3D Large-Scale Point Cloud Semantic Segmentation Using Optimal Feature Description Vector Network: OFDV-Net

JIAN LI$^{1}$, QUAN SUN$^{2}$, KERU CHEN$^{2}$, HAO CUI$^{1}$, KUAN HUANGFU$^{2}$, AND XIAOLONG CHEN$^{2}$

$^{1}$School of the Geo-Science and Technology, Zhengzhou University, Zhengzhou 450001, China
$^{2}$School of Water Conservancy Science and Engineering, Zhengzhou University, Zhengzhou 450001, China

Corresponding author: Quan Sun (sunquan@gs.zzu.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFC0406501, in part by the Foundation of Henan Educational Committee under Grant 15A420002 and Grant 14A420002, in part by the China Postdoctoral Science Foundation under Grant 2019M662534, and in part by the National Natural Science Foundation under Grant 42001405.

ABSTRACT Efficient semantic segmentation of large-scale 3D point clouds is a fundamental and essential capability for real-time intelligent systems, such as autonomous driving and augmented reality. The high dimension feature vector and the complex network structure are two major constraints to utilize the large-scale point cloud. This paper proposes an optimal feature description vector network (OFDV-Net) for 3D point cloud semantic segmentation. First, a multiscale point cloud feature extraction structure is constructed to generate an initial feature description vector (IFDV). Then, IFDV is selected by a feature selection unit to obtain the optimal feature description vector (OFDV). The OFDV encapsulates the best 3D features set of the points and can be used as the input of the deep neural network for training and testing. Finally, the OFDV-Net was applied to the standard public outdoor large-scale point cloud datasets Semantic3D and NPM3D, and the overall segmentation accuracy of 88.3% and 87.7% were obtained, respectively; moreover, the OFDV-Net requires less training time, which indicates that the algorithm can obtain high-precision semantic segmentation results on an outdoor large-scale point cloud while reducing model training time.

INDEX TERMS Optimal feature description vector, point cloud, feature selection, semantic segmentation, deep learning.

I. INTRODUCTION

With the development of 3D sensors, such as lidar and stereo cameras, point cloud acquisition is increasingly easier and more convenient. Efficient and accurate large-scarse point cloud semantic segmentation is particularly important, and has already become a substantial topic in many areas, such as geomatics spatial information technology, navigation positioning, computer vision, and pattern recognition [1]. In the point cloud, points are independent and lack contextual information, making it difficult to infer semantic information directly. A lot of traditional point cloud semantic segmentation methods firstly calculated the geometric features [2] of the point cloud based on the neighborhood [3], then used machine learning algorithms (random forest, support vector machine) to classify the point cloud [4]. These methods achieve good point cloud semantic segmentation results, but their precision needs to be improved.

In recent years, with the rapid development of deep learning, many attempts have been made to apply deep learning to point cloud semantic segmentation, but there are numerous challenges to overcome. The scattered properties of 3D point clouds and the lack of clear structure akin to the regular grid arrangement in images means that deep learning cannot be applicable to point cloud segmentation directly. Previous approaches mainly transform 3D point cloud into 2D images and regular voxel grid. However, converting the point clouds into 2D formats results in the loss of information. Voxelization often discards small details and leads to heavy calculation and low efficiency due to the sparsity of point clouds. PointNet [5] was the first work to directly process 3D point clouds. However, this work failed to capture the
local features of point clouds. To overcome this drawback, PointNet++ [6] was proposed. Inspired by this algorithm, many attempts have emerged to apply deep learning to point cloud semantic segmentation.

However, existing point clouds semantic segmentation methods do not make full use of common features in point cloud processing such as curvature and roughness. If some semantic information with advanced prior knowledge can be extracted prior to neural network training, a better segmentation result could be obtained. Inspired by this idea, we proposed a large-scale 3D point cloud semantic segmentation network using optimal feature description vector network (OFDV-Net). The network consists of three units: First, in the initial feature description vector (IFDV) preprocess unit, we extracted a series of features by setting different neighborhood sizes after random sampling and constructed the IFDV. Then, in the feature selection unit, the IFDV was optimized to the OFDV. Finally, in the semantic segmentation unit, the OFDV is input into the fully connected neural network to obtain the semantic segmentation results. Our network structure is very simple but can achieve a high precision semantic segmentation result while reducing model training time.

The main contributions of this study are listed as follows:

- We designed a new network OFDV-Net, which can fully incorporate traditional feature extraction methods, feature selection methods and deep learning methods for highly accurate point cloud semantic segmentation while reducing model training time. We first extracted the initial features from 3D points which are usually cannot be ‘seen’ directly by neural networks, and optimized the initial features into the optimal features based on feature selection methods. The OFDV was input into the fully connected neural network to obtain the semantic segmentation results.

- We constructed the optimal feature description vector (OFDV) by optimizing each feature vector in the initial feature description vector (IFDV). Our OFDV encapsulates the best 3D features set of the points and can be used as the input of the deep neural network for training and testing, which can be a guideline in similar work.

