Holistic 3D Scene Understanding from a Single Image with Implicit Representation: Supplementary Material

Cheng Zhang[2*] Zhaopeng Cui[1*] Yinda Zhang[3*] Bing Zeng[2] Marc Pollefeys[4] Shuaicheng Liu[2†]
[1] State Key Lab of CAD & CG, Zhejiang University
[2] University of Electronic Science and Technology of China [3] Google [4]ETH Zürich

In this supplementary material, we provide the detailed network architecture, implementation details, 3D detection on all categories, more qualitative results, and discussion of failure cases.

1. Architecture of Our Pipeline

We show the architecture of LEN, ODN, LIEN, and SCGN in Figure 1.

2D Detector, LEN, ODN. Following [6, 2], we use Faster RCNN [7] trained on COCO dataset [4] and fine-tuned on SUN RGB-D [8] as 2D detector. The 2D detection results on SUN RGB-D are filtered and matched with the ground-truth 3D object bounding box during the data preparation procedure provided by [6]. During the initialization stage, we use LEN and ODN architecture shown in Figure 1 similar with [6].

LIEN. Our proposed LIEN consists of an image encoder followed by a three-layer MLP to embed a single image into a code. When evaluating on SUN RGB-D, the category labels are mapped to the ones used by Pix3D and concatenated to the image feature following [6]. To construct the shape elements for LDIF decoder, we follow [1] to reshape the 1344-dim vector into a 32x42 array, which corresponds to 42-dim (10 for analytic code and 32 for latent code) codes of the 32 shape elements.

SGCN. Before being fed into each node, the features from different sources are flattened, concatenated, and embedded into a 512-dim representation using FC layers. The weights of the embedding network for layout, object, and relationship nodes are independent of each other. After updated with four steps of message passing, the representations of layout and object nodes are decoded into parameters with the networks specially designed for each of the node types. The decoding networks follow the design of LEN and ODN, and refine the parameterized initial outputs of them.

*Equal contribution
†Corresponding author

2. Implementation details

Data Processing. For the training of LIEN, watertight meshes [1] must be used to retrieve the ground-truth values of inside-outside labels. However, the models of Pix3D [9] are not that clean with inverted surface normals and holes occasionally, which causes failure with the traditional flood fill algorithm. To get more robust results, we utilize the mesh fusion pipeline [5] which generates watertight meshes by fusing signed distance fields from several virtual cameras and applying the marching cube algorithm on it. Although the mesh fusion pipeline makes the model thicker and introduces noise to the ground-truth sample points, we evaluate it on the original mesh to directly compare with previous works.

SGCN Outputs. As mentioned in main paper Section 3.1, our SGCN predicts residuals to refine the parameters of object bounding boxs, layout box, and camera pose. We follow [6] to set the origin of the world coordinate frame at the camera center, with the y-axis up and perpendicular to the floor, and the x-axis aligned to the orientation of the camera forward. Thus the camera pose can be parameterized as $\mathbf{R}(\beta, \gamma)$, where β is the camera pitch and γ is the camera roll. Also, a bounding box can be parameterized as 3D center $C \in \mathbb{R}^3$, size $s \in \mathbb{R}^3$ and orientation $\theta \in [-\pi, \pi)$. Specifically, a layout box can be represented as $\left(C, s^l, \theta^l\right)$, and a object box can be represented as (δ, d, s, θ), where $\delta \in \mathbb{R}^2$ is the offset between 2D projection of 3D center and detected 2D object bounding box center, and d is the distance between 3D center and camera center.

Hyper Parameters. When training LIEN, we use 1024 near-surface samples and 1024 uniformly samples, and set their loss weight $\lambda_{ns} = 0.1$ and $\lambda_{us} = 1$. For shape element center loss, we let $\lambda_c = 0.2$. Following [6, 3], classification and regression loss is used for parameters of both LEN and ODN, which we denote as $\mathcal{L}_x^{cls,reg} = \mathcal{L}_x^{cls} + \lambda_x^{reg}\mathcal{L}_x^{reg}, \forall x \in \{\beta, \gamma, \theta^l, d, \theta\}$. Other parameters of LEN and ODN are using only regression loss. For camera parameters, we set $\lambda_\beta = 0.25, \lambda_\beta^{reg} = 40, \lambda_\gamma = 0.25$, and $\lambda_\gamma^{reg} = 20$. For layout box parameters, we set $\lambda_C = 10, \lambda_{s^l} = 10, \lambda_{\theta^l} = 0.25$, and $\lambda_{\theta^l}^{reg} = 30$. For object box

