DeepPanoContext: Panoramic 3D Scene Understanding with Holistic Scene Context Graph and Relation-based Optimization

Cheng Zhang[1] Zhaopeng Cui[2*] Cai Chen[1] Shuaicheng Liu[1*] Bing Zeng[1] Hujun Bao[2] Yinda Zhang[3*]
[1] University of Electronic Science and Technology of China
[2] State Key Lab of CAD & CG, Zhejiang University [3] Google

Abstract

Panorama images have a much larger field-of-view thus naturally encode enriched scene context information compared to standard perspective images, which however is not well exploited in the previous scene understanding methods. In this paper, we propose a novel method for panoramic 3D scene understanding which recovers the 3D room layout and the shape, pose, position, and semantic category for each object from a single full-view panorama image. In order to fully utilize the rich context information, we design a novel graph neural network based context model to predict the relationship among objects and room layout, and a differentiable relationship-based optimization module to optimize object arrangement with well-designed objective functions on-the-fly. Realizing the existing data are either with incomplete ground truth or overly-simplified scene, we present a new synthetic dataset with good diversity in room layout and furniture placement, and realistic image quality for total panoramic 3D scene understanding. Experiments demonstrate that our method outperforms existing methods on panoramic scene understanding in terms of both geometry accuracy and object arrangement. Code is available at https://chengzhag.github.io/publication/dpc.

1. Introduction

Image-based holistic 3D indoor scene understanding is a long-lasting challenging problem in computer vision, due to scene clutter and 3D ambiguity in perspective geometry. Over decades, the scene context, which encodes high-order relations across multiple objects following certain design rules, has been widely utilized to improve the scene understanding [49, 6]. However, it is still arguable and unclear if the top-down context is more or less important than bottom-up local appearance-based approaches for the scene parsing task, especially with the rapidly emerging deep learning methods that have achieved great success on object classification and detection. One possible reason could be that the field of view of a standard camera photo is normally less than 60°, and thus only limited context can be utilized among a small number of objects co-existing in the image. Zhang *et al.* [49] proposed a 3D scene parsing method that takes a 360° full-view panorama as the input, where almost all major objects are visible. They showed that the context became significantly stronger with more objects in the same image, which enables accurate 3D scene understanding even with less engineered local features.

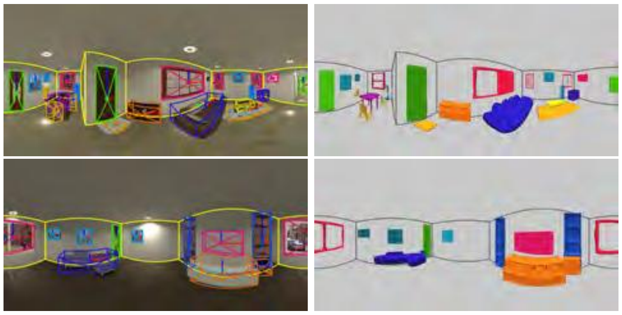

Figure 1: From a single panorama image as input, our proposed pipeline estimates layout and object poses, then reconstructs the scene with object reconstruction, to achieve total scene understanding.

In this paper, we empower the panoramic scene understanding task with stronger 3D perception capability and aim to predict the objects' shapes, 3D poses, and semantic categories as well as the room layout by taking a single color full-view panorama image as the input. To fulfill this goal, we propose a novel deep learning based framework that leverages both local image information and global context for panoramic 3D scene understanding. Specifically, we first extract room layout and object hypothesis from local image regions with the algorithms customized for panorama images, and rely on a global graph-based context model to effectively refine the initial estimations. Overall, our method achieves phenomenal performance on both geometry accuracy and object arrangement for 3D panoramic scene understanding.

Besides renovating the predecessor [49] with a more advanced deep learning algorithm, the key to the significant

[*] Corresponding author

performance gain is **a novel context model that predicts relations across objects and room layout including supporting, attaching, relative orientation, *etc*., which are then fed into an optimization to adjust the object arrangement.** This is inspired by the common sense that we, humans, tend to place objects tightly against the wall, *e.g*., beds, or side-by-side with consistent orientation, *e.g*., nightstands, and these relations could provide critical information to fix the object arrangement errors that may be minor in traditional metrics but obviously wrong judged by human perception. To leverage the predicted relations, we propose a novel differentiable optimization with carefully designed objective functions to adjust the initial object arrangement w.r.t. the predicted relations, which further enables joint training of relation prediction and object arrangement. The optimization is fully differentiable, which can be attached with our graph based context model, and conceptually any neural network, for joint training.

Unfortunately, the panoramic scene datasets for holistic 3D scene parsing are still missing in the literature. Existing panorama datasets are either with overly simplified scenes [49], purely 2D-based [40], or missing important 3D ground truth such as object poses [1, 5]. Since annotating real data with accurate 3D shapes is extremely challenging, we resort to synthetic data and create a new dataset for holistic panoramic 3D scene understanding. The dataset provides high-quality ground truth for object location, pose, shape, and pairwise relations, and serves well for training and rigorous evaluation. Though purely synthetic, we find the learned context model, which relies mainly on indoor scene context but not heavily on the image appearance, can be naturally generalized to real images by retraining bottom-up models that provide the initialization.

In summary, our contributions are as follows. We propose the first deep learning based pipeline for holistic 3D scene understanding that recovers 3D room layout and detailed shape, pose, location for objects in the scene from a single color full-view panorama image. To fully exploit context, we design a novel context model that predicts the relationship among objects and room layout, followed by a new differentiable relationship-based optimization module to refine the initial results. To learn and evaluate our model, a new dataset is created for total panoramic 3D scene understanding. Our model achieves the state-of-the-art performance on both geometry accuracy and 3D object arrangement.

2. Related Work

3D Scene Understanding Scene understanding in 3D world is a trending topic in the vision community. The task includes a volley of interesting sub-tasks including layout estimation, 3D object detection and pose estimation, and shape reconstruction. Various methods estimate the layout by adopting Manhattan World assumption [33, 8, 35, 51, 44] or cuboid assumption [9, 29, 25, 17]. 3D bounding boxes and object poses can be predicted from 2D representation with CNN-based methods [6, 19, 12, 38, 36, 4]. Object shapes can also be recovered by matching similar models, with geometrical or implicit representations [14, 26, 28, 39, 15, 23, 21, 20].

Total3D [31] is the first work to jointly solve multiple scene understanding tasks, including estimating the scene layout, object poses, and shapes. Recently, Zhang *et al*. [47] improves the performance of all three tasks via the implicit function and scene graph neural network. However, they still suffer from the insufficient exploitation of relationships among objects in the scene. In this work, we study the problem using panorama images which contain rich context information compared with the perspective ones with limited field of views.

Context for Scene Understanding Context priors can be employed for the scene understanding, *e.g*., a bed is placed on the floor and aligned with the wall. For perspective images, some methods [10, 11] adopt explicit constraints to avoid object overlaps. Zhang *et al*. [48] proposes to exploit scene context with a 3D context network. Recently, panoramic images have been exploited by optimization-based methods [32, 13, 41, 42, 46] designed over geometric or semantic cues, and learning-based methods [51, 24, 44, 35] with drastically advantageous representation of local context. Zhang *et al*. [49] achieves several tasks of 3D scene understanding by generating 3D hypotheses based on contextual constraints to exploit rich context information provided from large field of view (FOV). However, none of them provides a complete understanding of the scene. Instead, we propose a learning-based framework to jointly predict object shapes, 3D poses, semantic categories, and the room layout from a single panorama image, which takes full advantages of the scene context.

Panoramic Dataset For real-world scenes, the first panorama dataset is published by Xiao *et al*. [40], namely SUN360, and is later annotated for indoor scene understanding by Zhang *et al*. [49]. It contains high-resolution color panoramas with diverse objects, layout, and axis-aligned object boxes. However, it lacks object poses as well as shapes and only includes 700 images which is not adequate for the neural network training. 2D-3D-S [1] and Matterport3D [5] are also real-world datasets with more data and richer annotations, but poses are absent. Some datasets [7, 45] with the bounding FOV annotations are published for the purpose of panoramic object detection. Recently, a large photo-realistic dataset is proposed for structured 3D modeling, namely Structured3D [50], but mesh ground truths are not published. A panoramic scene dataset contains complete ground truth, including shape, object arrangement, and room layout is still missing.

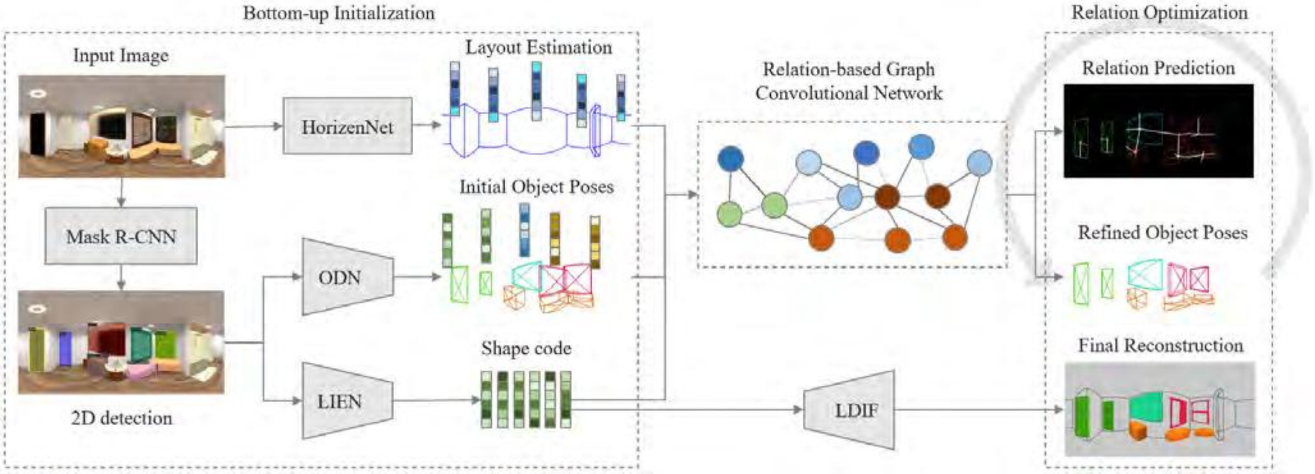

Figure 2: Our proposed pipeline. We first do a bottom-up initialization with several SoTA methods [47, 31, 16, 35] and provide various features, including geometric, semantic, and appearance features of objects and layout. These are then fed into our proposed RGCN network to refine the initial object pose and estimate the relation among objects and layout. A relation optimization is adopted afterward to further adjust the 3D object arrangement to align with the 2D observation, conform with the predicted relation, and resolve physical collision.

