Towards Weakly Supervised Semantic Segmentation in 3D Graph-Structured Point Clouds of Wild Scenes

Haiyan Wang[1]
hwang005@citymail.cuny.edu
Xuejian Rong[1]
xrong@ccny.cuny.edu
Liang Yang[1]
lyang1@ccny.cuny.edu
Shuihua Wang[2]
sw546@le.ac.uk
Yingli Tian (Corresponding author)[1]
ytian@ccny.cuny.edu

[1] The City College, City University of New York, New York, NY 10031, USA
[2] University of Leicester, LE1 7RH, UK

Abstract

The deficiency of 3D segmentation labels is one of the main obstacles to effective point cloud segmentation, especially for wild scenes with varieties of different objects. To alleviate this issue, we propose a novel graph convolutional deep framework for large-scale semantic scene segmentation in point clouds with solely 2D supervision. Different with numerous preceding multi-view supervised approaches focusing on single object point clouds, we argue that 2D supervision is also capable of providing enough guidance information for training 3D semantic segmentation model of natural scene point clouds while not explicitly capturing their inherent structures, even with only single view per sample. Specifically, a Graph-based Pyramid Feature Network (GPFN) is designed to implicitly infer both global and local features of point sets, and a perspective rendering and semantic fusion module are proposed to provide refined 2D supervision signals for training along with a 2D-3D joint optimization strategy. Extensive experimental results demonstrate the effectiveness of our 2D supervised framework, which achieves comparable results with the state-of-the-art approaches trained with full 3D labels, for semantic point cloud segmentation on the popular S3DIS benchmark.

1 Introduction

The last decade has witnessed quick advancement in 3D scanning technologies which have become increasingly ubiquitous and paved the way for generating highly accurate point cloud data. This includes sensors such as laser scanners, time-of-flight sensors such as *Microsoft Kinect* or *Intel RealSense* device, structural light sensors such as *iPhone X* and *Structure Sensor*, and outdoor sensors such as *LiDA*.

3D information could significantly contribute to fine-grained scene understanding. For instance, depth information could drastically reduce the segmentation ambiguities from 2D imagery, and surface normal could provide important cues of the scene geometry. However, 3D data are typically formed with point clouds (geometric point sets in Euclidean space), which is a set of unordered 3D points with or without additional information such as RGB on each point. These 3D points do not conform to the regular lattice grids as in 2D images. Directly converting point clouds to 3D regular volumetric grids might bring computation intractability due to the issues of unnecessary sparsity and high-resolution volumes. The work in PointNet[15] and PointNet++ [16] have pioneered the use of deep learning for 3D point cloud processing with handling the permutation invariance problem, including reconstruction and semantic segmentation tasks. However, these methods still heavily depend on 3D aligned point-wise labels as strong supervision signals for training, which are difficult and cumbersome to prepare and annotate.

In this paper, unlike existing methods which typically require expensive point-wise 3D annotations, we tackle the task of semantic point cloud segmentation for natural scenes by utilizing easily available 2D data such as segmentation masks to supervise the training process. However, the occluded objects may not be assigned correct labels in generating 2D segmentation masks from a given viewpoint. Therefore, we propose a novel filtering strategy to refine the 2D projection map for providing better supervision signal to the proposed model.

We argue that 2D supervision is capable of providing enough guidance information for training 3D semantic scene segmentation model for point clouds while not explicitly capturing inherent structures of 3D point clouds. Different with some recent 2D multi-view supervision-based single object 3D reconstruction approaches [8, 10, 11] (enforcing cycle-consistency or not) which solely focus on single objects and require 2D data in multiple viewpoints, our approach works on the large-scale scene segmentation of point clouds with requiring only single view per sample. The unified architecture illustrated in Figure 1 comprises a Graph-based Pyramid Feature Network (GPFN), a 2D perspective rendering module, and a 2D-3D joint optimizer. Specifically, the graph convolutional feature pyramid encoder works to hierarchically infer the semantic meaning of a scene in both local and global levels. The 2D perspective rendering works along with the distance filter to generate effective refined 2D masks for loss computation. And the 2D-3D joint optimizer supports the complete end-to-end training.

