arXiv:1904.02199v1 [cs.CV] 3 Apr 2019

3D-BEVIS: Birds-Eye-View Instance Segmentation

Technical Report

Cathrin Elich, Francis Engelmann, Jonas Schult, Theodora Kontogianni, Bastian Leibe
RWTH Technical University Aachen
{firstname.lastname}@rwth-aachen.de

Abstract

Recent deep learning models achieve impressive results on 3D scene analysis tasks by operating directly on unstructured point clouds. A lot of progress was made in the field of object classification and semantic segmentation. However, the task of instance segmentation is less explored. In this work, we present 3D-BEVIS, a deep learning framework for 3D semantic instance segmentation on point clouds. Following the idea of previous proposal-free instance segmentation approaches, our model learns a feature embedding and groups the obtained feature space into semantic instances. Current point-based methods scale linearly with the number of points by processing local sub-parts of a scene individually. However, to perform instance segmentation by clustering, globally consistent features are required. Therefore, we propose to combine local point geometry with global context information from an intermediate bird's-eye view representation.

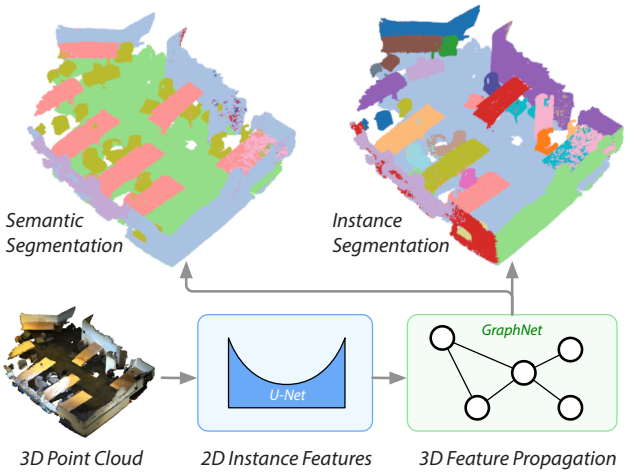

Figure 1. We present a 2D-3D deep model for semantic instance segmentation on 3D point clouds. Bottom: input point cloud and sketch of our architecture. Top: actual predictions from this work.

1. Introduction

The recent progress in deep learning techniques along with the rapid availability of commodity 3D sensors has allowed the community to leverage classical tasks such as semantic segmentation and object detection from the 3D image space into the 3D world. In this work, we tackle the joint tasks of semantic segmentation and instance segmentation of 3D point clouds. Specifically, given a 3D reconstruction of a scene in the form of a raw point cloud, our goal is not only to estimate for each point a semantic label but also to identify every object instance. A number of computer vision applications such as automatic scene parsing, robot navigation and virtual- or augmented-reality can benefit from progress in these challenging tasks.

Currently, there are two main basic concepts to tackle semantic instance segmentation. *Proposal-based* methods look for interesting regions first and then segment the main object in the detected proposal [19, 8, 11]. Alternatively, *proposal-free* approaches learn a feature space for the pixels within the image. The pixels are subsequently grouped into instances according to their feature vector [1, 13, 24]. In this work, we follow the latter direction as only here it is possible to jointly perform semantic- and instance-segmentation for every point int the scene.

There are two fundamental issues that need to be addressed to solve proposal-free instance segmentation on point clouds: First, we need to learn instance features, *i.e.*, a point representation that allows us to group object instances. Although some attempts have been made to learn such instance features for 2D instance segmentation [24, 3, 1], it remains unclear what is the best path for learning instance features on 3D point clouds.

This strongly relates to the second issue, which deals with the scale of point clouds. A typical point cloud can have multiple millions of points along with high dimensional features, including position, color and normals. The usual approach to process large scenes consists of splitting the point cloud into chunks that are processed separately [20, 21, 28]. This is problematic for instance segmentation, as large instances can extend over multiple chunks. The alternative consists in downsampling the original point cloud to a manageable size [26], which leads to obvious drawbacks (*e.g.* loss of details) and can still fail with very large

point clouds, such as in dense outdoor scenes.

In this work, we introduce a hybrid network architecture (see Figure 1) that learns global instance features on a 2D representation of the full scene and then propagates the learned features onto subsets of the full 3D point cloud. In order to achieve this effect, we need a network architecture that supports propagation over unstructured data. Graph neural networks became very popular [27, 16, 22, 29] and are an adequate choice for this purpose. We present results for our model on the Stanford Indoor 3D scenes dataset [2] and the more recent ScanNet dataset [6].

