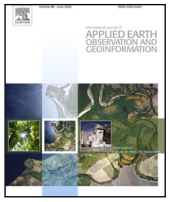
International Journal of Applied Earth Observations and Geoinformation 113 (2022) 102983

Contents lists available at ScienceDirect

International Journal of Applied Earth Observation and Geoinformation

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



Spherical coordinate transformation-embedded deep network for primitive instance segmentation of point clouds

Wei Li [a,b,c], Sijing Xie [a,b], Weidong Min [a,b,*], Yifei Jiang [a,b], Cheng Wang [c], Jonathan Li [d]

[a] School of Software, Nanchang University, 235 Nanjing Road East, Nanchang, JX 330047, China
[b] Jiangxi Key Laboratory of Smart City, Nanchang University, Nanchang, China
[c] Fujian Key Laboratory of Sensing and Computing for Smart Cities, School of Information Science and Engineering, Xiamen University, 422 Siming Road South, Xiamen, FJ 361005, PR China
[d] Departments of Geography and Environmental Management and Systems Design Engineering, University of Waterloo, 200 University Avenue West, Waterloo, ON N2L 3G1, Canada

ARTICLE INFO

MSC:
00-01
99-00

Keywords:
Primitive instance segmentation
Spherical coordinate transformation
Relation matrix

ABSTRACT

In this research, a primitive prediction network embedding Spherical Coordinate Transformation (named SCT-Net), which is a simple and end-to-end deep neural network, is proposed for primitive instance segmentation of point clouds. The key point of SCT-Net is to excavate the relationship between local neighborhood points. First, in order to enhance the compacted expression of local feature, a spherical coordinate transformation is embedded to a deep network. Second, the embedded network is constructed to predict the point grouping proposals and classify the primitives corresponding to each proposal, which can segment primitive instance directly. Third, the feature relationship between each two points is revealed by the constructed relation matrix. The designed loss function not only encourages the embedded network to describe local surface properties, but also produces a grouping strategy accurately for each point. Experiments show that the proposed SCT-Net achieves the state-of-the-art performance than representative methods. At the same time, the capability of spherical coordinate transformation has been demonstrated to improve primitive instance segmentation.

1. Introduction

With the rapid development of 3D scanning technology, the acquisition of point clouds, which records the spatial information of the scene or object surface, is becoming more and more convenient. However, these point clouds lack topological relationships and tend to require larger storage capacity, which increases challenges for applications in real environments. The triangulation of point cloud enables the discrete points to obtain the topological relationship between the neighborhoods (Lafarge and Alliez, 2013; Holzmann et al., 2018), but the mesh decimation iteratively folds edges that cannot preserve the important structure. Primitive assembly, which requires primitive instances based on the point segmentation, is therefore limited by the quality of point cloud segmentation. Thus, our goal is to improve the quality of point cloud segmentation and generate a lightweight primitive model.

Indeed, a lot of works have been proposed for the primitive instance segmentation. There are two representative solutions: RANSAC and region growing. For RANSAC (Derpanis, 2010), many algorithms of computing inlier points have varying degrees of sensitivity to density, noise, and occlusion. How to find the appropriate parameter remains a big challenge. In addition, region growing (David and Gabor, 2001) is a kind of non-global methods for primitive instance extraction. The main challenge is how to fit appropriate parameters so as to preserve boundaries of similar primitives robustly. Both two methods are highly dependent on the choice of parameters. Recently, a number of methods based on deep learning have emerged. For example, Li et al. (2019) introduced a Supervised Primitive Fitting Network (SPFN) that can predict primitives at different varying scales automatically. However, this method assumes that several primitives are known and are sensitive to the incomplete data obtained by scanning in the real environment. Apart from that, Sharma et al. (2020) presented a novel decomposition strategy that considers primitive patches as the parametric fitting of simple geometric patches. Nevertheless, the method's decomposition module still relies on the given surface patch primitives. Besides, Huang et al. (2021) proposed an adversarial network (PrimitiveNet), decomposed the global segmentation problem into local tasks, and fitted them with geometric primitives. It is noteworthy that they still need the help of RANSAC or region growing, and the designed deep neural network greatly increases the complexity of such methods.

Primitive extraction is aimed at extracting high-quality primitive instances and clears boundaries from point clouds corrupted by noise

* Corresponding author.
E-mail address: minweidong@ncu.edu.cn (W. Min).

https://doi.org/10.1016/j.jag.2022.102983
Received 23 April 2022; Received in revised form 31 July 2022; Accepted 13 August 2022
Available online 5 September 2022

and outliers. In order to distinguish different primitive instances robustly, we propose an end-to-end deep neural network that reinforces the extraction of point-to-point relationships based on spherical coordinate transformation, and derives a smooth border at the junction of primitives. Although the problem of primitive segmentation is generally considered as the instance segmentation, it is different from the traditional 3D instance segmentation. Primitive does not contain sharp features that make the relationship of points more important. In general, a cloud saves with the artesian coordinate system, the relationship between points such as distance is calculated, and the square root or trigonometric functions is unavoidable. Floating-point arithmetic tends to generate error and high time complexity. Notably, spherical coordinate is a good way to solve this problem, because a point and its neighbor always have a similar elevation angle or azimuth. It is convenient and efficient for us to introduce spherical coordinate transformation, and point deep learning methods such as PointNet and PointNet++. Thus, the feature similarity of two points reaches a specific value, indicating that the selected two points belong to a same primitive. Beyond that, the distance of pairwise points is measured to construct a relation matrix making an initial proposal of each primitive instance. Combining with a learned confidence map and primitive estimation, we can finally get an accurate group.

Experiments show the convenience and effectiveness of spherical coordinate transformation-based method. Besides, SCT-Net achieved better performance on the ABC dataset than the existing representative methods. Notably, the time consumption is also lower than the state-of-the-art methods. Ablation studies show that spherical coordinate transformation significantly improves the standard point cloud feature extraction network.

