PlaneMVS: 3D Plane Reconstruction from Multi-View Stereo – Supplementary Material

1. Hypothesis selection for slanted planes

Fig. 1 shows the distribution of the three axes of plane $\boldsymbol{n}^T/e$ sampled from $10,000$ training images. Based on the distribution, we select $(-2, 2)$, $(-2, 2)$, $(-2, 0.5)$ as the range of x, y, z axis for $\boldsymbol{n}^T/e$, respectively, to ensure at least 95% of the groundtruth planes lie within the ranges. Since our plane hypothesis is a three-dimensional vector, the computational cost of the cost volume is cubic w.r.t. the number of hypothesis per axis. To reach a balance between accuracy and memory consumption, we sample 8 hypotheses uniformly along every axis and finally have $N = 8^3 = 512$ plane hypotheses in total.

2. Semantic classes on ScanNet

After merging the semantically-similar categories in NYU40 [7] labels, we pick 11 classes: wall, floor, door, chair, window, picture, desk & table, bed & sofa, monitor & screen, cabinet & counter, box & bin, which are likely to contain planar structures in indoor scenes. Please refer to Fig. 2 for some visualization examples of the generated planar instance and semantic groundtruth from ScanNet [2].

3. Benchmark setup

7-Scenes. 7-Scenes [4] collects posed RGB-D camera frames of seven indoor scenes. We sample stereo pairs in the same manner as in ScanNet [2] and follow the official split to get finetuning and evaluation data. We finally have $26,358$ pairs for finetuning and $15,508$ pairs for evaluation.

TUM-RGBD. TUM-RGBD [8] is an indoor RGB-D monocular SLAM dataset with calibrated cameras. We randomly select 4 scenes (*i.e.*, fr1-desk, fr1-room, fr1-desk2, fr3-long-office-household) with $5,013$ pairs for finetuning and 2 scenes (*i.e.*, fr2-desk, fr3-long-office-household-validation) containing $4,817$ pairs for evaluation.

4. Results on 7-Scenes and TUM-RGBD

We have discussed how we deal with 7-Scenes and have demonstrated its quantitative results in the main paper. Here we introduce our simple but effective strategy to perform finetuning with only groundtruth depth. We first generate pseudo groundtruths of plane masks by getting the predictions with the ScanNet-pretrained model on the testing images. Then we train our model without plane parameter losses but maintain other losses. We simply set each loss weight to 1 instead of adopting the loss term uncertainty during finetuning since we find it cannot bring much improvement. We finetune the model for 5 epochs. The planar depth gets much improved and we find that the plane detection results also tend to be visually better, which may be accredited to multi-task training and our soft-pooling loss to associate 2D with 3D. The same applies to the TUM-RGBD [8] dataset. Some qualitative examples of 7-Scenes are shown in Fig. 3.

As shown in Tab. 1 and Fig. 4, similar to 7-Scenes, our approach generalizes much better on TUM-RGBD compared with PlaneRCNN [5], thanks to the learned multi-view geometric relationship. By performing the proposed finetuning strategy, the results get further improved on both 3D planar geometry and 2D planar detection.

Method	AbsRel↓	SqRel↓	$\delta < 1.25$↑
PlaneRCNN [5]	0.243	0.105	0.655
Ours	0.143	0.07	0.795
Ours-FT	**0.120**	**0.054**	**0.851**

Table 1. Reconstructed depth on TUM-RGBD dataset [8] of different methods. "Ours" means directly testing with the ScanNet-pretrained model. "Ours-FT" means testing with the TUM-RGBD-finetuned model.

5. More Ablation studies

In this section, we discuss the impact of applying different hyper-parameters or settings in our experiments. Then we show qualitative examples on the two components of our proposed method to intuitively demonstrate their effects.

