Total3DUnderstanding: Joint Layout, Object Pose and Mesh Reconstruction for Indoor Scenes from a Single Image

Yinyu Nie[1,2,3,†], Xiaoguang Han[2,3,*], Shihui Guo[4], Yujian Zheng[2,3], Jian Chang[1], Jian Jun Zhang[1]
[1]Bournemouth University [2]The Chinese University of Hong Kong, Shenzhen
[3]Shenzhen Research Institute of Big Data [4]Xiamen University

Abstract

Semantic reconstruction of indoor scenes refers to both scene understanding and object reconstruction. Existing works either address one part of this problem or focus on independent objects. In this paper, we bridge the gap between understanding and reconstruction, and propose an end-to-end solution to jointly reconstruct room layout, object bounding boxes and meshes from a single image. Instead of separately resolving scene understanding and object reconstruction, our method builds upon a holistic scene context and proposes a coarse-to-fine hierarchy with three components: 1. room layout with camera pose; 2. 3D object bounding boxes; 3. object meshes. We argue that understanding the context of each component can assist the task of parsing the others, which enables joint understanding and reconstruction. The experiments on the SUN RGB-D and Pix3D datasets demonstrate that our method consistently outperforms existing methods in indoor layout estimation, 3D object detection and mesh reconstruction.

1. Introduction

Semantic reconstruction from an indoor image shows its unique importance in applications such as interior design and real estate. In recent years, this topic has received a rocketing interest from researchers in both computer vision and graphics communities. However, the inherent ambiguity in depth perception, the clutter and complexity of real-world environments make it still challenging to fully recover the scene context (both semantics and geometry) merely from a single image.

Previous works have attempted to address it via various approaches. **Scene understanding** methods [38, 14, 3] obtain room layout and 3D bounding boxes of indoor objects without shape details. **Scene-level reconstruction** methods recover object shapes using contextual knowledge (room layout and object locations) for scene reconstruction, but most methods currently adopt depth or voxel representations [39, 22, 46, 19]. Voxel-grid presents better shape description than boxes, but its resolution is still limited, and the improvement of voxel quality exponentially increases the computational cost, which is more obvious in scene-level reconstruction. **Mesh-retrieval** methods [17, 15, 16] improve the shape quality in scene reconstruction using a 3D model retrieval module. As these approaches require iterations of rendering or model search, the mesh similarity and time efficiency depend on the size of the model repository and raise further concerns. **Object-wise mesh reconstruction** exhibits the advantages in both efficiency and accuracy [50, 10, 30, 18, 9], where the target mesh is end-to-end predicted in its own object-centric coordinate system. For scene-level mesh reconstruction, predicting objects as isolated instances may not produce ideal results given the challenges of object alignment, occlusion relations and miscellaneous image background. Although Mesh R-CNN [9] is capable of predicting meshes for multiple objects from an image, its object-wise approach still ignores scene understanding and suffers from the artifacts of mesh generation on cubified voxels. So far, to the best of authors' knowledge, few works take into account both mesh reconstruction

Figure 1: From a single image (left), we simultaneously predict the contextual knowledge including room layout, camera pose, and 3D object bounding boxes (middle) and reconstruct object meshes (right).

[†] Work done during visiting CUHKSZ and SRIBD.
[*] Corresponding author: hanxiaoguang@cuhk.edu.cn

and scene context (room layout, camera pose and object locations) for total 3D scene understanding.

To bridge the gap between scene understanding and object mesh reconstruction, we unify them together with joint learning, and simultaneously predict room layout, camera pose, 3D object bounding boxes and meshes (Figure 1). The insight is that object meshes in a scene manifest spatial occupancy that could help 3D object detection, and the 3D detection provides with object alignment that enables object-centric reconstruction at the instance-level. Unlike voxel grids, coordinates of reconstructed meshes are differentiable, thus enabling the joint training by comparing the output mesh with the scene point cloud (e.g. on SUN RGB-D [41]). With the above settings, we observe that the performance on scene understanding and mesh reconstruction can make further progress and reach the state-of-the-art on the SUN RGB-D [41] and Pix3D [42] datasets. In summary, we list our contributions as follows:

- We provide a solution to automatically reconstruct room layout, object bounding boxes, and meshes from a single image. To our best knowledge, it is the first work of end-to-end learning for comprehensive 3D scene understanding with mesh reconstruction at the instance level. This integrative approach shows the complementary role of each component and reaches the state-of-the-art on each task.