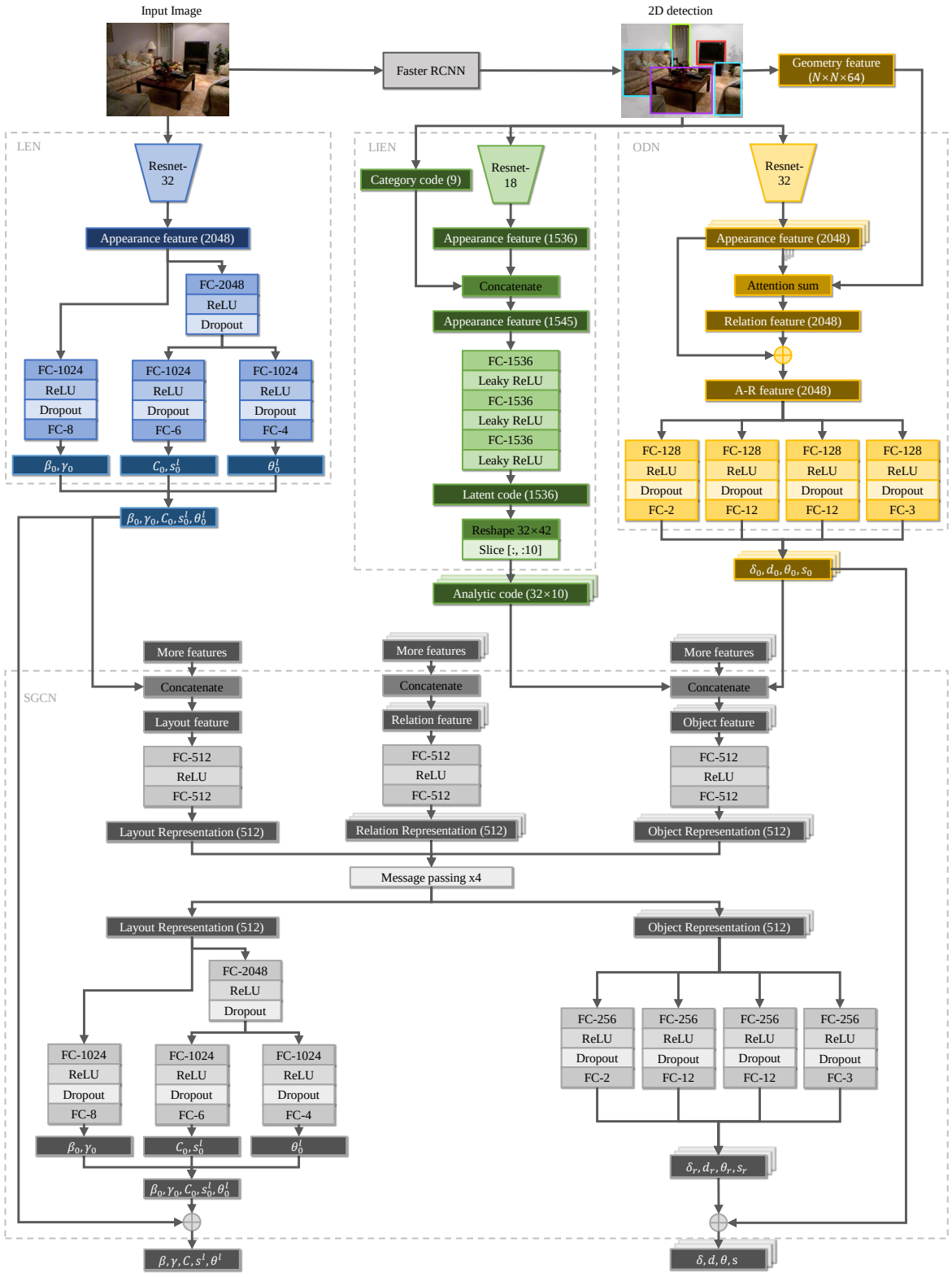


Figure 1: Architecture of LEN, ODN, LIEN, and SGCN. Our pipeline takes features from LEN, ODN, LIEN and other sources and embeds them into node representations. The parameter x_0 initialized by LEN and ODN is then refined with residual x_r decoded from updated node representations, $\forall x \in \{\beta, \gamma, C, s^l, \theta^l, \delta, d, \theta, s\}$. Variables $\beta, \gamma, \theta^l, d, \theta$ are parameterized following [6, 3]. We set dropout rate to 0.5 for all dropout blocks.

parameters, we set $\lambda_\delta = 1, \lambda_d = 0.75, \lambda_d^{reg} = 6.7, \lambda_s = 10, \lambda_\theta = 0.33$, and $\lambda_\theta^{reg} = 30$. When training with cooperative loss and object physical violation loss, we set $\lambda_{co} = 150, \lambda_{phy} = 20, \alpha = 100, k = 4$.

Scene Mesh Reconstruction. Since our LIEN is trained on Pix3D with only 9 categories like MGN of Total3D, we suffer from the same problem with them when testing on SUN RGB-D, that our LIEN can not generalize to some of the categories. For the accuracy of the scene reconstruction, we follow [6] to only consider certain categories of objects (i.e. cabinet, bed, chair, sofa, table, door, bookshelf, desk, shelves, dresser, refrigerator, television, box, whiteboard, nightstand). As a result, the reconstructed scene mesh has fewer objects than 3D detections.

3. 3D Detection on all categories

In this section, we report the average precision of 3D object detection on all categories of SUN RGB-D for a full comparison in Table 1. Our method achieves the best performance for 27 over 33 categories and a significantly better mean average precision.