5.1. Hyper-parameters and settings

Plane hypothesis range. We first study the effect of the plane hypothesis range we set. We compare the results of different hypothesis ranges while keeping the hypothesis number N unchanged: (i) use the same range of $(-2, 2)$ for the x, y, z axes; (ii) broaden the range to $(-2.5, 2.5)$; (iii)

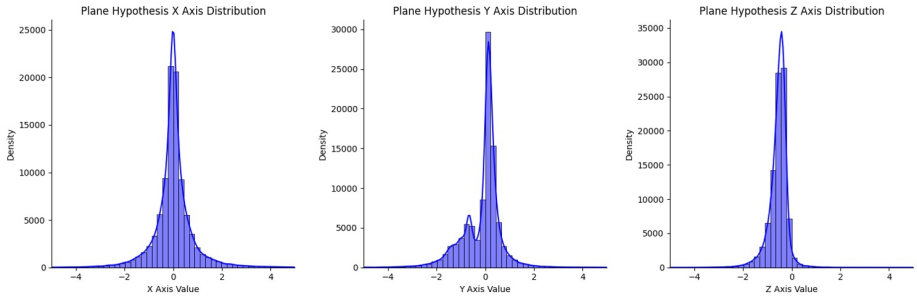

Figure 1. Plane hypothesis distribution of the three axes.

Figure 2. Examples of planar semantic and instance groudtruths on ScanNet [2]. Different colors represent different plane instances (2^{nd} column) or semantic categories (3^{rd} column).

shorten the range to $(-1.75, 1.75)$; (iv) employ the same range of $(-2, 2)$ for the x, y axes and a different range of $(-2, 0.5)$ for the z axis. As shown in Tab. 2, setting (iv), which serves as our default setting, achieves the best result. The performance drops when using the same range for all axes as (i), since z values mainly distribute between $(-2, 0.5)$. Using a broader range, *e.g.* (i) and (ii), covers some marginal values but decreases the density of the plane hypothesis, thus leading to less accurate results. In setting (iv), although shortening ranges can increase the hypothesis density, some non-negligible groundtruth values are not well covered, thus also leading to worse results.

Hypos range	AbsRel↓	$\delta < 1.25$↑
(-2, 2) for x,y,z	0.093	0.920
(-1.75, 1.75) for x,y,z	0.094	0.921
(-2.5, 2.5) for x,y,z	0.096	0.919
(-2, 2) for x,y; (-2, 0.5) for z	**0.088**	**0.926**

Table 2. Ablation study on the range of slanted plane hypothesis.

Plane hypothesis number. When keeping the plane hypothesis range constant, varying hypothesis number N changes the hypothesis density. We test our model using $6, 8, 10$ hypotheses per axis, *i.e.*, $N = 216, 512$ and $1,000$ respectively. The results are listed in Tab. 3. As expected, in general, the higher density we set, the better geometry performance we achieve. The performance gaps among different numbers are small, which demonstrates that our model is robust to these hyper-parameters to some extent. Note that using $N = 1,000$ will substantially increase the memory consumption. So we choose $N = 512$ in our default setting.

Hypos number per axis	AbsRel↓	$\delta < 1.25$↑
6 hypos (216 in total)	0.091	0.924
8 hypos (512 in total)	**0.088**	0.926
10 hypos (1,000 in total)	**0.088**	**0.927**

Table 3. Ablation study on plane hypothesis number.

Method	AbsRel↓	$\delta < 1.25$↑
Pixel-planar w/o pooling	0.091	0.920
Pooling with predicted masks	0.088	0.925
Soft-pooling with predicted masks	0.088	0.926
Pooling with groundtruth masks	**0.087**	**0.932**

Table 4. Ablation study on plane instance pooling with plane masks during testing.

Plane instance-aware soft pooling. We now evaluate the recovered depths among different pooling strategies reflecting the efficacy of plane detection on the learned 3D planar geometry. As shown in Tab. 4, when evaluating the depth reconstructed from pixel-level plane parameters, it underperforms the results with plane instance pooling since the generated depth maps cannot capture piece-