3. Method

In this section, we introduce our method for 3D panoramic scene understanding. As shown in Fig. 2, we first extract the whole-room layout under Manhattan World assumption and the initial object estimates including locations, sizes, poses, semantic categories, and latent shape codes. These, along with extracted features, are then fed into the Relation-based Graph Convolutional Network (RGCN) for refinement and to estimate relations among objects and layout simultaneously. Then, a differentiable Relation Optimization (RO) based on physical violation, observation, and relation is proposed to resolve collisions and adjust object poses. Finally, the 3D shape is recovered by feeding the latent shape code into Local Implicit Deep Function (LDIF) [14], and combined with object pose and room layout to achieve total scene understanding.

3.1. Bottom-up Initialization

We first estimate the room layout, initial objects' poses and shape codes for the panoramic scene from local image appearance. Similar to Zhang *et al*. [47], we run a Mask R-CNN to detect 2D objects, an Object Detection Network (ODN) [31] to generate initial pose, and a Local Implicit Embedding Network (LIEN) [47] to embed implicit 3D representation for each object. All the networks are retrained or customized for equirectangular panorama images.

Specifically, we first fine-tune the Mask R-CNN on our data such that it learns to handle the distortion and runs directly on panorama. We then fit a bounding box for each detected object mask represented as a Bounding FoV (BFoV) [7, 45], which is defined with the latitude and longitude of the center and the horizontal and vertical field of view. Since the left and right borders of full-view panorama images are actually connected, we extend the panorama by half of the width (concatenating the left half to the right) before feeding into the detector, then offset the detections of the extended part back to the left, following a standard non-maximum suppression (NMS) to merge overlapped or cross-border object detection. Images in each BFoV are then projected to the perspective view and fed into ODN and LIEN for 3D pose and latent shape representation. Note that for simplicity we assume that the object only rotates around y axis and ODN predicts the yaw angle of the object in the cropped perspective image coordinate as the object rotation. We empirically found this representation benefit the pose estimation performance, and the result can easily be converted to panorama (*i.e*., world) coordinates. Regarding the room layout, we use the SoTA HorizonNet [35].

3.2. Relation-based Graph Convolutional Network

After having the initial estimation, similar to Zhang *et al*. [47], we model the whole scene with a graph and refine the results via a Graph R-CNN [43]. Thanks to the full-view panorama, our GCN can now model all the objects in the room, which is able to encode and leverage stronger context than that in a perspective view [47]. Different than Zhang *et al*. [47], our model not only refines object poses but also predicts relations between objects and room layouts. Therefore we call our model Relation-based Graph Convolutional Network (RGCN).

Graph Construction Besides modeling each object as a node as in Zhang *et al*. [47], we further represent each wall in the estimated room layout via HorizonNet into a cuboid with a certain thickness and model them as separate nodes. This facilitates the learning of the relation between objects

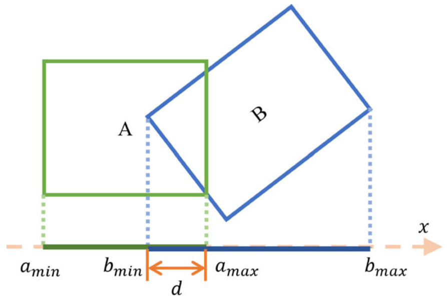

Figure 3: Object-object collision term defined with Separating Axis Theorem. We calculate separation distance d on all the separation axis x of object A and B.

to each wall without additional complexity. For each pair of wall/object nodes, we connect them with an undirected edge to form a complete graph with self circles. Then two relation nodes with directed edges are added to connect the wall/object nodes. Each node, including wall, object, and relation, is embedded with a latent vector, which is updated by GCN through message passing [47, 43].

Input Features For each type of node, we collect different features from various sources, concatenate, and embed them with Multi-Layer Perceptron (MLP) into initial node latent vectors. Following [47], we take bounding box parameters for object/wall nodes, category/analytic code and blob centers of LDIF in the world frame [47, 14] for object nodes, and geometry feature of 2D bounding boxes [18, 37] for relation nodes. Besides, we propose to further take geometric features from the room layout and the initial 3D object pose estimations to favor the relation estimation. Specifically speaking, on relation nodes, we add rotation (same definition as object-object rotation) and separation distance (to be further discussed in Sec. 3.3) between each pair of object/wall 3D bounding boxes. On object nodes, we add height differences between object 3D box corners and the floor/ceiling plane, and the 2D distances from bounding box corners to the layout polygon.

Relation Estimation Besides refining initial object pose objects' poses, our RGCN also outputs the relations between objects and the layout. The purpose of the relation estimation is to learn valuable context information which may have not been captured by the pose refinement branch. Specifically, we design two categories of relations between a pair of elements: object-object and object-layout. For object-object (including walls since they are also represented as nodes) relation, we define 1) the relative rotation between the front face of two objects; 2) whether the two 3D bounding boxes contact with a certain tolerance; and 3) if the center of one object's 3D bounding box is further than that of the other w.r.t. the camera center. For object-layout relation, we define 1) whether the object is supported by the floor or contacts with ceiling; and 2) if the 3D bounding box is fully inside the room. The later one is required to disable certain terms in relation optimization (to be further discussed in Sec. 3.3) for objects visible but outside the room. Motivated by [2], we design relation estimation as binary classification tasks for binary relations. For the angular differences, we formulate it into multi-class classification by making a decision on one of the 8 discretized bins in 360° considering that most furniture in the room is well arranged. All the relations are estimated by an additional MLP that takes the node representation as inputs.

3.3. Relation Optimization

While RGCN refines object poses, some numerically tiny errors may severely violate the context and thus be obvious in human perception, such as physical collision, flying objects, or small gaps to the wall. To fix these, we propose a differentiable optimization to update the refined poses w.r.t. the predicted relation as introduced in Fig. 3.2. Specifically, we use a gradient descend to minimize a loss function including three major components measuring physical collision, conformity to relation, and consistency with bottom-up observations.

3.3.1 Collision Term

At first, we define collision terms, which measures the amount of collision between objects, walls, ceiling, and floor. Two types of collision terms are defined according to the node types.

Object-Object Collision Since the object pose is represented by a cuboid, we use Separating Axis Theorem (SAT) [27, 22], which measures the collision between convex polygons to penalize the collision between two objects. As explained in Fig. 3, two oriented bounding boxes A and B collide with each other if their projections overlap along all separating axes (directions perpendicular to edges). Specifically, the projection of bounding box A on separating axis x can be defined as $a_{min} = min\{c \cdot x | c \in \mathbb{C}_{\mathrm{A}}\}$ and $a_{max} = max\{c \cdot x | c \in \mathbb{C}_{\mathrm{A}}\}$, where $\mathbb{C}_{\mathrm{A}}$ is the set of corners of the bounding box A and x is represented as vector. Thus the sum of overlaps d on every separating axis of A and B can be treated as a measurement for their collision. It is also true in 3D space for convex polyhedrons with separating axes defined as the directions perpendicular to faces. Based on this, we define the object-object collision term between object i and j as:

$$e^{oc}_{ij} = \begin{cases} \sum_{x \in \mathbb{S}_{ij}} d_x, & \text{if } i, j \text{ have collision} \\ 0, & \text{otherwise} \end{cases}, \qquad (1)$$

where $\mathbb{S}_{ij}$ is the set of separating axes, and $d_x = min(|a_{max} - b_{min}|, |a_{min} - b_{max}|)$ is the separation distance along axis x.

Object-Layout Collision Since the room layout is under Manhattan World assumption, we define 1) object-wall collision e^{wc} of each object as the sum of the distances between its bounding box corners and the layout floor map; and 2) object-floor/ceiling collision, e^{fc} and e^{cc}, as the distance between the lower/upper surface of the bounding box and the floor/ceiling. All of these terms are zero if no collision happens. As mentioned in Sec. 3.2, some objects may still be visible even they are outside the room, which should not be considered for our task. Therefore, we weight e^{wc} with in-room likelihood l^{in} to avoid pulling out-room objects inside.

The scene collision term with objects $\mathbb{O}$ can be defined as:

$$E^c = \sum_{i,j\in\mathbb{O},i\neq j} \lambda^{oc} e^{oc}_{ij} + \sum_{i\in\mathbb{O}} (\lambda^{wc} l^{in}_i e^{wc}_i + \lambda^{fc} e^{fc}_i + \lambda^{cc} e^{cc}_i), \tag{2}$$

where λ^* are preset weights.

3.3.2 Relation Term

We then define relation terms to measure the conformity of object poses with regard to the predicted relations from RGCN in Sec. 3.2.

For the relative rotation, we define the term e^{rr} as the absolute error between the observed and predicted relative angle. For the object attachment relation (*i.e.*, contact), we define the term e^{oa} similar to e^{oc} but only penalize sum of separation distances when there is no collision. The terms e^{fa} and e^{ca} are defined as the distance from the lower/upper surface of the bounding box to the floor/ceiling, and are respectively set to zero if the object is already attaching with the floor/ceiling. For relative distance, we calculate a view distance for each object as the distance from camera center to object center, and define the term e^{rd} as the difference between view distance if their relative order disobeys with the prediction and zero otherwise. Overall, the relation term is defined as:

$$\begin{aligned} E^r = &\sum_{i\in\mathbb{O}} \lambda^{rr} e^{rr}_i + \sum_{i\in\mathbb{O},j\in\mathbb{O}\cup\mathbb{W},i\neq j} \lambda^{oa} l^{oa}_{ij} e^{oa}_{ij} \\ &+ \sum_{x\in\{ft,ct,rd\}} \sum_{i\in\mathbb{O}} \lambda^x l^x_i e^x_i \end{aligned} \tag{3}$$

where $\mathbb{W}$ is the set of walls, l^* are the relation labels predicted by RGCN, and λ^* are weights for each term.