Method	CooP [2]	Total3D [6]	Ours
cabinet	10.47	14.51	**33.93**
bed	57.71	60.65	**89.34**
chair	15.21	17.55	**35.14**
sofa	36.67	44.90	**69.10**
table	31.16	36.48	**57.37**
door	0.14	0.69	**5.82**
window	0.00	**0.62**	0.00
bookshelf	3.81	4.93	**18.33**
picture	0.00	0.37	**1.04**
counter	27.67	32.08	**57.02**
blinds	**2.27**	0.00	1.69
desk	19.90	27.93	**49.03**
shelves	2.96	3.70	**16.68**
curtain	1.35	3.04	**7.38**
dresser	15.98	21.19	**29.27**
pillow	2.53	4.46	**11.41**
mirror	0.47	0.29	**0.87**
clothes	0.00	0.00	0.00
books	3.19	2.02	**5.44**
fridge	21.50	24.42	**39.12**
tv	5.20	5.60	**11.17**
paper	0.20	**0.97**	0.03
towel	2.14	2.07	**7.73**
shower curtain	**20.00**	**20.00**	0.00
box	2.59	2.46	**6.71**
whiteboard	0.16	0.61	**2.39**
person	20.96	**31.29**	20.82
nightstand	11.36	17.01	**41.34**
toilet	42.53	44.24	**70.81**
sink	15.95	18.50	**33.81**
lamp	3.28	5.04	**11.90**
bathtub	24.71	21.15	**53.64**
bag	1.53	2.47	**6.82**
mAP	12.23	14.28	**24.10**

Table 1: Average precision of 3D object detection on all categories. For CooP, we report the better results from [6] trained on NYU-37 object labels.

4. More Qualitatively Comparison with MGN on Object Mesh Reconstruction

In this section, we show more results on the object reconstruction in Figure 4. Compared to MGN in Total3D[6], our method produces more accurate geometry preserving high-quality details especially on chairs, bookshelves, and those shapes with relatively more complex topology.

