# PlaneSegNet: Fast and Robust Plane Estimation Using a Single-stage Instance Segmentation CNN

Yaxu Xie Jason Rambach Fangwen Shu Didier Stricker

*Abstract*— **Instance segmentation of planar regions in indoor scenes benefits visual SLAM and other applications such as augmented reality (AR) where scene understanding is required. Existing methods built upon two-stage frameworks show satisfactory accuracy but are limited by low frame rates. In this work, we propose a real-time deep neural architecture that estimates piece-wise planar regions from a single RGB image. Our model employs a variant of a fast single-stage CNN architecture to segment plane instances. Considering the particularity of the target detected, we propose Fast Feature Non-maximum Suppression (FF-NMS) to reduce the suppression errors resulted from overlapping bounding boxes of planes. We also utilize a Residual Feature Augmentation module in the Feature Pyramid Network (FPN) . Our method achieves significantly higher frame-rates and comparable segmentation accuracy against two-stage methods. We automatically label over 70,000 images as ground truth from the Stanford 2D-3D-Semantics dataset. Moreover, we incorporate our method with a state-of-the-art planar SLAM and validate its benefits.**

## I. Introduction

Detection of 3D geometry features in scenes supports tasks such as 3D scene understanding, robot navigation and Simultaneous Localization and Mapping (SLAM). Planes, as one of the most fundamental geometry features, widely present in most man-made scenes. In indoor applications, piece-wise plane estimation benefits building modeling [33], visual SLAM [24] and robot navigation [23]. Planar information is also valuable for mobile Augmented Reality (AR) applications. In outdoor urban environments, plane estimation is used for object level reconstructing of buildings [15] and 6-DoF pose estimation of object [25].

Geometry model based plane estimation algorithms have been extensively studied for many years. Some approaches use Random Sample Consensus (RANSAC) [11] or Hough Transformation [32] achieving solid results and near real-time running speed. However, such techniques consume large computational resources when dealing with dense inputs (point cloud or depth image), and require additional sensors for depth estimation. Their run-time and segmentation precision are also highly correlated with the complexity of the scene. Nevertheless, geometry model based plane estimation methods are indispensable for generating the ground truth, which is needed for training machine learning approaches.

Instance segmentation is the task of detecting and delineating each distinct object of interest appearing in an image. The problem of piece-wise planar region estimation from a single RGB image can be reverted to a binary-class (planar and non-planar) instance segmentation problem. In recent years, great progress has been made in instance segmentation with the help of convolutional neural networks (CNN). State-of-the-art two-stage approaches like FCIS [16] and Mask R-CNN [7] depend on feature localization to produce masks of the objects. PlaneRCNN [18] is a multi-task plane estimation model, whose detection network is built upon Mask R-CNN. It inherits not only the high accuracy advantage of the latter, but also its run-time limitations.

In this paper, we present a novel single-stage instance segmentation architecture for piece-wise planar regions estimation which reaches significantly higher frame-rates and improved segmentation accuracy against two-stage methods. In detail our contributions in this paper are:

- We propose PlaneSegNet, the first real-time single-stage detector for piece-wise plane segmentation. PlaneSegNet is shown in our experiments to outperform the state-of-the-art both in segmentation accuracy and run-time.
- We improve the localization and segmentation accuracy of the network by enhancing spatial context features and introducing Fast Feature Non-maximum Suppression (FF-NMS).
- We annotate with piece-wise plane masks and make available 70,000 images from the dataset Stanford 2D-3D-Semantics [1] using an NDT-RANSAC method.

The rest of the paper is organized as follows: First we discuss related work in Section II. Our proposed network architecture, specific improvements and ground truth generation method are discussed in Section III. The evaluation of the approach (quantitative and qualitative comparison, testing with Planar SLAM) is given in Section IV. Finally, we give concluding remarks in Section V.

## II. Related Work

In this section, we summarize the relevant research done on deep learning method for piece-wise planar reconstruction and SLAM system using semantic planar regions as cues.

### A. Piece-wise Plane Segmentation

PlaneNet [19] is the first end-to-end neural network for piece-wise planar reconstruction from a single RGB image. It is built upon Dilated Residual Networks (DRNs) [36] and has three prediction branches: plane parameters estimation, non-planar depth map estimation and plane segmentation. The segmentation branch starts with a pyramid pooling module followed by CRFasRNN [39] layers to refine the prediction.

DFKI - German Research Center for Artificial Intelligence, Kaiserslautern, Germany. E-mail: {first_name}.{last_name}@dfki.de

**Acknowledgment:** This research has been partially funded by the German BMBF projects MOVEON (01IS20077) and RACKET (01IW20009).

In PlaneRecover [34], Yang and Zhou discuss the difficulty of obtaining ground truth plane annotations in real dataset. Instead, they utilize a synthetic dataset [26] of urban scenes and introduce a novel plane structure-induced loss to train the network without direct supervision. However, both PlaneNet and PlaneRecover require the maximum number of planes in an image as prior. Instead of using a fully CNN network, Z. Yu et al. [37] utilize an encoder-decoder architecture to first perform semantic segmentation, and then use associative embedding model with mean shift clustering method to further segment the planar region into piece-wise plane instance.