3.3.3 Observation Term

Not only abide to the predicted relation and physics, the object pose refinement should also respect the initial predictions observed from the input image.

We first define a loss term that measures the consistency with the raw image observation. For each object, we fit a 2D bounding box to the projection of the 3D cuboid on the tangent plane centered at cuboid center, and compare it with the results from Mask-RCNN. We define e^{bp} as the intersection over union between two boxes. We then define a loss term to measure the consistency between the optimized cuboid with the initial estimation, which is the L1 loss of the cuboid parameters, including the offset from the 2D detection center to the cuboid center projection δ, distance from camera center to the cuboid center d, size s, and orientation θ as defined in previous work [31, 20]. The total scene observation term is then defined as:

$$E^o = \sum_{x\in\{bp,\delta,d,s,\theta\}} \sum_{i\in\mathbb{O}} \lambda^x e^x_i. \tag{4}$$

3.3.4 Optimization

We minimize the sum of the three terms:

$$\min E(\delta, d, s, \theta) = E^c + E^r + E^o. \tag{5}$$

are chosen according to the confidence of estimated relation and bottom-up observations. More details can be found in Supp. Materials. Note that the optimization can be achieved via gradient decent such that is differentiable and can be added to the RGCN for joint training.

3.4. Loss Function

We adopt the loss from Nie *et al.* [31] to train the ODN:

$$\mathcal{L}_{ODN} = \sum_{x\in\{\delta,d,s,\theta\}} \lambda_x \mathcal{L}_x, \tag{6}$$

where $\mathcal{L}_*$ are the classification and regression loss for the object pose parameters. To train RGCN, we first train pose refinement branch with $\mathcal{L}_{ODN}$, then add the losses for the relation branch:

$$\mathcal{L}_{RGCN} = \mathcal{L}_{ODN} + \sum_{x\in\{rr,oa,fa,ca,rd\}} \lambda_x \mathcal{L}_x, \tag{7}$$

where $\mathcal{L}_{rr}$ is 8-class cross-entropy loss of rotation classification, and $\mathcal{L}_x, x \in oa, fa, ca, rd$ are binary cross-entropy loss. When training ODN, RGCN with RO end-to-end, we define the joint loss as:

$$\mathcal{L} = \mathcal{L}_{ODN} + \mathcal{L}_{RGCN} + \sum_{x\in\{\delta,d,s,\theta\}} \lambda'_x \mathcal{L}'_x, \tag{8}$$

where $\mathcal{L}'_x$ is the L_1 loss of the optimized pose parameters.

3.5. Panoramic Datasets

As there is no panorama dataset with complete ground truth for room layout, object poses, and object shapes, we propose to synthesize a panoramic dataset that provides the detailed 3D shapes, poses, positions, semantics of objects as well as the room layout by utilizing the latest simulation environment iGibson [34]. iGibson contains 500+ objects of 57 categories, and 15 fully interactive scenes with 100+

Method	chair	sofa	table	fridge	sink	door	floor lamp	bottom cabinet	top cabinet	sofa chair	dryer	mAP
Total3D-Pers	13.71	68.06	30.55	36.02	69.84	11.88	12.57	35.56	19.19	64.29	41.36	36.64
Total3D-Pano	20.84	69.65	31.79	43.13	68.42	10.27	16.42	34.42	20.83	62.38	33.78	37.45
Im3D-Pers	30.23	**75.23**	44.16	52.56	76.46	14.91	9.99	45.51	23.37	**80.11**	53.28	45.98
Im3D-Pano	33.08	72.15	37.43	70.45	75.20	11.58	6.06	43.28	18.99	78.46	41.02	44.34
Ours (w/o. RO)	**33.57**	75.18	38.65	71.97	**80.66**	19.94	18.29	50.67	29.05	79.42	**60.07**	50.68
Ours (Full)	27.78	73.96	**46.85**	**74.22**	75.29	**21.43**	**20.69**	**52.03**	**50.39**	77.09	59.91	**52.69**

Table 1: 3D object detection. Following [31, 19], we evaluate on common object categories and use mean average precision (mAP) with the threshold of 3D bounding box IoU set at 0.15 as the evaluation metric. Please refer to supplementary material for evaluation on full 57 categories.

Method	background	bed	painting	window	mirror	desk	wardrobe	tv	door	chair	sofa	cabinet	mIoU
PanoContext	86.90	**78.58**	38.70	35.58	38.15	29.55	27.44	34.81	19.40	9.61	11.10	5.46	31.38
Ours (Full)	**87.48**	62.99	**56.33**	**65.36**	**40.48**	**52.86**	**53.50**	**46.88**	**49.70**	**34.21**	**48.59**	**10.36**	**50.73**

Table 2: Semantic segmentation IoU. Following [49] we calculate IoU with uniformly sampled points on sphere surface.

rooms in total, and 75 objects on average. Before rendering, we run a physical simulation [34] to resolve bad placement (*e.g*., floating objects) and randomly replace objects with models from the same semantic category for each scene. Then we set the cameras with height of 1.6m looking at random directions in the horizontal plane. By building a 2D occupation map of objects, we avoid setting cameras inside, over, or too close to objects. Finally, we render 1,500 panorama images with semantic/instance segmentation, depth images, room layout, and the oriented 3D object bounding boxes from the physical simulator. Among 15 provided scenes, we use 10 for training and 5 for testing, generating 100 images per scene.

We crop each object separately to train LIEN and LDIF. In total, we collect 19,245 object crops from the training set and 7,753 from the test set. Besides these, we also render extra object-centric images, which contain 51,285 for training and 5,715 for testing. To generate implicit signed distance field ground truth, we process the 3D object CAD model [30, 14] to make sure the objects are watertight. Please refer to supplementary material for examples of our synthesized dataset.

4. Experiments

To our best knowledge, we are the first to achieve total 3D scene understanding on panorama images with scene level reconstruction. Thus to make comparisons with the SoTA methods Total3D [31] and Im3D [47] which work with perspective cameras, we divide the panorama camera into a set of cameras with horizontal FoV of 60°. We then retrieve the detection results on panorama from our 2D detector and group them by camera splits then feed them into Total3D and Im3D. The results of object pose and shape are transformed from camera coordinates to world coordinates to make the final results (Total3D-Pers and Im3D-Pers). Besides the perspective version, we also extend Total3D and Im3D to work directly on panorama images (Total3D-Pano and Im3D-Pano). Specifically, we change the representation of 2D bounding box into BFoV and input object detection results as a whole to provide richer scene context information. Since Total3D and Im3D are designed to do cuboid layout estimation, for a fair comparison, we replaced their layout estimation network with HorizonNet and only compare with them on 3D object detection and scene reconstruction. All the models are fine-tuned on our proposed dataset following the same process. Please refer to supplementary material for more details.

4.1. Comparison with SoTA

3D object detection We evaluate our method with mean average precision (mAP) on 3D object detection and scene understanding. Following [47, 31, 19], we consider a predicted 3D bounding box with IoU (with ground truth) more than 0.15 as a true detection. As shown in Tab. 1, our method has a great improvement over the SoTA even without relation optimization, which mainly benefits from the novel geometric features extracted from initial estimates as well as the constraints between more objects. To show the generalization ability of our model, we compare it with PanoContext [49] on their proposed dataset in Tab. 2. Since the PanoContext dataset has no object orientation label, we fine-tune our Full model only up to 2D detector. The results show that our context model can also generalize to real data with only bottom-up models fine-tuned. We also show corresponding qualitative results in Fig. 4 and Fig. 5.

Physical violation To highlight the improvement benefit from relation optimization, *i.e*., collision avoidance, we calculate average collision times per scene and the average number of objects with collision. We also report the number of collision of each kind between object to object/ceiling/floor/wall. The results shown in Tab. 3 indicate that our method outperforms SoTA methods from all perspectives in preventing physical collision, while the gap from the ablated version further illustrating the importance of relation optimization. The analysis in ablation study (Sec. 4.2) further demonstrates the importance of relation optimization in delivering context plausible results.

Holistic Scene Reconstruction We compare the reconstruction results qualitatively with the Total3D-Pano and Im3D-Pano in Fig. 4. Our method shows overall the best