- We propose a novel density-aware topology modifier in object mesh generation. It prunes mesh edges based on local density to approximate the target shape by progressively modifying mesh topology. Our method directly tackles the major bottleneck of [30], which is in the requirement of a strict distance threshold to remove detached faces from the target shape. Compared with [30], our method is robust to diverse shapes of indoor objects under complex backgrounds.

- Our method takes into account the attention mechanism and multilateral relations between objects. In 3D object detection, the object pose has an implicit and multilateral relation with surroundings, especially in indoor rooms (e.g., bed, nightstand, and lamp). Our strategy extracts the latent features for better deciding object locations and poses, and improves 3D detection.

2. Related Work

Single-view scene reconstruction presents a challenging task since the first work [37] in monocular shape inference. For indoor reconstruction, the difficulties increase with the complexity of clutter, occlusion and object diversity, etc.

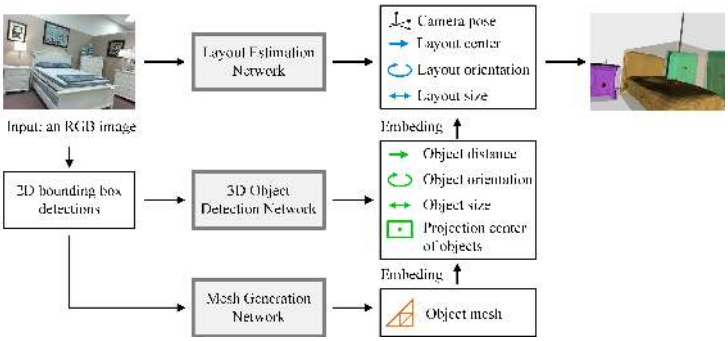
Early works focus on room layout estimation [12, 21, 25, 5, 35] to represent rooms with a cuboid. With the advance of CNNs, more methods are developed to estimate object poses beyond the layout [7, 14, 1]. Still, these methods are limited to the 3D bounding box prediction of each furniture. To recover object shapes, some methods [17, 16, 15] adopt shape retrieval to search for appearance-similar models from a dataset. However, its accuracy and efficiency directly depend on the size and diversity of the dataset.

Scene reconstruction at the instance level remains problematic because of the large number of object categories with diverse geometry and topology. To first address single object reconstruction, approaches represent shapes in the form of point cloud [8, 26, 20, 29], patches [10, 51] and primitives [45, 47, 32, 6] which are adaptable to complex topology but require post-processing to obtain meshes. The structure of voxel grids [4, 23, 49] is regular while suffering from the balance between resolution and efficiency, demanding for Octree to improve local details [36, 44, 51]. Some methods produce impressive mesh results using signed distance functions [31] and implicit surfaces [2, 28, 52, 27]. However, they are time-consuming and computationally intensive, making it impractical to reconstruct all objects in a scene. Another popular approach is to reconstruct meshes from a template [50, 10, 18], but the topology of the reconstructed mesh is restricted. So far, the state-of-art approaches modify the mesh topology to approximate the ground-truth [30, 43]. However, existing methods estimate 3D shapes in the object-centric system, which cannot be applied to scene reconstruction directly.

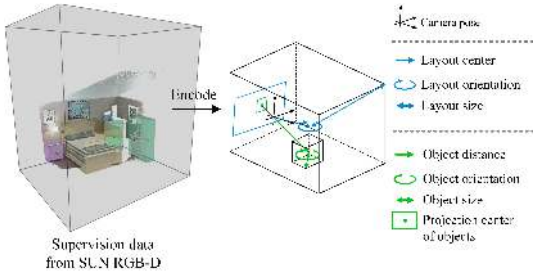
The most relevant works to us are [22, 46, 19, 9], which take an image as input and predict multiple object shapes in a scene. However, the methods [22, 46, 19] are designed for voxel reconstruction with limited resolution. Mesh R-CNN [9] produces object meshes, but still treats objects as isolated geometries without considering the scene context (room layout, object pose, etc.). It uses cubified voxels as an intermediate representation and suffers from the limited resolution. Different from them, our method connects the object-centric reconstruction with 3D scene understanding, enabling joint learning of room layout, camera pose, object bounding boxes, and meshes from a single image.