2. Model

In the following, we will present the architecture of our model for semantic instance segmentation on 3D point clouds as visualized in Figure 2. Our model consumes a point cloud $\mathcal{P} = \{x_i\}_{i=1}^N$ *i.e.* a set of points $x_i \in \mathbb{R}^F$ where F is the dimension of the input point features *e.g.* $F = 6$ for XYZ-position and RGB-color. The model predicts semantic labels $\mathcal{L} = \{l_i\}_{i=1}^N$ and instance features $\mathcal{F}^{\text{inst}} = \{f_i\}_{i=1}^N$ with $f_i \in \mathbb{R}^E$, which are grouped to extract the semantic instance labels $\mathcal{I} = \{\mathcal{I}_i\}_{i=1}^N$. The entire framework consists of the three stages, as explained next.

2D Instance Feature Network. To efficiently process the entire scene at once, we consider an intermediate representation $\mathcal{B}$ in the form of a bird's-eye view projection of the point cloud $\mathcal{P}$ (see Figure 3). In contrast to previous methods [28] which independently process small chunks of the full point cloud, we are thereby able to learn instance features, which are globally consistent across the point cloud.The 2D representation $\mathcal{B} \in \mathbb{R}^{H \times W \times F_B}$ is the input to a fully convolutional network (FCN) [25], which predicts the instance feature map $\mathcal{B}' \in \mathbb{R}^{H \times W \times E}$. The FCN can process rooms of changing size during test. We utilizes a simple encoder-decoder architecture inspired by U-Net [23]. There are two output branches, one for semantic segmentation and one for instance segmentation. The corresponding losses are $\mathcal{L}_{\text{inst}}^{2D}$ and $\mathcal{L}_{\text{sem}}^{2D}$. $\mathcal{L}_{sem}^{2D}$ is the cross-entropy loss for semantic segmentation. The instance segmentation loss $\mathcal{L}_{\text{inst}}^{2D}$ is based on a similarity measure for pairs of pixels: $s_{i,j} = \|x_i - x_j\|_2$. From this, we define the entire loss as

$$\mathcal{L}_{\text{inst}}^{2D} = \mathcal{L}_{var} + \mathcal{L}_{dist} \tag{1}$$

with

$$\begin{aligned} \mathcal{L}_{var} &= \sum_{c=1}^{C} \sum_{x_i, x_j \in S_c} s_{i,j}, \\ \mathcal{L}_{dist} &= \sum_{\substack{c,c'=1 \\ c \neq c'}}^{C} \sum_{\substack{x_i \in S_c \\ x_j \in S_{c'}}} [\delta_{dist} - s_{i,j}]_+ \end{aligned} \tag{2}$$

This ensures feature vectors of points belonging to the same object to be similar while encouraging a large distance in the feature space between features corresponding to different instances. To compute the instance loss, we use the same sampling strategy as applied in [9, 18]. Instead of comparing all pairs of feature vectors, we sample a subset S_c containing M pixels for each instance c. The projections $\mathcal{B}$ are precomputed offline. We use color and height-above-ground as input features, thus $F_B = 4$.

3D Feature Propagation Network. At this stage, we have instance features $\mathcal{B}'$ for all the points $\mathcal{P}_{\mathcal{B}} \subset \mathcal{P}$ visible in $\mathcal{B}$. These features are globally consistent and can thus be used as a basis for later grouping. Due to occlusion in the bird's-eye view projection, a fraction of the points was unregarded so far. Therefore, in the next step, we use a graph neural network to propagate existing features and predict instance features for all points in P. Specifically, we concatenate the initial point cloud features x_i with the learned instance features from $\mathcal{B}$ to obtain $\mathcal{P}'$. When generating $\mathcal{B}$, we keep track of point indices to map the learned instance features back to the point cloud $\mathcal{P}$. The instance feature of unseen points in $\mathcal{P} \setminus \mathcal{P}_{\mathcal{B}}$ is set to zero. As graph neural network, we use the architecture from DGCNN [29] which was originally presented for learning a semantic segmentation on point clouds. Similar to the 2D instance feature network, the graph neural network has two output branches, each with an assigned loss function. The semantic segmentation loss $\mathcal{L}_{sem}^{3D}$ is again the cross-entropy loss. The instance segmentation loss $\mathcal{L}_{inst}^{3D}$ is defined as:

$$\mathcal{L}_{inst}^{3D} = \left\| \mathcal{F}^{\text{inst}} - \mathcal{F}^{\text{target}} \right\| \tag{3}$$

where $\mathcal{F}^{\text{target}} \in \mathbb{R}^{N \times E}$ are *target instance features*. The target instance feature for a point x_i is the mean over all instance features in $\mathcal{B}'$ which lie in the same ground truth instance $\mathcal{I}_j$.