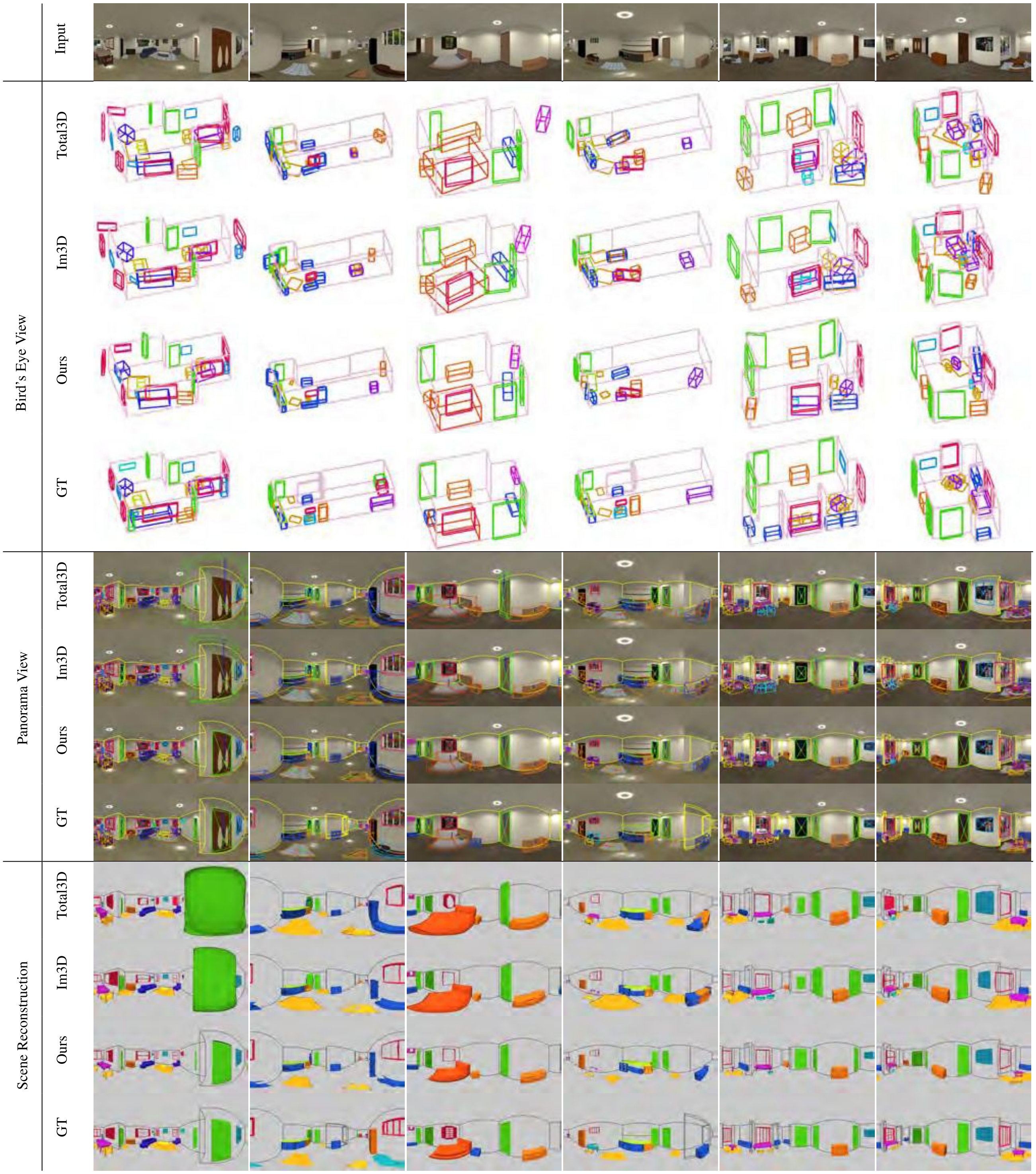

Figure 4: Qualitative comparison on 3D object detection and scene reconstruction. We compare object detection and compare scene reconstruction results with Total3D-Pers and Im3D-Pers in both bird's eye view and panorama format.

performance on object pose estimation and shape reconstruction. We also achieve more reasonable object-wall relations looking from the bird's eye view. For example in the third column, our method places the bed in the bottom-left corner right next to the wall, while Im3D and Total3D all fail with considerable error.

4.2. Ablation Study

To evaluate the proposed relation and object features and different parts of the proposed relation optimization, we

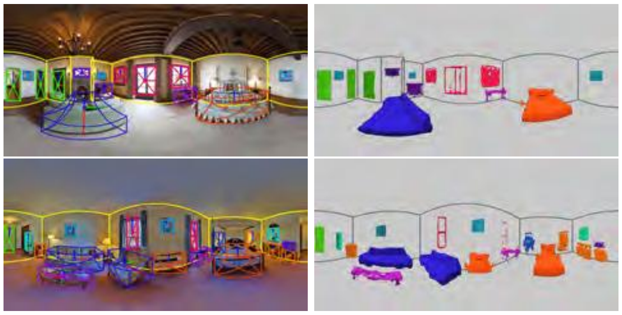

Figure 5: Generalization examples on PanoContext dataset.

Method	collision times among objects	objects having collision with			
		object	ceil	floor	wall
Total3D-Pers	3.45	4.96	0.09	2.70	2.68
Total3D-Pano	3.41	4.87	0.14	2.81	2.66
Im3D-Pers	3.16	4.54	0.03	1.79	2.42
Im3D-Pano	2.62	3.98	0.02	2.36	2.26
Ours (w/o. RO)	2.68	4.08	**0.01**	1.76	2.23
Ours (Full)	**0.86**	**1.50**	0.04	**0.45**	**1.33**

Table 3: Physical violation. We compare our methods with average collision times per scene to verify the effect of the proposed relation optimization. The collision detection is done with a toleration of 0.1m.

conduct ablation studies by removing different parts of our method and make comparisons on 3D object detection, collision, and relation estimation. For binary relations, *e.g.*, contact and relative distance, we compare Truth Positive Rate (TPR) and Truth Negative Rate (TNR). For rotation relation, we compare mean absolute error in degrees.

Do the Proposed Features Matter? As described in Sec. 3.2, we propose separate features for relation nodes and object nodes to encode explicit collision information and 3D geometry priors to RGCN. To show the necessity and effectiveness of these features, we remove the relation features (w/o. Fr) and the object features (w/o. Fo) respectively. As shown in Tab. 4, removing any feature will cause a drop in object detection mAP, as well as attachment classification. This is as expected as both relation feature and object feature provides critical information to measure the distance/collision among objects and layouts.

Does Relation Optimization Matter and How Each Loss Term Contributes? Our proposed relation optimization provides an end-to-end solution to hard code collision, contact, and rotation constraints into RGCN, with the purpose of obtaining more physically plausible and accurate detection results. By removing it (w/o. RO), we observe a great drop on mAP and average collision times per scene. We found our RO also improves Total3D and Im3D with predicted relation, and see supp. materials for more details. We also conduct study (w/o. E^c, w/o. E^r, w/o. E^o) on different terms to see how they contributes to the final improvement. The missing of collision term hurts the average collision times most, which further illustrates its importance on collision avoidance. We also observe even greater drops on mAP when removing E^r and E^o independently, which shows that our proposed terms collaborate together to improve 3D detection.

Method	obj attach	wall attach	obj rot (°)	wall rot (°)	mAP	avg col
w/o. Fr	0.30	0.71	64.21	52.03	33.22	0.74
w/o. Fo	0.46	0.74	**62.83**	**43.46**	33.32	0.83
w/o. RO	-	-	-	-	30.91	2.68
w/o. E^c	-	-	-	-	30.42	2.05
w/o. E^r	-	-	-	-	29.88	0.46
w/o. E^o	-	-	-	-	25.30	**0.09**
Full	**0.47**	**0.76**	62.97	43.72	**33.59**	0.86

Table 4: Ablation study. We compare F1 on binary-classified object-object attachment and object-wall attachment relations. For rotation relation classification, we compare mean absolute error in degrees. We evaluate 3D object detection with mAP of all 57 categories and physical violation with average collision times per scene.

FoV(°)	360	180	120	90	60	30
w/o. RO	30.91	27.4	24.77	25.90	26.16	25.51
Full	33.92	26.09	24.34	23.01	21.72	18.52

Table 5: mAP vs FoV. By narrowing the FoV of our model, the performance drops greatly, especially for our full model.

Is Panorama 360°FoV Helping RGCN and RO? Following the same procedure of splitting detection results by horizontal FoVs when making Total3D and Im3D working on panorama, we conduct ablation experiments on our proposed method by narrowing the FoV of each split. We compare our Full model, with or without RO, with different FoVs with mAP on full 57 categories in Tab. 5. The results show that limiting the message flow within small FoVs hurts the performance, which means that our RGCN and RO are really taking advantage of the whole scene context to estimate relations and optimize object detection.

5. Conclusion

This paper presents a novel method for holistic 3D scene understanding from a single full-view panorama image, which recovers the 3D room layout and the shape, pose, position, and semantic category of each object in the scene. To exploit the rich context information in the panorama image, we employ the graph neural network and design a novel context model to predict the relationship among objects and room layout, which will be further utilized by a novel differentiable relationship-based optimization module to refine the initial estimation. Due to the limitation of existing datasets for holistic 3D scene understanding, we present a new synthetic dataset. Experiments validate the effectiveness of each module in our method, and show that our method reaches the SoTA performance. Future directions could include simplifying the terms of RO and unifying different modules into a single framework.

Acknowledgement: This research was supported in part by National Natural Science Foundation of China (NSFC) under grants No.61872067 and No.61720106004.

References

[1] Iro Armeni, Sasha Sax, Amir R Zamir, and Silvio Savarese. Joint 2d-3d-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*, 2017. 2

[2] Armen Avetisyan, Tatiana Khanova, Christopher Choy, Denver Dash, Angela Dai, and Matthias Nießner. Scenecad: Predicting object alignments and layouts in rgb-d scans. In *Eur. Conf. Comput. Vis.*, pages 596–612, 2020. 4

[3] Lukas Biewald. Experiment tracking with weights and biases, 2020. Software available from wandb.com. 13

[4] Romain Brégier, Frédéric Devernay, Laetitia Leyrit, and James L Crowley. Symmetry aware evaluation of 3d object detection and pose estimation in scenes of many parts in bulk. In *Int. Conf. Comput. Vis. Worksh.*, pages 2209–2218, 2017. 2

[5] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niebner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. In *Int. Conf. 3D Vis.*, pages 667–676, 2017. 2

[6] Yixin Chen, Siyuan Huang, Tao Yuan, Siyuan Qi, Yixin Zhu, and Song-Chun Zhu. Holistic++ scene understanding: Single-view 3d holistic scene parsing and human pose estimation with human-object interaction and physical commonsense. In *Int. Conf. Comput. Vis.*, pages 8648–8657, 2019. 1, 2

[7] Shih-Han Chou, Cheng Sun, Wen-Yen Chang, Wan-Ting Hsu, Min Sun, and Jianlong Fu. 360-indoor: Towards learning real-world objects in 360deg indoor equirectangular images. In *IEEE Wint. Conf. Appli. Comput. Vis.*, pages 845–853, 2020. 2, 3, 11

[8] James M Coughlan and Alan L Yuille. Manhattan world: Compass direction from a single image by bayesian inference. In *Int. Conf. Comput. Vis.*, volume 2, pages 941–947, 1999. 2