5. More Qualitatively Comparison on 3D Detection and Scene Reconstruction

In main paper Section 4.2, we show qualitative results of the 3D object detection and scene reconstruction. Here, we show more results in Figure 5, Figure 6, and Figure 7. We can observe that compared to the state-of-the-art method [6], our method produces significantly more accurate object pose estimation with fewer flying objects (Figure 5e, Figure 6a), fewer objects intersected with each other (Figure 5a, Figure 6d, Figure 7e), and more accurate object orientation estimation (Figure 5c, Figure 6e, Figure 7c). We also observe fewer objects intersected with the layout box (Figure 5d, Figure 6e, Figure 7a).

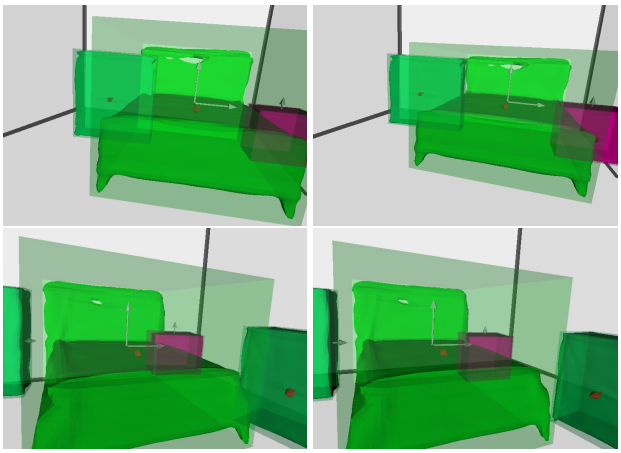
6. Qualitative Comparison of Ablation Study

In main paper Section 4.3, we quantitatively compare the improvement of our proposed $\mathcal{L}_{phy}$. While exhibiting a small gap from the metric, we show in qualitative results (Figure 2) that the visual difference is relatively large. Objects are more likely to intersect with each other when trained without $\mathcal{L}_{phy}$, which disobeys physical context severely. On the contrary, training with $\mathcal{L}_{phy}$ effectively prevents these errors in the results.

We also quantitatively compare the supporting relation, in main paper Section 4.3. Here in Figure 3, we qualitatively compare the understanding of supporting relation in the front view of object 3D detection.

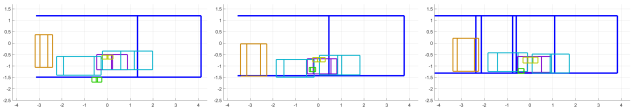
7. Failure Cases

We also show some failure cases in Figure 8. We observe that although our LIEN performs well on Pix3D and is generalized to SUN RGB-D, it still cannot make plausible reconstruction for some objects in rarely seen shapes (i.e. the



(a) Ours-$\mathcal{L}_{phy}$ (b) Full: with $\mathcal{L}_{phy}$

Figure 2: Scene reconstruction samples of Ours-$\mathcal{L}_{phy}$ and Full. We observe more intersections between objects without physical violation loss in some scenes.



(a) Total3D (b) Ours (c) GT

Figure 3: Qualitative comparison of supporting relation. We take the front view of object 3D detection of main paper Figure 5 column 4 as an example. We observe fewer flying objects in our results than Total3D, which shows a better understanding of supporting relation.

desks of (a) and (b), the bookshelves of (b) and (c), the bed of (d)). For object detection, our pipeline fails to correctly estimate the pose of the bed in (e), which might result from the clustered scenes. Also, in some extreme cases, heavy occlusion might cause our pipeline to fail like in (f).

References

[1] Kyle Genova, Forrester Cole, Avneesh Sud, Aaron Sarna, and Thomas Funkhouser. Local deep implicit functions for 3d shape. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4857–4866, 2020. 1

[2] Siyuan Huang, Siyuan Qi, Yinxue Xiao, Yixin Zhu, Ying Nian Wu, and Song-Chun Zhu. Cooperative holistic scene understanding: Unifying 3d object, layout, and camera pose estimation. In *Adv. Neural Inform. Process. Syst.*, 2018. 1, 3

[3] Siyuan Huang, Siyuan Qi, Yixin Zhu, Yinxue Xiao, Yuanlu Xu, and Song-Chun Zhu. Holistic 3d scene parsing and reconstruction from a single rgb image. In *Eur. Conf. Comput. Vis.*, 2018. 1, 2

