Towards In-the-wild 3D Plane Reconstruction from a Single Image

Supplementary Material

1. Details on Plane Annotation Generation

In this section, we present more details about our dense plane annotation generation pipeline on the new benchmark indoor [18, 20, 21, 23, 24] and outdoor datasets [2, 7, 8, 10, 14, 15, 22]. Figure 1 shows examples of our plane annotation on different datasets.

Point cloud lifting. For RGB-D datasets containing precise ground-truth depth maps, we lift depth map to 3D point cloud for plane fitting. For stereo data such as ApolloScape, we first transform the disparity map into depth map using the provided camera baseline and intrinsic parameters, then lift the depth map and fit planes.

Panoptic segmentation. For datasets without dense semantic instance ground truth, we employ the state-of-the-art image segmentation approach Mask2Former [4] to obtain the panoptic segmentation results to assist the plane fitting process. We leverage their released models pretrained on ADE20K [25] and Cityscapes [5] to run on our indoor and outdoor datasets, respectively.

Plane number ranges. We select the obtained masks from categories likely to contain planar structures into our plane fitting stage, and perform instance-wise plane fitting. Moreover, we empirically set different plane number range (minimum and maximum number of planes) contained in each mask from either a background stuff or a foreground instance. For instance, for outdoor scenes we set [1, 2] for roads and walls, [1, 5] for buildings, and [0, 2] for vehicles. For indoor scenes, we set [0, 1] for floors and [0, 5] for other furniture.

Plane fitting with RANSAC. We follow previous works [11, 12] to fit planes with RANSAC. Specifically, we run RANSAC for 200 iterations for each plane. In each iteration, we randomly sample three points from the instance mask to fit a plane hypothesis then compute and record the number of point inliers over the instance point set. We select the plane hypothesis with maximum inliers as the final plane proposal, and use least square algorithm to refit the plane onto the entire set of its inliers and update its parameter. After getting proposals for each instance independently, we merge the neighbouring planes from the same semantic instance if their plane parameters are close to each other. Please refer to the implementation of [11, 12] for more details.

Distance-aware fitting error thresholds. Since the geometric scale variation of outdoor data is much larger than that of indoor scenes, we set a more tolerant fitting error (the average distance of all inlier points to the fitted plane proposal) threshold for the distant points while employing RANSAC. Our motivation is to make the threshold proportional to the average depth of these points. In this way, close and distant points are treated in a roughly equal manner. We set $0.05m$ as the reference fitting error and $10m$ as the reference average depth. Then, the adapted fitting error E of a plane proposal with an average depth d_m, is computed as:

$$E = \max(\frac{0.05 * d_m}{10}, 0.05) \tag{1}$$

A plane proposal will be rejected from the RANSAC process if its average fitting error exceeds the corresponding error threshold E.

Filtering tiny planes. After RANSAC fitting, we filter out tiny planes (those smaller than 200 pixels), as they are too challenging to be reliably detected by our annotation model.

User evaluation on our generated groundtruth. To intuitively validate the groundtruth quality of our pipeline, we have invited 10 volunteers to give rating on the plane segmentation quality from 500 randomly sampled images from all datasets as good, borderline, or bad. We received ratings of 84% 'good', 15% 'borderline', and 1% 'bad', verifying the convincing quality of our generated data.

Limitations on current pipeline. Although achieving desirable annotation quality over most of the scenes, we acknowledge that our current pipeline still exists a few limitations over some scenarios. First, on real-world data, the depth maps captured by sensors are sometimes incomplete, leading to missing planar mask annotation in our annotation since we leverage the point map lifted by depth. Second, the instance segmentation categories and plane number ranges are pre-defined prior to plane fitting, leading to some undefined regions on some not-well-defined cases. A potential solution is to leverage the SOTA segmentation model such as SAM for open-set segmentation to cover more planes.