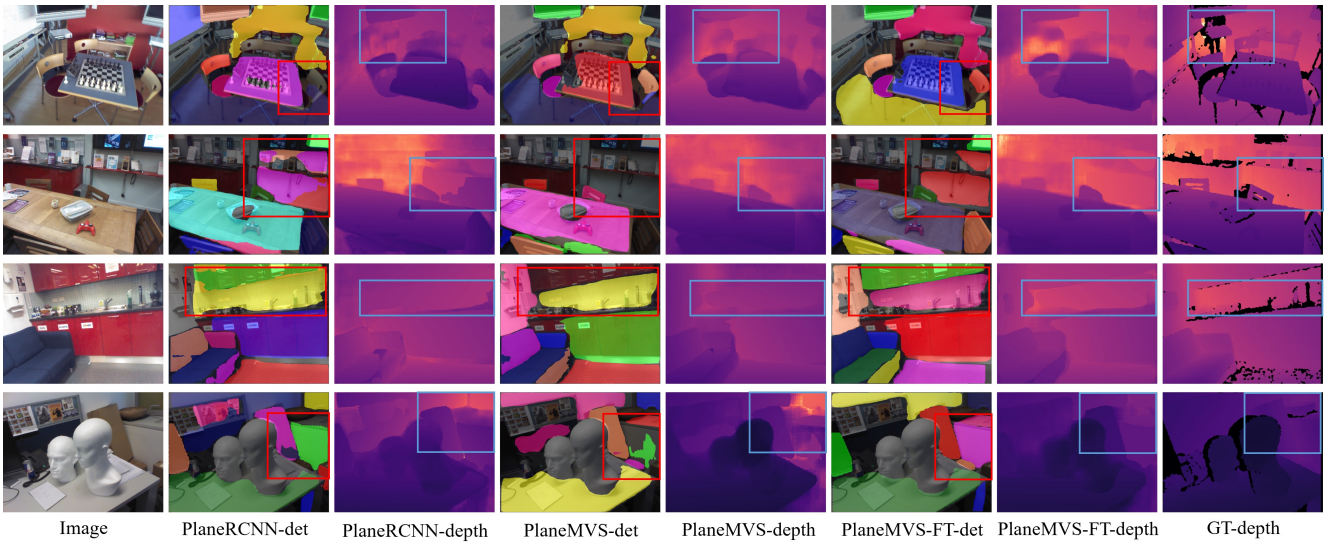

Figure 3. The plane reconstruction results on 7-scenes [4] among different methods. "FT" denotes "finetuned" and "det" is short for "detection". Regions with salient differences are highlighted with blue and red boxes. Best viewed on screen with zoom-in.