2. Related work

2.1. Fitting-based methods

Primitive fitting is to sample points and fit them with basic primitives such as planes and cylinders. The representative methods are reviewed as follows. Fischler and Bolles (1981) presented a classic algorithm namely Random Sampling Consensus (RANSAC) algorithm that iteratively operates between randomized sampling and estimates the fitting parameters from a given data containing outliers, which has been applied for computer vision and image processing. Variants of RANSAC (Chum and Matas, 2005; Derpanis, 2010) are efficient in outlier detection and its selection strategy decides the precision of results. For dense point clouds, Schnabel et al. (2007) proposed a more robust algorithm to detect different types of primitives. Li et al. (2011), who extended Schnabel's method, introduced subsequent optimizations to extract primitives according to their relationships. As an extension of the RANSAC-based approach, Wu et al. (2018) and Du et al. (2018) presented a method to reverse a mesh or point cloud from which a solid geometry was constructed. Although the experiments of these RANSAC variants have improved the accuracy significantly in the corresponding respective fields, it often relays on unstable and laboratory parameters adjusted for different types of objects. Furthermore, these methods usually require point normals as input, which are not directly available from 3D scans. Some of the learning-based algorithms have been constructed for fitting primitive. Fang et al. (2018) proposed a framework to detect planar shapes at structural scales. Then, they presented a hybrid approach that connects and slices planes for reconstructing 3D objects (Fang and Lafarge, 2020). However, these two methods have an ill-posed problem with no guarantee to adequately describe the observed objects. Lin et al. (2020) presented a fast regularity-constrained fitting method for planar segmentation of point cloud, but merely focused on plane fitting. Jiang et al. (2021) presented a Non-Watertight PolyFit (NW-PolyFit) algorithm to simplify polygonal modeling from incomplete data, yet NW-PolyFit still uses RANSAC to detect planar points.

2.2. Clustering-based methods

Region growing refers to a classic method that segments the point cloud into homogeneous regions according to the local indicators (Besl and Jain, 1988; David and Gabor, 2001; Rabbani et al., 2006a). After selection of the seed, an intelligent algorithm is designed to iteratively compute the similarity of between the category and each point, until all the points are classified into a certain class, the regional growth stops. Region growing, as a local-based method, is easy to implement. However, it suffers from the uncertainty issue of seed selection and the interference of noise. Besides, the selection of the neighborhood size, the pre-set merging rules and the initial seed selection are all crucial. A lot of methods still use the k-nearest neighbor or fixed neighbor algorithm (Rabbani et al., 2006b), but they are usually affected by the point density. There are two types of methods to improve the accuracy of the algorithm. The first method is to improve the ability of local feature description (Che and Olsen, 2018). For instance, Nurunnabi et al. (2016)proposed a robust method based on normal estimation to analyze adjacent points, which made the algorithm significantly improve the results of planar segmentation of cylinders. Furthermore, the second one is to reduce the algorithm's dependence on the threshold. For the other instance, Maalek et al. (2015) tried to identify outliers before cluster segmentation. In addition, local features can also be used as an indicator of growth, and Lin et al. (2017) made use of geometric model to reconstruct plane primitive that successfully fits the plane under outliers and noise. While these methods are usually not robust caused of noises or complex structures of the object's surface. Besides, Poullis (2019) proposed a tensor-based clustering algorithm to divide the tensor into basic 3D graphs with different thresholds elements, such as curves, surfaces, or intersections. At the same time, Chen et al. (2017) proposed a partial improvement by initial clustering so as to 100 to eliminate erroneous segmentation due to inappropriate neighborhood size and threshold choices. Additionally, Xu et al. (2019) presented a novel hierarchical method to cluster point clouds with the bipartite graph theory, which allowing plane primitives to retain desired parts after processing, such as rule-based merging. Nonetheless, the clustering method is still affected by the selection of seed points and noise interference.

2.3. Deep learning-based methods

Many supervised or unsupervised deep learning-based methods have been proposed for primitive data extraction. For example, Zou et al. (2017) presented a 3D Primitive Recurrent Deep Neural Network (3D-PRNN) that encodes symmetry characteristics on the common man-made objects, which significantly reduced parameter space. Beyond that Tulsiani et al. (2017) presented an unsupervised deep learning framework to generate simple geometric 3D volumetric primitives. Although the method can predict shape representation in a very simple setting, these two methods focus on man-made objects and are composed of several simple cuboids that are difficult to adapt to the fitting of real-world 3D scene data. Apart from that, Li et al. (2019) presented a Supervised Primitive Fitting Network (SPFN) that fits geometric parameters of various primitives robustly based on the primitive types. Furthermore, Yan et al. (2021) used hybrid feature representations to separate points of different primitives. Despite that the above two methods can accurately predict various primitives, a finite number of primitives and object completeness are needed. Sharma et al. (2020) presented a decomposition method that transforms the surface patches of 3D point cloud into the parametric fitting of geometric patches. However, the decomposition module by this method still relies on given surface patch primitives. Loizou et al. (2020) proposed a graph convolutional network framework to detect boundaries of parts in 3D objects, which is applied to detect probabilistic boundaries in the ABC dataset. However, the segmented results are sensitive to wrong predictions. Lê et al. (2021) presented a Cascaded Primitive Fitting

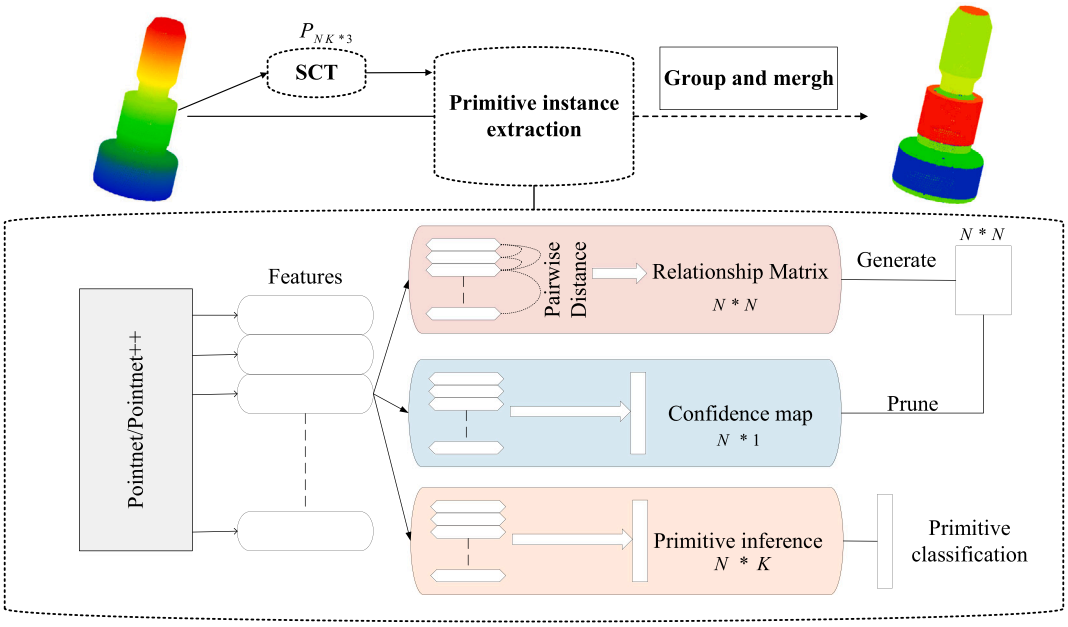


Fig. 1. Primitive Predict Network based on spherical coordinate transformation. We embed (a) spherical coordinate transformation into (b) 3D feature extraction backbone of PointNet++, where implicit and explicit features are supervised by the real labels. The training loss is designed based on the global point set and local planar primitives.