2. Details on Our Method

Loss Weights. On the weight coefficients of different loss terms, we empirically set $\lambda_c = 2.0$ on plane classification, $\lambda_m = 5.0$ on plane mask for both dice and cross entropy losses, $\lambda_{n_c} = 1.0$ for normal classification, $\lambda_{n_r} = 5.0$ for

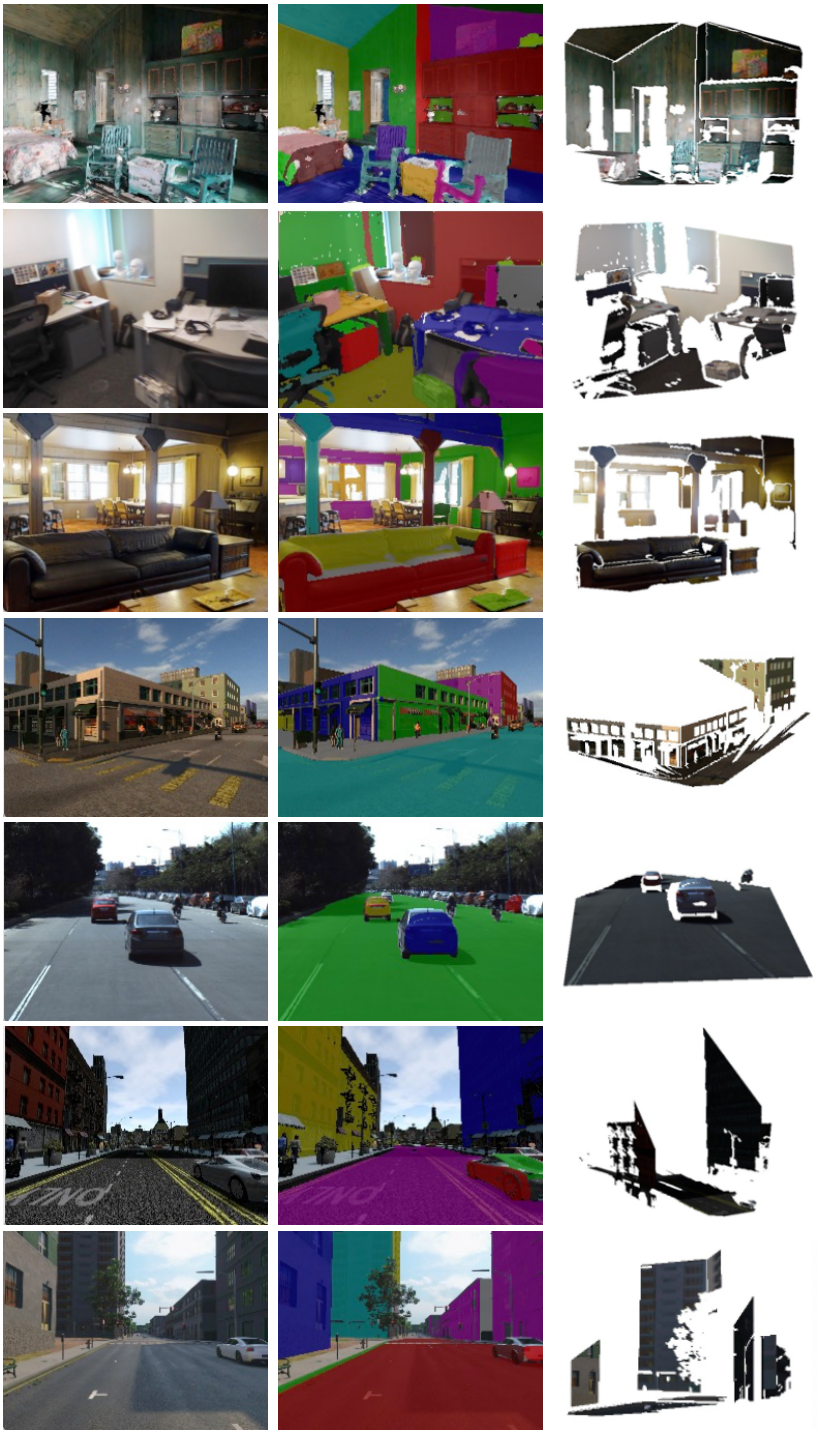

Figure 1. From top to bottom: our annotated ground-truth planes on HM3D [23], 7-Scenes [18], Taskonomy [24], ParallelDomain [8, 14], ApolloScape [10], Synthia [15] and Sanpo [22] datasets.

normal residual regression, $\lambda_{d_c} = 1.0$ for offset classification, $\lambda_{d_r} = 2.0$ for offset residual regression, $\lambda_{p_d} = 0.5$, $\lambda_{p_{n\_l1}} = 1.0$ for pixel normal L_1 loss and $\lambda_{p_{n\_cos}} = 5.0$ for pixel normal cosine distance loss.