[9] Saumitro Dasgupta, Kuan Fang, Kevin Chen, and Silvio Savarese. Delay: Robust spatial layout estimation for cluttered indoor scenes. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 616–624, 2016. 2

[10] Luca Del Pero, Joshua Bowdish, Daniel Fried, Bonnie Kermgard, Emily Hartley, and Kobus Barnard. Bayesian geometric modeling of indoor scenes. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2719–2726, 2012. 2

[11] Luca Del Pero, Joshua Bowdish, Bonnie Kermgard, Emily Hartley, and Kobus Barnard. Understanding bayesian rooms using composite 3d object models. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 153–160, 2013. 2

[12] Yilun Du, Zhijian Liu, Hector Basevi, Ales Leonardis, Bill Freeman, Josh Tenenbaum, and Jiajun Wu. Learning to exploit stability for 3d scene parsing. In *Adv. Neural Inform. Process. Syst.*, pages 1733–1743, 2018. 2

[13] Kosuke Fukano, Yoshihiko Mochizuki, Satoshi Iizuka, Edgar Simo-Serra, Akihiro Sugimoto, and Hiroshi Ishikawa. Room reconstruction from a single spherical image by higher-order energy minimization. In *Int. Conf. Pattern Recog.*, pages 1768–1773, 2016. 2

[14] 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. 2, 3, 4, 6

[15] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan Russell, and Mathieu Aubry. AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 216–224, 2018. 2

[16] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Int. Conf. Comput. Vis.*, pages 2961–2969, 2017. 3

[17] Varsha Hedau, Derek Hoiem, and David Forsyth. Recovering the spatial layout of cluttered rooms. In *Int. Conf. Comput. Vis.*, pages 1849–1856, 2009. 2

[18] Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, and Yichen Wei. Relation networks for object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3588–3597, 2018. 4

[19] 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.*, pages 206–217, 2018. 2, 6

[20] 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.*, pages 187–203, 2018. 2, 5

[21] Moos Hueting, Pradyumna Reddy, Vladimir Kim, Ersin Yumer, Nathan Carr, and Niloy Mitra. Seethrough: finding chairs in heavily occluded indoor scene images. *arXiv preprint arXiv:1710.10473*, 2017. 2

[22] Johnny Huynh. Url: https://jkh.me/files/tutorials/separating axis theorem for oriented bounding boxes.pdf. 2009. 4

[23] Hamid Izadinia, Qi Shan, and Steven M Seitz. Im2cad. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5134–5143, 2017. 2

[24] Chen-Yu Lee, Vijay Badrinarayanan, Tomasz Malisiewicz, and Andrew Rabinovich. Roomnet: End-to-end room layout estimation. In *Int. Conf. Comput. Vis.*, pages 4865–4874, 2017. 2

[25] David C Lee, Martial Hebert, and Takeo Kanade. Geometric reasoning for single image structure recovery. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2136–2143, 2009. 2

[26] Jun Li, Kai Xu, Siddhartha Chaudhuri, Ersin Yumer, Hao Zhang, and Leonidas Guibas. Grass: Generative recursive autoencoders for shape structures. *ACM Trans. Graph.*, 36(4):1–14, 2017. 2

[27] Cheng Liang and Xiaojian Liu. The research of collision detection algorithm based on separating axis theorem. *Int. J. Sci*, 2(10):110–114, 2015. 4

[28] Chen-Hsuan Lin, Chen Kong, and Simon Lucey. Learning efficient point cloud generation for dense 3d object reconstruction. In *AAAI*, pages 7114–7121, 2018. 2

[29] Arun Mallya and Svetlana Lazebnik. Learning informative edge maps for indoor scene layout prediction. In *Int. Conf. Comput. Vis.*, pages 936–944, 2015. 2

[30] 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. 6

[31] 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.*, pages 55–64, 2020. 2, 3, 5, 6, 11, 12

[32] Giovanni Pintore, Valeria Garro, Fabio Ganovelli, Enrico Gobbetti, and Marco Agus. Omnidirectional image capture on mobile devices for fast automatic generation of 2.5 d indoor maps. In *IEEE Wint. Conf. Appli. Comput. Vis.*, pages 1–9, 2016. 2

[33] Srikumar Ramalingam, Jaishanker K Pillai, Arpit Jain, and Yuichi Taguchi. Manhattan junction catalogue for spatial reasoning of indoor scenes. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3065–3072, 2013. 2

[34] Bokui Shen, Fei Xia, Chengshu Li, Roberto Martín-Martín, Linxi Fan, Guanzhi Wang, Shyamal Buch, Claudia D'Arpino, Sanjana Srivastava, Lyne P Tchapmi, et al. igibson, a simulation environment for interactive tasks in large realisticscenes. *arXiv preprint arXiv:2012.02924*, 2020. 5, 6, 11

[35] Cheng Sun, Chi-Wei Hsiao, Min Sun, and Hwann-Tzong Chen. Horizonnet: Learning room layout with 1d representation and pano stretch data augmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1047–1056, 2019. 2, 3

[36] Jonathan Tremblay, Thang To, Balakumar Sundaralingam, Yu Xiang, Dieter Fox, and Stan Birchfield. Deep object pose estimation for semantic robotic grasping of household objects. In *Conf. Rob. Learn.*, pages 306–316, 2018. 2

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Adv. Neural Inform. Process. Syst.*, pages 5998–6008, 2017. 4

[38] Johanna Wald, Armen Avetisyan, Nassir Navab, Federico Tombari, and Matthias Nießner. Rio: 3d object instance relocalization in changing indoor environments. In *Int. Conf. Comput. Vis.*, pages 7658–7667, 2019. 2

[39] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1912–1920, 2015. 2

[40] Jianxiong Xiao, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Recognizing scene viewpoint using panoramic place representation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2695–2702, 2012. 2

[41] Jiu Xu, Björn Stenger, Tommi Kerola, and Tony Tung. Pano2cad: Room layout from a single panorama image. In *IEEE Wint. Conf. Appli. Comput. Vis.*, pages 354–362, 2017. 2

[42] Hao Yang and Hui Zhang. Efficient 3d room shape recovery from a single panorama. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5422–5430, 2016. 2

[43] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *Eur. Conf. Comput. Vis.*, pages 670–685, 2018. 3, 4

[44] Shang-Ta Yang, Fu-En Wang, Chi-Han Peng, Peter Wonka, Min Sun, and Hung-Kuo Chu. Dula-net: A dual-projection network for estimating room layouts from a single rgb panorama. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3363–3372, 2019. 2

[45] Wenyan Yang, Yanlin Qian, Joni-Kristian Kämäräinen, Francesco Cricri, and Lixin Fan. Object detection in equirectangular panorama. In *Int. Conf. Pattern Recog.*, pages 2190–2195, 2018. 2, 3, 11

[46] Yang Yang, Shi Jin, Ruiyang Liu, Sing Bing Kang, and Jingyi Yu. Automatic 3d indoor scene modeling from single panorama. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3926–3934, 2018. 2

[47] Cheng Zhang, Zhaopeng Cui, Yinda Zhang, Bing Zeng, Marc Pollefeys, and Shuaicheng Liu. Holistic 3d scene understanding from a single image with implicit representation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8833–8842, 2021. 2, 3, 4, 6, 11, 12

[48] Yinda Zhang, Mingru Bai, Pushmeet Kohli, Shahram Izadi, and Jianxiong Xiao. Deepcontext: Context-encoding neural pathways for 3d holistic scene understanding. In *Int. Conf. Comput. Vis.*, pages 1192–1201, 2017. 2

[49] Yinda Zhang, Shuran Song, Ping Tan, and Jianxiong Xiao. Panocontext: A whole-room 3d context model for panoramic scene understanding. In *Eur. Conf. Comput. Vis.*, pages 668–686, 2014. 1, 2, 6

[50] Jia Zheng, Junfei Zhang, Jing Li, Rui Tang, Shenghua Gao, and Zihan Zhou. Structured3d: A large photo-realistic dataset for structured 3d modeling. In *Eur. Conf. Comput. Vis.*, pages 519–535, 2020. 2

[51] Chuhang Zou, Alex Colburn, Qi Shan, and Derek Hoiem. Layoutnet: Reconstructing the 3d room layout from a single rgb image. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2051–2059, 2018. 2

Supplementary Material

In this supplementary material, we provide synthetic dataset examples, network architecture details, and implementation details. We also provide visualization of relation optimization, 3D detection performance on all categories, more qualitative results, more comparison on Structured3D, and discussion of failure cases.

A. Dataset Examples

Our synthetic dataset provides various ground truth along with the RGB panorama images, including 2D object bounding boxes/BFoVs, watertight scene/object meshes, oriented 3D object bounding boxes, and 3D room layout. Our synthetic panorama scene understanding dataset also provides depth maps and semantic/instance segmentation images, which can be used by others. Some examples of our panorama dataset are shown in Fig. C. We also show object crops collected from the panorama images and extra object-centric images rendered from object models used for single image object reconstruction in Fig. D. The data generation code is built upon iGibson [34] and fully customized for panorama images.

B. Implementation Details

Dealing with Panorama Image As mentioned in the main paper, to deal with the continuity of panorama images, we parameterize the 2D bounding box with Bounding FoV (BFoV) [7, 45], and extend the panorama boundary before running 2D detector. Moreover, we change the object orientation θ to be the yaw angle of the object in the cropped perspective image coordinate. Compared to directly estimating the orientation in the world frame as in Im3D [47] and Total3D [31], our representation is more intuitive because it explicitly codes the transformation from the camera coordinates to the world coordinates. When calculating bounding box projection term e^{bp} in relation optimization, we rotate the camera to each detected bounding box center then do the projection of 3D bounding boxes, which avoids cross-border situations.

RGCN relation branch We design a relation branch for our RGCN to facilitate the relation estimation from the 512-dim representation vectors of object/relation nodes. We design the relation branch of RGCN as two-layer MLPs for each relation, which consist of a 256-dim FC layer, followed by a ReLU and Dropout layer with a drop factor of 0.5, and an output layer. The output layer is 1-dim for binary relations (*i.e.*, object-object/wall/floor/ceiling contact, inside or outside room, closer and farther to camera center between a pair of objects), and 8-dim for multi-class relations (*i.e.*, the angular difference between object and object/wall).