[4] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Eur. Conf. Comput. Vis.*, pages 740–755. Springer, 2014. 1

[5] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4460–4470, 2019. 1

[6] Yinyu Nie, Xiaoguang Han, Shihui Guo, Yujian Zheng, Jian Chang, and Jian Jun Zhang. Total3dunderstanding: Joint layout, object pose and mesh reconstruction for indoor scenes from a single image. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 1, 2, 3, 5

[7] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(6):1137–1149, 2016. 1

[8] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015. 1

[9] Xingyuan Sun, Jiajun Wu, Xiuming Zhang, Zhoutong Zhang, Chengkai Zhang, Tianfan Xue, Joshua B Tenenbaum, and William T Freeman. Pix3d: Dataset and methods for single-image 3d shape modeling. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2974–2983, 2018. 1

Figure 4: More qualitative comparisons on object reconstruction. We compare with MGN from Total3D [6].

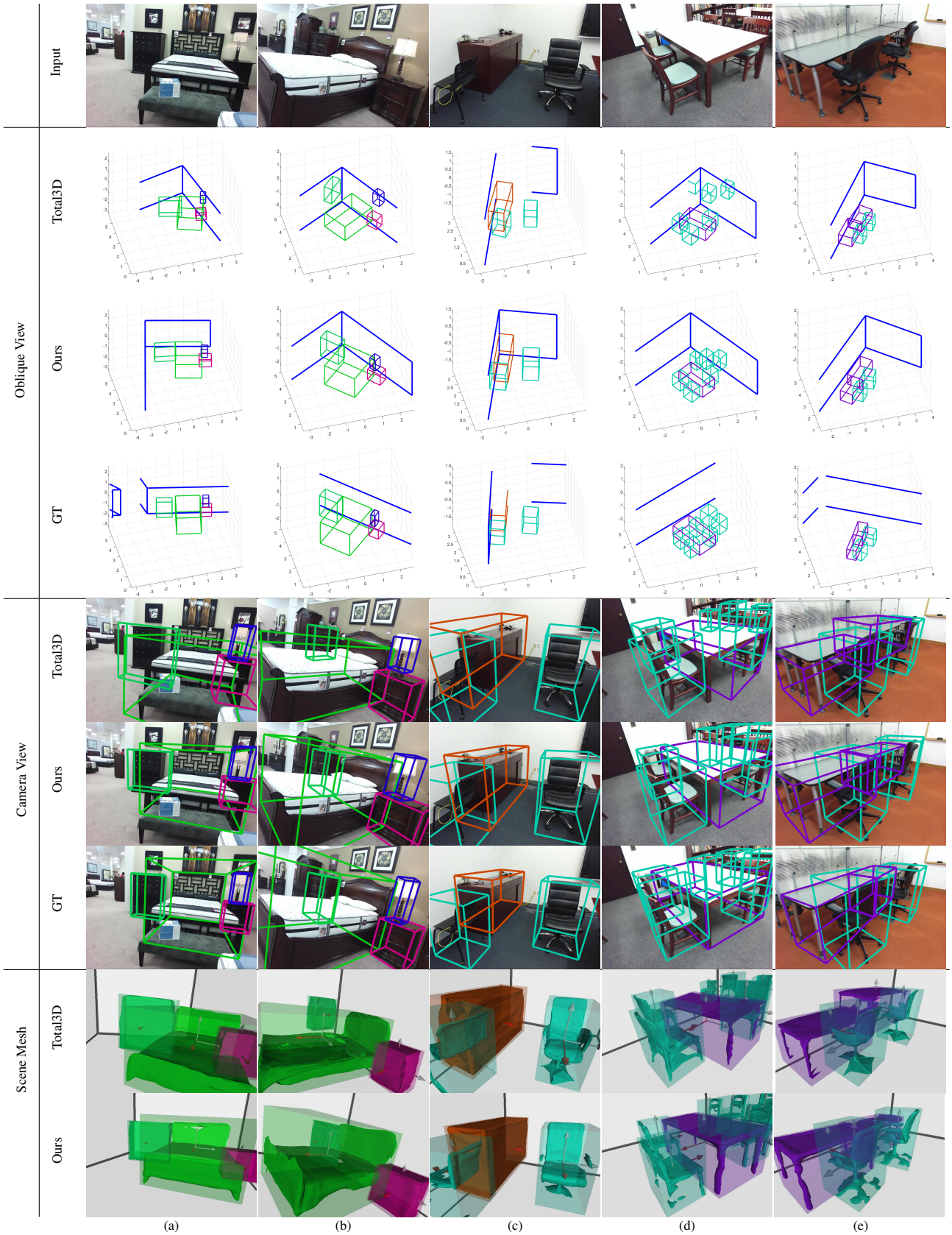


Figure 5: Qualitative comparisons on object detection and scene reconstruction.

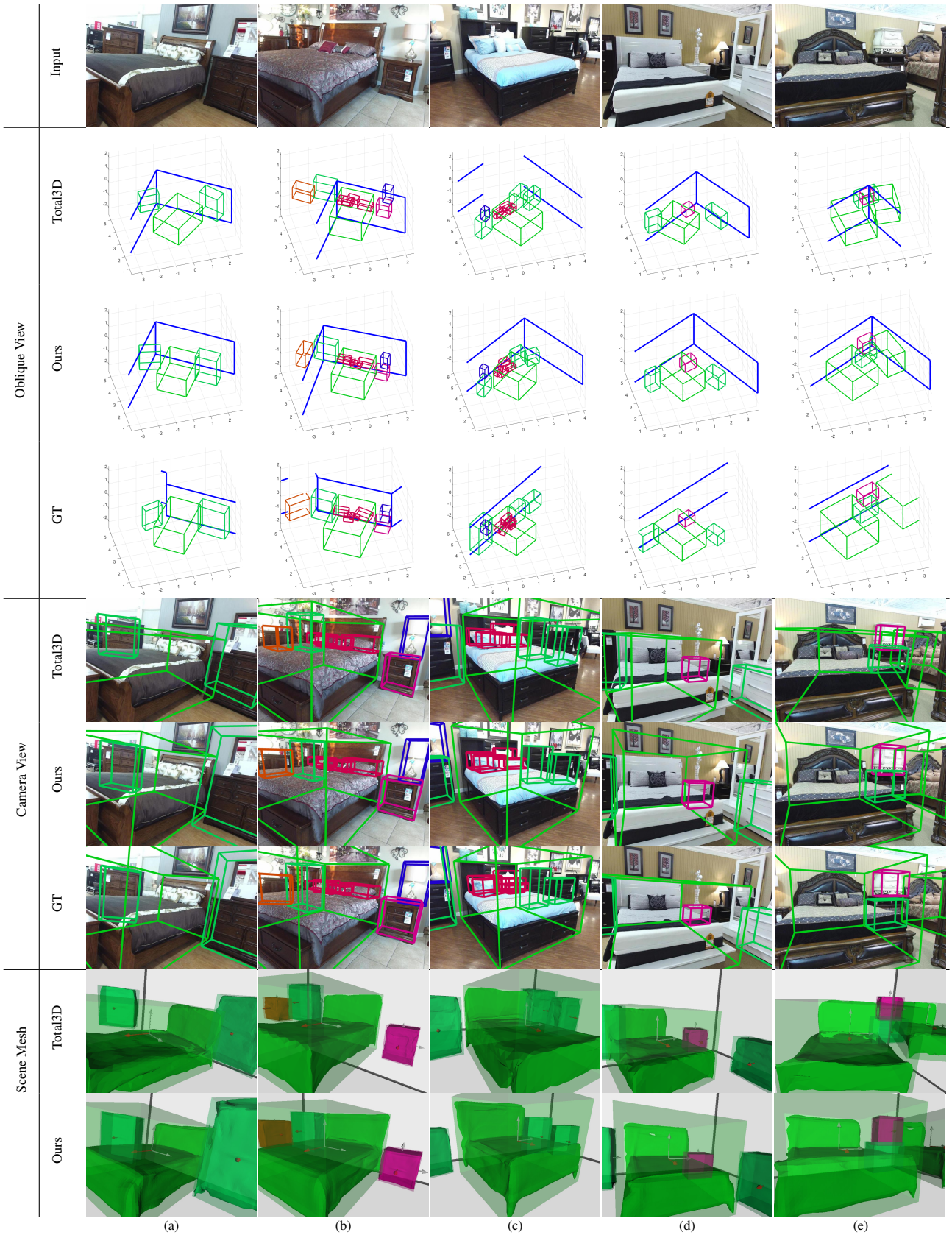


Figure 6: Qualitative comparisons on object detection and scene reconstruction.