II. RELATED WORK

Traditional point cloud semantic segmentation methods are based on the features extracted from local neighborhoods, and use machine learning methods to perform point cloud semantic segmentation. The precision of this scheme mainly depends on the quality of the extracted features. Therefore, a feature set with rich characterization capabilities is needed to describe the geometric figures in a point neighborhood. Demantke et al. [4] first proposed features based on the local covariance matrix for point cloud semantic segmentation. Weinamann et al. [9] added descriptors such as verticality and height distribution on this basis. Chehata et al. [10] realized the airborne point cloud semantic segmentation by using random forests with 21-dimensional features. Kragh et al. [11] used the support vector machine model to perform point cloud semantic segmentation with the extracted 13-dimensional features. Thomas et al. [12] extracted multi-scale point cloud features by setting different neighborhood sizes and used random forest classifiers to achieve point cloud segmentation, and demonstrated that the multi-scale spherical neighborhood of 3D point clouds can significantly improve the precision of 3D scene semantic segmentation.

With the development of deep learning, there are an increasing number of attempts being made to apply deep learning to point cloud semantic segmentation. According to the different point cloud conversion forms, we divided these approaches into three groups. (1) Multiview-based approaches: Inspired by 2D images, Su et al. [13] proposed an image-based multiview convolutional neural network (MVCNN), which projected 3D point clouds into 2D images under multiple perspectives, and then used 2D convolution to process the classification and segmentation task. Boulch et al. [14] proposed the SnapNet network, which uses a 2D neural network to process pairs of RGB images and depth images to construct 3D scene images. (2) Voxelization-based approaches: Wu et al. [15] converted the 3D point cloud into voxel representations and extended 2D convolution to 3D convolution to handle the classification and segmentation tasks. Graham [16] designed a sparse convolution network and applied it to the 3D point cloud segmentation task. Tchapmi et al. [17] subdivided the large-scale point cloud into voxel grids and used trilinear interpolation and conditional random field for post-processing to obtain the semantic segmentation results. Liu et al. [18] developed a 3D object classification system using a broad learning system (BLS) with a feature extractor called VB-Net. (3) Point-based approaches: Qi et al. [5] pioneered the PointNet, a deep learning network that directly processes unstructured point cloud data to obtain classification and segmentation results. However, this work failed to capture the local features of point clouds. To solve this drawback, Qi et al. proposed an upgraded version, PointNet++ [6], by improving the original network framework. PointNet++ added a local area division module based on PointNet, and recycled PointNet to extract local features. Li et al. [19] proposed a convolutional network PointCNN suitable for point cloud models, which used X-conv operator to extract permutation invariant features, input them into a 2D convolutional network, and finally obtained classification results through full connection and a classifier. Landrieu and Simonovsky [20] divided the input point cloud into geometrically simple shapes called super points, and constructed a graph neural network called super-point graph (SPG) to deal with the large-scale segmentation problem. Liu et al. [21] proposed a simple yet effective Point Context Encoding (PointCE) module to capture semantic contexts features of a point cloud. Their PointCE module can be integrated into any point cloud.
segmentation network to improve its segmentation precision with only marginal extra overheads.

The above work has made positive contributions to point cloud classification and segmentation based on deep learning, inspired more scholars to invest time in this field, and achieved good classification and segmentation precision [19], [22]. However, the network structures of these methods are generally more complicated, and the basic feature information of the point cloud is not fully utilized, so a longer training time is required. This paper intends to establish an efficient large-scale 3D point cloud semantic segmentation network—OFDV-Net—based on the optimal feature description vector. The network generates the optimal feature description vector, so that the neural network has advanced semantic information before training, which can reduce the training time of the model and increase the efficiency of point cloud semantic segmentation.

III. METHODOLOGY
In this section, we mainly describe the details of our proposed OFDV-Net framework. First, we summarize the architecture of our framework; then the IFDV preprocessing unit and feature selection unit are introduced. Finally, a 6-layer fully connected neural network is employed to perform point cloud semantic segmentation.

A. OVERVIEW OF THE PROPOSED FRAMEWORK
Our proposed framework includes three stages: IFDV preprocessing unit, feature selection unit, and semantic segmentation unit. As shown in Figure 1, we first divide the raw input point cloud \( P_r \) into a sparse point cloud \( P_s \) and construct a multi-scale spherical neighborhood to extract the neighborhood features \( F_n \). We concatenate \( F_n \) and raw point features as the initial feature description vector (IFDV). Next, we design a two-step feature selection unit to convert IFDV to the optimal feature description vector (OFDV). Lastly, a 6-layer fully connected neural network is constructed to train the input OFDV and perform high-precision large-scale point cloud semantic segmentation.

The reason for designing the IFDV preprocess unit and feature selection unit is the limitation of point-based deep learning methods. The feature vectors of point-based framework, e.g., PointNet and PointWise CNN are composed of XYZ coordinates, RGB color, and XYZ normalized coordinates. They didn’t make full use of curvature, roughness and other commonly features used in point cloud processing. If some semantic information with advanced prior knowledge can be extracted prior to neural network training, a better segmentation result could be obtained. Thus, we design the IFDV preprocess unit to extract the neighborhood feature using different neighborhood sizes, and optimize IFDV to optimal feature description vector (OFDV) through the feature selection unit. The OFDV encapsulates the best 3D features set of the points and can be used as the input of the deep neural network for training and testing.