Hyper parameters For the weights of $\mathcal{L}_{ODN}$, we refer to Total3D [31] for detailed settings. For the loss weights of $\mathcal{L}_{RGCN}$ and the joint loss $\mathcal{L}$, we show the description of each output and its corresponding loss weight in Tab. A. For relation optimization, we weight each term by its confidence and importance. For example, 2D observations should be more confident, and collision terms should be weighted more if we consider physically plausible object poses important. We show the description of each term and its corresponding weight in Tab. B. In optimization, we use a gradient descend optimizer and set the learning rate to 1, steps to 100, and momentum to 0.9.

RGCN Output		Loss Weight	
Symbol	Description	Symbol	Value
rr	Object-object/wall relative rotation relation	λ_{rr}	10
oa	Object-object/wall attachment relation	λ_{oa}	10
fa	Object-floor attachment relation	λ_{fa}	10
ca	Object-ceiling attachment relation	λ_{ca}	10
rd	Object-object relative distance relation	λ_{rd}	10
δ	3D bounding box center offset	λ'_{δ}	10
d	3D bounding box distance	λ'_{d}	10
s	3D bounding box size	λ'_{s}	10
θ	Object orientation	λ'_{θ}	10

Table A: RGCN outputs and loss weights of $\mathcal{L}_{RGCN}$ and $\mathcal{L}$.

Term		Weight	
Symbol	Description	Symbol	Value
oc	Object-object collision	λ^{oc}	1
wc	Object-wall collision	λ^{wc}	1
fc	Object-floor collision	λ^{fc}	1
cc	Object-ceiling collision	λ^{cc}	1
rr	Object-object/wall relative rotation relation	λ^{rr}	0.1
oa	Object-object/wall attachment relation	λ^{oa}	1
fa	Object-floor attachment relation	λ^{fa}	1
ca	Object-ceiling attachment relation	λ^{ca}	1
rd	Object-object relative distance relation	λ^{rd}	0.01
δ	3D bounding box center offset	λ^{rd}	0.0001
d	3D bounding box distance	λ^{d}	0.01
s	3D bounding box size	λ^{s}	1
θ	Object orientation	λ^{θ}	0.001
bp	3D bounding box projection	λ^{bp}	10

Table B: Terms in relation optimization and the weight of each term.

Training All the borrowed networks (*i.e.*, Mask-RCNN, HorizonNet, ODN, LIEN, LDIF) are fine-tuned individually on our proposed dataset. Specifically, Mask-RCNN is fine-tuned from the weights pre-trained on COCO dataset, with batch size of 8 and learning rate of 2e-3 for 1e5 steps. HorizonNet is fine-tuned from the weights pre-trained on Structured3D dataset, with batch size of 6 and learning rate of 1e-4 for 50 epochs. ODN is fine-tuned from the weights pre-trained on SUN RGB-D, with batch size of 6 and learning rate of 1e-4 for 15 epochs. LIEN and LDIF are fine-tuned from the weights pre-trained on Pix3D, with batch size of 24 and learning rate of 2e-4 for 100 epochs. To make a fair comparison, all variation of Total3D [31] and Im3D [47] including the perspective and panorama version are also fine-tuned on our proposed dataset following the above process. For Total3D-Pers and Im3D-Pers, the

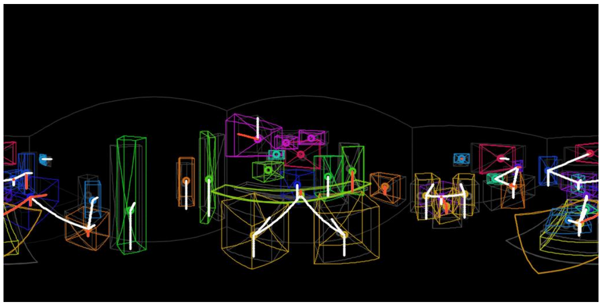

Figure A: Visualization of our proposed relation optimization. A PDF reader like **Adobe Acrobat Reader / KDE Okular** might be needed for displaying animated sequences. We also include more animation as a video along with this pdf file. The ground truth object bounding boxes are visualized with gray color for reference, while the current states are colorized. The attachment relations among objects, walls, floor, and ceiling are indicated by thick white lines, while the collisions are in red.

ODN and Scene Graph Convolutional Network (SGCN) are fine-tuned and tested with detection results obtained from split views. In addition, MGNet used by Total3D is fine-tuned from the weights pre-trained on Pix3D, with batch size of 16 and learning rate of 1e-4 for 100 epochs. To train our proposed RGCN, we generate the attachment relation ground truth by doing collision detection with a tolerance of 0.1m (*i.e.*, before collision detection, we expand the bounding box by 0.05m) on the ground-truth 3D object bounding boxes and the estimated layout walls. The other relations are calculated according to their definition directly. We first train our RGCN with only the pose refinement branch, with batch size of 16 and learning rate of 1e-4 for 35 epochs. Then we fine-tune it with relation estimation branch for 20 epochs using the same settings. Finally, we do an end-to-end training of ODN and RGCN with RO, with batch size of 1 and learning rate of 1e-5 for 10 epochs.

C. Visualization of Relation Optimization

To visualize the process of our proposed relation optimization, we present an animation in Fig. A. For demonstration, we add random noises to the ground truth object poses as the initial state, which simulates the inaccuracy of the initial pose estimation. We then use the relation generated from the ground truth poses to optimize the current poses (colorized) using our proposed method. We observe that as the optimization goes on, the position and orientation of the objects become closer to the ground truth, while the collisions are gradually resolved.

Metric	Total3D		Im3D	
	w/o. RO	w. RO	w/o. RO	w. RO
mAP (57 categories, ↑)	25.79	32.46	27.25	33.54
avg col (↓)	3.41	0.89	2.62	0.90

Table C: The improvement of RO on different methods. The improvement of 3D object detection is evaluated with mAP of all 57 categories and physical violation is evaluated with average collision times per scene.

Methods (Pano)	Initial Estimation	Object Reconstruction	GCN	RO	Total
Total3D	0.66 (Mask R-CNN, HorizonNet, ODN)	0.23 (MGN)	-	-	0.89
Im3D		5.92 (LIEN+LDIF)	0.03 (Scene GCN)	-	6.62
Ours			0.06 (Relation-based GCN)	4.74	11.38

Table D: Efficiency comparison. We use average time per scene in seconds to compare efficiency of different methods and modules (tested on a single GTX 1080Ti).

D. 3D Detection mAP on all 57 categories

In Tab. 1 of the main paper, we show 3D object detection results for 11 common categories. Here we show a complete quantitative evaluation on all 57 categories in Tab. G. Same as the conclusion made in the main paper, our method outperforms the SoTA with a large margin.

E. More Qualitative Comparison on 3D Detection and Scene Reconstruction

In Sec. 4.1 of the paper, we show qualitative comparisons on 3D detection and reconstruction. Here we provide more results in Fig. E. Compared to the SoTA methods [31, 47], our method produces significantly better 3D detection and reconstruction results. From the 3D detection and reconstruction results in panorama view, we observe that our method generates more accurate projections of reconstructed objects (*e.g.*, the mirror of (a), the sofa of (b) and (d), the door of (c)). From the 3D detection results in Bird's Eye View, we can see that our method generates more reasonable and physically plausible object poses (*e.g.*, (c), (e) have less object-wall collision and better rotation relations with walls).

F. Would RO Improve Other Methods?

In order to further evaluate the proposed relation optimization, we apply our RO on Total3D and Im3D using our predicted relation and their final results, and show the results in Tab. C. We can see that both methods still significantly benefit from the RO, which demonstrates that our RO is effective and robust to different initial estimates.

G. Run-time Efficiency

The efficiency comparison is shown in Tab. D. It is worth mentioning that implicit representation LDIF and RO are all

Method (Pano)	door	picture	table	sofa	chair	window	bed	bottom cabinet	chest
Total3D	28.65	0.06	38.83	31.64	23.71	4.78	74.09	37.08	62.07
Im3D	37.59	0.14	**49.47**	**37.24**	29.34	6.35	77.66	45.18	70.03
Ours (w/o. RO)	54.74	0.69	48.39	36.05	29.85	**13.49**	81.13	48.33	72.08
Ours (Full)	**57.73**	**1.24**	49.10	37.02	**29.95**	12.28	**81.15**	**48.76**	**74.26**

Method (Pano)	sink	fridge	bathtub	shelf	mirror	toilet	counter	standing tv	mean
Total3D	28.24	68.82	69.36	10.36	0.04	19.88	19.17	2.12	30.52
Im3D	**28.57**	71.39	**73.93**	9.78	0.92	15.04	19.17	2.52	33.78
Ours (w/o. RO)	27.43	**73.35**	**73.93**	**15.84**	1.47	32.87	19.17	9.61	37.55
Ours (Full)	27.93	**73.35**	**73.93**	15.76	**3.19**	**65.54**	19.17	**13.20**	**40.21**

Table E: 3D object detection comparison on Structured3D. We evaluate on the 17 iGibson categories mapped from 20 Structured3D categories and use mean average precision (mAP) with the threshold of 3D bounding box IoU set at 0.15 as the evaluation metric.

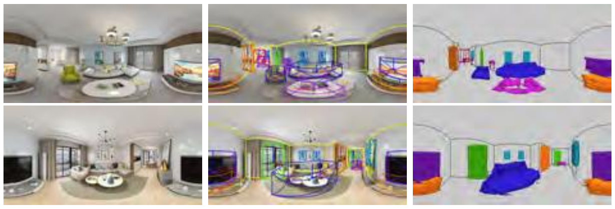
(a) Input (b) 3D Detection (c) Reconstruction

Figure B: Qualitative results of our model on Structured3D.

implemented with PyTorch, and can be further optimized, *e.g.*, using CUDA, to improve the efficiency.