Network Architecture. For the use of DINOv2 encoder and DPT pixel decoder, we follow their official implementation. On the pixel depth and normal heads, we feed the pixel features into three consecutive convolutional layers with ReLU activation except for the output layer for depth and normal respectively. For the pixel-geometry enhanced plane embedding module, we first pass the predicted depth and normal separately to a convolutional layer to derive the pixel geometric embeddings, then employ cross-attention, self-attention, and feed-forward network (FFN) between the plane query embeddings and the obtained pixel geometric embeddings to obtain the enhanced plane embeddings. This procedure is similar to the computational manner between query embeddings and pixel features used in query-based transformer detectors, as detailed in Mask2Former [4]. Regarding normal and offset classification and residual regression, we use two MLPs which take the instance-level plane embeddings as input and decode the plane class logits and residual vector, respectively. To achieve a better trade-off between precision and computational cost, we decrease every embedding layers dimension used in original [4] from 256 to 64, where we do not observe a great impact on plane reconstruction performance.

Computational overhead. We compare our computational overhead with PlaneRecTR [17] which shares similar overall architecture with ours. Under our default setting with DINO-B as our encoder, our model has 107.8M parameters and our FLOPS is 285M, whereas PlaneRecTR has 107M parameters and the FLOPS is 265M. We achieve comparable computational cost while significantly better zero-shot generalizability compared with this competitive counterpart.

3. Additional Experimental Results and Ablation Studies

In this section, we incorporate more ablation studies to demonstrate the robustness of our model, including the selections of exemplar number, the design of disentangled plane normal and offset used in our system, the robustness of our model on potential data bias, and the employment of SOTA monocular depth estimation with RANSAC as a competitive baseline method.

In-domain evaluation. Besides zero-shot evaluation, we provide the evaluation results of our model on the validation split of in-domain datasets (ScanNet [6], Synthia [15]) for both single-dataset training and mix-dataset training settings. As shown in Table 1, in both settings, our method achieves notable improvement for most of the metrics, especially on planar geometry.

Table 1. In-domain evaluation of both single-dataset-trained model (denoted as S) and mix-trained model (denoted as M) on ScanNet [6] and Synthia [15].

Evaluation Dataset	Method	Plane Segmentation			Plane Recall (depth)			Plane Recall (normal)		
		RI(↑)	VOI(↓)	SC(↑)	@0.05m / 1m	@0.1m / 3m	@0.6m / 10m	@5°	@10°	@30°
ScanNet [6]	PlaneRecTR (S) [17]	**0.94**	0.68	0.86	27.47	47.94	**77.21**	49.37	65.83	**75.24**
	Ours (S)	**0.94**	**0.65**	**0.87**	**29.62**	**48.79**	74.76	**58.18**	**68.52**	73.64
	PlaneRecTR (M) [17]	**0.91**	**0.88**	**0.80**	18.01	37.62	75.22	37.69	59.53	72.11
	Ours (M)	0.90	0.93	0.78	**21.3**	**40.43**	**75.5**	**55.7**	**66.78**	**73.64**
Synthia [15]	PlaneRecTR (S) [17]	**0.99**	0.22	0.94	**61.52**	71.32	73.80	66.46	72.87	75.37
	Ours (S)	**0.99**	**0.13**	**0.97**	61.45	**77.04**	**79.92**	**79.16**	**81.37**	**82.20**
	PlaneRecTR (M) [17]	0.97	0.50	0.87	40.85	50.44	57.38	41.62	52.84	59.54
	Ours (M)	**0.99**	**0.17**	**0.96**	**49.49**	**62.61**	**71.23**	**67.89**	**72.48**	**73.66**

Table 2. Quantitative results on employing coupled or disentangled plane normal and offset on NYUv2 [19] dataset.

Settings	Plane Recall (depth)			Plane Recall (normal)		
	@0.05m	@0.1m	@0.6m	@5°	@10°	@30°
Coupled normal and offset	7.9	**17.94**	**55.76**	34.48	46.63	56.61
Disentangled normal and offset	**8.54**	17.86	55.08	**37.29**	**47.58**	**57.19**

Table 3. Quantitative results on employing different numbers of normal and offset exemplars on NYUv2 [19] dataset.