B. IFDV PREPROCESS UNIT
1) RANDOM SAMPLING OF POINT CLOUD
Considering the huge amount of data in outdoor large-scale 3D point cloud, the raw point cloud is sampled first. Hu et al. [23] analyzed the time and memory consumption of the existing point sampling approaches in outdoor large-scale point clouds and demonstrated that random sampling is better than others in terms of time and memory consumption. Inspired by this idea, we selected the random sampling approach to process the raw point cloud. The comparison results of the point cloud data volume before and after sampling are shown in Figure 2. The large-scale point cloud is significantly subsampled by three orders of magnitude, but basic features can still be retained.

2) MULTI-SCALE FEATURE CALCULATION
Multi-scale point cloud feature calculation is mainly divided into two steps: Neighborhood selection and feature extraction. Neighborhood selection can be achieved by limiting the number of neighboring points (KNN) or restricting the range of neighborhoods (spherical neighborhood).
The KNN neighborhood and spherical neighborhood construction forms under different point cloud densities are shown in Figure 3. Compared with the KNN neighborhood, the spherical neighborhood is relatively less affected by the point cloud density. Thus, we adopted the spherical neighborhood for neighborhood building. Let $C \subset \mathbb{R}$ be a point cloud, the spherical neighborhood of point $p_0, C$ is defined by:

$$\vartheta_r(p_0, C) = \{p \in C \mid ||P - P_0|| \leq r\}$$

where $r \subset \mathbb{R}$ is the radius of the spherical neighborhood.

During feature extraction, if the search value of the spherical neighborhood radius $r$ is too small, the number of contained points in the neighborhood is too small, so the calculated features are not discriminative. If the $r$ value is too large, neighborhood calculation efficiency will be particularly low, and the calculated features lose their clear descriptive meaning. Considering this, we selected two surface features and six covariance matrix features with $r$ values of 0.05 m, 0.1 m, 0.25 m, 0.5 m, and 1 m. The surface features mainly include roughness (R) and gaussian curvature (Gau). Roughness represents the ratio of the surface area to its projected area projection in a certain area. Gaussian curvature is defined by the ratio of the first basic invariant $CN - M^2$ of the surface to the second basic invariant $EG - F^2$ of the surface, reflecting the degree of curvature of the surface.

The covariance matrix can be used as the structure tensor of the 3D point cloud. It can reflect the spatial distribution characteristics of the point set and effectively distinguish the urban features [23]. Let $P \subset \mathbb{R}$ be a point cloud, the covariance matrix of $P$ is defined by:

$$\text{cov}(N) = \frac{1}{|N|} \sum_{p \in N} (P - \overline{P})(P - \overline{P})^T$$

where $\overline{P}$ is the centroid of the neighborhood, $N$. By calculating the covariance matrix of the point cloud in the neighborhood, the object eigenvalues $\lambda_1, \lambda_2, \lambda_3 \in \mathbb{R}$ ($\lambda_1 > \lambda_2 > \lambda_3$) and the corresponding eigenvectors $e_1, e_2, e_3 \in \mathbb{R}$ can be obtained. From the eigenvalues, we can compute six covariance matrix features: omnivariance (O), planarity (P), linearity (L), surface variation (Sur), sphericity (Sp), and verticality (Ve). The detailed calculation formulas are shown in Table 1. We show the example visualizations of the 8-features calculation results when the radius $r$ of the spherical neighborhood is 1 m in Figure 4.

### C. FEATURES SELECTION UNIT

In this section, we introduce the feature selection unit, which performs two-step screening on the IFDV obtained by the IFDV preprocessing unit to obtain the OFDV. The specific algorithm information flow is shown in Figure 5. We first...
used the XGBoost algorithm to rank the feature importance of IFDV, and removed the features with an F-Score lower than N to obtain the IFDV subset T2. Then, we used the sequential backward selection (SBS) algorithm to iteratively fit the random forest classifier which had adjusted the hyperparameters by IFDV. We removed the last feature from the rest of feature set each time. When the evaluation function value reaches the best value, the feature set at this time is the optimal feature description vector (OFDV).

1) XGBOOST IMPORTANCE RANKING INITIAL SCREENING

XGBoost [25] (Extreme Gradient Boosting) algorithm is a boosting algorithm for classifying regression tree models which is coming from the gradient lifting decision tree. Comparing with the gradient lifting decision tree that only use first-order derivative information in the optimization process, XGBoost performs a second-order Taylor expansion on the cost function and uses the first-order derivative and the second-order derivative at the same time, so that XGBoost has good results. The XGBoost algorithm mainly has the following characteristics: (1) it has good anti-overfitting characteristics; (2) it has high calculation efficiency. In addition, the XGBoost algorithm can count the F-score importance of each feature variable, and a feature importance ranking is obtained based on this. The F-score of the feature i is defined by:

\[ F_i = \frac{1}{N} \sum_{j=1}^{l} \left( \bar{x}_i^j - \bar{x}_i \right)^2 \]

where \( \bar{x} \) is the average value, \( j \) is the category, and \( k \) is the sample.