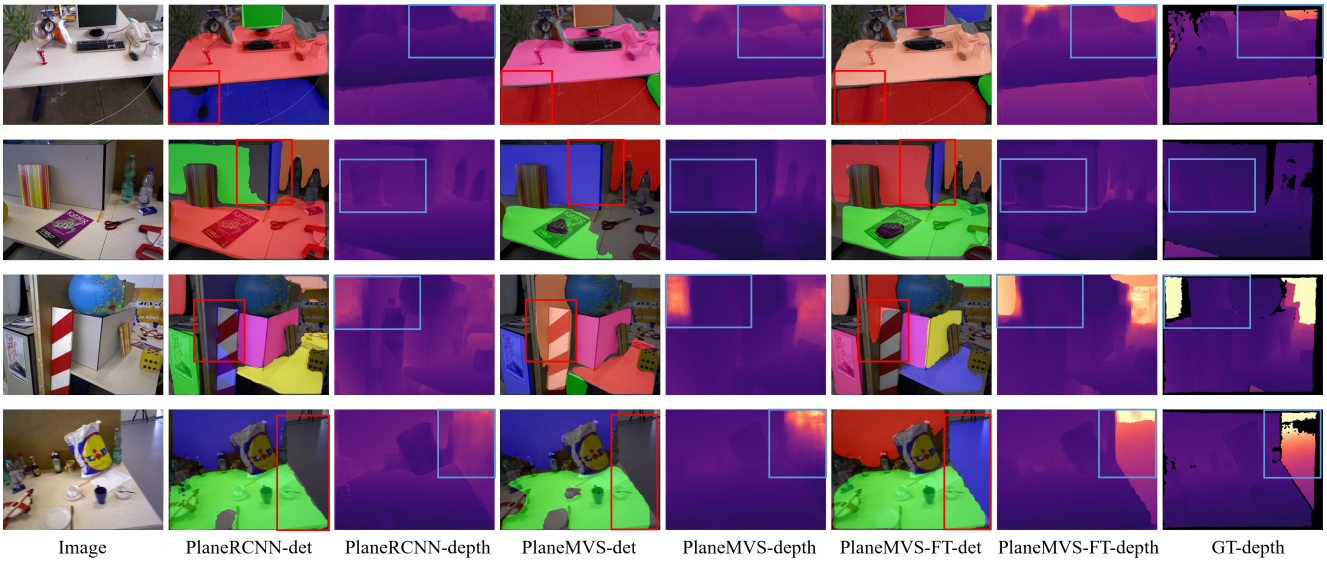

Figure 4. The plane reconstruction results on TUM-RGBD [8] among different methods. Regions with salient differences are highlighted with blue and red boxes. Best viewed on screen with zoom-in.

wise planarity. The result improves when we apply hard-pooling with predicted plane masks over the pixel-level plane parameters. Applying soft-pooling weighted with pixel-level probability further brings a minor improvement since the probability reflects the confidence of a pixel belonging to a plane instance. Finally, we use groundtruth plane masks to perform pooling, which represents the upper bound of the impact of plane detection on geometry. As expected, it achieves the best result among the settings. Since groundtruth plane masks are not available during testing, we always apply the soft-pooling with predicted masks in other experiments.

Depth on planar region. We further compare the reconstructed depth over only planar regions *v.s.* the whole image. Specifically, we conduct experiments only evaluating depth on the pixels that belong to any of the groundtruth planes. As shown in Tab. 5, compared with the depth over the whole image, the quantitative result over planar regions is better, no matter whether plane-instance-pooling is applied or not. This demonstrates that our proposed method's geometry improvement mainly comes from the pixels of planar regions, which conforms to our initial motivation and

objective.

Method	AbsRel↓	$\delta < 1.25$↑
Depth over whole image w/o pooling	0.091	0.920
Depth over planar region w/o pooling	0.086	0.929
Depth over whole image	0.088	0.926
Depth over planar region	**0.081**	**0.938**

Table 5. Ablation study on the evaluations over planar region.

Training dataset scale. In our default setting, we only sample $20,000$ stereo pairs for training. To analyze the impact of the scale of training data, we sample a larger training set with $66,000$ stereo pairs from the same scene split but keep the evaluation split unchanged. As shown in Tab. 6, our performance can be further improved with more training data on both plane detection and geometry metrics.

Dataset Scale	AbsRel↓	$\delta < 1.25$↑	$AP^{0.2m}$ ↑	AP↑
20,000 training pairs	0.088	0.926	0.456	0.564
66,000 training pairs	**0.082**	**0.934**	**0.470**	**0.570**

Table 6. Ablation study on the scale of training dataset.

5.2. Qualitative ablation analysis

This section gives some qualitative ablation analysis on the two components (*i.e.*, convex upsampling and the soft-pooling loss) used in our method. Fig. 5 shows the efficacy of convex upsampling. We show the depth map recovered from pixel-level parameters to eliminate the effect of plane instance pooling. It is clear that the results upsampled by convex combination have sharper boundaries and fewer artifacts than using bilinear upsampling.

Fig. 6 shows the effectiveness of the proposed soft-pooling loss. The detected planes from the model trained with the soft-pooling loss are much more complete and align better with their boundaries.

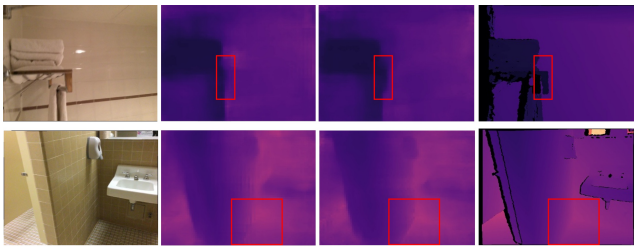

Figure 5. Effects of the convex upsampling on the depth map from pixel-level plane parameters. Regions with salient differences are highlighted with red boxes. Best viewed on screen with zoom-in.

6. Additional visualizations

We provide additional visualizations on predicted instance plane detection, planar semantic map, reconstructed planar depth map and 3D point cloud in Fig. 7, from our testing set on ScanNet [2].

Figure 6. Effects of the soft-pooling loss on plane detection. Regions with salient differences are highlighted with red boxes. Best viewed on screen with zoom-in.