PlaneRCNN [18] proposes a more effective plane segmentation branch built upon Mask R-CNN [7]. The network shows high generalization ability across both indoor and outdoor scenes, but fails to reach real-time frame rate. Mask R-CNN based methods generate candidate region-of-interests (ROIs) and segment them sequentially. Therefore, the run-time is strongly influenced by scene complexity and object size, which further harms the frame rate robustness in real-time applications.

### B. Planar SLAM

Most of the existing methods of SLAM systems are based only on points to describe the scenes and estimate the camera poses, which encounter various problems in practical application such as low-texture environments and changing light. Semantic cues are added in SLAM as geometric regularization regarding to different landmarks and optimize the geometric structure jointly, such as the distance between plane and associated 3D points, or the perpendicular plane layout of an indoor scene (walls and floor) under Manhattan assumption [3]. An early work [31] presented a RGB-D SLAM system for hand-held 3D sensor using both point and plane as primitives. Followed by the works [27], [12], [10], [9], [38] which tackle the problem similarly by extracting plane from depth image and optimize the poses of keyframes and landmarks (point and plane) in Bundle Adjustment (BA). More recently, [35] employs high-level object and plane landmarks with Monocular ORB-SLAM2 [22], the built map is dense, compact and semantically meaningful compared to the classic feature-based SLAM. Similarly, SlamCraft [24] presented an efficient planar monocular SLAM which fuses the detected plane from PlaneNet iteratively with the point cloud, resulting in higher accuracy of camera localization.

## III. METHODOLOGY

In this section, we describe our proposed network architecture, the motivation behind our design choices, and the steps we take to reduce inference run-time.

### A. PlaneSegNet Overview

Our proposed PlaneSegNet is build upon the YOLACT++ [2] instance segmentation network with several modules optimized for the task at hand (see Figure 1 for an illustration). A ResNet [8] backbone with Feature Pyramid Network (FPN) serves as the encoder and provides a set of multi-scale feature maps $\{P_3, P_4, P_5, P_6, P_7\}$. Layer $C_5$ of the backbone is connected to the prediction layer $P_5$ of the FPN through a regular lateral path and an extra Residual Feature Augmentation path (see Sec. III-C) to enhance spatial context information. The Protonet is a set of fully convolutional layers, which predicts $k$ channels for instance independent prototype masks from the feature maps $P_3$. The prediction heads provide 4 bounding box regressors, $c$ class confidences and $k$ mask coefficients for $k$ prototype masks from every level of the pyramid feature maps $\{P_3, P_4, P_5, P_6, P_7\}$. We propose the Fast Feature NMS (see Sec. III-D) instead of the Fast NMS to improve the detection robustness on overlapping instances. Finally we assemble those prototype masks with coefficients prediction with a linear combination of both followed by a sigmoid non-linearity function:

$$M = \sigma(P\boldsymbol{C}^T) \tag{1}$$

where $P$ is an $h \times w \times k$ tensor of prototype masks and $\boldsymbol{C}$ is a $n \times k$ matrix of mask coefficients for $n$ instances surviving from Fast Feature NMS (see Sec. III-D) and score thresholding.

The loss function is defined as a weighted sum of the localization loss (bounding box loss), the confidence loss and the mask loss as

$$\begin{aligned}\mathscr{L}(x,C,B,M) = \frac{1}{N}(\mathscr{L}_{conf}(x,C) + \alpha\mathscr{L}_{loc}(x,B,B_{gt}) \\ +\beta\mathscr{L}_{mask}(x,M,M_{gt}))\end{aligned} \tag{2}$$

where $N$ is the number of positive matches, and $C, B, M$ represent confidence, bounding box and mask, respectively. $\mathscr{L}_{confs}$ and $\mathscr{L}_{loc}$ are the same as the loss functions of a single-shot detector [20]. $\mathscr{L}_{mask}$ is the pixel-wise binary cross entropy between predicted masks and the ground truth masks.

### B. Piece-wise Planes as Instance Segmentation

When we consider piece-wise planar regions as instances in scenes, our segmentation target shows particularities different from common instance segmentation tasks. In Table I, we analysed the bounding box overlapping frequency (the percentage of frames with overlapping objects through the data sample), bounding box IoU distribution and the instance size distribution of COCO [17] (a common multi-class instance segmentation) with the piece-wise plane annotations of ScanNet (given by [18]) and Stanford 2D-3D-S (labeled by us). We observed the fact that the bounding boxes of plane instances overlap in most of the frames in indoor datasets, which is much more common than that of object instances from COCO (only 0.015%). We also found out that instances with a mask area greater than 10% of the frame size appear more frequently in the plane annotations of ScanNet [4] and 2D-3D-S [1]. It is also intuitively obvious that planes in indoor scenes appear as background room structures and surfaces of foreground objects, which leads to dense and nested spatial distribution of their bounding boxes. Therefore, we are motivated to improve our approach from two aspects:

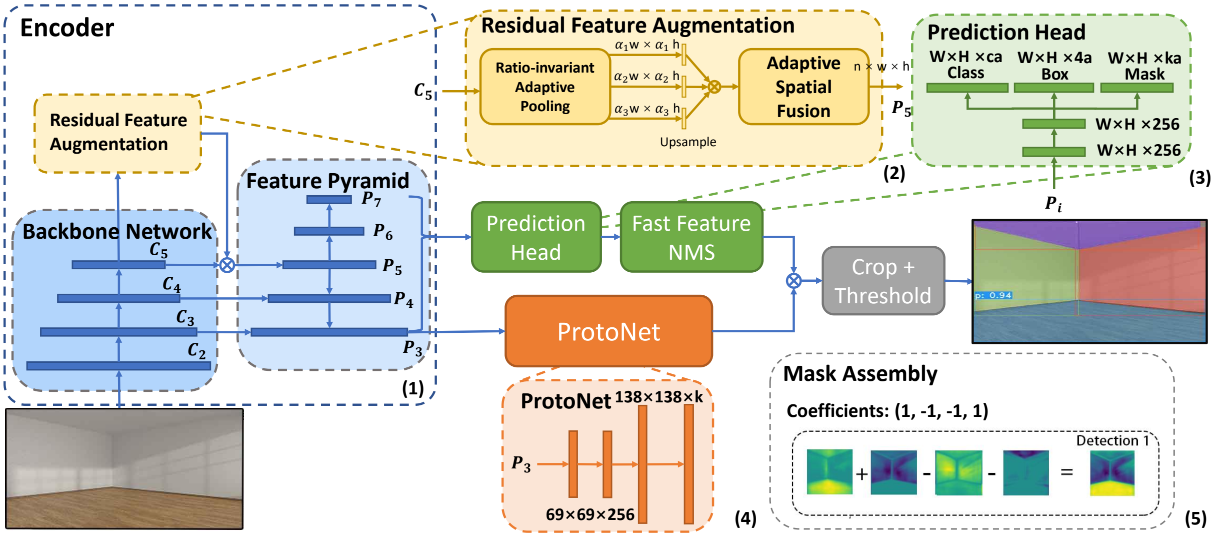


Fig. 1: **PlaneSegNet Architecture: (1) The Encoder:** a ResNet backbone combined with a Feature Pyramid Network. **(2) Residual Feature Augmentation:** an extra path from $C_5$ to $P_5$ to enhance spatial context. **(3) Prediction Heads:** provides classes, boxes and mask coefficients' prediction. **Fast Feature NMS** suppresses redundant proposals. **(4) ProtoNet:** predicts $k$ instance independent prototype masks. **(5) Assembly:** the instance masks come from a linear combination of the prototype masks and the mask coefficients.

| Dataset | Overlap(%) | IoU(25-50%) | IoU(≥50%) | Large-Obj.(%) |
|---|---|---|---|---|
| COCO | 0.015% | 9.01% | 4.55% | 12.27% |
| ScanNet | 69.01% | 10.72% | 2.94% | 23.41% |
| 2D-3D-S | 80.08% | 18.11% | 7.18% | 32.84% |

TABLE I: **Instance Feature Comparison** for COCO (multi-class annotations), Stanford 2D-3D-S (plane annotations) and ScanNet (plane annotations), using 2,000 random samples of each. Metrics: Overlap(%) - percentage of frames with overlapping bounding boxes in the sample dataset, IoU in range of $(a\%, b\%)$ - percentage of overlaps with a IoU within $(a\%, b\%)$ among all overlapping bounding boxes in the sample, Large-Obj.(%) - percentage of large instance (whose area is greater than 10% of the frame size) among all instances in the sample.

- Optimize the mask quality of large size instances by compensation for spatial information loss on prediction layers. As accurate segmentation of dominant planar regions benefits our primary targeted application, visual SLAM.
- Decrease the false suppression on overlapping bounding boxes of different instances by introducing multi-steps NMS method (Fast Feature NMS). This target shares similarity with dense pedestrian segmentation.

### C. Residual Feature Augmentation

To improve the mask quality of large size instances, we combine the low-resolution, semantically strong features with the high-resolution, semantically weak features from different depths of the backbone network via a top-down pathway and lateral connections within the Feature Pyramid Network (FPN). This feature pyramid enriches the semantics at all levels without sacrificing representational speed or memory.

Nevertheless the feature channels are reduced to 256-D by applying a convolution when building lateral connections, which causes information loss at the highest pyramid level. Furthermore, as the layer of the top down path way, $P_5$ only contains single scale context information. We utilize the Residual Feature Augmentation based on [6] to build an extra connection from highest feature layer to its correspondent prediction layer, so that the lost information can be compensated by incorporating the spatial context information into $P_5$. The structural details of the Residual Feature Augmentation are illustrated in Figure 1 (2).