Settings	Plane Recall (depth)			Plane Recall (normal)		
	@0.05m	@0.1m	@0.6m	@5°	@10°	@30°
$K_n = 14, K_d = 20$	8.27	**17.98**	54.9	36.46	47.18	56.67
$K_n = 7, K_d = 10$	8.21	17.84	54.67	36.71	47.47	56.67
$K_n = 7, K_d = 20$ (our default setting)	**8.54**	17.86	**55.08**	**37.29**	**47.58**	**57.19**

The use of disentangled normal and offset. In Tab.2, we show the result of an ablation study that compares between without disentanglement (using n/d to represent the plane parameter for classification-then-regression while keeping all the other proposed modules) and with disentanglement. It shows that disentanglement brings remarkable improvements in most of the metrics. This verifies the necessity of applying decoupled representation on normal and offset, whose physically meanings are distinct.

The selection of normal and offset exemplar numbers. We then investigate the impact of varying the number of exemplars on normal and offset in Table 3. One can see that, our model is generally robust to the selection of K_n and K_d, where the gaps on different selections are relatively small. Empirically, changing solely normal or offset exemplars does not lead to much gain and our default parameters achieve the best overall performance.

Robustness on the source of plane exemplar. To verify the robustness on how we obtain the clusters of plane normal and offsets on classification-then-regression, we conduct an ablation study by using only 2 indoor and 2 outdoor datasets, as opposed to using all 10 mixed training datasets, for clustering the normal and offset exemplars while still training on the full set of 10 mixed datasets. As shown, although suboptimal clusters led to a marginal performance drop, our model still demonstrated clear robustness over the source of plane examplar clusters.

Evaluation Dataset	Cluster source	@0.05m	@0.1m	@0.6m	@5°	@10°	@30°
NYUv2	partial (4 datasets)	7.73	17.2	54.59	37.02	47.56	56.28
	full (10 datasets)	**8.54**	**17.86**	**55.08**	**37.29**	**47.58**	**57.19**

@X6w8 **Robustness to pixel-level depth and normal prediction.** To validate whether bad pixel depth&normal prediction can lead to a performance gap on final plane reconstruction, We did an ablation study by adding random Gaussian noise with variation 0.05 w.r.t the original pixel and depth prediction values. As shown in the following table, there are only minor changes, demonstrating the robustness of our framework on depth and normal predictions.

Evaluation Dataset	Pixel depth & normal	@0.05m	@0.1m	@0.6m	@5°	@10°	@30°
NYUv2	Adding noise	8.52	17.88	55.04	37.31	47.62	57.21
	Model Prediction	8.54	17.86	55.08	37.29	47.58	57.19

The bias introduced by Mask2Former [4] on groundtruth fitting and model design. One potential concern raised from our proposed plane annotation pipeline and our framework is that, we use Mask2former's panoptic segmentation predictions for instance segmentation then plane fitting during groundtruth generation for a couple of datasets, while our framework is also partially based on Mask2former. This will introduce bias during both training and evaluation especially on the datasets whose groundtruth is involved by Mask2former. To this end, we conduct an ablation experiment, where we use the rest of datasets whose annotation pipeline does not involve Mask2former to train both the baseline counterpart [17] and our system, which eliminates the effect brought by Mask2former's involvement on groundtruth labels. As shown in Table 4, our method still significantly outperforms the parallel version of PlaneRecTR, which demonstrates the robustness of our model on this potential bias.

Employing SOTA monocular depth estimation and segmentation as a competitive baseline. Inspired by the recent success of foundation models on depth estimation and image segmentation, we apply the SOTA monocular metric depth estimation methods Metric3D-v2 [9] and Depth-Pro [1] to get dense pixel-wise monocular depth, and use Mask2former [4] for panoptic segmentation. Then, we apply the same RANSAC pipeline as we used on groundtruth plane generation to fit planes. We regard this as a training-free baseline which leverages foundation model inputs to tackle this task. As shown in Table 5, which achieving admissible performance of these two counterparts, we still beat their performance by a large margin, demonstrating our advantage over directly applying foundation models to solve this problem.

Table 4. Quantitative results on training without Mask2former-produced datasets, then evaluating on NYUv2.