7. Discussions and limitations

Our method *v.s.* patchmatch stereo. Our method shares high-level ideas with traditional patchmatch stereo works [1, 3] which aim to estimate a slanted plane for each pixel on the stereo reconstruction problem. However, our method differs from them in several aspects. (i) They perform patch matching around a pixel within a squared support window, where the patch size requires to be carefully set, thus not flexible and adaptive across various real-world cases. Instead of explicitly defining a patch, we associate and match the multi-view deep features. This is based on the observation that a pixel's receptive field on the feature map is far beyond itself because of stacked CNNs. The model can automatically learn the appropriate field for matching local features with end-to-end training. (ii) These methods usually first initialize pixels with random slanted plane hypotheses, then undergo sophisticated, multi-stage schemes with iterative optimizations. In contrast, we generate more reliable slanted plane hypotheses based on a data-driven approach (*i.e.*, analyzing the groundtruth plane distribution), and learn the pixel-wise plane parameters in an end-to-end manner, which is much easier to optimize. (iii) They usually adopt the photometric pixel dissimilarity as the matching cost function, which is sensitive to illumination changes and motion blurs across views. In contrast, we apply a feature-metric matching strategy, which is more robust to potential noises compared with applying photometric distance.

Potential limitations. Although we have achieved superior performance in most images, our system generates some failure cases as well. Firstly, as shown in Fig. 8, because of the large temporal gap, there exist areas in the target image which are invisible in the source image and thus