Instance Grouping. The last component obtains the final instance labels $\mathcal{I}$ by clustering the predicted instance features $\mathcal{F}^{\text{inst}}$ using the MeanShift [4] algorithm. MeanShift does not require a pre-determined number of clusters and is thus suited for the task of instance segmentation with an arbitrary number of instances. As a final post-processing step, we found it beneficial to split up instances with an inconsistent semantic labeling. This helps to distinguish between objects from different classes that are hardly identified from the bird's-eye view like windows and walls.

3. Experiments

We evaluate our method using two benchmark datasets on which we conduct experiments on the task of semantic- and instance-segmentation. We show qualitative and quantitative results on both tasks.

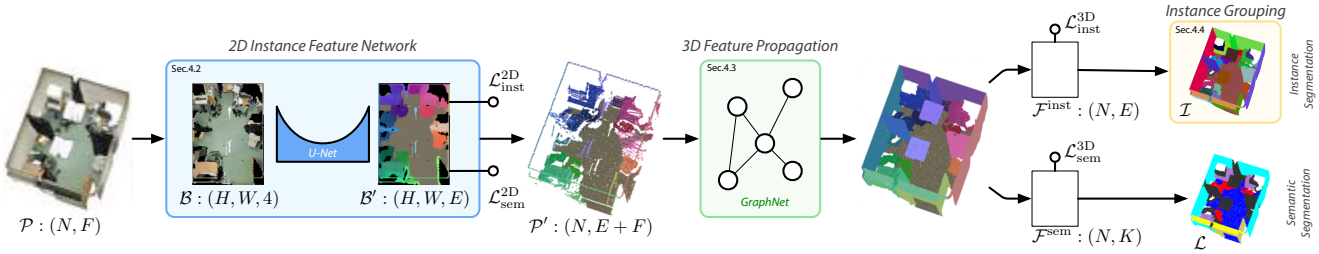

Figure 2. **Our model** consumes a point cloud $\mathcal{P}$ in order to predict instance labels $\mathcal{I}$ and semantic labels $\mathcal{L}$. The 2D instance feature network predicts instance features $\mathcal{B}'$ from a bird's-eye-view $\mathcal{B}$ of the scene. After concatenating the instance features to the original point cloud features, a graph neural network propagates and predicts instance features for all points in the scene. Our model finally predicts semantic labels $\mathcal{L}$ and instance features $\mathcal{F}^{\text{inst}}$, which are clustered to instance labels $\mathcal{I}$.

	ScanNet [6]					**S3DIS** [2]				
	AP	AP 50%	AP 25%	mIoU	mAcc	AP	AP 50%	AP 25%	mIoU	mAcc
PointNet [20]+Conn.Comp.	3.9	9.7	25.9	20.8	45.3	10.1	19.9	34.9	39.5	60.0
SGPN [28]	3.2	7.8	19.9	28.0	50.9	10.1	20.3	31.6	41.8	59.7
DGCNN [29] + Conn.Comp.	8.8	18.3	36.1	34.6	64.8	16.5	29.8	42.6	48.0	**69.9**
SGPN$_{(\text{DGCNN})}$	11.2	20.8	34.6	43.9	**69.3**	18.7	29.7	40.3	53.0	67.3
Ours (3D-BEVIS)	**16.8**	**31.5**	**44.6**	**49.9**	60.5	**21.8**	**33.3**	**44.7**	**58.4**	68.9

Table 1. **Summary Table.** Instance and semantic segmentation results on the ScanNet [6] dataset and the S3DIS [2] dataset. In this table, we compare methods that jointly predict semantic labels and instance labels. On both ScanNet and S3DIS, our presented method yields the best results for instance segmentation compared to SGPN and our connected-components baseline.