Settings	Plane Recall (depth)			Plane Recall (normal)		
	@0.05m	@0.1m	@0.6m	@5°	@10°	@30°
PlaneRecTR w/o Mask2former data	5.01	13.47	49.29	19.16	36.69	50.77
Ours w/o Mask2former data	**7.22**	**16.37**	**49.8**	**33.66**	**43.68**	**52.05**

Table 5. Quantitative results on training without Mask2former-produced datasets, then evaluating on NYUv2.

Settings	Plane Recall (depth)			Plane Recall (normal)		
	@0.05m	@0.1m	@0.6m	@5°	@10°	@30°
Metric3D + Mask2former + RANSAC	2.72	6.76	47.02	14.09	34.11	47.56
Depth-Pro + Mask2former + RANSAC	3.61	9.14	47.91	20.11	37.41	49.52
Ours	**8.54**	**17.86**	**55.08**	**37.29**	**47.58**	**57.19**

4. More Qualitative Results

In Fig. 2, we showcase more qualitative results on testing images from diverse benchmarks or newly sampled in-the-wild data. Our model consistently demonstrates effectiveness and robustness across various environments.

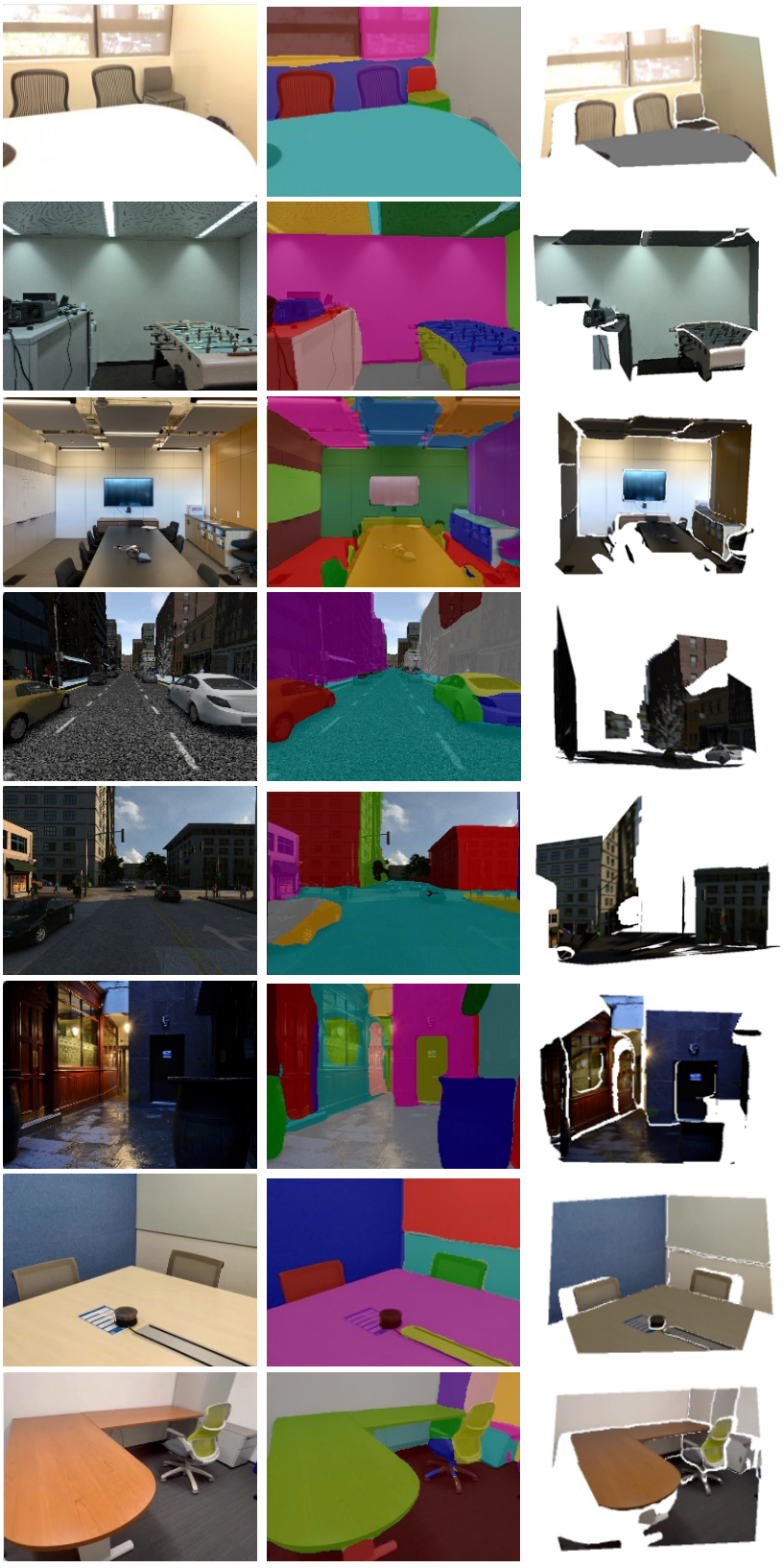

Figure 2. From top to bottom: the plane segmentation and reconstruction visualization of our model on ScanNet [6], ETH3D [16], LLFF [13], Synthia [15], ParallelDomain [8, 14], OASIS [3] and two in-the-wild images captured by ourselves.

References

[1] Aleksei Bochkovskii, Amaël Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. *arXiv preprint arXiv:2410.02073*, 2024. 4

[2] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2. *arXiv preprint arXiv:2001.10773*, 2020. 1

[3] Weifeng Chen, Shengyi Qian, David Fan, Noriyuki Kojima, Max Hamilton, and Jia Deng. Oasis: A large-scale dataset for single image 3d in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 679–688, 2020. 5

[4] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1290–1299, 2022. 1, 3, 4

[5] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 1

[6] 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. 3, 5

[7] Adrien Gaidon, Qiao Wang, Yohann Cabon, and Eleonora Vig. Virtual worlds as proxy for multi-object tracking analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4340–4349, 2016. 1

[8] Vitor Guizilini, Jie Li, Rareș Ambruș, and Adrien Gaidon. Geometric unsupervised domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8537–8547, 2021. 1, 2, 5

[9] Mu Hu, Wei Yin, Chi Zhang, Zhipeng Cai, Xiaoxiao Long, Hao Chen, Kaixuan Wang, Gang Yu, Chunhua Shen, and Shaojie Shen. Metric3d v2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation. *arXiv preprint arXiv:2404.15506*, 2024. 4