3. Method

We illustrate our overview in Figure 2a. The network architecture follows a 'box-in-the-box' manner and consists of three modules: 1. Layout Estimation Network (LEN); 2. 3D Object Detection Network (ODN); 3. Mesh Generation Network (MGN). From a single image, we first predict 2D object bounding boxes with Faster R-CNN [34]. LEN takes the full image to produce the camera pose and the layout bounding box. Given the 2D object detections, ODN predicts their 3D bounding box in the camera system, while MGN generates the mesh geometry in their object-centric system. We reconstruct the full scene mesh by embedding the outputs of all networks together with joint training



(a) Architecture of the scene reconstruction network



(b) Parameterization of the learning targets

Figure 2: Overview of our approach. (a) The hierarchy of our method follows a 'box-in-the-box' manner using three modules: the Layout Estimation Network (LEN), 3D Object Detection Network (ODN) and Mesh Generation Network (MGN). A full scene mesh is reconstructed by embedding them together with joint inference. (b) The parameterization of our learning targets in LEN and ODN [14].

and inference, where object meshes from MGN are scaled and placed into their bounding boxes (by ODN) and transformed into the world system with the camera pose (by LEN). The details of each network are described below.

3.1. 3D Object Detection and Layout Estimation

To make the bounding box of layout and objects learnable, we parameterize a box as the prior work [14] (Figure 2b). We set up the world system located at the camera center with its vertical (y-) axis perpendicular to the floor, and its forward (x-) axis toward the camera, such that the camera pose $\mathbf{R}(\beta, \gamma)$ can be decided by the pitch and roll angles (β, γ). In the world system, a box can be determined by a 3D center $\boldsymbol{C} \in \mathbb{R}^3$, spatial size $\boldsymbol{s} \in \mathbb{R}^3$, orientation angle $\theta \in [-\pi, \pi)$. For indoor objects, the 3D center $\boldsymbol{C}$ is represented by its 2D projection $\boldsymbol{c} \in \mathbb{R}^2$ on the image plane with its distance $d \in \mathbb{R}$ to the camera center. Given the camera intrinsic matrix $\mathbf{K} \in \mathbb{R}^3$, $\boldsymbol{C}$ can be formulated by:

$$\boldsymbol{C} = \mathbf{R}^{-1}(\beta, \gamma) \cdot d \cdot \frac{\mathbf{K}^{-1}[\boldsymbol{c}, 1]^T}{\|\mathbf{K}^{-1}[\boldsymbol{c}, 1]^T\|_2}. \quad (1)$$

The 2D projection center $\boldsymbol{c}$ can be further decoupled by $\boldsymbol{c}^b + \boldsymbol{\delta}$. $\boldsymbol{c}^b$ is the 2D bounding box center and $\boldsymbol{\delta} \in \mathbb{R}^2$ is the offset to be learned. From the 2D detection $\boldsymbol{I}$ to its 3D bounding box corners, the network can be represented as a function by $\mathbf{F}(\boldsymbol{I}|\boldsymbol{\delta}, d, \beta, \gamma, \boldsymbol{s}, \theta) \in \mathbb{R}^{3\times 8}$. The ODN estimates the box property $(\boldsymbol{\delta}, d, \boldsymbol{s}, \theta)$ of each object, and the LEN decides the camera pose $\mathbf{R}(\beta, \gamma)$ with the layout box $(\boldsymbol{C}, \boldsymbol{s}^l, \theta^l)$.

Object Detection Network (ODN). In indoor environments, object poses generally follow a set of interior design principles, making it a latent learnable pattern. Previous works either predict 3D boxes object-wisely [14, 46] or only consider pair-wise relations [19]. In our work, we assume each object has a *multi-lateral relation* between its

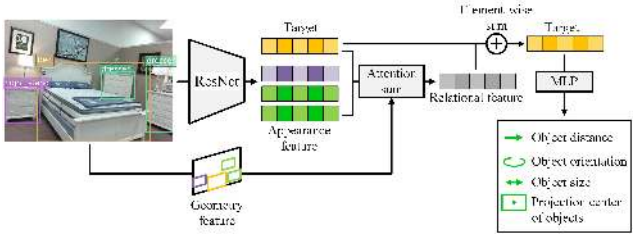


Figure 3: 3D Object Detection Network (ODN)