Input: $\mathcal{B}$ GT instance labels Output: $\mathcal{B}'$

Figure 3. Left to Right: Input bird's-eye view $\mathcal{B}$, ground truth instance labels, predicted instance features $\mathcal{B}'$ colored according to the GT instance labels. For visualization, we project the E-dimensional instance features $\mathcal{B}'$ to 2D with PCA.

	AP	AP 50%	AP 25%
ResNet-backbone [14]	26.2	45.9	69.5
MASC [15]	25.4	44.7	61.5
PanopticFusion-inst [17]	21.4	47.8	69.3
UNet-backbone [14]	16.1	31.9	60.5
3D-SIS [12]	16.1	38.2	55.8
GSPN [30]	15.8	30.6	54.4
PMRCC	2.1	5.3	22.7
SGPN [28]	4.9	14.3	39.0
Our method	11.0	22.5	35.0

Table 2. **ScanNet v2 Benchmark Challenge.** Scores from [7].

Stanford Large-Scale 3D Indoor Spaces (S3DIS) [2]. contains dense 3D point clouds from 6 large-scale indoor areas consisting of 271 rooms from 3 different buildings. The points are annotated with 13 semantic classes and grouped into instances. We follow the usual 6-fold cross validation strategy for training and testing as used in [2, 20].

ScanNet v2 [6]. contains 3D meshes of a wide variety of indoor scenes including apartments, hotels, conference rooms and offices. The dataset contains 20 semantic classes. We use the public training, validation and test split of 1201, 312 and 100 scans, respectively.

Metrics. For semantic segmentation, we adopt the predominant metrics from the field: intersection over union and overall accuracy. To report scores on instance segmentation we follow the ScanNet [6] benchmark evaluation metric, which is adapted from the CityScapes [5] evaluation. We report the mean average precision (AP [10]) and AP with overlap of 50 % and AP with 25 % overlap.

Baselines. We compare our method to **SGPN** [28], the only published work so far in the field of semantic instance segmentation operating directly on point clouds. SGPN uses PointNet [20] as initial feature extraction network. We conducted an additional baseline experiment **SGPN$_{\text{DGCNN}}$**

which replaces PointNet by DGCNN [29].We used the source code provided by the authors of [28], although it required some modifications to run and reproduce the numbers provided in their paper.

Further, we present a naive baseline **Conn.Comp.** where we conduct a connected component analysis on a semantically labeled point cloud. We use the semantic labels from PointNet and DGCNN. Two points are connected in the same instance if they have the same semantic label and their relative distance is below a threshold $\tau_d = 6\,\text{cm}$. We set the minimal number of points in an instance to $n_{\text{min}} = 50$ points. A similar baseline is presented in [28] called SegCluster.

Main Results. We present quantitative and qualitative results for semantic instance segmentation. Table 1 summarizes all our results. Our method outperforms all baselines and SGPN on both datasets. We see that DGCNN is a powerful method, it can help to significantly improve existing approaches.Combined with Conn.Comp. it even outperforms the original SGPN. Please note that our scores differ from the ones reported in SGPN [28] as we use the stricter ScanNet metric for all evaluations. Specifically, this metric penalizes wrong semantic labels even if the instance labels are predicted correctly. This is not the case in [28]. Also, we evaluate on all semantic classes including 'clutter' and 'other furniture'.

In Table 2, we report our scores on the ScanNet v2 benchmark instance segmentation challenge. We get decent results compared to our baseline SGPN. Other recently submitted scores are included as well.

We show qualitative results of our method and SGPN [28] for instance and semantic segmentation on ScanNet [6] in Figure 4 and S3DIS [2] in Figure 5.

4. Discussion

The bird's-eye view used in this work has proven to be very powerful to compute globally consistent features. However, there are intrinsic limitations *e.g.* vertically oriented objects are not well visible in this 2D representation. The same is true for scenes including numerous occluded objects. An obvious extension could be to include multiple 2D views of the scene. Compared to previous work [28], our model is able to learn global instance features which are consistent over a full scene. Thus, the presented method overcomes the necessity for a heuristic post-processing step to merge instances.

5. Conclusion

In this work, we explored the relatively new field of instance segmentation on 3D point clouds. We have proposed a 2D-3D deep learning framework combining a U-shaped fully convolution network to learn globally consistent instance features from a bird's-eye view in combination with a graph neural network to propagate and predict point features in the 3D point cloud. Future work could look at alternative 2D representations to overcome the limitations of the bird's-eye view.