[10] Xinyu Huang, Xinjing Cheng, Qichuan Geng, Binbin Cao, Dingfu Zhou, Peng Wang, Yuanqing Lin, and Ruigang Yang. The apolloscape dataset for autonomous driving. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 954–960, 2018. 1, 2

[11] Chen Liu, Jimei Yang, Duygu Ceylan, Ersin Yumer, and Yasutaka 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. 1

[12] 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

[13] Ben Mildenhall, Pratul P Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*, 38(4):1–14, 2019. 5

[14] 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. 1, 2, 5

[15] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio 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. 1, 2, 3, 5

[16] Thomas Schops, Johannes L Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3260–3269, 2017. 5

[17] Jingjia Shi, Shuaifeng Zhi, and Kai Xu. Planerectr: Unified query learning for 3d plane recovery from a single view. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9377–9386, 2023. 3, 4

[18] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2930–2937, 2013. 1, 2

[19] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part V 12*, pages 746–760. Springer, 2012. 3

[20] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 1

[21] Igor Vasiljevic, Nick Kolkin, Shanyi Zhang, Ruotian Luo, Haochen Wang, Falcon Z Dai, Andrea F Daniele, Mohammadreza Mostajabi, Steven Basart, Matthew R Walter, et al. Diode: A dense indoor and outdoor depth dataset. *arXiv preprint arXiv:1908.00463*, 2019. 1

[22] Sagar M Waghmare, Kimberly Wilber, Dave Hawkey, Xuan Yang, Matthew Wilson, Stephanie Debats, Cattalyya Nuengsigkapian, Astuti Sharma, Lars Pandikow, Huisheng Wang, et al. Sanpo: A scene understanding, accessibility, navigation, pathfinding, obstacle avoidance dataset. *arXiv preprint arXiv:2309.12172*, 2023. 1, 2

[23] Karmesh Yadav, Ram Ramrakhya, Santhosh Kumar Ramakrishnan, Theo Gervet, John Turner, Aaron Gokaslan, Noah Maestre, Angel Xuan Chang, Dhruv Batra, Manolis Savva, et al. Habitat-matterport 3d semantics dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4927–4936, 2023. 1, 2

[24] Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3712–3722, 2018. 1, 2

[25] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017. 1