In the feature screening unit, we first used the XGBoost algorithm to rank the importance of IFDV, and removed the features with an F-Score lower than 500 to obtain the IFDV subset T2, as shown in Figure 6.

2) SEQUENCE BACKWARD SELECTION ITERATIVE FITTING RANDOM FOREST

First, we took the IFDV as input, used the random forest classifier of the scikit-learn library to perform point cloud segmentation, and performed cross-validation with the following five hyper-parameters: Number of parameter base classifiers (n_estimators), maximum depth (max_depth), minimum number of samples for internal node division (min_samples_split), minimum number of leaf nodes (min_samples_leaf), and maximum number of features (max_features). The hyperparameter adjustment results are shown in Table 2. The hyperparameters were used for subsequent random forest model training.

The sequential backward selection (SBS) algorithm involves the removal one feature from the full feature set each time, so that after removing the feature, the evaluation function value reaches the optimal value [26]. In order to ensure reliability and stability of the model, this paper used 11 stations of the 15 station training data published by the Semantic3D dataset [27] as the training dataset, and the remaining 4 stations as the validation dataset. We set 3 stations of the 4 station training data published by NPM3D dataset [28] as the training dataset, and used the remaining station as the validation dataset and the specific distribution of each dataset is shown in Table 3. According to the feature importance ranking based on XGBoost, we used the SBS to iteratively fit the random forest classifier which had adjusted the hyperparameters, and finally obtained the OFDV, as shown in Figure 7.
FIGURE 7. Sequential backward selection-random forest (SBS-RF) feature selection, the input IFDV Subset T2 was selecting by SBS-RF to obtain OFDV.

FIGURE 8. Semantic segmentation unit neural network structure. OFDV is the input layer, the numbers under each fully connected layer represent the number of neurons, N represents the number of points in the input layer, T represents the dimension of the input point cloud, C represents the number of final classification categories, C is 8 in the Semantic3D dataset, and C is 9 in the NPM3D dataset. The learning rate is 0.005 and epoch is 250.

D. SEMANTIC SEGMENTATION UNIT

We took the OFDV obtained after feature selection unit as the input and constructed a 6-layer fully connected layer neural network to perform point cloud semantic segmentation, as shown in Figure 8. The loss function of the neural network model adopts the cross-entropy loss function, and is defined by:

$$\text{loss} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{K} y^{(i)}_k \log(p^{(i)}_k)$$  \hspace{1cm} (4)

where $N$ is the total number of training samples, and $K$ is the number of categories to be classified. If the category of the training sample $i$ is $k$, $y^{(i)}_k = 1$, otherwise $y^{(i)}_k = 0$. $p^{(i)}_k \in [0, 1]$ is the probability that training sample $i$ is category $k$.

To evaluate the results, we used the evaluation indicators given by Semantic3D and NPM3D official websites, namely, average intersection over union (AIOU) and overall accuracy (OA). The metrics IoU and OA are computed by:

$$\text{IoU}_i = \frac{c_{ij}}{c_{ij} + \sum_{j \neq i} c_{ij} + \sum_{k \neq j} c_{kj}}$$  \hspace{1cm} (5)

and

$$A_{IoU} = \frac{\sum_{i=1}^{L} \text{IoU}_i}{L}$$  \hspace{1cm} (6)

$$OA = \frac{\sum_{i=1}^{L} \sum_{j=1}^{L} c_{ij}}{\sum_{j=1}^{L} \sum_{k=1}^{L} c_{jk}}$$  \hspace{1cm} (7)

where $c$ is an $L \times L$ confusion matrix of the semantic segmentation method and $c_{ij}$ is the number of samples from the ground-truth class $i$ that are predicted as class $j$.

IV. MATERIALS AND EXPERIMENTS

A. DATASETS AND CONFIGURATION

To verify the effectiveness and robustness of the OFDV-Net proposed in this paper, as shown in Table 4, we used two large-scale standard public point cloud datasets (Semantic3D [27] and NPM3D [28]) to evaluate the proposed approach. The ground-truth visualizations of the files are shown in Figure 9. All experiments were implemented with Python 3.6, TensorFlow platform, and the hardware environment was an Intel Xeon W-2145@ 3.70GHz, 64 G RAM, GPU NVIDIA TITAN RTX 24G workstation.

The Semantic3D benchmark dataset is composed of dense point clouds obtained by a static ground laser scanner. We used 11 of the 15 stations with ground-truth training data as the training dataset, and 4 stations as the validation dataset to adjust the parameters of our model; then, we tested the online test data of 4 stations without ground-truth to obtain the predicted labels, and uploaded the labels to the Semantic3D website to get the segmentation precision. The Semantic3D dataset consists of 8 types of features, and the distribution of each type is shown in Table 5.