References

[1] N. Alejandro, H. Zhiao, and D. Jia. Associative Embedding: End-to-end Learning for Joint Detection and Grouping. In *Neural Information Processing Systems (NIPS)*, 2017.

[2] I. Armeni, O. Sener, A. R. Zamir, H. Jiang, I. Brilakis, M. Fischer, and S. Savarese. 3D Semantic Parsing of Large-Scale Indoor Spaces. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[3] B. D. Brabandere, D. Neven, and L. V. Gool. Semantic Instance Segmentation with a Discriminative Loss Function. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017.

[4] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002.

[5] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The Cityscapes Dataset for Semantic Urban Scene Understanding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[6] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner. ScanNet: Richly-annotated 3D Reconstructions of Indoor Scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[7] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner. Scannet benchmark challenge - 3d semantic instance benchmark. `http://kaldir.vc.in.tum.de/scannet_benchmark/semantic_instance_3d`, 2018. [Online; accessed 02-April-2019].

[8] J. Dai, K. He, and J. Sun. Instance-aware Semantic Segmentation via Multi-task Network Cascades. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[9] A. Fathi, Z. Wojna, V. Rathod, P. Wang, H. O. Song, S. Guadarrama, and K. P. Murphy. Semantic Instance Segmentation via Deep Metric Learning. *CoRR*, abs/1703.10277, 2017.

[10] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Simultaneous Detection and Segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014.

[11] K. He, G. Gkioxari, P. Dollar, and R. B. Girshick. Mask R-CNN. In *International Conference on Computer Vision (ICCV)*, 2017.

[12] J. Hou, A. Dai, and M. Nießner. 3d-sis: 3d semantic instance segmentation of RGB-D scans. *CoRR*, abs/1812.07003, 2018.

[13] Y.-C. Hsu, Z. Xu, Z. Kira, and J. Huang. Learning to Cluster for Proposal-Free Instance Segmentation. *International Conference on Neural Networks (IJCNN)*, 2018.