Networks (CPFN) using a network that can adaptively sample patches, but focused on human-made objects. Huang et al. (2021) presented a solution to extract primitive that significantly reduces the high computing resources. However, it still needs the help of region growing.

3. Approach

3.1. Overview

The primitive extraction is considered as instance segmentation assembled from the module of Wang et al. (2018) in our proposed SCT-Net. The input of this problem is formulated as the combining set $\mathcal{I} =< P, C, \mathcal{E} >$, where $P = \{p_i\}_{i=1}^{N}$ represents the point coordinates, N represents the number of points, $C = \{c_i\}_{i=1}^{N}$ represents the normal vector of a point, $\mathcal{E} = \{< p_i, p_j > | i = 1, \ldots, N, j = 1, \ldots, N\}$ represents that the point p_i and point p_j are neighborhood. We aim at predicting the output as $\mathcal{O} =< P', \mathcal{V}, \mathcal{L} >$, where $P' = \{p_k\}_{k=1}^{K}$ represents the point set without noise and outliers, K represents the corresponding number, $\mathcal{V} = \{v_i\}_{i=1}^{K}$ represents normal vector set and primitive instance labels $\mathcal{L} = \{l_k\}_{k=1}^{K}$. Here, the points belonging to the same primitive have the same label. To acquire the indicators of primitive instance segmentation, the relationship degree matrix is built as $R = \{r_{ij}\}_{N\times N}$. As shown in Fig. 1, instead of directly learning features, the input point clouds are transformed based on spherical coordinate transformation to distinguish the boundary points. The operation of primitive extraction is achieved based on a deep network module. The backbone of PointNet/PointNet++ is firstly used for learning point features. Then, the generated features are divided into three parts: constructing relationship matrix, generating confidence map and inferring primitive points. Here, a confidence map is learnt for pruning the final relationship matrix. After the extraction of instance primitives, the operation of grouping and merging are used for generating the final instance primitives. The loss function is constructed according to the point's features as well as the difference of point and planar primitives.

3.2. Spherical coordinate transformation

In order to improve the local planar description of the primitive instances, a method of spherical coordinate transformation has been embedded into primitive predicting network. The local point set R_i of a point $p_i = \{x_i, y_i, z_i\}$ is gathered at a given radius for an inputting point cloud P. Fig. 2 shows that a point p_i combining normal vector is transformed according to a spherical coordinate system. Here, we consider the point p_i as the center of a sphere, and r represents the searching radius. The point $p_j = \{x_j, y_j, z_j\}$ in the neighborhood of point p_i in the rectangular coordinate system can be formulated as

$$\begin{cases} x = r sin\varphi cos\theta \\ y = r sin\varphi sin\theta \\ z = r cos\theta \end{cases} \tag{1}$$

Therefore, the point $p_j = \{x_j, y_j, z_j\}$ in the neighborhood of point p_i in the spherical coordinate system can be formulated as

$$\begin{cases} r = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + (z_j - z_i)^2} \\ \varphi = arccos\frac{(z_j - z_i)}{r} \\ \theta = arctan\frac{(y_j - y_i)}{(x_j - x_i)} \end{cases} \tag{2}$$

The local patch around a selected point is gathered based on a spherical field and transformed by spherical coordinate transformation for expression. On the one hand, because of the local coordinate system of each scanning cloud, large differences have existed between two local scanning point clouds. On the other hand, the angles of elevation and azimuth in the spherical coordinate system are greatly affected by the attitude. Therefore, to alleviate the sensitive of posture changes, a transformation network is introduced to correct relative pose to some extent. Thus, with the transformation of the spherical coordinate system, the mutual difference of elevation angles between two points on the same plane is quite small, especially less than the points of a threshold θ on a same primitive. Besides, the cosine distances are less than a threshold γ in the same plane. Secondly, for primitive instance segmentation, it is important to distinguish that the boundary points are at the intersection of two primitives. The difficulty of boundary point identification mainly lies in that it belongs to two planes at the same time. Due to the fact that these boundary points belong to two plane primitives, the angle differences of the elevation and azimuth between the points of their K-nearest neighbors are larger, which is considered as key description of the boundary point. Besides, to detect primitive instance more accurately, the receptive field of each point is

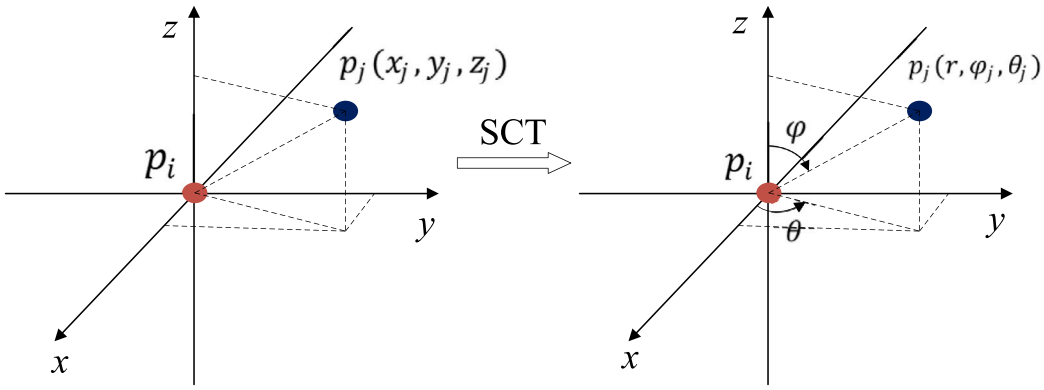


Fig. 2. A spherical transformed example that a selected point in the (a) Cartesian Coordinate System is transformed into the (b) Spherical Coordinate System.

set as small as possible, such that the points belong to the same surface. In the experiment test, the selection of the ball radius is 10% of the diagonal length of the corresponding point cloud 3Dbox.

3.3. Relation matrix

In order to indicate segment results more accurately, the corresponding transformation matrix is formulated as $Matrix_{Spherical} \in R^{N\times 3}$, where R represents a matrix of dimension $N \times N$, and the value of r_{ij} indicates whether a point pair p_i and p_j have the same label. Each row can be regarded as a set of points that are formed as a plane. The distance of two features corresponding to two points of the same plane should be smaller than a given threshold. For a corresponding points (p_i, p_j), the L_2-norm of corresponding features is used to measure the planar similarity of each pair of points, i.e., $d_{ij} = \|F_i - F_j\|_2$, which makes the distance between points on the same plane smaller. Conversely, the distance between points on different planes becomes larger in feature space. Compared with plane points, a smaller angle of elevation locates in the boundary points. From their own perspective, the difference in the elevation angles of the points around them is also larger. Such special points have very few connections with other points in the feature space, so that they are usually classified as one category. Therefore, there is no need to get an exact regression value of the feature, and only the points in the same plane need to be transformed and optimized to a feature space.