surroundings, and take all in-room objects into account in predicting its bounding box. The network is illustrated in Figure 3. Our method is inspired by the consistent improvement of attention mechanism in 2D object detection [13]. For 3D detection, we first object-wisely extract the appearance feature with ResNet-34 [11] from 2D detections, and encode the relative position and size between 2D object boxes into geometry feature with the method in [13, 48]. For each target object, we calculate its **relational feature** to the others with the object relation module [13]. It adopts a piece-wise feature summation weighted by the similarity in appearance and geometry from the target to the others, which we call '**attention sum**' in Figure 3. We then element-wisely add the relational feature to the target and regress each box parameter in $(\boldsymbol{\delta}, d, \boldsymbol{s}, \theta)$ with a two-layer MLP. For indoor reconstruction, the object relation module reflects the inherent significance in the physical world: objects generally have stronger relations with the others which are neighboring or appearance-similar. We demonstrate its effectiveness in 3D object detection in our ablation analysis.

Layout Estimation Network (LEN). The LEN predicts the camera pose $\mathbf{R}(\beta, \gamma)$ and its 3D box $(\boldsymbol{C}, \boldsymbol{s}^l, \theta^l)$ in the world system. In this part, we employ the same architecture as ODN but remove the relational feature. $(\beta, \gamma, \boldsymbol{C}, \boldsymbol{s}^l, \theta^l)$

are regressed with two fully-connected layers for each target after the ResNet. Similar to [14], the 3D center $\boldsymbol{C}$ is predicted by learning an offset to the average layout center.

3.2. Mesh Generation for Indoor Objects

Our Mesh Generation Network directly tackles the major issue with one recent work, Topology Modification Network (TMN) [30]: TMN approximates object shapes by deforming and modifying the mesh topology, where a predefined distance threshold is required to remove detached faces from the target shape. However, it is nontrivial to give a general threshold for different scales of object meshes (see Figure 5e). One possible reason is that indoor objects have a large shape variance among different categories. Another one is that complex backgrounds and occlusions often cause the failure of estimating a precise distance value.

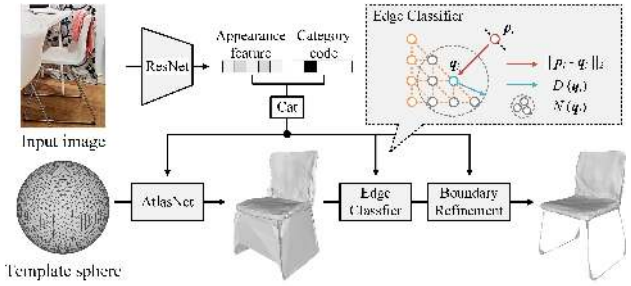


Figure 4: Mesh Generation Network (MGN). Our method takes as input a detected object which is vulnerable to occlusions, and outputs a plausible mesh.

Density v.s. Distance. Different from TMN where a strict distance threshold is used for topology modification, we argue that whether to reserve a face or not should be determined by its local geometry. In this part, we propose an adaptive manner that modifies meshes based on the local **density** of the ground-truth. We set $\boldsymbol{p}_i \in \mathbb{R}^3$ as a point on our reconstructed mesh, and $\boldsymbol{q}_i \in \mathbb{R}^3$ corresponds to its nearest neighbor on the ground-truth (see Figure 4). We design a binary classifier $f(*)$ to predict whether $\boldsymbol{p}_i$ is close to the ground-truth mesh in Equation 2:

$$f(\boldsymbol{p}_i) = \begin{cases} \text{False} & \|\boldsymbol{p}_i - \boldsymbol{q}_i\|_2 > D(\boldsymbol{q}_i) \\ \text{True} & \text{otherwise} \end{cases}, \quad (2)$$
$$D(\boldsymbol{q}_i) = \max_{\boldsymbol{q}_m, \boldsymbol{q}_n \in N(\boldsymbol{q}_i)} \min \|\boldsymbol{q}_m - \boldsymbol{q}_n\|_2, m \neq n$$

where $N(\boldsymbol{q}_i)$ are the neighbors of $\boldsymbol{q}_i$ on the ground-truth mesh, and $D(\boldsymbol{q}_i)$ is defined as its local density. This classifier is designed by our insight that: in shape approximation, a point should be reserved if it belongs to the neighbors $N(*)$ of the ground-truth. We also observe that this classifier shows better robustness with different mesh scales than using a distance threshold (see Figure 5).