We first perform ratio-invariant adaptive pooling on $C_5$ to obtain multiple context features with different scales of $(\alpha_1 \times S, \alpha_2 \times S, .., \alpha_n \times S)$, which in our case $(\alpha_1, \alpha_2, \alpha_3) = (0.1, 0.2, 0.3)$. Each context feature will then independently pass through a convolution layer to reduce the channel dimension to 256. After that, we upsample these features to the same scale as $C_5$ through bilinear interpolation and fuse them subsequently.

The subsequent Adaptive Spatial Fusion (ASF) module combines these upsampled context features and produces spatial weight maps for each of them. Different to the original design [6], we decrease the channel of spatial weight branch to its half to accelerate the network. The weights are used to generate a set of residual feature maps, which contains multi-scale context information. Subsequently, we combine them with the feature map from top-down lateral layers by summation, and perform a $3 \times 3$ convolution to build the beginning $(P_5)$ of prediction layers $\{P_3, P_4, P_5, P_6, P_7\}$.

### D. Fast Feature Non-maximum Suppression

For piece-wise plane instance in indoor scenario, bounding boxes with high overlap ($IoU \geq 50\%$) do not necessarily belong to the same object in our scenario. Therefore, a classic NMS method is not optimal to solve the redundant proposals from the prediction head.

Addressing this issue, we introduce a novel method, Fast Feature Non-maximum Suppression (see Algorithm 1), which is inspired by [28]. Similar as Fast NMS, a $c \times n \times n$ pairwise IoU matrix $X$ is computed for the top $n$ detections sorted descending by score for each of $c$ classes (in our case, $c = 1,\ n = 200$). The column-wise maximum $K$ is computed from the upper triangle matrix $X^{triu}$.

**Algorithm 1:** Fast Feature NMS

```
**input:** $P \leftarrow Sort(Proposals)$ with Scores, $D \leftarrow \varnothing$;
$X^{triu} \leftarrow GetPairwiseIoU(P)$;
$K \leftarrow \max(X^{triu})$ column-wise;
if $K_i \leq N_1$ then
    $PUSH(p_i, D)$;
else
    if $K_i \leq N_2$ then
        $C^{triu} \leftarrow GetCosineSim(p, D)$;
        $S \leftarrow \max(C^{triu})$ column-wise;
        if $S_i \leq T$ then
            $PUSH(p_i, D)$
end
return $D$
```

If the maximal IoU value is less or equal than threshold $N_1$, the detection is considered to belong to different objects, and when the IoU is larger than threshold $N_2$ the detection is considered to belong to the same object. In our network the final mask segmentation is assembled from the prototype masks with mask coefficients prediction. Therefore, we compute the cosine similarity matrix $S$ of mask coefficients vectors between the detections $p$ whose $IoU \in (N_1, N_2)$ with marked detections $d$. We set a similarity threshold $T_1$ to indicate which highly overlapping detection to keep for each class. The algorithm is very efficient since it only involves matrix operation. Compared to Fast NMS, our method introduce an overhead less than 1 ms.

## IV. Experiments and Results

There are three different experiments conducted in our work: first we compare our PlaneSegNet with other state-of-the-art quantitatively and qualitatively. Second we ablate our added components in terms of their performance and the trade-off between run-time and accuracy. Third we present an application of our network combined with a monocular planar SLAM system.

### A. Benchmarking and Experimental Setup

We present a new benchmark from RGB-D image on dataset 2D-3D-S [1] with improved RANSAC method based on Normal Distribution Transformation (NDT) [21], following the work of Li et al. [14]. Comparing to simple RANSAC, the approach shows better performance as illustrated in Figure 2. The main reasons are: the method considers both the normal direction and the location of NDT cells while estimating planes, in order to minimize incorrect fitting on step-wise object, and non-planar cells are filtered out with a threshold before RANSAC. Thus, we avoid hard-fitting on curve surface.

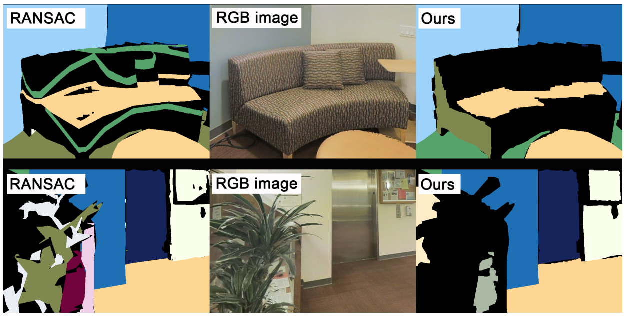


Fig. 2: **Comparison of ground truth generation of the applied NDT-RANSAC against simple RANSAC:** in the first column, the example (top row) shows the simple RANSAC failed due to the absence of normal direction of points, or (bottom row) failed by attempting to fit planes in curve surface. Raw RGB images are extracted from 2D-3D-S [1].