Here two cases are considered in each correspondence (p_i, p_j) as follows: (a) the point p_i and point p_j both belong to a same plane b_k; (b) the point p_i and point p_j belong to two different planes. Thus, the loss function is regarded as the measurement of pairwise features corresponding to two points, and is formulated as

$$L_R = \sum_{i}^{N}\sum_{j}^{N} l(i,j) \tag{3}$$

Where $l(i,j)$ is expressed as

$$l(i,j) = \begin{cases} \|f_i - f_j\|_2, & p_i, p_j \in b_k \\ max(0, k - \|f_i - f_j\|_2), & p_i \in b_k, p_j \notin b_k \end{cases} \tag{4}$$

Where b_k represents the kth plane, and $\{k\}_{k=1}^{M}$, M represents the number of extracted planes.

Since the difference between points is distinguished by two losses, the accuracy and convergence speed of the network can also be improved. Because in the smaller spherical coordinate system, the point-to-point not in the same plane is reduced to a great extent. Therefore, the constraint $0 < a < 1$ is added to control the gradient in the feature space.

The features obtained from the blackbone module of pointnet++ are used to predict an $N \times 1$ confidence matrix through MLP and fully connected layers. This confidence matrix reflects an evaluation of the model for this grouping as a candidate for face segmentation. Likewise, the confidence of points located at the edge of the face is low, it is regressed according to the ground truth. In terms of the relationship matrix in B, its size is $N * N$, for any point P_i, if it is an outlier or noise, then the ith row in the relationship matrix should be all 0, so in our In the confidence matrix of, we define the value of its ith row as the ratio of the predicted value which is relative to the ground-truth value. As for Loss, it should also be the L_2 loss of the predicted confidence map and ground truth. Although the matrix of this step is dependent heavily on the output of the relation matrix in B, we also run this branch in parallel and set a threshold to make its output more accurate.

3.4. Primitive inference

In this module, each primitive is considered independently. Here, a plane primitive is expressed as $\boldsymbol{\Phi} = (\mathbf{n}, \upsilon)$, where $\mathbf{n}$ represents the normal vector of the local primitive, and $\|\mathbf{n}\| = 1$. Inner product of the normal vector $\mathbf{n}$ and the point p_i on the local primitive can be denoted as $\mathbf{n}^T p_i = \upsilon$. Thus, the distance between a point p_i and a plane primitive $\boldsymbol{\Phi}$ can be formulated as $d_{p_i \to \boldsymbol{\Phi}} = (\mathbf{n}^T p_i - \upsilon)$. Combining with the relationship matrix R, the minimized cost of point p_i to plane primitive $\boldsymbol{\Phi}$ can be defined as

$$L_{local} = \sum_{i=1}^{N} R_i(\mathbf{n}^T p_i - \upsilon)^2 \tag{5}$$

The solution of the equation $\frac{\partial \boldsymbol{\Phi}}{\partial \upsilon} = 0$ is computed by lagrange multiplier method. Then, $\boldsymbol{\Phi}$ can be deduced as

$$\boldsymbol{\Phi}' = \|diag(R_i X \mathbf{n})\| \tag{6}$$

Where

$$X_i = p_i - \frac{\sum_{i=1}^{N} R_i p_i}{\sum_{i=1}^{N} R_i} \tag{7}$$

This formulate represents the relationship of point p_i and expected plane primitive. Similarly, when the normal vector $\|a\| = 1$, the corresponding solution is actually the right singular value vector, so that the gradient can be back-propagated by the SVD decomposition method. Thus, it is used as a point-wise classifier to get a preliminary primitive classification.

The relation matrix R generates N sets of proposals, many of which represent the same plane as noise. Hence, further trimming is required to obtain accurate, non-overlapping, or angles (less than 15°). By using Non-Maximum Suppression, take the point set with a larger IoU as the benchmark, then try to assign each point to this point set and merge them. In some special cases, a point belongs to multiple groups, which means that the point is at the intersection of the face and the face. This does not affect the final result because the points on the intersection can be its classification to any one plane. Therefore, we assign these points randomly and arbitrarily without affecting the accuracy as much as possible.

4. Experiments

In order to demonstrate the superiority of proposed SCT-Net in primitive instance segmentation, all of our experiments have been tested on a PC with Ubuntu 20.04.2, Intel(R) XCore(R)E5-2678 v3 CPU @3.30 GHz and 16.0 GB RAM and with a graphics card model of NVIDIA RTX2080 and 8G memory.

4.1. Dataset

The datasets used in the experimental analysis including CAD models which are provided by ABC dataset (Koch et al., 2019), and the indoor's dataset, which are from SUN3D dataset (Xiao et al., 2013), are applied to test the performance on the real-word scene. Since our method has focused on the detection of planar primitives, the network is trained on data that does not contain spherical surfaces. Second, to increase our sample size, we scale each model data around the center so that they all lie within a unit cube of different scales. To test the anti-noise performance, Gaussian noise is added to the data along with the point normal direction, and the noise range is between [-0.01, 0.01]. Therefore, the normal vector of each point receives uniformly distributed noise interference. In this way, it deviates from the original direction, and the deviation angle is within 3°. Afterwards, the dataset includes the training and testing dataset at the ratio of 3:1.

4.2. Evaluation metrics for algorithmic analysis

Here, the segmentation performance of the SCT-Net is tested by the following evaluation indicators:

(a) Segmentation Mean IoU (Seg mIoU)

The Seg mIoU measures the similarity from the results of predicted and ground-truth, the higher the similarity, the higher the value. For a given ground truth set W, the result set $\overline{W}$ can be predicted by the network, below:

$$SegmIoU = \frac{1}{N}\sum_{k=1}^{N} iou(\overline{W}[:,k], h(W[:,k])) \tag{8}$$

Where $\overline{W}[:,k]$ represents the kth column of matrix $\overline{W}$, h represents a one-hot encoding transformation, and N represents the number of ground-true plane segmentation.