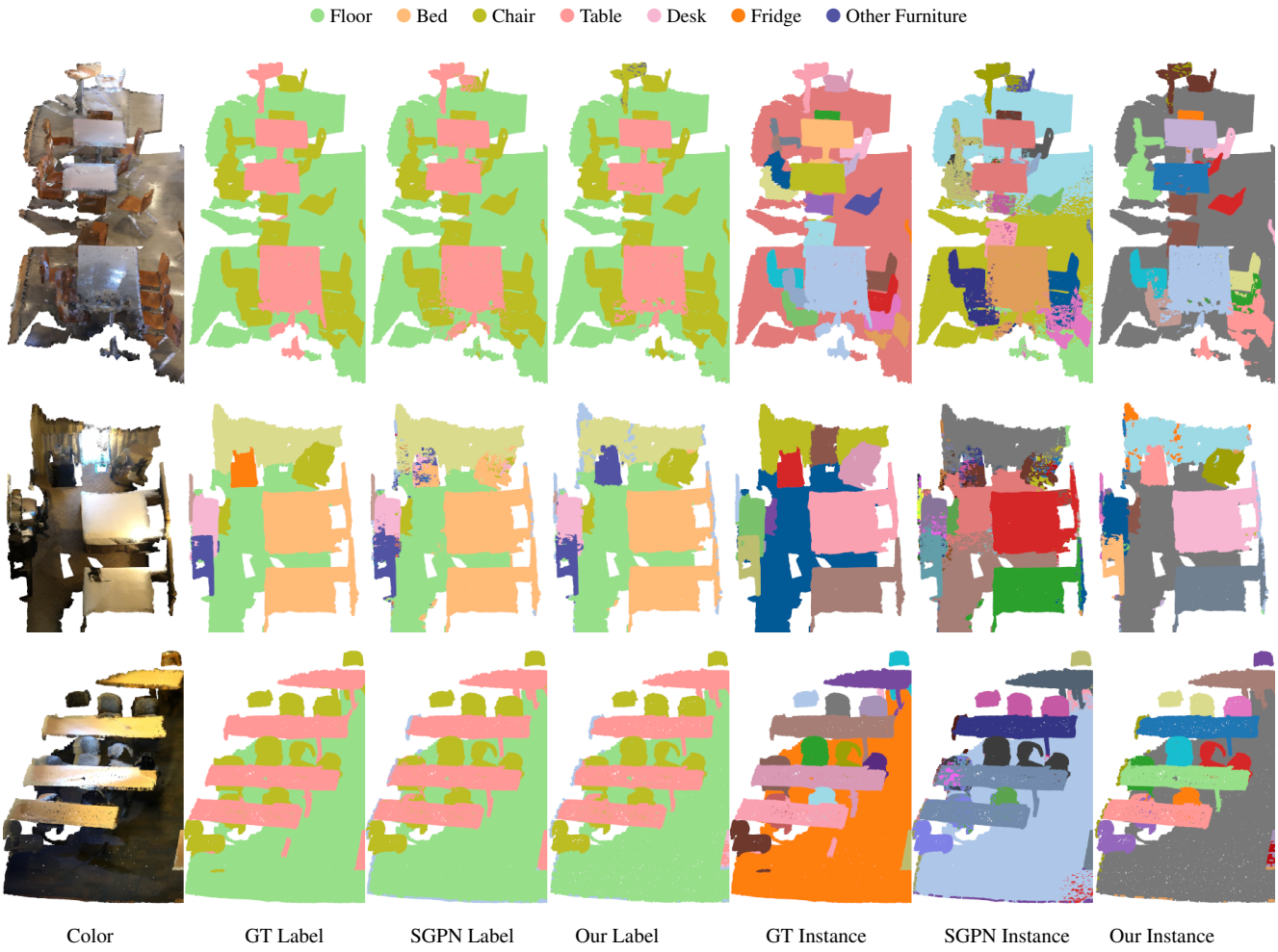

Figure 4. **Qualitative results on ScanNet [6].** We compare our method to SGPN for instance and semantic label prediction. For semantic segmentation, both methods produce visually similar results. On instance segmentation merging problems for SGPN become obvious for large instances such as floor. This could be due to the block merging algorithm which has problems combining large instances that extend over multiple blocks. Our method produces globally consistent instance segmentations. Still, our method is not always able to separate neighboring instances such as the chairs in the above examples.

[14] Z. Liang, M. Yang, and C. Wang. 3d graph embedding learning with a structure-aware loss function for point cloud semantic instance segmentation. *CoRR*, abs/1902.05247, 2019.

[15] C. Liu and Y. Furukawa. MASC: multi-scale affinity with sparse convolution for 3d instance segmentation. *CoRR*, abs/1902.04478, 2019.

[16] P. V. M Defferrard, X Bresson. Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. In *Neural Information Processing Systems (NIPS)*, 2016.

[17] G. Narita, T. Seno, T. Ishikawa, and Y. Kaji. Panopticfusion: Online volumetric semantic mapping at the level of stuff and things. volume abs/1903.01177, 2019.

[18] A. Newell and J. Deng. Pixels to Graphs by Associative Embedding. In *Neural Information Processing Systems (NIPS)*, 2017.

[19] P. O. Pinheiro, R. Collobert, and P. Dollar. Learning to Segment Object Candidates. In *Neural Information Processing Systems (NIPS)*, 2015.

[20] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[21] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In *Neural Information Processing Systems (NIPS)*, 2017.

[22] X. Qi, R. Liao, J. Jia, S. Fidler, and R. Urtasun. 3D Graph Neural Networks for RGBD Semantic Segmentation. In *International Conference on Computer Vision (ICCV)*, 2017.

[23] O. Ronneberger, P.Fischer, and T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015.