Our main contributions are summarized as follows:

- A joint 2D-3D deep architecture is designed to compute hierarchical and spatially-aware features of the input point cloud by integrating graph-based convolution and pyramid structure for encoding, which further compensates the weak 2D supervision information.
- A novel reprojection method, named *perspective rendering*, is proposed to enforce the

2D and 3D geometric correspondence. Our approach significantly alleviates the need of 3D point-wise annotations for training, while only the 2D ground truth segmentation mask is used to calculate loss with the reprojection. And if necessary, it is also capable to transfer the learned prior of 2D semantic segmentation model to the 3D counterpart, i.e. adopting predicted 2D segmentation masks for loss computation instead of the ground truth.

- To the best of our knowledge, this is the first work to apply 2D supervision for 3D semantic point cloud segmentation of wild scenes without using any 3D point-wise annotations. Extensive experiments are conducted and the proposed method achieves comparable performance with the state-of-the-art 3D supervised methods on the popular S3DIS benchmark.

2 Related Work

Deep Learning for 3D Point Clouds. In the deep learning era, early attempts at using deep learning for large 3D point cloud data usually replicated successful convolutional architecture by converting point sets to regular grid-alike voxels. Recent emerging approaches go beyond this and are able to directly process point clouds. Starting with PointNet [15], more methods such as PointNet++ [16] and Frustum PointNets [14] were proposed to directly feed point clouds to networks with fulfilling permutation invariance, and achieved good performance for 3D tasks for point clouds such as classification [6, 17, 18, 20], segmentation [23], reconstruction [12], completion [1], and etc. This paper focuses on the task of large-scale semantic scene segmentation for point clouds.

3D Semantic Segmentation. Before PointNet was proposed, deep learning-based methods have already become popular in solving 3D semantic segmentation for other data formats including voxel and mesh. Previous methods proposed by Song et al. [19] and Dai et al. [3] tackled the semantic scene completion from the 3D volume perspective, as well as explored the relationship between scene completion and semantic scene parsing. However, the limitations of these volume-based 3D methods are that they have to sacrifice the representation accuracy and cause huge redundancies. Recently, deep learning methods based on points were proposed [15, 16] to handle 3D semantic segmentation from the perspective of points and take the point cloud data as input which are permutation invariant, and output the class labels for each point. Dai et al. [9] proposed a graph-based method to handle large scale point clouds or super points. And frameworks proposed by Engelmann et al. [4, 5] aimed to enlarge the receptive field of 3D scene and explored the spatial context information for semantic segmentation. Wang et al. proposed a method to find the promotion between instance segmentation and semantic segmentation [21]. Our approach, however, focuses on effectively utilizing easily accessible 2D training data for 3D large-scale scenes.

2D Supervision for 3D Tasks. While 3D supervision semantic segmentation has made great progress, many researchers started to explore using 2D labels to train networks for 3D tasks in order to reduce the heavy workload of labeling 3D annotations (point clouds, voxels, meshes, etc.), albeit most of which are designed for single objects. The work proposed by Lin et al. [11] attempted to generate point clouds for object reconstruction and applied 2D