(b) Mean Point Normal Difference

This indicator measures the standard deviation of the absolute difference between the point normal angle of the point set in the ground truth and the point normal angle of the point set predicted by the model (hereinafter referred to as MPND). For any two points belonging to the same point set, we have

$$MPND = |\angle(n(p_i), n(p_j)) - \angle(n(p_i^{GT}), n(p_j^{GT}))| \tag{9}$$

Among them, $n(p_i)$ represents the point normal at point p_i, and the point normal is closely related to the attitude of the corresponding plane. This item can be used to evaluate the attitude of the plane.

(c) Reprojection Degree

Since the output of our network is actually a division of points, and these points are ultimately used as the basis for generating a plane. Then, projecting the predicted set of points onto their corresponding fitted planes should be close to their projections are on the ground truth fitted plane. Whether the model restores the correlation between point sets is measured by calculating how many points and their projection points on the corresponding fitted plane. Here, the angle of the predicted line and the ground-truth fitted primitive is less than $0.1\pounds$ in our experiments.

Table 1
Quantitative comparative analysis of different algorithms.

Methods	Seg mIoU	Point normal	Reprojection degree
Efficient Ransac	45.9	10.6°	35.1%
Pearl	47.2	12.4°	34.2%
Global L_0	56.1	9.4°	73.9%
SCT-Net	**69.8**	10.1°	**75.4%**

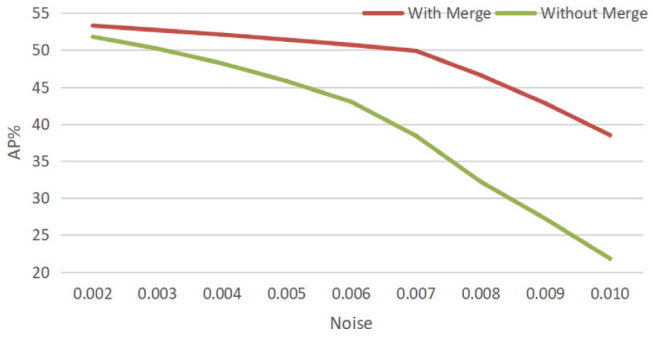


Fig. 3. Quantitative test of robustness against merge or not with varying Gaussian noise scales.

4.3. Experimental analysis

To highlight the superiority of SCT-Net, the representative methods such as Efficient Ransac (Schnabel et al., 2007), Pearl (Delong et al., 2012) and Global L0 (Lin et al., 2020) were selected for testing the performance based on the relationship matrix of the spherical coordinate system. Table 1 shows the comparison of the differences between the results of these methods.

As shown in Table 1, it can be clearly observed that the primitive extraction algorithm of point cloud based on the relationship matrix of the spherical coordinate system has the highest mean IoU, which indicating that the method can better classify the points of primitive instance, and satisfy the reprojection requirements. The proportion of points with the threshold is also the largest, suggesting that the method can analyze the relationship between points more accurately. The point normal vectors generated by Global L_0 are more accurate, indicating that the effect of noise on it is minimal. From the quantitative analysis given in the above table, the results obtained by the point cloud primitive extraction algorithm based on the spherical coordinate system relationship matrix are basically close to the ground truth. In some parts with apparent edges and corners, the boundary line segments are also smooth. In addition, the method does not produce over-segmentation.

Ablation experiment for spherical coordinate system analysis module is also carried out. At present, the main network skeletons for feature extraction of point clouds are PointNet++ and Spconv. In order to reflect the improvement of the network model by the spherical coordinate system analysis module, Spconv is used to replace the PointNet++ part of the network, and the network is trained on the ABC dataset. The two evaluation indicators below are added to evaluate the performance of the network, namely Label mIoU and average precision (Average Precision hereinafter referred to as AP). Among them, AP measures the degree of false detection and degree of missed detection of the model. The larger the value, the smaller the degree of false detection and missed detection.

As displayed in Table 2 that the spherical coordinate system has improved the performance of the two main point cloud feature extraction backbone networks to varying degrees. Here, the test dataset is same as the test dataset of PrimitiveNet. Therefore, the measurements including AP_{25}, AP_{50} and AP of PointNet++ and Spconv are directly cited from the (Huang et al., 2021). The advantages of the spherical

Table 2
Analysis of ablation experiments.

Methods	Seg mIoU	Point normal	Reprojection degree	AP_{25}	AP_{50}	AP
PoinNet++	71.8%	87.9%	12.7%	28.4%	16.5%	12.7%
PoinNet++ + SCT	72.0%	89.1%	13.0%	68.6%	55.3%	43.8%
Spconv	82.3%	91.8%	53.1%	73.6%	59.1%	53.1%
Spconv + SCT	82.7%	93.1%	54.4%	77.5%	66.3%	60.5%

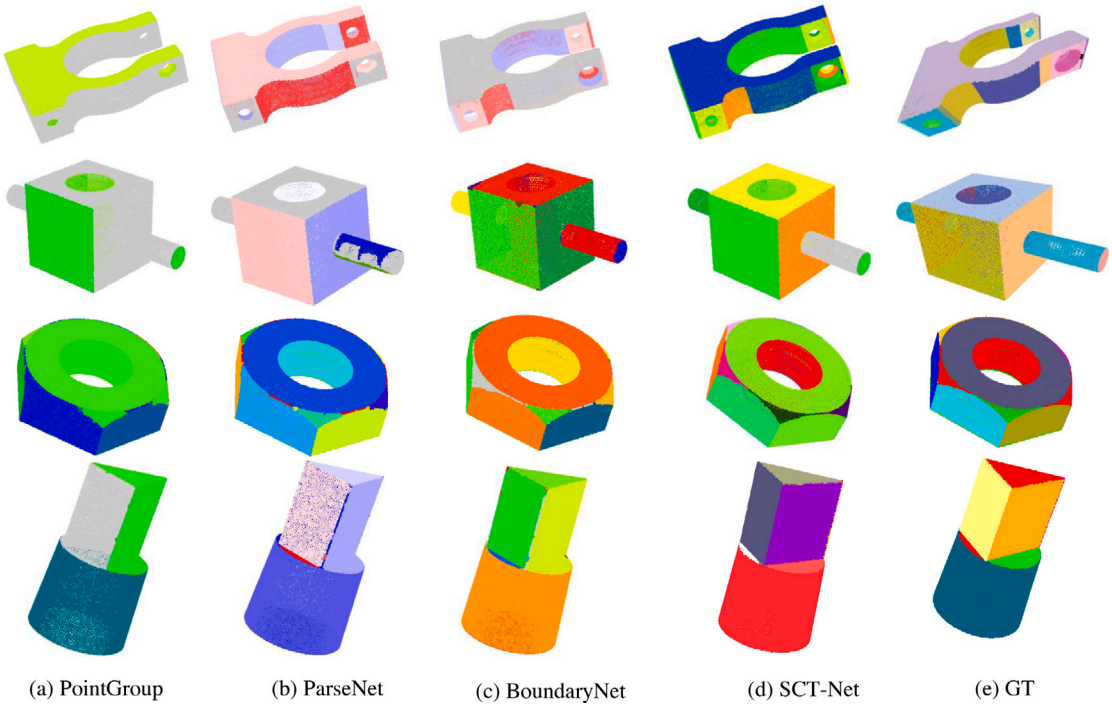


Fig. 4. Some select visual comparisons via the representative methods on ABC dataset (Koch et al., 2019). The proposed SCT-Net achieves the best performance.