The NPM3D benchmark dataset is generated by a mobile laser system that accurately scans two different cities in France (Paris and Lille). We used 3 of the 4 stations with ground-truth as the training data, and 1 station as the validation dataset to adjust the parameters of our proposed model. We then tested the online test data of 3 stations without ground-truth to obtain the predicted labels and uploaded the labels to the NPM3D website to obtain the segmentation precision. The NPM3D dataset consists of 9 types of features, and the specific distribution of each type is shown in Table 6.
FIGURE 9. Example visualizations for ground-truth on both datasets: (a) Semantic3D dataset, (b) NPM3D dataset, and different colors represent different features.

TABLE 5. Semantic3D dataset category distribution table. The training and validation datasets are a 1% random sample.

| Class  | Training (1%) | Validation (1%) | Test   |
|--------|---------------|-----------------|--------|
| road   | 295,686       | 90,957          | 14,317,309 |
| grass  | 196,941       | 40,378          | 10,694,746  |
| tree   | 228,035       | 66,468          | 4,675,871  |
| bush   | 44,271        | 32,515          | 3,021,894  |
| buildings | 544,737   | 234,812         | 30,911,206  |
| hardscape | 49,961    | 17,542          | 2,384,545  |
| artefacts | 5,263     | 6,053           | 233,839    |
| cars   | 20,888        | 3,144           | 851,944    |
| total  | 1,385,782     | 491,869         | 67,091,584 |

B. EXPERIMENTS AND ANALYSIS

First, we constructed a multi-scale spherical neighborhood for the Semantic3D dataset and the NPM3D dataset, calculated 8 features under 5 scale radii, and added the basic coordinate information X, Y, Z, the color information R, G, B, and the intensity information. A total of 47-dimensional features constituted the IFDV. Then we sorted IFDV based on the XGBoost importance, and the feature importance score is shown in Figure 10.

It can be seen from Figure 10 that the F score of surface variation under 0.25 m (515), and nearly 19 times lower than the linearity score under 1 m (1971), indicating the importance of different features at different scales has significant difference. According to the F-scores in Figure 10, we eliminated features with F-scores below 500 to construct the IFDV subset T2, and then sequential backward selection-random forest (SBS-RF) was used to fine-screen T2. The relationship between the correct rate and the number of features in the feature set is shown in Figure 11. It can be seen that in the Semantic3D dataset, when the number of features reaches 22, the precision at this time is the highest (83.80%) and when the number of features reaches 17 in the NPM3D dataset,
the precision is the highest (80.21%). This feature subset is carried forward as the OFDV.

The specific feature information contained in IFDV, T2 and OFDV was analyzed and compared, among which IFDV was the raw feature set containing 8 features under 5 scale radii. The feature types and scale information in T2 and OFDV are shown in Table 6.

For the Semantic3D dataset, it can be seen that in the T2 initially screened by the XGBoost classifier in the first step, all features with a spherical radius of 1 m were retained, and the verticality at each scale was also retained. In the OFDV, the verticality at each scale was retained, and only the surface variation under the 1 m scale radius was eliminated. It can be concluded that the verticality feature has a high degree of importance in this dataset. Therefore, the importance of verticality was enhanced in OFDV, that is, the value of verticality is multiplied by two. For the NPM3D dataset, in the T2 initially screened by the XGBoost classifier in the first step, all the features under the 0.05 m scale radius were eliminated. The main reason for this is that the NPM3D dataset is sparser than the Semantic3D point cloud. Therefore, most features under the 0.05 m scale radius cannot be calculated. Compared with other features, the linearity under the four scale radii are all retained, so the importance of linearity is enhanced in the OFDV, that is, the value of linearity is multiplied by two.

To compare the performance of the models trained on different feature subsets, the IFDVs, T2, and OFDV were used as the input of the neural network which is constructed in 3.C section. After this, the trained model was tested on the test dataset. Finally, the test dataset labels were uploaded to the official websites of the two benchmark datasets to get the final results. The results of the Semantic3D dataset are shown in Table 8. The visualization of the segmentation results of OFDV-Net in the Semantic3D dataset is shown in Figure 12.

According to Table 8, after going through the feature screening module, the training accuracy is significantly improved, and the final OFDV accuracy is the highest. The main reasons for this are as follows: (1) The optimal features obtained after feature selection can better characterize real scene compared to the initial features. In Figure 5, we can see that not all features have the recognition ability. Furthermore, the optimal features obtained through feature selection have been proven to have strong characterization capabilities in the feature selection unit. (2) If some semantic information with advanced prior knowledge can be extracted prior to neural network training, a better segmentation result could be obtained. It is inevitable that optimal features can achieve higher segmentation results due to its better prior knowledge.