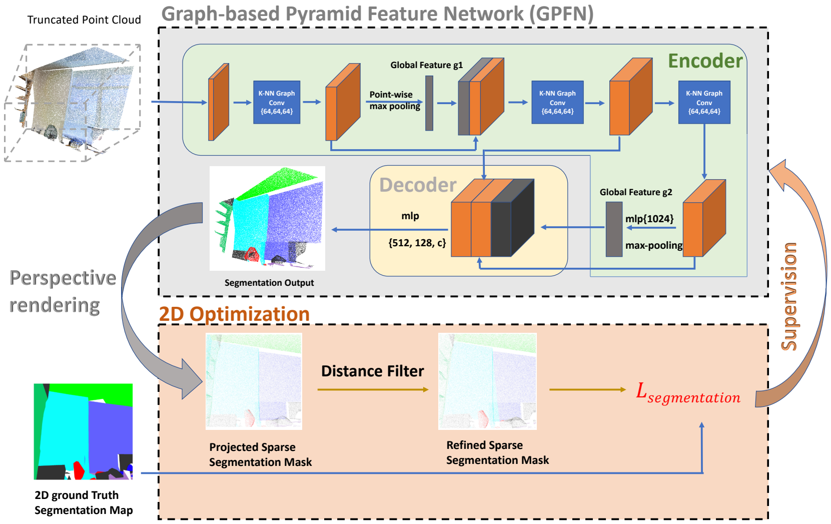

Figure 1: The main pipeline of the proposed graph convolutional deep framework of 2D-supervised 3D semantic point cloud segmentation. Brown line in the network means the identical propagation process.

projection mask and depth mask for joint optimization. The authors introduced a pseudo-rendering in the 2D image plane, which solves the collision within single object during projection. However, the simple up-sampling followed with a max-pooling strategy only works well with single object. When dealing with a more complex scene which contains multiple objects, the pseudo-rendering cannot guarantee to assign correct labels for the objects of different classes when they have collision. Navaneet et al. [13] proposed CAPNET for 3D point cloud reconstruction. The authors introduced a continuous approximation projection module and proposed a differentiable point cloud rendering to generate a smooth and accurate point cloud projection. Through the supervision of 2D projection, their method achieved comparable reconstruction results and generalizability in the real data. In this work, we propose an unprecedented method towards better 2D supervision for 3D point cloud semantic scene segmentation and prove its effectiveness on the S3DIS benchmark.

3 Methodology

3.1 Overview

Previous 3D-Supervised deep models for semantic point cloud segmentation, such as PointNet [15], PointNet++ [16], and DGCNN [22], usually require 3D point-wise class labels in training and achieve satisfying results. In order to reduce the expensive labeling effort for each point in 3D point cloud data, we propose a weakly 2D-supervised method by using the

2D ground truth segmentation map which is easier to obtain to supervise the whole training process. Inspired by DGCNN [22], we propose an effective encoder and decoder network to learn the representation of the point cloud and output corresponding segmentation prediction. Perspective rendering and distance filter are further designed to handle the object occlusion and point collision problem.

Figure 1 illustrates the proposed 2D-supervised graph convolutional deep framework for 3D semantic point cloud segmentation which comprises 2 main components: the Graph-based Pyramid Feature Network (GPFN) network and the 2D optimization module. The **GPFN** contains a graph-based feature pyramid encoder and decoder network. The encoder takes a truncated point cloud (see details in Section 3.2) from a given viewpoint as input and the decoder predicts the class label for each point. During the **2D Optimization** process, after perspective rendering, the projected sparse segmentation mask from the predicted point cloud segmentation is refined by distance filter and finally calculating the 2D sparse segmentation loss with 2D ground truth segmentation map as supervision to optimize the training network. To the best of our knowledge, we are the first to apply weakly 2D supervision to the point cloud semantic scene segmentation task.

3.2 Graph Convolutional Feature Pyramid Encoder

By casting rays from the camera through each pixel to the scene, the points under specific viewpoints are extracted respectively to obtain the truncated point cloud for multiple viewpoints. An encoder-decoder network (E_p, D_p) is trained which takes the truncated input point cloud of $X_p \in \mathbb{R}^{N\times 6}$ (N is the number of the points and 6 is the dimension of each point including XYZ and RGB) from a given viewpoint $v = \{R,t\}$ and predicts the class label of the point cloud with size of $\widehat{X}_p \in \mathbb{R}^{N\times C}$ (C is the number of classes.)