We implemented our PlaneSegNet using PyTorch, and use ResNet-101 [8] as backbone with ImageNet [5] pre-trained weights [8]. Newly added layers are initialized as random filters sampled from a normal distribution with zero mean and variance. We trained our network end-to-end on a NVIDIA RTX 3090 for 15 epochs with Adam [13] optimizer on the dataset 2D-3D-S [1] with piece-wise plane annotations, which consists of about 70,000 images. For comparison purposes we also trained a model of our network on ScanNet [4] (70,000 image samples, with the annotation provided by Liu et al. [18]).

Considering a major application of our work is visual-SLAM, we additionally utilize motion blur and Gaussian noise as data augmentation methods, to enhance our network's robustness against noisy input images.

### B. Instance Planar Segmentation Results

In Table II, we compare our approach to two piece-wise plane reconstruction networks: PlaneNet [19] and PlaneR-CNN [18]. We test on two dataset previously not used for this purpose, namely TUM RGB-D [30] (average 2.92 planes per image) and the NYU-V2 labeled subset [29] (average 11.63 planes per image) with the plane segmentation ground truth generated by us. In order to exclude the influence of different training datasets, we also give the result of our PlaneSegNet trained on ScanNet with the annotation given by Liu et al. [18]. We evaluate the run-time in terms of frame per second (FPS), the detection using bounding box average precision ($AP^b$) and the segmentation performance using mask average precision ($AP^m$), VOI, RI and SC. The frame rate of all

|  | Backbone | Training | FPS | TUM RGB-D [30] |  |  |  |  | NYU-V2 [29] |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
|  |  |  |  | $AP^b_{50}$ | $AP^m_{50}$ | $VOI\downarrow$ | $RI$ | $SC$ | $AP^b_{50}$ | $AP^m_{50}$ | $VOI\downarrow$ | $RI$ | $SC$ |
| **PlaneRCNN** [18] | ResNet101 | ScanNet | 3.0 | 37.13 | 31.93 | **1.502** | 0.746 | **0.652** | 20.02 | 18.89 | 2.861 | 0.724 | 0.458 |
| **PlaneSegNet-101** | ResNet101 | ScanNet | **30.2** | 34.82 | 33.80 | 1.759 | 0.733 | 0.602 | 26.36 | 21.69 | 2.794 | 0.768 | 0.481 |
| **PlaneSegNet-101** | ResNet101 | 2D-3D-S | **30.2** | **43.97** | **40.52** | 1.645 | **0.748** | 0.628 | **32.78** | **21.74** | **2.524** | **0.774** | **0.521** |
| **Baseline (Yolact++)** | ResNet101 | 2D-3D-S | 31.5 | 41.62 | 37.64 | 1.724 | 0.734 | 0.607 | 32.02 | 20.83 | 2.591 | 0.766 | 0.506 |
| **Baseline + FF-NMS** | ResNet101 | 2D-3D-S | 31.4 | 41.80 | 37.81 | 1.708 | 0.737 | 0.612 | 32.39 | 20.86 | 2.573 | 0.769 | 0.511 |
| **Baseline + Aug-FPN** | ResNet101 | 2D-3D-S | 30.7 | 43.72 | 40.33 | 1.673 | 0.744 | 0.620 | 32.14 | 21.55 | 2.557 | 0.770 | 0.513 |

TABLE II: **Quantitative comparison of our approach against other state-of-the-art methods and ablation study of our contributions** on dataset TUM RGB-D [30] (about 1,300 samples), dataset NYU V2 [29] evaluated in terms of mask Average Precision ($AP^m$), bounding box Average Precision ($AP^b$), Variation of Information (VOI), Rand Index (RI), Segmentation Covering (SC) and run-time (FPS).