Comparing the metrics of different features, it can be seen that the method in this paper has better semantic segmentation effects on three types of features: “man-made terrain”, “high vegetation”, and “building”. The segmentation accuracy for these three feature types was above 94%, and the segmentation accuracy of high vegetation reached 97.74%. The main reasons for this are as follows: (1) The elevation information and intensity information have high recognition ability for these three types of features. Furthermore, the multi-scale and multi-features calculated by using multi-scale

### Table 7. The feature information contained in IFDV Subset T2 and OFDV, where ✓ represents the feature retention on the Semantic3D dataset, and □ ✓ represents the feature retention on the NPM3D dataset. The r values of the spherical neighborhood were set to 0.05 m, 0.1 m, 0.25 m, 0.5 m, and 1 m.

| Feature       | IFDV Subset (T2) | OFDV |
|---------------|------------------|------|
|               | 0.05 | 0.1  | 0.25 | 0.5  | 1    | 0.05 | 0.1  | 0.25 | 0.5  | 1    |
| Roughness     | □ ✓  | □ ✓  | ✓    | ✓    |      |      |      |      |      |      |
| Gaussian Curvature | □ ✓  | □ ✓  | ✓    |      |      |      |      |      |      |      |
| Invariance    | □ ✓  | □ ✓  | ✓    |      |      |      |      |      |      |      |
| Planarity     |      |      |      |      |      |      |      |      |      |      |
| Linearity     |      |      |      |      |      |      |      |      |      |      |
| Surface variation | □ ✓  | □ ✓  | □ ✓  |      |      |      |      |      |      |      |
| Sphericity    |      |      |      |      |      |      |      |      |      |      |
| Verticity     |      |      |      |      |      |      |      |      |      |      |
| Total dim     | 21/21 |      |      |      |      |      | 15/13 |      |      |      |

### Table 8. Intersection over union metric of different input subsets for the different classes of the Semantic3D dataset (%).

| Class            | Accuracy | IOU  |
|------------------|----------|------|
|                  | IFDV     | IFDVSub | OFDV | IFDV     | IFDVSub | OFDV |
| Man-made terrain | 92.78    | 96.69  | 96.69 | 82.7    | 85.5    | 86.5 |
| Natural terrain  | 80.84    | 83.95  | 85.20 | 75.4    | 73.3    | 75.4 |
| High vegetation  | 91.70    | 83.35  | 97.74 | 66.6    | 65.9    | 76.1 |
| Low vegetation   | 50.52    | 47.72  | 46.37 | 39.5    | 37.7    | 40.3 |
| Building         | 95.06    | 94.69  | 94.68 | 88.3    | 88.5    | 89.2 |
| Hard scape       | 13.90    | 10.16  | 18.24 | 11.1    | 7.7     | 14.0 |
| Scanning artifacts | 42.65  | 30.90  | 30.90 | 30.4    | 27.9    | 27.9 |
| Cars             | 67.41    | 66.70  | 66.68 | 48.7    | 46.8    | 49.2 |
| Average          | 86.65    | 86.92  | 88.34 | 55.0    | 54.2    | 57.3 |
spherical neighborhoods further enhanced the model’s ability to recognize these three types of features. (2) The number of these three types of features is relatively large compared to other types of features. Thus, in the process of neural network model training, more parameters can be obtained to distinguish these three types of features. In addition, the misclassification of the same number of point clouds has little effect on the accuracy of these three types of features. The method in this paper has a poor classification effect for “artefacts”, mainly because the training samples of artefacts were too small, so the difference between them and other features could not be learned well. However, although “low vegetation” and “hard scape” have a sufficient number of training samples compared to artefacts, the method in this paper also has a poor segmentation effect for these two types of features, mainly due to their high similarity in the multi-scale and multi-features of the two types of features with the “natural terrain” and “building” types. Low vegetation and hardscape were often mistakenly divided into natural terrain and building, which reduced the classification’s precision.

To verify the efficiency of the method proposed in this paper, the time required to complete the training of the three feature sets was counted. As shown in Table 9, the training time of IFDV was about 80 minutes, and the OFDV obtained by feature screening can complete the training in about 50 minutes, which is 37% more efficient than IFDV.

We systematically evaluate the overall efficiency of our OFDV-Net on real-world large-scale point clouds for semantic segmentation. Particularly, we evaluate OFDV-Net on Semantic3D-reduced dataset, obtaining the total time consumption of our framework. We also evaluate the time consumption of recent representative works on the same dataset. Note that, all experiments are conducted on the same machine with an Intel Xeon W-2145@ 3.70GHz CPU and an NVIDIA TITAN RTX 24G GPU. The calculated consumption is shown in Table 10.

Table 10 quantitatively shows the preprocessing time, training time and total time of different approaches. It can be seen that: (1) SPG has an expensive preprocess cost due to its geometrical partitioning and super-graph construction; (2) RandLA-Net takes the lowest time to process the point cloud, but it is still very time costing in training. (3) Our OFDV-Net has the relatively lowest training time (3097 seconds on the semantic3D-reduced dataset).

To compare the segmentation results of OFDV-Net with others on the Semantic3D dataset, the semantic segmentation results of this IFDV, the IFDV Sub-Net, and the OFDV are marked as IFDV-Net, IFDVSub-Net, and OFDV-Net respectively. The comparison results are shown in Table 11 and show that the method in this paper has the highest overall segmentation accuracy and average intersection over union (IoU).