First, the truncated point cloud from a given viewpoint is fed into the encoder network E_p, which is comprised of several 1D convolution layers, edge graph convolution layers, and max pooling layers to map the input data to a latent representation space. Then the decoder network D_p processes the feature vector through several fully-connected layers $(512, 128, C)$ and finally output the class prediction of each point.

In order to conjunct with the weak 2D labels, a graph-based feature pyramid encoder is designed to subside the effect of weak labels to the point cloud segmentation. Combining the dynamic graph convolution model and pyramid structure, the network could globally understand the semantic meaning of a scene in both low level and high level layers. Inspired by [22], we introduce the K-NN dynamic graph edge convolution here. For each graph convolution layer, the K-NN graph is different and represented with $\mathcal{G}^{(l)} = (\mathcal{V}^{(l)}, \mathcal{E}^{(l)})$. $|\mathcal{V}|$ represents the nearest k points to x_i, and $\mathcal{E}$ stands for edges between (i, j_1), ... , (i, j_k). Through the graph convolution h_θ, the local neighbourhood information is aggregated by capturing edge features between k neighbors and center points, $h_\theta(x_i, x_j) = h_\theta(x_i, x_j - x_i)$. As shown in Figure 1, the pyramid of two global layers are added to the GPFN. The global features $g1, g2$ are concatenated with previous point feature in both low level and high level. This pyramid design and augmented point feature matrix are effective to improve the performance when using 2D supervision.

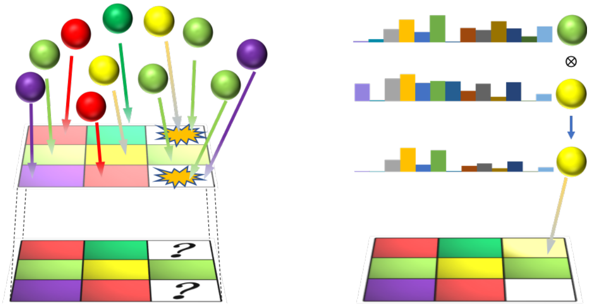

Figure 2: Concept illustration of the proposed perspective rendering and semantic fusion. During the projection, multiple points of different object classes (shown in different colors) are projected to grids (with corresponding colors) in the image plane. Each point has a probability distribution of the predicted classes. For the grid which has multiple projected points, the perspective rendering is applied by calculating the dot product of the probabilities for all the points according to the classes, and after normalization, the class label of this grid can be finally determined.

3.3 Perspective Rendering and Distance Filter

As one of our main contributions the whole process includes **Perspective Rendering** and **Distance Filter** which is used to generate the effective 2D prediction results.

Perspective Rendering. Point cloud in the world coordinate system is represented as $p_w = (x_w, y_w, z_w)$. Camera pose and 3D transformation matrix for a given viewpoint are (R_k, t_k). The projected point in the camera coordinate system $\widehat{p_c} = (x_c, y_c, z_c)$ can be obtained through Eq. (1).

$$\widehat{p_c} = (\widehat{x_c}, \widehat{y_c}, \widehat{z_c}) = (R_k p_w + t_k) = (R_k(x_w, y_w, z_w) + t_k) \tag{1}$$

However, as shown in the Figure 2 during this process, different points might be projected to the same pixel position in the image plane. Through Eq. (2), the perspective rendering is applied for semantic fusion by predicting the probability distribution across all classes and fusing the probability between all the N points which are projected to the same grid cell. At last, the probability distribution of this grid is obtained through semantic fusion, the largest probability such as yellow shown in Figure 2 will be assigned as the final label of this grid.

$$p(C_i|x_{grid}) = \prod_{n=1}^{N} p(C_i|x_n), \quad p(x_{grid}) = max\{p(C_1|x_{grid}), ..., p(x_{C_{13}|grid})\} \tag{2}$$

Distance Filter. In the projection, the collision problem may happen which means the points on hidden objects and visible objects of various classes might be projected to the same area and have intersection in the image plane. Apparently, as shown in the Figure 3, there exist collisions such as bookcase and wall, computer and window, chair and wall etc. To solve this problem, the boundaries of all of the instances are extracted from the 2D ground truth segmentation map using connect components algorithm. We notice that collision only occurs at the intersection area of 2 boundaries.