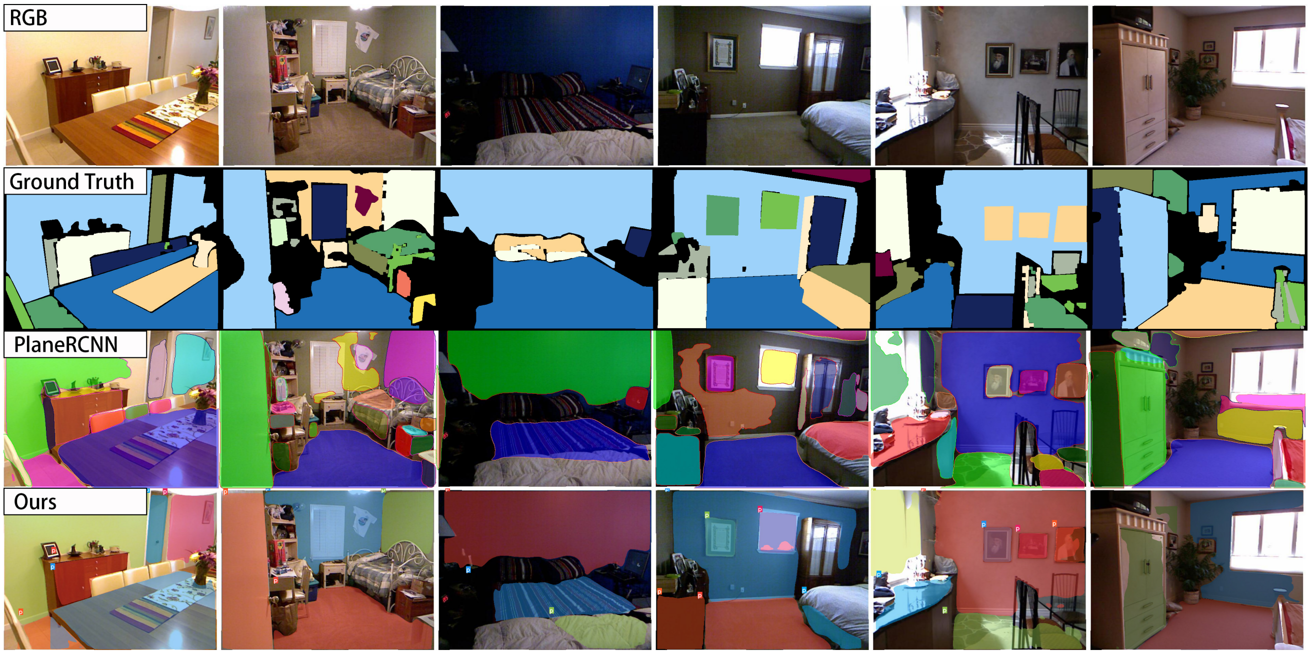
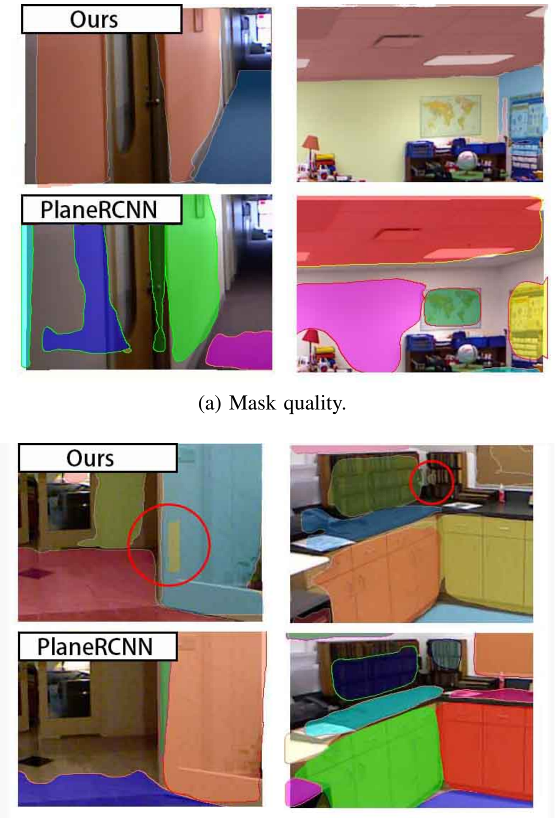


Fig. 3: **Qualitative results of instance planar segmentation of challenging cases** on dataset NYU v2 [29]. Notice that NYU dataset does not have planar semantic ground truth, the ground truth illustrated here was generated by our method (Sec. IV-A). The result shows our method provides in general more compact and more precise semantic masks, especially in favor of the plane boundary.

methods is tested with an indoor scene video with image size of 640 × 480, running all models on the same GPU (GTX 1080 Ti). Since PlaneNet and PlaneRCNN are multi-task networks, we disable their depth estimation and plane parameter estimation branches during the run-time testing. Our PlaneSegNet-101 trained on 2D-3D-S has the best result in terms of mask and bounding box Average Precision, while PlaneRCNN shows the best results in terms of VOI, RI and SC, however these metrics are strictly designed for evaluating semantic segmentation, instead of instance segmentation.

The qualitative results of planar segmentation are illustrated in Figure 3, where PlaneRCNN shows higher detecting ability on small objects, but suffers from the tendency to generalize as many positive detections as possible, which sometimes leads to false positive results. It also results in the average precision of PlaneRCNN being much lower than our method on the TUM RGB-D dataset, because PlaneRCNN divides many large scale planes into individual small instances which will not be considered as matched prediction when computing AP. Meanwhile our method provides better mask boundary quality on dominant planes in the image and maintains the instance completeness.

For visual SLAM applications, high quality segmentation prediction for large-size dominant planes significantly benefits the tracking accuracy. PlaneRCNN is based on two-stage segmentation method and local mask segmentation inside ROIs, while our PlaneSegNet is based on global prototype mask assembly. This improves our method's prediction accuracy on large scale instances (dominant plane), as shown in Figure 4 (a). Some failure cases of PlaneSegNet can also be seen in Figure 4 (b). The reason of these failures is that the prediction masks are cropped after assembly and no further suppression module is involved to filter out the noise from the cropped region.



(a) Mask quality.

(b) Failure cases.

Fig. 4: **Segmentation quality and failure cases.** In (a) we illustrate the quality. Our segmentation quality is better than PlaneRCNN in terms of the mask boundary completeness. In (b) we show some failure cases of feature leakage.