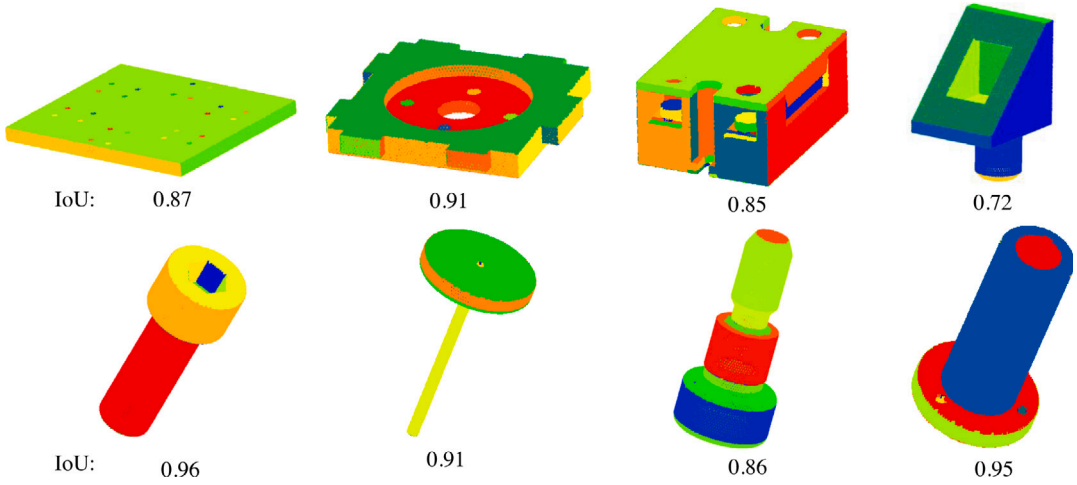


Fig. 5. Examples of some other models are listed. Our proposed method (.i.e SCT-Net) can accurately segment the primitive instances.

coordinate system are presented that it is more suitable for the following relationship matrix modules. The relationship between points can be represented accurately and clearly. Second, since the part of the experiment only uses a small part of the ABC dataset and does not add additional Gaussian noise to it, the performance of Spconv is better than that of PointNet++.

In order to better demonstrate the sensitivity of the primitive instance extraction, quantitative test of robustness against merge or not is designed with varying Gaussian noise scales. The algorithm based on the relationship matrix of the spherical coordinate system to noise and the effect of merging grouping in high noise, Gaussian noises of different scales are added to the ABC dataset. Fig. 3, which shows the optimized performance of the noise by final merge grouping, plots the accuracy variation curves of the quantitative results under different noise scales. In the case of high noise, if the final result is not merged and grouped, a large number of additional results will be obtained, which leads to a sharp drop in network performance.

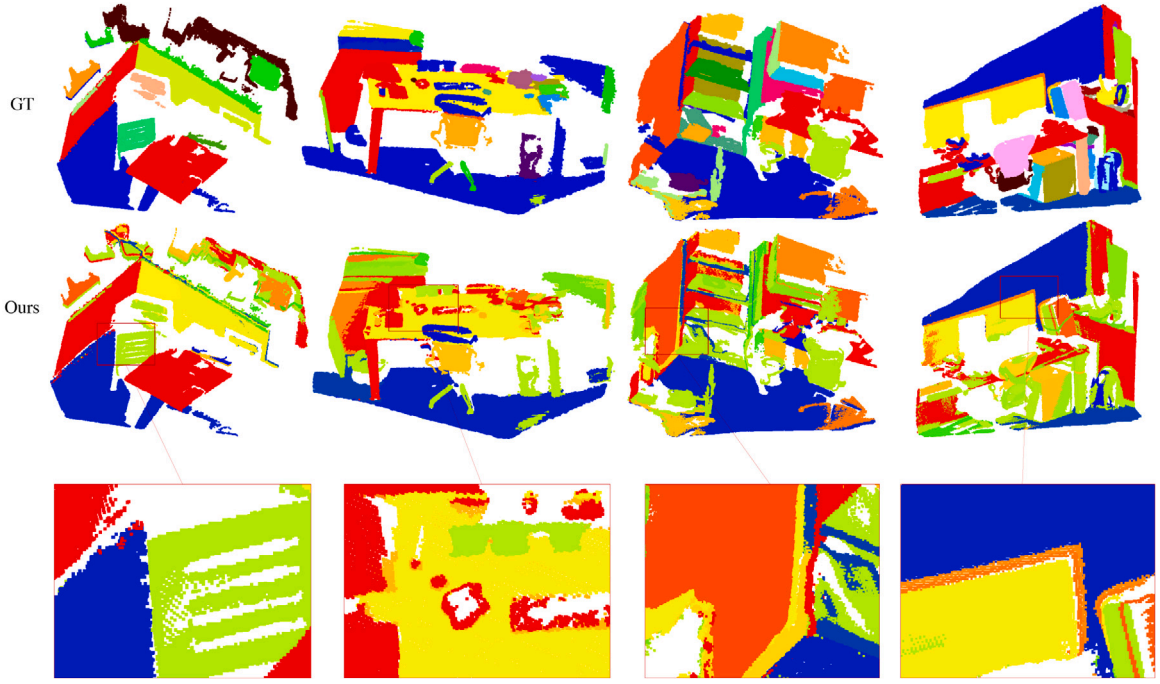


Fig. 6. Primitive instance segmentation on the real-world indoor dataset.

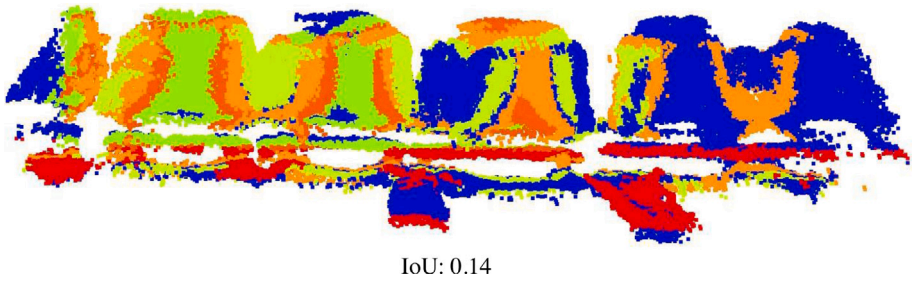


Fig. 7. The failure result of primitive instance segmentation.