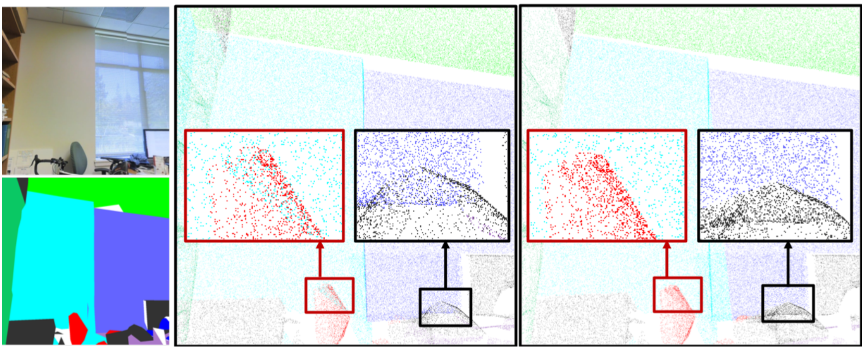

Figure 3: Illustration of the proposed distance filter. The left column shows the RGB image (just for visualization, it is not used in training) and 2D ground truth segmentation map under this viewpoint. The middle and right images demonstrate the comparison of the projected mask before and after applying the distance filter. Two areas with collision are zoomed in to view details: in the red box, points of different objects (chair and wall) are projected in the same region in 2D image plane before using distance filter. After applying the distance filter, they are correctly separated. The collision problem is also resolved as shown in the black box between the clutter (in black) and window (in purple).

Denote $p_{cn} = (x_{cn}, y_{cn}, z_{cn}, l_n)$, $n \in \{1, ..., N\}$, as the point sets at the intersection area of 2 boundaries (N is number of points and l is the label of point). Thus, along with depth value (z_c), we could use the distance filter to determine the smallest depth value and its corresponding label over the intersection area. $l_{inter} = l_{argmin(z_{cn})}$. Finally, the optimized projected map is obtained as shown in the rightmost column of Figure 3. As shown in the figure, the points of hidden objects will be filtered out, which will not lead to an error when calculating segmentation loss with the ground truth segmentation map.

3.4 Loss Function

The sparse loss is calculated for the projected segmentation result in the training process as the following equation:

$$L_{seg} = -\frac{1}{N}\sum_{i=1}^{N}[p_i \log(\hat{p}_i + (1-p_i)\log(1-\hat{p}_i))] \tag{3}$$

where p_i is the predicted point cloud label projected to the 2D image plane. According to the 2D coordinates of the projected points, $\hat{p}_i$ is obtained by finding the label of the corresponding points in the ground truth segmentation map.

4 Experiments

4.1 Datasets

The proposed weakly 2D-supervised semantic segmentation method is evaluated on the public and challenging 3D wild scene dataset: *Stanford Large-Scale 3D Indoor Spaces* (S3DIS)

Training data	mAcc	mIoU	oAcc
1/4	66.7	50.9	79.5
1/6	66.5	50.8	79.1
1/12	56.5	39.3	66.2
1/20	37.8	29.1	40.0

Table 1: Performance comparison of using different amount of training data (The capacity of our model is not representative enough to utilize more data and benefit the performance. It has huge potential to be improved in the future).

dataset [2]. This dataset consists of 3D scan point clouds for 6 indoor areas including total of 272 rooms. For each room, thousands of viewpoints are given, including camera pose, 2D RGB image, segmentation map and depth image under each specific viewpoint. For segmentation, there are 13 object categories including *ceiling, floor, wall, beam, column, window, door, table, chair, bookcase, sofa, board,* and *clutter.* The S3DIS dataset contains various larger-scale natural indoor environments and significantly more challenging than other real 3D datasets such as ScanNet [3] and SceneNN [7] datasets.