H. Experiment on Structured3D

Since Structured3D provides the ground truth of object pose and layout, we can train our model up to RGCN. Due to the lack of mesh ground truth, we load the object reconstruction model with weights trained on iGibson. Furthermore, since the object reconstruction model requires category label as input, we map the object categories from Structured3D to iGibson. We found overlapping categories between two datasets, which ends up with 20 structure3D categories mapped to 17 iGibson categories. Specifically, “cabinet”, “bookshelf”, “desk”, “shelves”, “dresser”, “floor mat”, “television”, “box”, “nightstand” in Structure3D are mapped to “bottom cabinet”, “shelf”, “table”, “shelf”, “bottom cabinet”, “carpet”, “standing tv”, “chest”, “chest” in iGibson, and others are mapped with the same category name. It is also worth mentioning that the bounding box GT of objects in Structured3D is not accurate or physical plausible, which makes it difficult to produce rich relation GT and to better refine the object poses with observation and collision terms. So the weights of relation optimization terms need to be tuned to match the condition. Specifically, we fix the weights of relation terms and auto-search the learning rate of gradient descend optimizer and other weights of relation optimization terms around the original settings used on iGibson. In summary, we train object detection on overlapping categories and set weights of RO terms with auto-search [3]. The auto-searched weights are shown in Tab. F. Qualitative results are shown in Fig. B. We can see that our method performs well with good layout, pose and shape estimation although there is no ground truth for shapes. We compare 3D object detection against existing methods quantitatively in Tab. E. The results show that our method still outperforms SoTA methods significantly, and RO plays a big role in improving the mAP.

Parameter	Value	Parameter	Value
λ^{rd}	0.0040	λ^{oc}	0.0157
λ^{d}	0.1404	λ^{wc}	0.2625
λ^{s}	6.0502	λ^{fc}	0.3182
λ^{θ}	0.0003	λ^{cc}	0.2036
λ^{bp}	0.2895	learning rate	0.0124

Table F: Auto-searched hyperparameters used on Structured3D, including weights of relation optimization terms and learning rate of relation optimization.

I. Failure Cases

We show failure cases in Fig. F. One scenario that our pipeline fails is when heavy occlusion happens (*i.e.*, one of the doors on the right in (d), the second door on the left in (a)), which tends to shrink the size of the object in order to favor the projection term with the partial 2D observation. A possible solution might be to understand the occlusion and learn the mask behind occluder. Another scenario is when the 2D detector has multiple detection results on a single object (*i.e.*, the wardrobe on the right in (a), the sofa on the right in (b), the drawer on the left in (c)), which lead to two overlapped object reconstructions in the same place but not sufficient to trigger non-maximum suppression. This might be solved by refining the category prediction of the 2D detector in the RGCN, which will presumably fix detected object categories with mistakes (or set reduplicated object to void) with a better understanding of the 3D scene context. The last scenario is when HorizonNet fails to generate layouts for rooms that don't satisfy the Manhattan-world assumption (*i.e.*, the wall on the left side in (e)), our pipeline will fail to optimize the object pose based on the wrong wall orientation. Also when object-wall rotation relation is estimated badly (*i.e.*, the window in (b)), the orientation cannot be optimized properly.

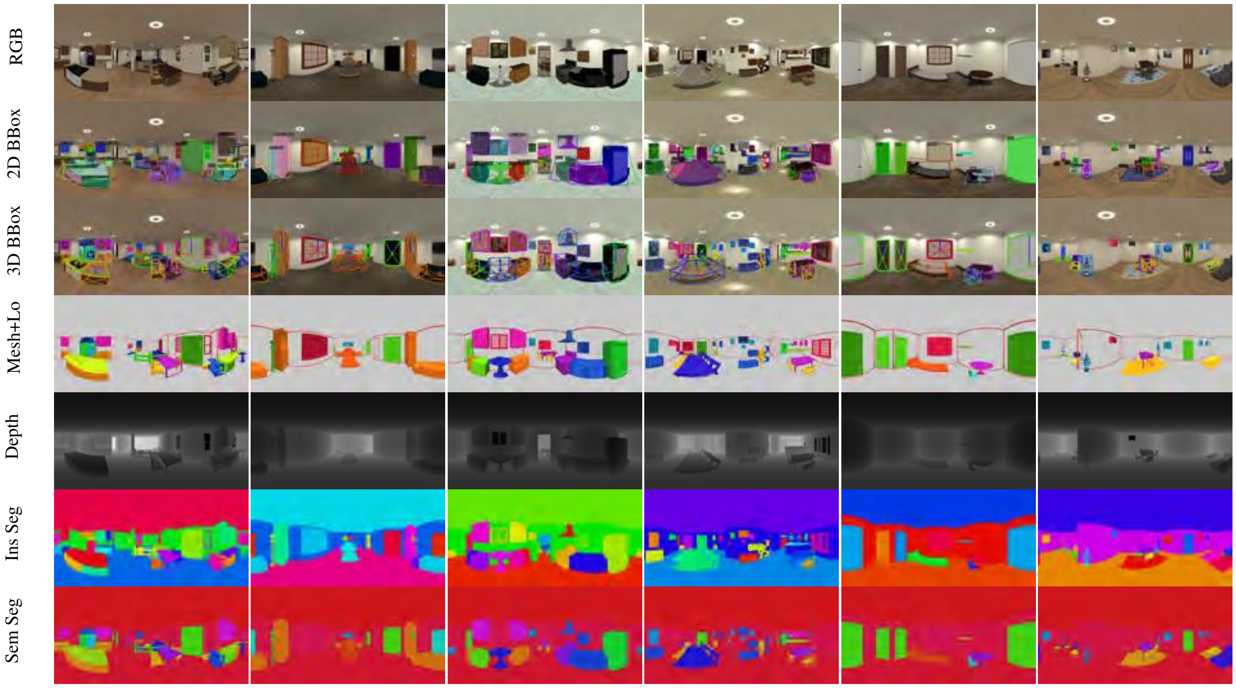

Figure C: Samples of our proposed panorama 3D scene understanding dataset.

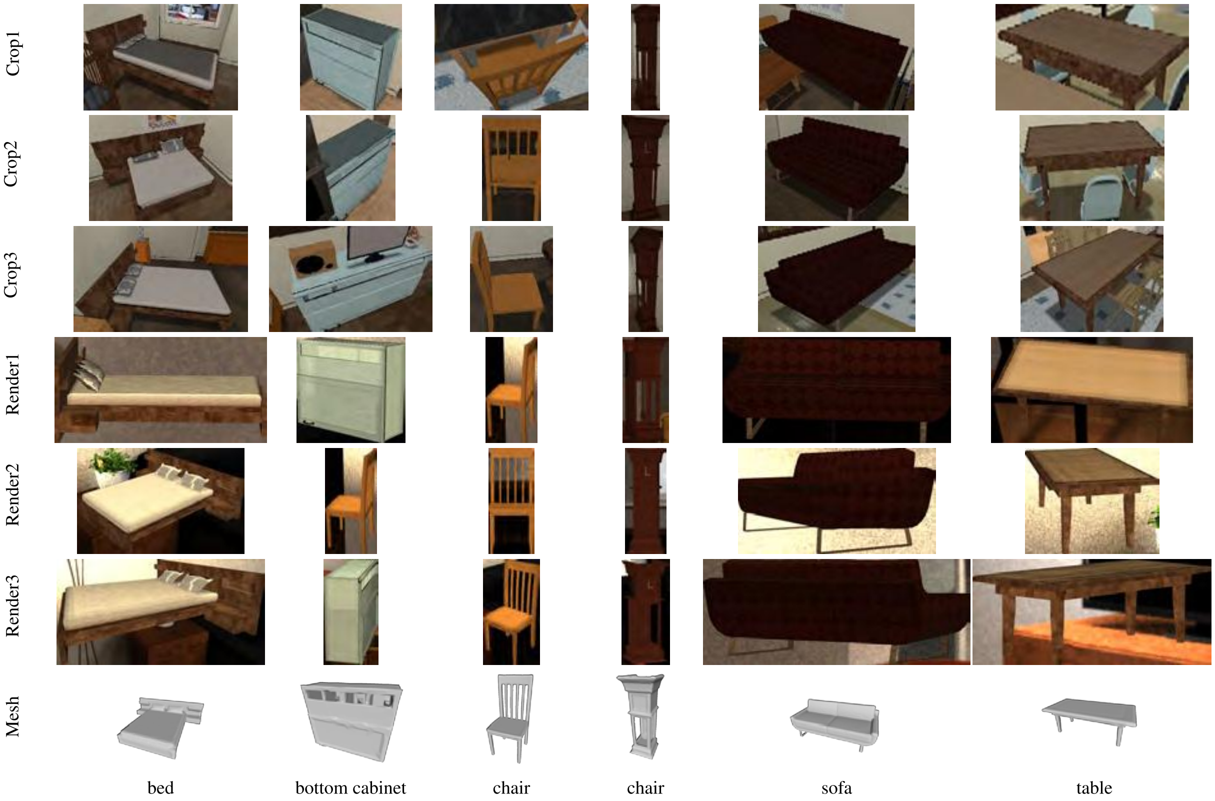

Figure D: Samples of dataset used for single image object reconstruction.