Edges v.s. Faces. Instead of removing faces, we choose to cut mesh edges for topology modification. We randomly sample points on mesh edges and use the classifier $f(*)$ to cut edges on which the average classification score is low. It is from the consideration that cutting false edges can reduce incorrect connections penalized by the edge loss [50] and create compact mesh boundaries.

Mesh Generation Network. We illustrate our network architecture in Figure 4. It takes a 2D detection as input and uses ResNet-18 to produce image features. We encode the detected object category into a one-hot vector and concatenate it with the image feature. It is from our observation that the category code provides shape priors and helps to approximate the target shape faster. The augmented feature vector and a template sphere are fed into the decoder in AtlasNet [10] to predict deformation displacement on the sphere and output a plausible shape with unchanged topology. The edge classifier has the same architecture with the shape decoder, where the last layer is replaced with a fully connected layer for classification. It shares the image feature, takes the deformed mesh as input and predicts the $f(*)$ to remove redundant meshes. We then append our network with a boundary refinement module [30] to refine the smoothness of boundary edges and output the final mesh.

3.3. Joint Learning for Total 3D Understanding

In this section, we conclude the learning targets with the corresponding loss functions, and describe our joint loss for end-to-end training.

Individual losses. ODN predicts $(\boldsymbol{\delta}, d, \boldsymbol{s}, \theta)$ to recover the 3D object box in the camera system, and LEN produces $\left(\beta, \gamma, \boldsymbol{C}, \boldsymbol{s}^l, \theta^l\right)$ to represent the layout box, along with the camera pose to transform 3D objects into the world system. As directly regressing absolute angles or length with L2 loss is error-prone [14, 33]. We keep inline with them by using the classification and regression loss $\mathcal{L}^{cls,reg} = \mathcal{L}^{cls} + \lambda_r \mathcal{L}^{reg}$ to optimize $\left(\theta, \theta^l, \beta, \gamma, d, \boldsymbol{s}, \boldsymbol{s}^l\right)$. We refer readers to [14] for details. As $\boldsymbol{C}$ and $\boldsymbol{\delta}$ are calculated by the offset from a pre-computed center, we predict them with L2 loss. For MGN, we adopt the Chamfer loss $\mathcal{L}_c$, edge loss $\mathcal{L}_e$, boundary loss $\mathcal{L}_b$ as [10, 50, 30] with our cross-entropy loss $\mathcal{L}_{ce}$ for classifying edges in mesh modification.

Joint losses. We define the joint loss between ODN, LEN and MGN based on two insights: 1. The camera pose estimation should improve 3D object detection, and vice versa; 2. object meshes in a scene present spatial occupancy that should benefit the 3D detection, and vice versa. For the first, we adopt the cooperative loss $\mathcal{L}_{co}$ from [14] to ensure the consistency between the predicted world coordinates of layout & object boxes and the ground-truth. For the second, we require the reconstructed meshes close to their point cloud in the scene. It exhibits global constraints by aligning mesh coordinates with the ground-truth. We define the global loss

as the partial Chamfer distance [10]:

$$\mathcal{L}_g = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{|\mathbb{S}_i|}\sum_{\boldsymbol{q}\in\mathbb{S}_i}\min_{\boldsymbol{p}\in\mathbb{M}_i}\|\boldsymbol{p}-\boldsymbol{q}\|_2^2, \tag{3}$$

where $\boldsymbol{p}$ and $\boldsymbol{q}$ respectively indicate a point on a reconstructed mesh $\mathbb{M}_i$ and the ground-truth surface $\mathbb{S}_i$ of i-th object in the world system. N is the number of objects and $|\mathbb{S}_i|$ denotes the point number on $\mathbb{S}_i$. Unlike single object meshes, real-scene point clouds are commonly coarse and partially covered (scanned with depth sensors), thus we do not use the Chamfer distance to define $\mathcal{L}_g$. All the loss functions in joint training can be concluded as:

$$\begin{aligned}\mathcal{L} = &\sum_{x\in\{\boldsymbol{\delta},d,\boldsymbol{s},\theta\}}\lambda_x\mathcal{L}_x + \sum_{y\in\{\beta,\gamma,\boldsymbol{C},\boldsymbol{s}^l,\theta^l\}}\lambda_y\mathcal{L}_y \\ &+ \sum_{z\in\{c,e,b,ce\}}\lambda_z\mathcal{L}_z + \lambda_{co}\mathcal{L}_{co} + \lambda_g\mathcal{L}_g,\end{aligned} \tag{4}$$

where the first three terms represent the individual loss in ODN, LEN and MGN, and the last two are the joint terms. $\{\lambda_*\}$ are the weights used to balance their importance.