4.2 Implementation Details

For the S3DIS dataset, each point is represented as a normalized flat vector (`XYZ,RGB`) with the dimension of 6. These truncated point clouds are used as training data as well as calculating the loss with 2D segmentation map under the same viewpoint. Followed the settings as [15], in which each point is represented as a 9D vector (`XYZ,RGB,UVW`), here `UVW` are the normalized spatial coordinates. In testing, the testing data is the points of the complete room similar to other 3D fully supervised methods. The experiment results are reported by testing on the 6-fold cross validation over the 6 areas (*area 1 - area 6*). Our proposed network is trained with 100 epochs with batch size 48, base learning rate 0.001 and then is divided by 2 for every 300k iterations. The Adam solver is adopted to optimize the network on a single GPU. A connected component algorithm is employed to calculate the boundary of each instance in the ground truth segmentation map. The performance of semantic segmentation results is evaluated by the standard metrics: mean accuracy of total classes (*mAcc*), mean per-class intersection-over-union (*mIOU*), and overall accuracy (*oAcc*).

4.3 Ablation Study of Amount of Training Data

In S3DIS dataset, the scene point clouds are constructed by thousands of viewpoints. Here the robustness of the proposed point cloud segmentation network is evaluated by using different amount of training data. Table 1 shows the performance of using various data proportion ($1/4, 1/6, 1/12, 1/20$ of the viewpoints, they are evenly random selected). There is no much difference between using $1/4$ and $1/6$ of all data. However, the performance is significantly decreased when using only $1/12$ or $1/20$ viewpoints. Considering the balance of training time and accuracy, $1/6$ data is used to conduct the following comparison with others.

	Method	**mAcc**	**mIoU**	**oAcc**
3D Supervision	*PointNet* [15]	66.2	47.6	78.5
	Engelmann et al. [4]	66.4	49.7	81.1
	PointNet++ [16]	67.1	54.5	81.0
	DGCNN [22]	-	56.1	84.1
	Engelmann et al. [5]	67.8	58.3	84.0
	SPG [9]	73.0	62.1	85.5
2D Supervision	*GPFN with DP (Ours)*	39.2	30.4	53.7
	GPFN with PR (Ours)	66.5	50.8	79.1

Table 2: Quantitative results of our proposed 2D supervised method on S3DIS by using only 1/6 of viewpoints in each room for training. "DP" indicates Direct Projection and "PR" indicates Perspective Rendering. The performance of our 2D supervised method is very close with most of the 3D supervised state-of-the-art methods.

4.4 Effectiveness of the Proposed Framework

As shown in the last two rows of Table 2, we conduct a series of experiments to evaluate the effectiveness of each component in our framework. During the training process, only $1/6$ of viewpoints in each room are used. The testing results are based on 6-fold cross validation, and a sparse cross entropy metric is applied to report the performance of semantic segmentation.

GPFN with 2D Supervision and Direct Projection. Instead of using 3D ground truth label as supervision, here only 2D segmentation ground truth map is adopted to optimize the training process together with the predicted 2D segmentation map in 2D by direct projection based on the basic camera model pose *(R,t)* from given viewpoint to re-project the predicted point cloud with labels to the image plane. Note that point collision might happen while the hidden object might be projected to the same area of the visible object. Obviously, the performance (39.2% mAcc, 30.4% mIoU, and 53.7% oAcc) is extremely lower than the 3D-supervised methods.

GPFN with the proposed 2D optimization. Similar to the above experiment, the *Perspective Rendering* is used to replace the direct projection. The points that are projected to the same intersection area of different classes are filtered by the proposed distance filter. Furthermore, the points that are projected to the same grid are selected by semantic fusion. As shown in Table 2, the segmentation results (66.5% mAcc, 50.8% mIoU, and 79.1% oAcc) are significantly improved than using direct projection and with small margins compared to 3D fully supervised results.