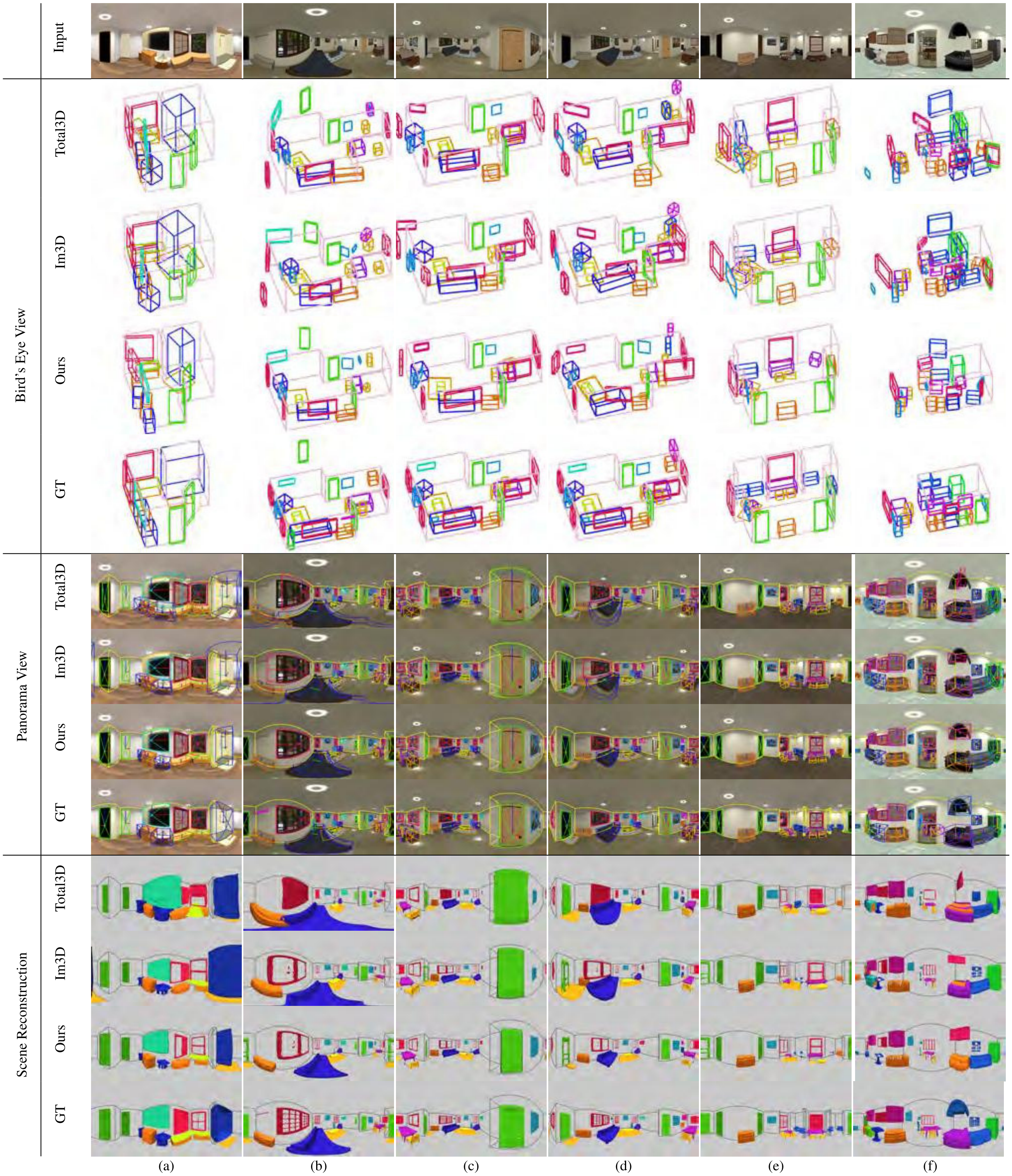

Figure E: More Qualitative comparison on 3D object detection and scene reconstruction.

Method	chair	sofa	table	fridge	sink	door	floor lamp	bottom cabinet	top cabinet	sofa chair	dryer
Total3D-Pers	13.71	68.06	30.55	36.02	69.84	11.88	12.57	35.56	19.19	64.29	41.36
Total3D-Pano	20.84	69.65	31.79	43.13	68.42	10.27	16.42	34.42	20.83	62.38	33.78
Im3D-Pers	30.23	**75.23**	44.16	52.56	76.46	14.91	9.99	45.51	23.37	**80.11**	53.28
Im3D-Pano	33.08	72.15	37.43	70.45	75.20	11.58	6.06	43.28	18.99	78.46	41.02
Ours (w/o. RO)	**33.57**	75.18	38.65	71.97	**80.66**	19.94	18.29	50.67	29.05	79.42	**60.07**
Ours (Full)	27.78	73.96	**46.85**	**74.22**	75.29	**21.43**	**20.69**	**52.03**	**50.39**	77.09	59.91

Method	window	carpet	picture	oven	bottom cabinet no top	counter	dish washer	shelf	coffee table	mirror	toilet
Total3D-Pers	2.92	0.05	0.01	31.33	34.40	0.78	43.54	10.93	39.72	0.11	90.00
Total3D-Pano	3.07	0.05	0.02	29.81	32.48	1.11	48.39	9.57	49.52	0.64	90.00
Im3D-Pers	3.52	0.12	0.00	31.28	**47.45**	**2.60**	51.47	15.01	59.02	0.81	90.00
Im3D-Pano	3.42	0.01	0.01	29.06	44.79	1.34	43.80	15.41	56.82	0.16	90.00
Ours (w/o. RO)	6.94	0.12	0.03	32.52	46.42	1.83	59.78	15.58	**61.17**	2.42	90.00
Ours (Full)	**9.56**	**0.65**	**0.21**	**34.50**	44.17	1.25	**63.19**	**22.65**	50.69	**6.12**	90.00

Method	wall mounted tv	loud speaker	console table	fence	chest	standing tv	table lamp	speaker system	bathtub	plant	treadmill
Total3D-Pers	**0.52**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.77	11.48	0.00
Total3D-Pano	0.24	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.77	3.10	0.00
Im3D-Pers	0.03	0.00	0.00	0.00	0.00	0.00	6.06	0.00	10.26	10.34	0.00
Im3D-Pano	0.08	0.00	0.00	0.00	0.00	0.00	3.17	0.00	10.26	12.69	0.00
Ours (w/o. RO)	0.24	0.00	0.00	0.00	0.00	0.00	**10.53**	0.00	10.26	8.35	0.00
Ours (Full)	0.14	0.00	0.00	0.00	0.00	0.00	2.79	0.00	**41.02**	**16.46**	0.00

Method	washer	stool	trash can	stove	bed	office chair	shower	towel rack	piano	mAP	
Total3D-Pers	35.06	29.09	24.45	44.44	71.87	0.00	**100.00**	25.00	55.56	25.11	
Total3D-Pano	32.21	29.09	25.84	44.44	**73.22**	0.00	72.73	**50.00**	75.00	25.79	
Im3D-Pers	**36.50**	29.09	22.02	44.44	**73.22**	0.00	81.82	**50.00**	83.33	29.86	
Im3D-Pano	**36.50**	29.09	39.13	44.44	**73.22**	0.00	80.17	0.00	43.33	27.25	
Ours (w/o. RO)	**36.50**	29.09	31.15	44.44	71.57	0.00	81.82	0.00	**100.00**	30.91	
Ours (Full)	**36.50**	29.09	**66.23**	44.44	71.57	0.00	**100.00**	0.00	**100.00**	**33.59**	

Table G: 3D object detection comparison on full 57 categories. Some categories existing in training scenes do not exist in testing scenes, or vice versa, which is the main reason for some of the 0 mAP cases.

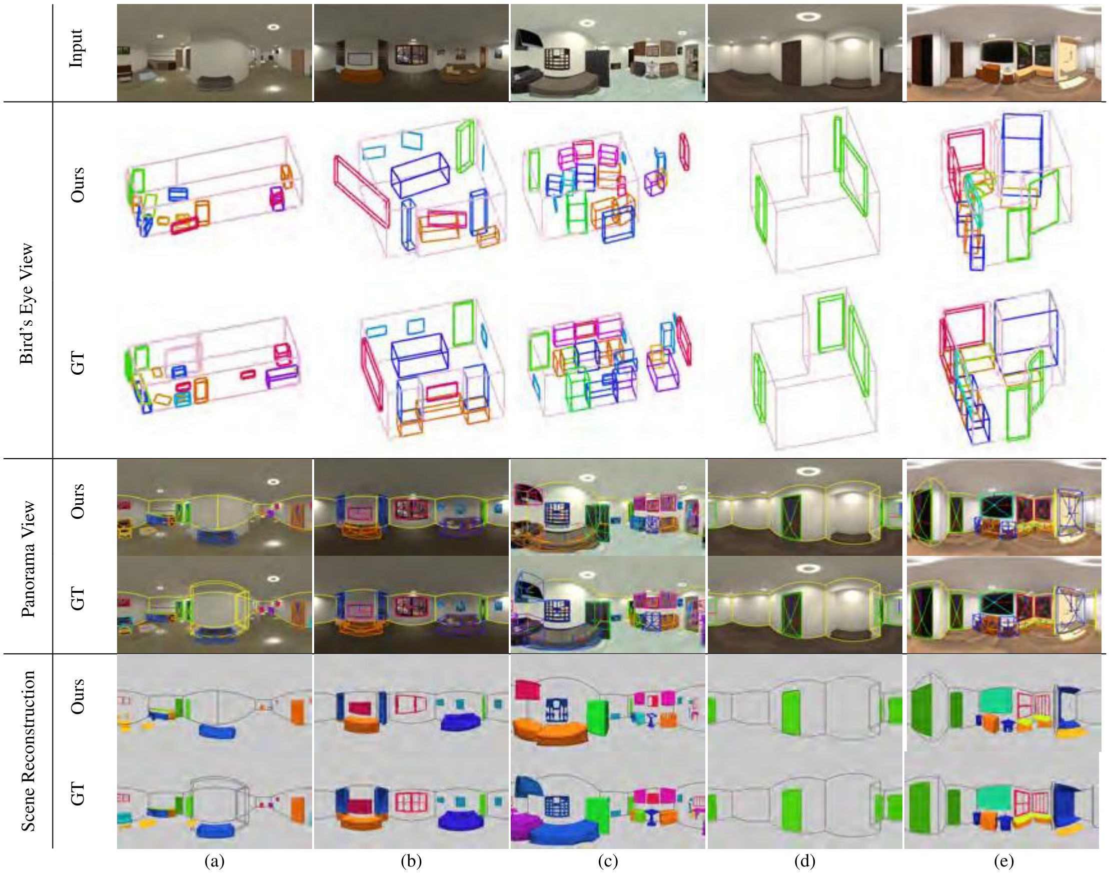

Figure F: Failure cases.