanjing Forestry University, Nanjing, China. His research interests include laser point cloud processing based on the deep-learning framework.