4.5 Comparison with the State-of-the-art Methods

Since there is no previous work of 2D supervised point cloud semantic segmentation for large-scale natural scenes, here, we compare the results with the state-of-the-arts of 3D supervised point cloud segmentation. As shown in Table 2, our method achieves comparable results with solely 2D supervision compared to most of the 3D supervised methods. Note that it even outperforms 3D fully supervised PointNet [15].The most recent top-performing 3D point cloud segmentation model, SPG [9], still leads a margin in terms of *mean IoU* by

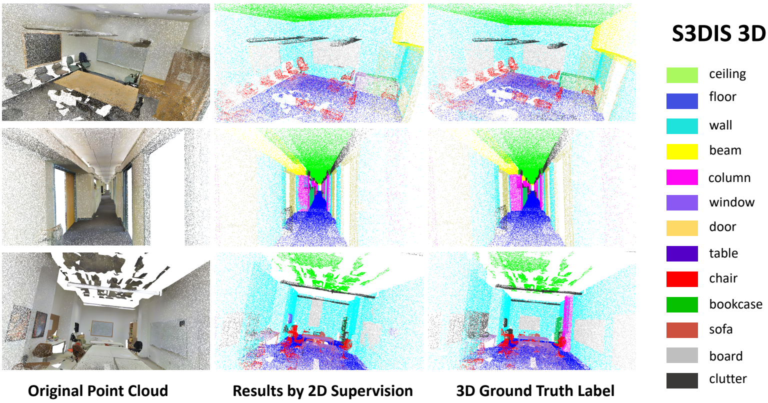

Figure 4: Qualitative results produced by our proposed method (middle column.)

applying a hierarchical architecture based on SuperPoints. However, the proposed approach achieves competitive performance in terms of *mean Accuracy* and *overall Accuracy*, without utilizing contextual relationship reasoning as in SPG. Figure 4 visualizes several example results on 3D point cloud semantic segmentation generated by our method. Overall, our proposed 2D supervised semantic segmentation method works well on the S3DIS dataset at various kinds of areas and rooms.

5 Conclusion

In this paper, we have proposed a novel graph convolutional deep framework for large-scale semantic scene segmentation in 3D point clouds of wild scenes with solely 2D supervision. Extensive experiments demonstrate the effectiveness of our approach. Different from numerous multi-view 2D-supervised methods focusing on single object point clouds, our proposed method can handle large-scale wild scenes with multiple objects and achieves encouraging performance, with even only single view per sample. The future directions include unifying the point cloud completion and segmentation tasks for natural scene point clouds.

6 Acknowledgement

This work was supported in part by NSF grant IIS-1400802.

References

[1] PCN: Point Completion Network, author=Wentao Yuan and Tejas Khot and David Held and Christoph Mertz and Martial Hebert. In *3DV*, 2018.

[2] Iro Armeni, Ozan Sener, Amir Roshan Zamir, Helen Jiang, Ioannis K. Brilakis, Martin A. Fischer, and Silvio Savarese. 3D Semantic Parsing of Large-Scale Indoor Spaces. *CVPR*, pages 1534–1543, 2016.

[3] Angela Dai, Daniel Ritchie, Martin Bokeloh, Scott E. Reed, Jürgen Sturm, and Matthias Nießner. ScanComplete: Large-Scale Scene Completion and Semantic Segmentation for 3D Scans. *CVPR*, pages 4578–4587, 2018.

[4] Francis Engelmann, Theodora Kontogianni, Alexander Hermans, and Bastian Leibe. Exploring Spatial Context for 3D Semantic Segmentation of Point Clouds. *ICCVW*, pages 716–724, 2017.