As shown in Table 12, the overall accuracy and average IoU of OFDV-Net on NPM3D dataset achieved 87.7% and 47.5%, respectively. The visualization results for the NPM3D dataset are shown in Figure 12.
TABLE 11. Intersection over union (IoU) metric of different methods for the different classes of the Semantic3D dataset. OA is the overall accuracy, while mIoU refers to the unweighted average of IoU of each class (%).

| Method            | OA mIoU | Man-made | Natural | High | Low | Buildings | Hard-scene | Scanning artefacts | Cars |
|-------------------|---------|----------|---------|------|-----|-----------|------------|-------------------|------|
| PointNet [5]      | 82.5    | 52.1     | 78.8    | 74.5 | 59.8| 60.8      | 81.7       | 33.2             | 15.4 | 12.7 |
| DeepVoxNet [29]   | 84.8    | 57.1     | 82.7    | 53.1 | 83.8| 28.7      | 89.9       | 23.6             | 29.8 | 65.0 |
| TMLC-MSR[30]      | 86.2    | 54.2     | 89.8    | 74.5 | 53.7| 26.8      | 88.8       | 18.9             | 36.4 | 44.7 |
| IFDV-Net          | 86.6    | 55.0     | 82.7    | 72.6 | 66.6| 39.5      | 88.3       | 11.1             | 30.4 | 48.7 |
| IFDVSub-Net       | 86.9    | 54.2     | 85.5    | 73  | 65.9| 37.7      | 88.5       | 7.7              | 27.9 | 46.8 |
| OFDV-Net          | **88.3**| **57.3** | 86.5    | **75.4**| 76.1| 40.3      | 89.2       | 14.0             | 27.9 | **49.2** |

TABLE 12. Intersection over union (IoU) metric for the different classes of the NPM3D dataset. OA is the overall accuracy, while mIoU refers to the unweighted average of IoU of each class (%).

| Method        | OA mIoU | Ground | Building | Pole | Bollard | Trash can | Barrier | Pedestrian | Car | Natural |
|---------------|---------|--------|----------|------|---------|-----------|---------|------------|-----|---------|
| OFDV-Net      | 87.7    | 47.5   | 95.2     | 62.1 | 35.6    | 75.4      | 4.0     | 14.9       | 12.3 | 82.1    | 45.7 |

FIGURE 13. Example semantic segmentation visualization on the NPM3D dataset. (a) ground truth of point clouds, (b) the prediction of semantic segmentation, and different colors represent different features.

V. CONCLUSION

In this paper, we proposed a semantic segmentation network OFDV-Net for a large-scale 3D point cloud. Our OFDV-Net consists of three important components. In the first unit, the initial feature description vector is constructed by setting different neighborhood sizes after random sampling, the IFDV encapsulates the multiscale neighbor features of each point. Then, in the second unit, the IFDV is optimized to the OFDV through a two-step feature selection method. The proposed method allows us to create the optimal features of each point. Finally, in the semantic segmentation unit, the OFDV is input into the fully connected neural network to obtain the semantic segmentation results. Experimental results from two public outdoor large-scale point cloud datasets (Semantic3D and NPM3D) show that our OFDV-Net can obtain better semantic segmentation results. The overall
accuracy for the Semantic3D and NPM3D datasets was 88.3% and 87.7%, and the average intersection over union was 57.3% and 47.5%, respectively. Compared with the existing deep learning networks, the overall accuracy was 2.1% higher, and the accuracy was 1.7% higher than that before feature screening. The training time was shortened by 37%.

However, there are also many problems that need to be solved in our network. Although the method in this paper can obtain high-precision segmentation results while reducing model training time, the IFDV preprocess unit computation is large due to the multiscale spherical neighborhood query, and it consumes more computation in the multi features computation. In the future, it is worthwhile to consider how to accelerate the preprocess time of our proposed network.

In addition, the convolutional layer is not well suited for deep feature extraction in our segmentation process unit, so optimizing the neural network structure to achieve better semantic segmentation results is a suitable direction of further research.

REFERENCES

[1] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, “Shapenet: An information-rich 3D model repository,” 2015, arXiv:1512.03012. [Online]. Available: https://arxiv.org/abs/1512.03012

[2] Z. Zhang, L. Zhang, X. Tong, P. T. Mathiopoulos, B. Guo, X. Huang, Z. Wang, and W. Yang, “A multilevel point-cluster-based discriminative feature for ALS point cloud classification,” IEEE Trans. Geosci. Remote Sens., vol. 54, no. 9, pp. 3390–3332, Jun. 2016.

[3] X. Roynard, J.-E. Deschaud, and F. Goulette, “Classification of point cloud scenes with multiscale voxel deep network,” 2018, arXiv:1804.03583. [Online]. Available: http://arxiv.org/abs/1804.03583

[4] J. Demanté, C. Mallet, N. David, and B. Vallet, “Dimensionality based scale selection in 3D lidar point clouds,” Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci., vol. XXXVIII-5/W2, pp. 97–102, Sep. 2012, doi: 10.5194/isprsarchives-XXXVIII-5-W2-97-2011.

[5] 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), Honolulu, HI, USA, Jul. 2017, pp. 77–85.

[6] 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., Long Beach, CA, USA, 2017, pp. 5099–5108.

[7] 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. III-3, pp. 177–184, Jun. 2016.

[8] 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, doi: 10.1016/j.isprsjprs.2015.01.2016.

[9] M. Weinmann, B. Jutzi, and C. Mallet, “Feature relevance assessment for the semantic interpretation of 3D point cloud data,” ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci., vol. II-5/W2, pp. 313–318, Oct. 2013.