### C. Ablation Study

To better understand how the components of our model contribute to the overall performance, in Table II we perform an ablation study by changing various components of our model. As shown in the Table, the Residual Feature Augmentation module enhances the spatial context and improves the segmentation and bounding box prediction accuracy in terms of all the metrics. By introducing the Fast Feature NMS the prediction accuracy of bounding box estimation is improved more obvious than mask segmentation. We also trained our network with the same amount of data and iteration on 2D-3D-S and ScanNet, to compare and to prove that our ground truth generation method is more reliable and precise than the method used in PlaneRCNN.

### D. Incorporating with Planar SLAM

In this work, we employ SlamCraft [24] which is an efficient planar SLAM using monocular image sequence and generating a compact surfel map representation based on semantic cues. We replace the plane segmentation network used in SlamCraft (PlaneNet) with our PlaneSegNet which is about 5 times faster. We present the comparison of tracking accuracy results from Monocular ORB-SLAM2, SlamCraft+Ours and SlamCraft+PlaneNet in Table III, respectively, as well as the qualitative results of dense planar map in Figure 5. Our work combined with SlamCraft shows better results in terms of providing robust segmentation mask and improved localization accuracy of the camera.

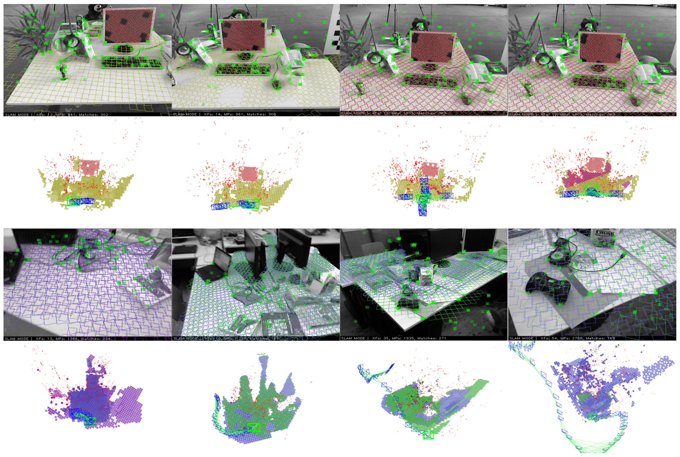


Fig. 5: **Qualitative results of dense planar map** generated from SlamCraft [24] using our PlaneSegNet. The two image sequences are fr2_xyz and fr1_desk from TUM RGB-D [30].

| Absolute KeyFrame Trajectory RMSE (cm) | | | | | | |
|---|---|---|---|---|---|---|
| **Seq.** | **ORB-SLAM2** | | **SlamCraft** | | **SlamCraft+Ours** | |
| # | Mean | Med. | Mean | Med. | Mean | Med. |
| fr1 xyz | 0.76 | 0.62 | **0.66** | **0.61** | 0.72 | 0.62 |
| fr2 xyz | 0.13 | 0.14 | 0.12 | 0.12 | **0.11** | **0.12** |
| fr1 desk | 0.68 | 0.68 | 0.60 | 0.47 | **0.58** | **0.46** |
| fr2 desk | 0.88 | 0.90 | 0.89 | 0.91 | **0.75** | **0.72** |
| fr2 sit xyz | 0.32 | 0.32 | 0.31 | 0.31 | **0.31** | **0.30** |
| fr3 str tex far | 0.86 | 0.87 | 0.84 | 0.84 | **0.76** | **0.68** |

TABLE III: **Comparison of Monocular ORB-SLAM2 [22] to SlamCraft [24]** in terms of trajectory RMSE (Root Mean Square Error) on dataset TUM RGB-D [30]. The original SlamCraft employs PlaneNet as their planar detector, while we replace it with our PlaneSegNet (SlamCraft+Ours) which improves accuracy in 5 out of the 6 image sequences.

## V. Conclusion

We present the first real-time single-stage instance segmentation method for piece-wise plane estimation. By optimizing the network structure and hyper parameters, we achieve a balance between accuracy and frame rate. Experiments against PlaneRCNN and PlaneNet demonstrated the effectiveness of our approach. Our method shows better segmentation quality in large scale planar regions while being real-time capable. Additionally, an application of our method on planar visual SLAM is presented in this work. Our approach still has some limitations in complex scenes with multiple scale instances. Further improving the bounding box localization accuracy and exploring more effective mask assembly solutions with acceptable run-time overhead will be interesting directions for future work.

# References

[1] M. Bassier, M. Bonduel, B. Van Genechten, and M. Vergauwen. Segmentation of large unstructured point clouds using octree-based region growing and conditional random fields. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42(2W8):25–30, 2017.

[2] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee. Yolact++: Better real-time instance segmentation. *ArXiv*, abs/1912.06218, 2019.

[3] A. Concha, M. W. Hussain, L. Montano, and J. Civera. Manhattan and piecewise-planar constraints for dense monocular mapping. In *Robotics: Science and systems*, 2014.