[5] Francis Engelmann, Theodora Kontogianni, Jonas Schult, and Bastian Leibe. Know What Your Neighbors Do: 3D Semantic Segmentation of Point Clouds. In *ECCV Workshops*, 2018.

[6] Joris Guerry, Alexandre Boulch, Bertrand Le Saux, Julien Moras, Aurélien Plyer, and David Filliat. SnapNet-R: Consistent 3D Multi-view Semantic Labeling for Robotics. *ICCVW*, pages 669–678, 2017.

[7] Binh-Son Hua, Quang-Hieu Pham, Duc Thanh Nguyen, Minh-Khoi Tran, Lap-Fai Yu, and Sai-Kit Yeung. SceneNN: A Scene Meshes Dataset with aNNotations. *3DV*, pages 92–101, 2016.

[8] Eldar Insafutdinov and Alexey Dosovitskiy. Unsupervised Learning of Shape and Pose with Differentiable Point Clouds. In *NeurIPS*, 2018.

[9] Loïc Landrieu and Martin Simonovsky. Large-Scale Point Cloud Semantic Segmentation with Superpoint Graphs. *CVPR*, pages 4558–4567, 2018.

[10] Yi-Lun Liao, Yao-Cheng Yang, and Yu-Chiang Frank Wang. 3D Shape Reconstruction from a Single 2D Image via 2D-3D Self-Consistency. *CoRR*, abs/1811.12016, 2018.

[11] Chen-Hsuan Lin, Chen Kong, and Simon Lucey. Learning Efficient Point Cloud Generation for Dense 3D Object Reconstruction. In *AAAI*, 2018.

[12] Priyanka Mandikal, L. NavaneetK., Mayank Agarwal, and Venkatesh Babu Radhakrishnan. 3D-LMNet: Latent Embedding Matching for Accurate and Diverse 3D Point Cloud Reconstruction from a Single Image. In *BMVC*, 2018.

[13] L NavaneetK, Priyanka Mandikal, Mayank Agarwal, and R. Venkatesh Babu. CAPNet: Continuous Approximation Projection For 3D Point Cloud Reconstruction Using 2D Supervision. *CoRR*, abs/1811.11731, 2019.

[14] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum PointNets for 3D Object Detection from RGB-D Data. *arXiv preprint arXiv:1711.08488*, 2017.

[15] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *CVPR*, 2017.

[16] Charles R Qi, Li Yi, Hao Su, and Leonidas J. Guibas. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In *NIPS*, 2017.

[17] Charles Ruizhongtai Qi, Hao Su, Matthias Nießner, Angela Dai, Mengyuan Yan, and Leonidas J. Guibas. Volumetric and Multi-view CNNs for Object Classification on 3D Data. *CVPR*, pages 5648–5656, 2016.

[18] Baoguang Shi, Song Bai, Zhichao Zhou, and Xiang Bai. DeepPano: Deep Panoramic Representation for 3-D Shape Recognition. *IEEE Signal Processing Letters*, 22:2339–2343, 2015.

[19] Shuran Song, Fisher Yu, Andy Zeng, Angel Xuan Chang, Manolis Savva, and Thomas A. Funkhouser. Semantic Scene Completion from a Single Depth Image. *CVPR*, pages 190–198, 2017.

[20] Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik G. Learned-Miller. Multi-view Convolutional Neural Networks for 3D Shape Recognition. *ICCV*, pages 945–953, 2015.

[21] Xinlong Wang, Shu Liu, Xiaoyong Shen, Chunhua Shen, and Jiaya Jia. Associatively Segmenting Instances and Semantics in Point Clouds. *CoRR*, abs/1902.09852, 2019.

[22] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic Graph CNN for Learning on Point Clouds. *CoRR*, abs/1801.07829, 2018.

[23] Xiaoqing Ye, Jiamao Li, Hexiao Huang, Liang Du, and Xiaolin Zhang. 3D Recurrent Neural Networks with Context Fusion for Point Cloud Semantic Segmentation. In *ECCV*, 2018.