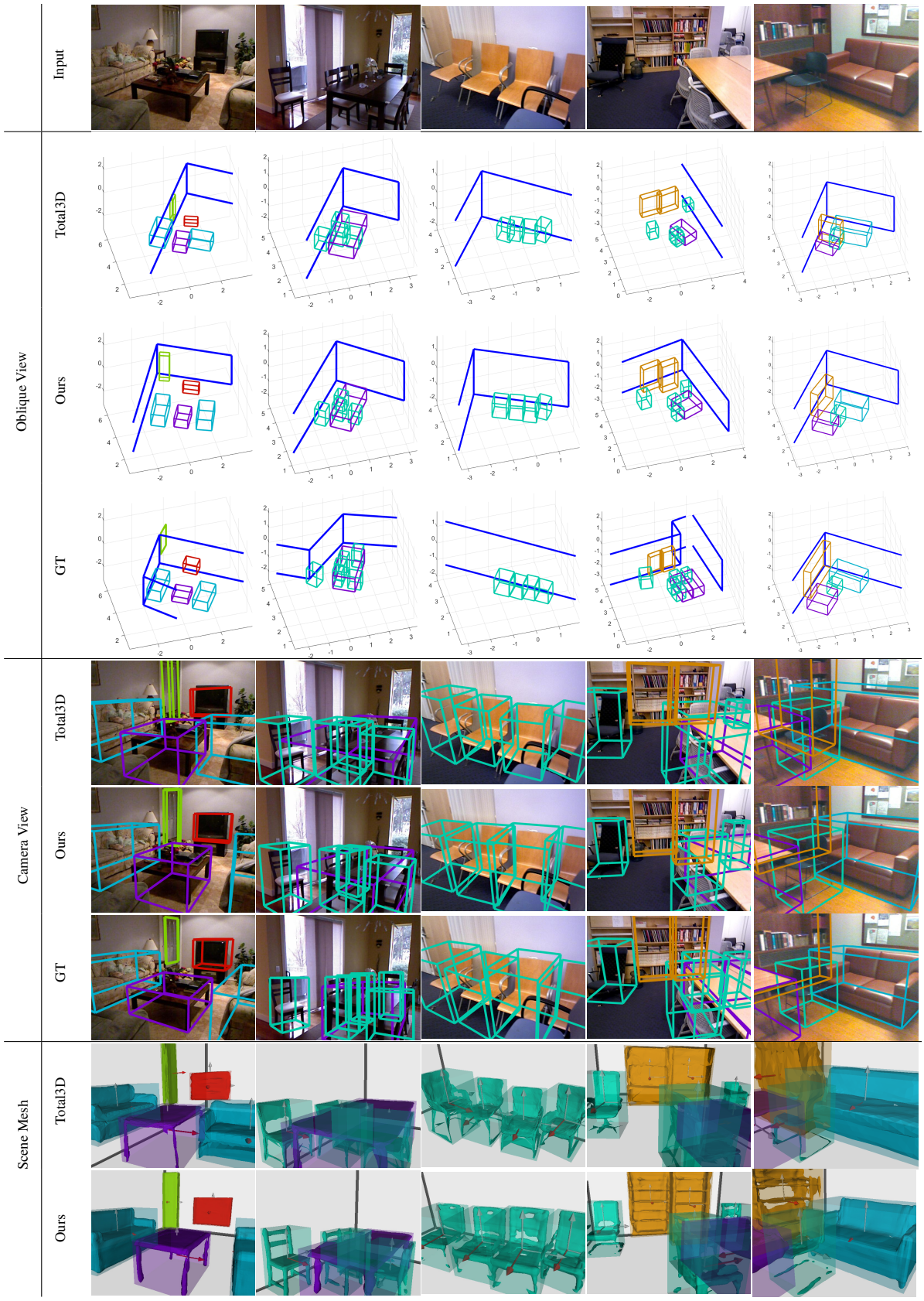


Figure 7: Qualitative comparisons on object detection and scene reconstruction.