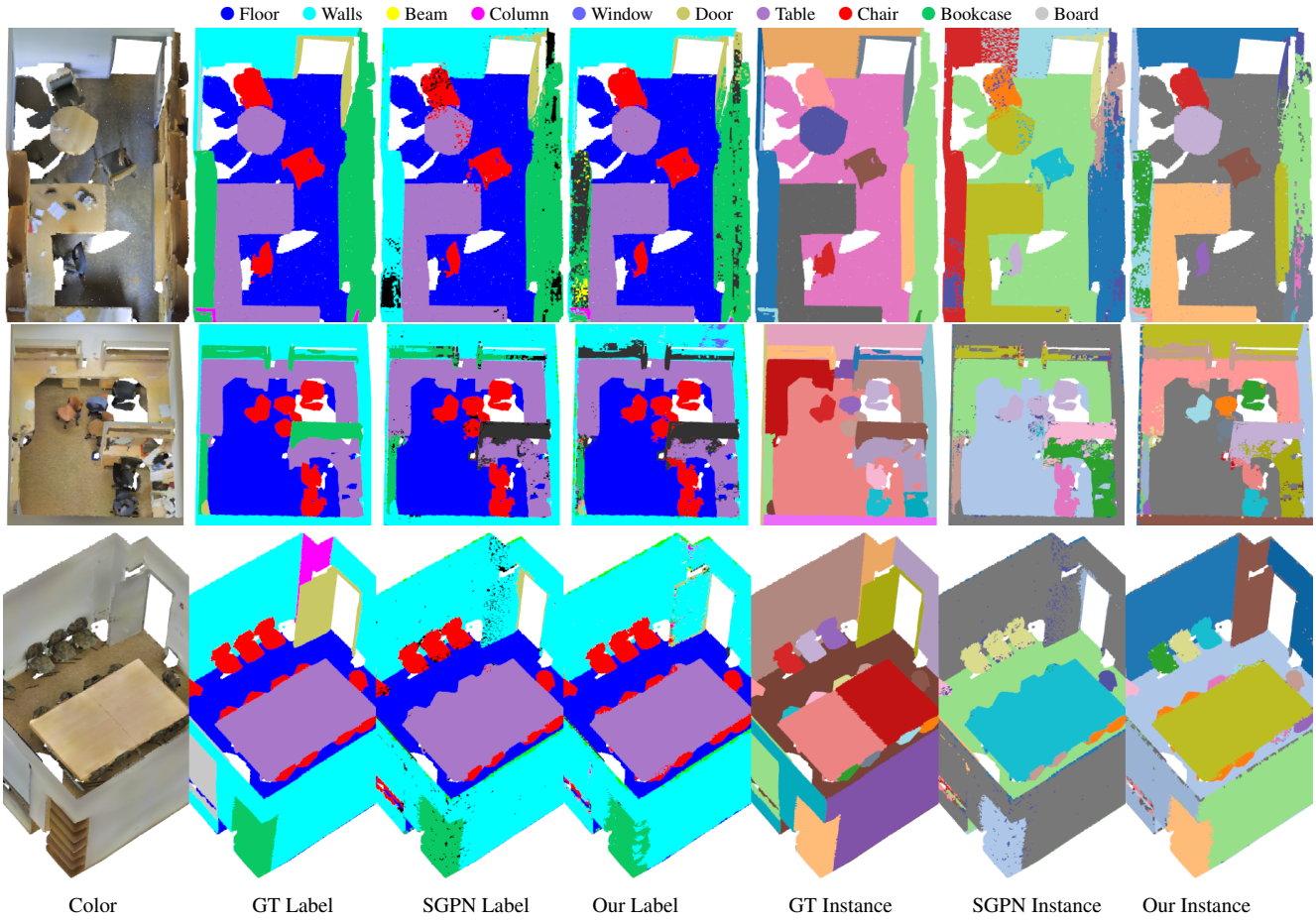

Figure 5. **Qualitative results on S3DIS [6].** We compare our method to SGPN for instance and semantic label prediction. For semantic segmentation, both methods predict reasonable results. They both have problems identifying the door and the column in the lower example. On instance segmentation both methods have difficulties distinguishing between the two tables next to each other. Our method performs better on identifying multiple instances of chairs. On the upper example, block artifacts become apparent from SGPN's merging step.

[24] C. F. S. Kong. Recurrent Pixel Embedding for Instance Grouping. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[25] E. Shelhamer, J. Long, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, (PAMI)*, 2017.

[26] M. Tatarchenko, J. Park, V. Koltun, and Q.-Y. Zhou. Tangent Convolutions for Dense Prediction in 3D. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[27] K. Thomas and W. Max. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations, (ICLR)*, 2017.

[28] W. Wang, R. Yu, Q. Huang, and U. Neumann. SGPN: Similarity Group Proposal Network for 3D Point Cloud Instance Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[29] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon. Dynamic Graph CNN for Learning on Point Clouds. *Computing Research Repository CoRR*, abs/1801.07829, 2018.

[30] L. Yi, W. Zhao, H. Wang, M. Sung, and L. J. Guibas. GSPN: generative shape proposal network for 3d instance segmentation in point cloud. *CoRR*, abs/1812.03320, 2018.