Comparative instance segmentation. Some representative methods such as PointGroup (Jiang et al., 2020), BoundaryNet (Loizou et al., 2020) and ParseNet (Sharma et al., 2020) have been selected for experimental comparison. A limited number (10,000) of inputting points was set such that all the networks can afford. Table 3 lists the tested results of representative algorithms on the ABC dataset. It is obvious that the proposed SCT-Net outperforms previous representative methods according to the scores of Seg mIoU. Here, the test dataset is same as the test dataset of PrimitiveNet that leads to the same results. Therefore, the compared measurements including AP_{25}, AP_{50} and AP are directly cited from the (Huang et al., 2021). Fig. 4 shows the corresponding visualization results, where the proposed method can accurately segment a variety of small primitive instances. It can be clearly observed that the proposed SCT-Net achieves the state-of-the-art performance. As shown in Fig. 5, many other results from ABC models are listed, revealing that the primitive instances have been detected accurately.

Application for indoor dataset. In order to test the proposed SCT-Net in real-world dataset, we have selected some indoor scenes from SUN3D dataset for some tests without any training. As far as we know, there is currently a lack of primitive-level benchmark datasets in the real-world scans. We collect the indoor dataset and applied it to show the experimental effect. The ground truth models are manually labeled. As shown in Fig. 6, the proposed method achieves specific effects in the primitive instance segmentation of real-world indoor point clouds.

Table 3
The results of Seg mIoU measurement which is tested on ABC dataset. The proposed SCT-Net achieves state-of-the-arts performance.

Methods	PointGroup	ParseNet	BoundaryNet	SCT-Net
Seg mIoU	61.4%	63.5%	71.1%	**82.7%**
AP_{25}	19.9%	25.7%	21.5%	**72.5%**
AP_{50}	12.4%	15.3%	13.6%	**63.5%**
AP	10.2%	11.4%	10.4%	**57.2%**

5. Discussion

In the real world, scenes or objects composed of many different types of primitives make primitive-based representation increasingly difficult. Due to the fact that the proposed method is designed based on a spherical coordinate system, the quality of primitive segmentation mainly focuses on simple primitives such as planar, cylinder and cone, which is relatively sensitive to complex objects in the scene. As shown in Fig. 7, the seatbacks are irregularity and lack supervised model reference. Therefore, the segmentation of primitives fails.

6. Conclusion

A novel primitive detection algorithm from point cloud is proposed based on spherical coordinate system relationship matrix. First of all, the current mainstream methods and adjustments in point cloud

plane detection are introduced in brief. Since most of the current mainstream methods require strict parameter adjustment and their application directions are very limited, an spherical coordinate transformation embedded-based deep network has been proposed with loose parameter settings. The advantage of the neural network is that its parameters are self-adjusted by learning. Thus, the input of the algorithm only needs the original point cloud, and the SCT-Net learns the characteristics of primitives from the unorganized point cloud. Therefore, compared with the traditional algorithm, it also has certain advantages in terms of the speed. Secondly, the whole network framework is introduced and analyzed in detail. At the same time, the reasons and functions of each module design are expounded. Finally, the experimental part expounds on the effectiveness and advancement of the method via the comparative experiments. In addition the ablation experiment part achieves the state-of-the-art performance than the representative methods, and has a specific robustness.

Declaration of competing interest

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

Data availability

No data was used for the research described in the article.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of Jiangxi Province under Grants 20212BAB212012 and the National Natural Science Foundation of China under Grant 62076117, Grant 41871380 and Grant 61762061. The authors would like to acknowledge the anonymous reviewers for their valuable comments.

References

Besl, P.J., Jain, R.C., 1988. Segmentation through variable-order surface fitting. IEEE Trans. Pattern Anal. Mach. Intell. 10 (2), 167–192. http://dx.doi.org/10.1109/34.3881.

Che, E., Olsen, M.J., 2018. Multi-scan segmentation of terrestrial laser scanning data based on normal variation analysis. ISPRS J. Photogramm. Remote Sens. 143, 233–248. http://dx.doi.org/10.1016/j.isprsjprs.2018.01.019.

Chen, D., Wang, R., Peethambaran, J., 2017. Topologically aware building rooftop reconstruction from airborne laser scanning point clouds. IEEE Trans. Geosci. Remote Sens. 55 (12), 7032–7052. http://dx.doi.org/10.1109/TGRS.2017.2738439.

Chum, O., Matas, J., 2005. Matching with PROSAC-progressive sample consensus. In: Proc. CVPR. pp. 220–226. http://dx.doi.org/10.1109/CVPR.2005.221.

David, M., Gabor, L., 2001. Robust segmentation of primitives from range data in the presence of geometric degeneracy. IEEE Trans. Pattern Anal. Mach. Intell. http://dx.doi.org/10.1109/34.910883.

Delong, A., Osokin, A., Isack, H.N., Boykov, Y., 2012. Fast approximate energy minimization with label costs. Int. J. Comput. Vis. 96 (1), 1–27. http://dx.doi.org/10.1109/CVPR.2010.5539897.

Derpanis, K.G., 2010. Overview of the RANSAC algorithm. Image Rochester NY 4 (1), 2–3.

Du, T., Inala, J.P., Pu, Y., Spielberg, A., Schulz, A., Rus, D., Solar-Lezama, A., Matusik, W., 2018. Inversecsg: Automatic conversion of 3d models to csg trees. ACM Trans. Graph. 37 (6), 1–16. http://dx.doi.org/10.1145/3272127.3275006.

Fang, H., Lafarge, F., 2020. Connect-and-slice: an hybrid approach for reconstructing 3D objects. In: Proc. CVPR. pp. 13490–13498. http://dx.doi.org/10.1109/CVPR42600.2020.01350.

Fang, H., Lafarge, F., Desbrun, M., 2018. Planar shape detection at structural scales. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2965–2973. http://dx.doi.org/10.1109/CVPR.2018.00313.

Fischler, M.A., Bolles, R.C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM 24 (6), 381–395. http://dx.doi.org/10.1145/358669.358692.

Holzmann, T., Maurer, M., Fraundorfer, F., Bischof, H., 2018. Semantically aware urban 3d reconstruction with plane-based regularization. In: Proc. ECCV. pp. 468–483. http://dx.doi.org/10.1007/978-3-030-01264-9-29.