[10] N. ChehataL, L. Guo, and C. Mallet, “Airborne Lidar feature selection for urban classification using random forests,” Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci., vol. 38, no. 1, pp. W8. 2009.

[11] M. Kragh, R. N. Jørgensen, and H. Pedersen, “Object detection and terrain classification in agricultural fields using 3D Lidar data,” in Proc. Comput. Vis. Syst. Cham, Switzerland: Springer, 2015, pp. 188–197.

[12] H. Thomas, F. Goulette, J.-E. Deschaud, B. Marcotegui, and Y. LeGall, “Semantic classification of 3D point clouds with multiscale spherical neighborhoods,” in Proc. Int. Conf. 3D Vis. (3DV), Sep. 2018, pp. 390–398.

[13] 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.

[14] A. Boulch, B. Le Saux, and N. Audebert, “Unstructured point cloud semantic labeling using deep segmentation networks,” in Proc. Eurogaphics Workshop 3D Object Retr., 2017, pp. 1254–1268.

[15] 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.

[16] B. Graham, “Spatially-sparse convolutional neural networks,” 2014, arXiv:1409.6070. [Online]. Available: http://arxiv.org/abs/1409.6070

[17] L. Tchapmi, C. Choy, I. Armeni, J. Gwak, and S. Savarese, “SEGCloud: Semantic segmentation of 3D point clouds,” in Proc. Int. Conf. 3D Vis. (3DV), Qingdao, China, Oct. 2017, pp. 537–547.

[18] Z. Liu, W. Song, Y. Tian, S. J. Yi, S. Wang, L. Wen, T. Zhang, L. Song, and A. Gozho, “VB-net: Voxel-based broad learning network for 3D object classification,” Appl. Sci., vol. 10, no. 19, pp. 6735, Sep. 2020.

[19] Y. Li, R. Bu, M. Sun, W. Xu, D. Bi, and C. Chen, “PointCNN: Convolution on X-transformed points,” in Proc. Adv. Neural Inf. Process. Syst., Montreal, CA, USA, 2018, pp. 820–830.

[20] 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.

[21] H. Liu, Y. Guo, Y. Ma, Y. Lei, and G. Wen, “Semantic context encoding for accurate 3D point cloud segmentation,” IEEE Trans. Multimedia, early access, Jul. 8, 2020, doi: 10.1109/TMM.2020.3007313.

[22] W. Wu, Z. Qi, and L. Fuxin, “PointConv: Deep convolutional networks on 3D point clouds,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 9621–9630.

[23] 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,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 11108–11117.

[24] I. Lee and T. Schenk, “Perceptual organization of 3D surface points,” Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., vol. 34, no. 3A, pp. 193–198, 2002.

[25] T. Chen and C. Guestrin, “XGB-Boost: A scalable tree boosting system,” in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Aug. 2016, pp. 785–794.

[26] F. Ruan, J. Qi, C. Yan, H. Tang, T. Zhang, and H. Li, “Quantitative detection of harmful elements in alloy steel by LIBS technique and sequential backward selection-random forest (SBS-RF),” J. Anal. At. Spectrometry, vol. 32, no. 11, pp. 2194–2199, 2017.

[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] 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.

[29] X. Roynard, J.-E. Deschaud, and F. Goulette, “Classification of point cloud scenes with multiscale voxel deep network,” 2018, arXiv:1804.03583. [Online]. Available: http://arxiv.org/abs/1804.03583.

[30] 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. III-3, pp. 177–184, Jun. 2016.
QUAN SUN received the B.S. degree from the School of Water Conservancy Science and Engineering, Zhengzhou University, Zhengzhou, China, where he is currently pursuing the master’s degree. His current research interests include 3D-data analysis, remote sensing, and deep learning.

KERU CHEN received the B.S. degree from the School of Water Conservancy Science and Engineering, Zhengzhou University, Zhengzhou, China, where she is currently pursuing the master’s degree. Her current research interests include remote sensing, image processing, and machine learning.

HAO CUI was born in China, in 1992. He received the B.S. degree from the School of Information Engineering, Northeastern University, in 2013, the M.S. degree from the Mining Engineering Research Institute, Beijing General Research Institute of Mining and Metallurgy, in 2016, and the Ph.D. degree from the School of Remote Sensing and Information Engineering, Wuhan University, in 2019. He is currently doing Postdoctoral Research with Zhengzhou University. His research interests include intelligent processing of point cloud and its application in highway and railway inspection and precision detection.

KUAN HUANGFU received the B.S. degree from the School of Water Conservancy Science and Engineering, Zhengzhou University, Zhengzhou, China, where she is currently pursuing the master’s degree. Her current research interests include remote sensing, image processing, and machine learning.

XIAOLONG CHEN received the B.S. degree from the School of Water Conservancy and Hydropower Engineering, China Three Gorges University, in 2018. He is currently pursuing the master’s degree with the School of Water Conservancy Science and Engineering, Zhengzhou University, Zhengzhou, China. His current research interests include 3D laser scanning technique, digital image processing, and damage detection of concrete.