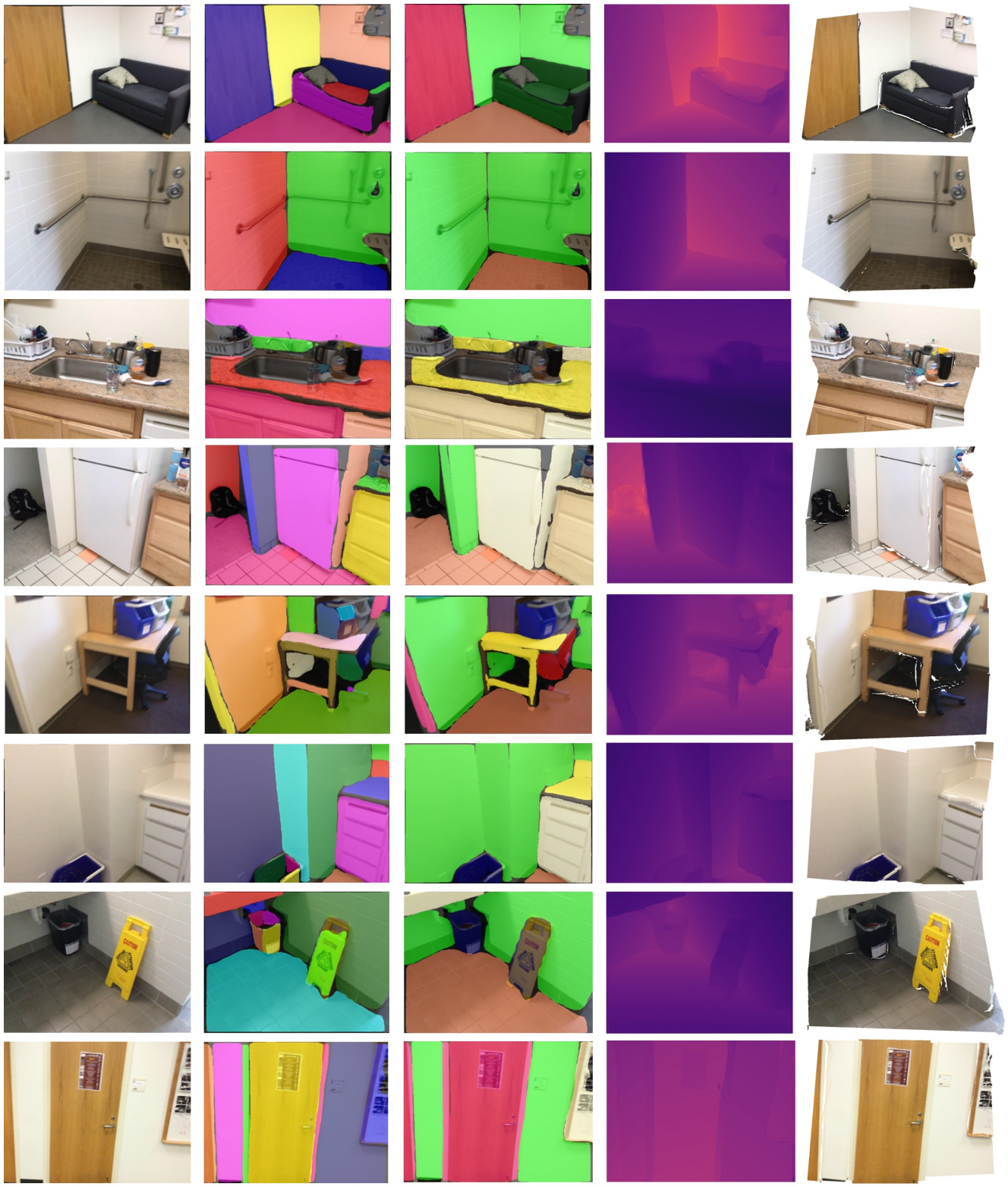

Figure 7. More qualitative results on ScanNet [2], including the instance planar masks, planar semantic map, planar depth map and the reconstructed 3D point cloud.

do not follow the planar homography relationship. This issue may be mitigated by introducing a network to learn the pixel-wise visibility or uncertainty [9]. Secondly, as in Fig. 9, there exist holes on some adjacent planes reconstructed from our method. An existing work [6] proposes to infer and enforce the inter-plane relationship from single images. This approach may solve the second issue and could be incorporated to further improve the final plane reconstruction. We also leave it into future work to explore.

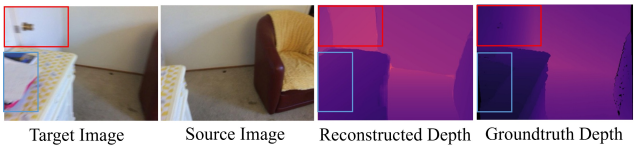

Figure 8. Failure case I: large temporal gap between two views. Problematic regions are highlighted with blue and red boxes. Best viewed on screen with zoom-in.

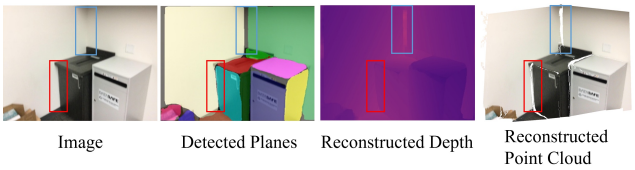

Figure 9. Failure case II: holes between adjacent planes. Problematic regions are highlighted with blue and red boxes. Best viewed on screen with zoom-in.

References

[1] Michael Bleyer, Christoph Rhemann, and Carsten Rother. Patchmatch stereo-stereo matching with slanted support windows. In *Bmvc*, volume 11, pages 1–11, 2011. 4

[2] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 1, 2, 4, 5

[3] Silvano Galliani, Katrin Lasinger, and Konrad Schindler. Massively parallel multiview stereopsis by surface normal diffusion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 873–881, 2015. 4

[4] Ben Glocker, Shahram Izadi, Jamie Shotton, and Antonio Criminisi. Real-time rgb-d camera relocalization. In *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 173–179. IEEE, 2013. 1, 3

[5] Chen Liu, Kihwan Kim, Jinwei Gu, Yasutaka Furukawa, and Jan Kautz. PlaneRCNN: 3d plane detection and reconstruction from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4450–4459, 2019. 1

[6] Yiming Qian and Yasutaka Furukawa. Learning pairwise inter-plane relations for piecewise planar reconstruction. In *European Conference on Computer Vision*, pages 330–345. Springer, 2020. 6

[7] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, pages 746–760. Springer, 2012. 1

[8] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of rgb-d slam systems. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 573–580. IEEE, 2012. 1, 3

[9] Jingyang Zhang, Yao Yao, Shiwei Li, Zixin Luo, and Tian Fang. Visibility-aware multi-view stereo network. *British Machine Vision Conference (BMVC)*, 2020. 6