4. Results and Evaluation

4.1. Experiment Setup

Datasets: We use two datasets in our experiments: 1) **SUN RGB-D** dataset [41] consists of 10,335 real indoor images with labeled 3D layout, object bounding boxes and coarse point cloud (depth map). We use the official train/test split and NYU-37 object labels [40] for evaluation on layout, camera pose estimation and 3D object detection. 2) **Pix3D** dataset [42] contains 395 furniture models with 9 categories, which are aligned with 10,069 images. We use this for mesh generation and keep the train/test split inline with [9]. The object label mapping from NYU-37 to Pix3D for scene reconstruction is listed in the supplementary file.

Metrics: Our results are measured on both scene understanding and mesh reconstruction metrics. We evaluate layout estimation with average 3D Intersection over Union (IoU). The camera pose is evaluated by the mean absolute error. Object detection is tested with the average precision (AP) on all object categories. We test the single-object mesh generation with the Chamfer distance as previous works [9, 30], and evaluate the scene mesh with Equation 3.

Implementation: We train the 2D detector (Figure 2a) on the COCO dataset [24] first and fine-tune it on SUN RGB-D. In MGN, the template sphere has 2562 vertices with unit radius. We cut edges whose average classification score is lower than 0.2. Since SUN RGB-D does not provide full instance meshes for 3D supervision, and Pix3D is only labeled with one object per image without layout information. We first train ODN, LEN on SUN-RGBD, and train MGN on Pix3D individually. We then combine Pix3D into SUN RGB-D to provide mesh supervision and jointly train all networks with the loss $\mathcal{L}$ in Equation 4. Here we use one hierarchical batch (each batch contains one scene image with N object images) in joint training. We explain the full architecture, training strategies, time efficiency and parameter setting of our networks in the supplementary file.

4.2. Qualitative Analysis and Comparison

In this section, we evaluate the qualitative performance of our method on both object and scene levels.

Object Reconstruction: We compare our MGN with the state-of-the-art mesh prediction methods [9, 10, 30] on Pix3D. Because our method is designed to accomplish scene reconstruction in real scenes, we train all methods inputted with object images but without masks. For AtlasNet [10] and Topology Modification Network (TMN) [30], we also encode the object category into image features enabling a fair comparison. Both TMN and our method are trained following a 'deformation+modification+refinement' process (see [30]). For Mesh R-CNN [9], it involves an object recognition phase, and we directly compare with the results reported in their paper. The comparisons are illustrated in Figure 5, from which we observe that indoor furniture are often overlaid with miscellaneous backgrounds (such as books on the shelf). From the results of Mesh R-CNN (Figure 5b), it generates meshes from low-resolution voxel grids (24^3 voxels) and thus results in noticeable artifacts on mesh boundaries. TMN improves from AtlasNet and refines shape topology. However, its distance threshold τ does not show consistent adaptability for all shapes in indoor environments (e.g. the stool and the bookcase in Figure 5e). Our method relies on the edge classifier. It cuts edges depending on the local density, making the topology modification adaptive to different scales of shapes among various object categories (Figure 5f). The results also demonstrate that our method keeps better boundary smoothness and details.

Scene Reconstruction: As this is the first work, to our best knowledge, of combing scene understanding and mesh generation for full scene reconstruction, we illustrate our results on the testing set of SUN RGB-D in Figure 6 (see more samples in the supplementary file). Note that SUN RGB-D does not contain ground-truth object meshes for training. We present the results under different scene types and diverse complexities to test the robustness of our method. The first row in Figure 6 shows the scenes with large repetitions and occlusions. We exhibit the cases with disordered object orientations in the second row. The third and the fourth rows present the results under various scene types, and the fifth row shows the performance in handling cluttered and 'out-of-view' objects. All the results manifest that, with different complexities, our method maintains visually appealing object meshes with reasonable object placement.

(a) (b) (c) (d) (e) (f)

Figure 5: Mesh reconstruction for individual objects. From left to right: (a) Input images and results from (b) Mesh R-CNN [9], (c) AtlasNet-Sphere [10], (d, e) TMN with $\tau = 0.1$ and $\tau = 0.05$ [30], (f) Ours.