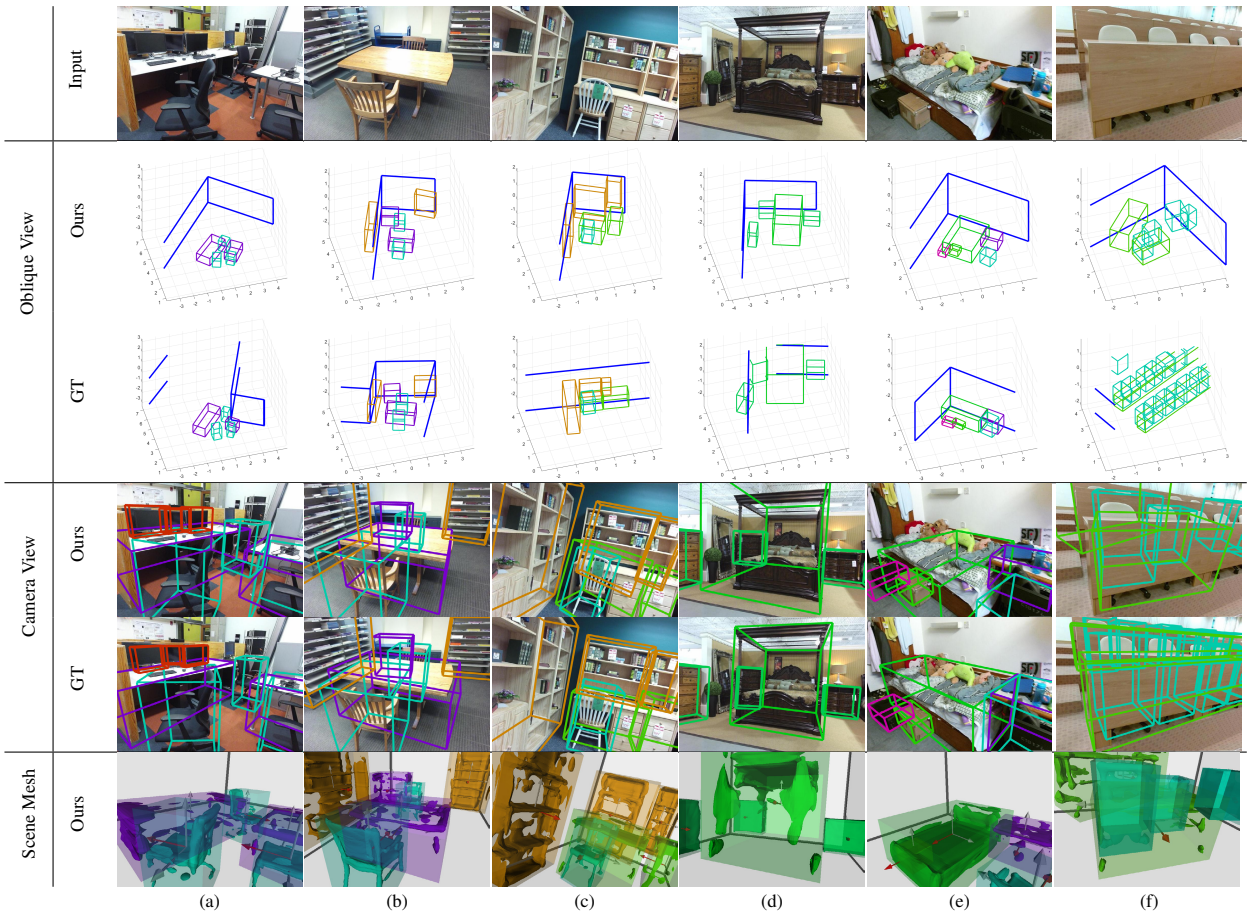


Figure 8: Failure Cases. Possible reasons might be unseen object shapes (a, b, c, d), heavy occlusion (f), cluttered scene (e).