Huang, J., Zhang, Y., Sun, M., 2021. PrimitiveNet: Primitive instance segmentation with local primitive embedding under adversarial metric. In: Proc. ICCV. pp. 15343–15353. http://dx.doi.org/10.1109/ICCV48922.2021.01506.

Jiang, Y., Dai, Q., Min, W., Li, W., 2021. Non-watertight polygonal surface reconstruction from building point cloud via connection and data fit. IEEE Geosci. Remote Sens. Lett. 19, 1–5. http://dx.doi.org/10.1109/LGRS.2021.3113662.

Jiang, L., Zhao, H., Shi, S., Liu, S., Fu, C.-W., Jia, J., 2020. Pointgroup: Dual-set point grouping for 3d instance segmentation. In: Proc. CVPR. pp. 4867–4876. http://dx.doi.org/10.1109/CVPR42600.2020.00492.

Koch, S., Matveev, A., Jiang, Z., Williams, F., Artemov, A., Burnaev, E., Alexa, M., Zorin, D., Panozzo, D., 2019. Abc: A big cad model dataset for geometric deep learning. In: Proc. CVPR. pp. 9601–9611. http://dx.doi.org/10.1109/CVPR.2019.00983.

Lafarge, F., Alliez, P., 2013. Surface reconstruction through point set structuring. Comput. Graph. Forum 32 (2pt2), 225–234. http://dx.doi.org/10.1111/cgf.12042.

Lê, E.-T., Sung, M., Ceylan, D., Mech, R., Boubekeur, T., Mitra, N.J., 2021. CPFN: Cascaded primitive fitting networks for high-resolution point clouds. In: Proc. ICCV. pp. 7457–7466. http://dx.doi.org/10.1109/ICCV48922.2021.00736.

Li, L., Sung, M., Dubrovina, A., Yi, L., Guibas, L.J., 2019. Supervised fitting of geometric primitives to 3d point clouds. In: Proc. CVPR. pp. 2652–2660. http://dx.doi.org/10.1109/CVPR.2019.00276.

Li, Y., Wu, X., Chrysathou, Y., Sharf, A., Cohen-Or, D., Mitra, N.J., 2011. Globfit: Consistently fitting primitives by discovering global relations. In: Proc. ACM SIGGRAPH. pp. 1–12, 10.11673.1938.

Lin, Y., Li, J., Wang, C., Chen, Z., Wang, Z., Li, J., 2020. Fast regularity-constrained plane fitting. ISPRS J. Photogramm. Remote Sens. 161, 208–217. http://dx.doi.org/10.1016/j.isprsjprs.2020.01.009.

Lin, Y., Wang, C., Chen, B., Zai, D., Li, J., 2017. Facet segmentation-based line segment extraction for large-scale point clouds. IEEE Trans. Geosci. Remote Sens. 55 (9), 4839–4854. http://dx.doi.org/10.1109/TGRS.2016.2639025.

Loizou, M., Averkiou, M., Kalogerakis, E., 2020. Learning part boundaries from 3d point clouds. 39, (5), Wiley Online Library, pp. 183–195. http://dx.doi.org/10.1111/cgf.14078,

Maalek, R., Lichti, D., Ruwanpura, J., 2015. Robust classification and segmentation of planar and linear features for construction site progress monitoring and structural dimension compliance control. ISPRS Ann. Photogrammetry Remote Sens. Spatial Inf. Sci. 2, 10.1424-8220/18/3/819.

Nurunnabi, A., Belton, D., West, G., 2016. Robust segmentation for large volumes of laser scanning three-dimensional point cloud data. IEEE Trans. Geosci. Remote Sens. 54 (8), 4790–4805. http://dx.doi.org/10.1109/tgrs.2016.2551546.

Poullis, C., 2019. Large-scale urban reconstruction with tensor clustering and global boundary refinement. IEEE Trans. Pattern Anal. Mach. Intell. 42 (5), 1132–1145. http://dx.doi.org/10.1109/TPAMI.2019.2893671.

Rabbani, T., Heuvel, F., Vosselman, G., 2006a. Segmentation of point clouds using smoothness constraint. 10.11221.7999.

Rabbani, T., Van Den Heuvel, F., Vosselmann, G., 2006b. Segmentation of point clouds using smoothness constraint. Int. Arch. Photogrammetry Remote Sens. Spatial Inf. Sci. 36 (5), 248–253, 10.11221.7999.

Schnabel, R., Wahl, R., Klein, R., 2007. Efficient RANSAC for point-cloud shape detection. Comput. Graph. Forum 26 (2), 214–226. http://dx.doi.org/10.1111/j.1467-8659.2007.01016.x.

Sharma, G., Liu, D., Maji, S., Kalogerakis, E., Chaudhuri, S., Měch, R., 2020. Parsenet: A parametric surface fitting network for 3d point clouds. In: Proc. ECCV. Springer, pp. 261–276. http://dx.doi.org/10.48550/arXiv.2003.12181.

Tulsiani, S., Su, H., Guibas, L.J., Efros, A.A., Malik, J., 2017. Learning shape abstractions by assembling volumetric primitives. In: Proc. CVPR. pp. 2635–2643. http://dx.doi.org/10.1109/CVPR.2017.160.

Wang, W., Yu, R., Huang, Q., Neumann, U., 2018. Sgpn: Similarity group proposal network for 3d point cloud instance segmentation. In: Proc. CVPR. pp. 2569–2578. http://dx.doi.org/10.1109/CVPR.2018.00272.

Wu, Q., Xu, K., Wang, J., 2018. Constructing 3D CSG models from 3D raw point clouds. 37 (5), 221–232. http://dx.doi.org/10.1111/cgf.13504.

Xiao, J., Owens, A., Torralba, A., 2013. Sun3d: A database of big spaces reconstructed using sfm and object labels. In: Proc. ICCV. pp. 1625–1632. http://dx.doi.org/10.1109/ICCV.2013.458.

Xu, S., Wang, R., Wang, H., Zheng, H., 2019. An optimal hierarchical clustering approach to mobile LiDAR point clouds. IEEE Trans. Intell. Transp. Syst. 21 (7), 2765–2776. http://dx.doi.org/10.1109/TITS.2019.2912455.

Yan, S., Yang, Z., Ma, C., Huang, H., Vouga, E., Huang, Q., 2021. Hpnet: Deep primitive segmentation using hybrid representations. In: Proc. ICCV. pp. 2753–2762. http://dx.doi.org/10.1109/ICCV48922.2021.00275.

Zou, C., Yumer, E., Yang, J., Ceylan, D., Hoiem, D., 2017. 3D-prnn: Generating shape primitives with recurrent neural networks. In: Proc. ICCV. pp. 900–909. http://dx.doi.org/10.1109/ICCV.2017.103.