4.3. Quantitative Analysis and Comparison

We compare the quantitative performance of our method with the state-of-the-arts on four aspects: 1. layout estimation; 2. camera pose prediction; 3. 3D object detection and 4. object and scene mesh reconstruction. The object mesh reconstruction is tested on Pix3D, and the others are evaluated on SUN RGB-D. We also ablate our method by removing joint training: each subnetwork is trained individually, to investigate the complementary benefits of combining scene understanding and object reconstruction.

Layout Estimation: We compare our method with existing layout understanding works [3, 15, 14]. As shown in Table 1, joint training with room layout, object bounding boxes and meshes helps to improve the layout estimation, providing a gain of 2 points than the state-of-the-arts.

Camera Pose Estimation: Camera pose is defined by $\mathbf{R}(\beta, \gamma)$, hence we evaluate the pitch β and roll γ with the mean absolute error with the ground-truth. The results are show in Table 1, where we observe that joint learning also benefits the camera pose estimation.

Method	3D Layout	Cam pitch	Cam roll
3DGP [3]	19.2	-	-
Hedau [12]	-	33.85	3.45
HoPR [15]	54.9	7.60	3.12
CooP [14]	56.9	3.28	2.19
Ours (w/o. joint)	57.6	3.68	2.59
Ours (joint)	**59.2**	**3.15**	**2.09**

Table 1: Comparisons of 3D layout and camera pose estimation on SUN RGB-D. We report the average IoU to evaluate layout prediction (higher is better), and the mean absolute error of pitch and roll angles (in degree) to test camera pose (lower is better). Note that our camera axes are defined in a different order with [14] (see the supplementary file).

3D Object Detection: We investigate the object detection with the benchmark consistent with [14], where the mean average precision (mAP) is employed using 3D bounding box IoU. A detection is considered true positive if its IoU with the ground-truth is larger than 0.15. We compare our method with existing 3D detection works [3, 15, 14] on the shared object categories in Table 2. The full table on all object categories is listed in the supplementary file. The comparisons show that our method significantly improves over the state-of-the-art methods, and consistently advances the ablated version. The reason could be two-fold. One is that the global loss $\mathcal{L}_g$ in joint learning involves geometry constraint which ensures the physical rationality, and the other is that multi-lateral relational features in ODN benefit the 3D detection in predicting spatial occupancy.

We also compare our work with [46] to evaluate object pose prediction. We keep consistent with them by training on the NYU v2 dataset [40] with their six object categories and ground-truth 2D boxes. The results are reported in Table 3. Object poses are tested with errors in object translation, rotation and scale. We refer readers to [46] for the definition of the metrics. The results further demonstrate that our method not only obtains reasonable spatial occupancy (mAP), but also retrieves faithful object poses.

Mesh Reconstruction: We evaluate mesh reconstruction on both the object and scene levels. For object reconstruction, we compare our MGN with the state-of-the-arts [10, 30] in Table 4. We ablate our topology modification method with two versions: 1. removing faces instead of edges (w/o. edge); 2. using distance threshold [30] instead of our local density (w/o. dens) for topology modification. The results show that each module improves the mean accuracy, and combining them advances our method to the state-of-the-art. A possible reason is that using local density keeps small-scale topology, and cutting edges is more robust in avoiding incorrect mesh modification than removing

Figure 6: Scene reconstruction on SUN RGB-D. Given a single image, our method end-to-end reconstructs the room layout, camera pose with object bounding boxes, poses and meshes.

Method	bed	chair	sofa	table	desk	dresser	nightstand	sink	cabinet	lamp	mAP
3DGP [3]	5.62	2.31	3.24	1.23	-	-	-	-	-	-	-
HoPR [15]	58.29	13.56	28.37	12.12	4.79	13.71	8.80	2.18	0.48	2.41	14.47
CooP [14]*	**63.58**	17.12	41.22	26.21	9.55	4.28	6.34	5.34	2.63	1.75	17.80
CooP [14]**	57.71	15.21	36.67	31.16	19.90	15.98	11.36	15.95	10.47	3.28	21.77
Ours (w/o. joint)	59.03	15.98	43.95	35.28	23.65	19.20	6.87	14.40	11.39	3.46	23.32
Ours (joint)	60.65	**17.55**	**44.90**	**36.48**	**27.93**	**21.19**	**17.01**	**18.50**	**14.51**	**5.04**	**26.38**

Table 2: Comparisons of 3D object detection. We compare the average precision of detected objects on SUN RGB-D (higher is better). [14]* shows the results from their paper, which are trained with fewer object categories. CooP [14]** presents the model trained on the NYU-37 object labels for a fair comparison.