[4] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[6] C. Guo, B. Fan, Q. Zhang, S. Xiang, and C. Pan. Augfpn: Improving multi-scale feature learning for object detection, 2019.

[7] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[8] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[9] M. Hosseinzadeh, Y. Latif, T. Pham, N. Suenderhauf, and I. Reid. Structure aware slam using quadrics and planes. In *Asian Conference on Computer Vision*, pages 410–426. Springer, 2018.

[10] M. Hsiao, E. Westman, G. Zhang, and M. Kaess. Keyframe-based dense planar slam. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5110–5117. IEEE, 2017.

[11] S. Isa, S. Abdul Shukor, I. Maarof, Z. R. Yahya, A. Zakaria, A. H. Abdullah, and R. Wong. Point cloud data segmentation using ransac and localization. *IOP Conference Series: Materials Science and Engineering*, 705:012004, 12 2019.

[12] M. Kaess. Simultaneous localization and mapping with infinite planes. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4605–4611. IEEE, 2015.

[13] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[14] L. Li, F. Yang, H. Zhu, D. Li, Y. Li, and L. Tang. An improved ransac for 3d point cloud plane segmentation based on normal distribution transformation cells. *Remote Sensing*, 9(5):433, 2017.

[15] M. Li, L. Nan, N. Smith, and P. Wonka. Reconstructing building mass models from uav images. *Computers & Graphics*, 54:84–93, 2016.

[16] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei. Fully convolutional instance-aware semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2359–2367, 2017.

[17] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. arxiv 2014. *arXiv preprint arXiv:1405.0312*, 2014.

[18] C. Liu, K. Kim, J. Gu, Y. Furukawa, and J. Kautz. Planercnn: 3d plane detection and reconstruction from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4450–4459, 2019.

[19] C. Liu, J. Yang, D. Ceylan, E. Yumer, and Y. Furukawa. Planenet: Piece-wise planar reconstruction from a single rgb image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2579–2588, 2018.

[20] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[21] M. Magnusson, A. Lilienthal, and T. Duckett. Scan registration for autonomous mining vehicles using 3d-ndt. *Journal of Field Robotics*, 24:803–827, 10 2007.

[22] R. Mur-Artal and J. D. Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Transactions on Robotics*, 33(5):1255–1262, 2017.

[23] N. Pears and B. Liang. Ground plane segmentation for mobile robot visual navigation. In *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No. 01CH37180)*, volume 3, pages 1513–1518. IEEE, 2001.

[24] J. Rambach, P. Lesur, A. Pagani, and D. Stricker. Slamcraft: Dense planar rgb monocular slam. In *16th International Conference on Machine Vision Applications (MVA)*, pages 1–6. IEEE, 2019.

[25] A. Rangesh and M. M. Trivedi. Ground plane polling for 6dof pose estimation of objects on the road. *IEEE Transactions on Intelligent Vehicles*, pages 1–1, 2020.

[26] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3234–3243, 2016.

[27] R. F. Salas-Moreno, B. Glocken, P. H. Kelly, and A. J. Davison. Dense planar slam. In *2014 IEEE international symposium on mixed and augmented reality (ISMAR)*, pages 157–164. IEEE, 2014.

[28] N. O. Salscheider. Featurenms: Non-maximum suppression by learning feature embeddings. *arXiv preprint arXiv:2002.07662*, 2020.

[29] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, pages 746–760. Springer, 2012.

[30] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A benchmark for the evaluation of rgb-d slam systems. In *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.

[31] Y. Taguchi, Y.-D. Jian, S. Ramalingam, and C. Feng. Point-plane slam for hand-held 3d sensors. In *2013 IEEE international conference on robotics and automation*, pages 5182–5189. IEEE, 2013.

[32] Y. Tian, W. Song, L. Chen, Y. Sung, J. Kwak, and S. Sun. Fast planar detection system using a gpu-based 3d hough transform for lidar point clouds. *Applied Sciences*, 10(5):1744, 2020.

[33] R. Wang, L. Xie, and D. Chen. Modeling indoor spaces using decomposition and reconstruction of structural elements. *Photogrammetric Engineering & Remote Sensing*, 83(12):827–841, 2017.

[34] F. Yang and Z. Zhou. Recovering 3d planes from a single image via convolutional neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 85–100, 2018.

[35] S. Yang and S. Scherer. Monocular object and plane slam in structured environments. *IEEE Robotics and Automation Letters*, 4(4):3145–3152, 2019.

[36] F. Yu, V. Koltun, and T. Funkhouser. Dilated residual networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 472–480, 2017.

[37] Z. Yu, J. Zheng, D. Lian, Z. Zhou, and S. Gao. Single-image piece-wise planar 3d reconstruction via associative embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1029–1037, 2019.

[38] X. Zhang, W. Wang, X. Qi, Z. Liao, and R. Wei. Point-plane slam using supposed planes for indoor environments. *Sensors*, 19(17):3795, 2019.

[39] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. Torr. Conditional random fields as recurrent neural networks. In *International Conference on Computer Vision (ICCV)*, 2015.