Method	Translation (meters)			Rotation (degrees)			Scale		
	Median (lower is better)	Mean	(Err≤0.5m)% (higher is better)	Median (lower is better)	Mean	(Err≤30°)% (higher is better)	Median (lower is better)	Mean	(Err≤0.2)% (higher is better)
Tulsiani *et al.*[46]	0.49	0.62	51.0	14.6	**42.6**	63.8	0.37	0.40	18.9
Ours (w/o. joint)	0.52	0.65	49.2	15.3	45.1	64.1	0.28	0.29	42.1
Ours (joint)	**0.48**	**0.61**	**51.8**	**14.4**	43.7	**66.5**	**0.22**	**0.26**	**43.7**

Table 3: Comparisons of object pose prediction. The difference values of translation, rotation and scale between the predicted and the ground-truth bounding boxes on NYU v2 are reported, where the median and mean of the differences are listed in the first two columns (lower is better). The third column presents the correct rate within a threshold (higher is better).

Category	bed	bookcase	chair	desk	sofa	table	tool	wardrobe	misc	mean
AtlasNet [10]	9.03	6.91	8.37	8.59	6.24	19.46	6.95	4.78	40.05	12.26
TMN [30]	7.78	**5.93**	6.86	7.08	4.25	17.42	4.13	4.09	**23.68**	9.03
Ours (w/o. edge)	8.19	6.81	6.26	5.97	4.12	15.09	3.93	4.01	25.19	8.84
Ours (w/o. dens)	8.16	6.70	6.38	**5.12**	4.07	16.16	3.63	4.32	24.22	8.75
Ours	**5.99**	6.56	**5.32**	5.93	**3.36**	**14.19**	**3.12**	**3.83**	26.93	**8.36**

Table 4: Comparisons of object reconstruction on Pix3D. The Chamfer distance is used in evaluation. 10K points are sampled from the predicted mesh after being aligned with the ground-truth using ICP. The values are in units of 10^{-3} (lower is better).

faces. Mesh reconstruction of scenes is evaluated with $\mathcal{L}_g$ in Equation 3, where the loss is calculated with the average distance from the point cloud of each object to its nearest neighbor on the reconstructed mesh. Different from single object reconstruction, scene meshes are evaluated considering object alignment in the world system. In our test, $\mathcal{L}_g$ decreases from 1.89e-2 to 1.43e-2 with our joint learning.

4.4. Ablation Analysis and Discussion

To better understand the effect of each design on the final result, we ablate our method with five configurations:
$\boldsymbol{C_0}$**:** without relational features (in ODN) and joint training (Baseline).
$\boldsymbol{C_1}$**:** Baseline + relational features.
$\boldsymbol{C_2}$**:** Baseline + (only) cooperative loss $\mathcal{L}_{co}$ in joint training.
$\boldsymbol{C_3}$**:** Baseline + (only) global loss $\mathcal{L}_g$ in joint training.
$\boldsymbol{C_4}$**:** Baseline + joint training ($\mathcal{L}_g$ + $\mathcal{L}_{co}$).
Full: Baseline + relational features + joint training.

We test the layout estimation, 3D detection and scene mesh reconstruction with 3D IoU, mAP and $\mathcal{L}_g$. The results are reported in Table 5, from which we observe that:
$\boldsymbol{C_0}$ v.s.$\boldsymbol{C_4}$ and $\boldsymbol{C_1}$ v.s. **Full**: Joint training consistently improves layout estimation, object detection and scene mesh reconstruction no matter using relational features or not.
$\boldsymbol{C_0}$ v.s.$\boldsymbol{C_1}$ and $\boldsymbol{C_4}$ v.s. **Full**: Relational features help to improve 3D object detection, which indirectly reduces the loss in scene mesh reconstruction.
$\boldsymbol{C_0}$ v.s.$\boldsymbol{C_2}$ and $\boldsymbol{C_0}$ v.s. $\boldsymbol{C_3}$: In joint loss, both $\mathcal{L}_{co}$ and $\mathcal{L}_g$ in joint training benefit the final outputs, and combing them further advances the accuracy.

We also observe that the global loss $\mathcal{L}_g$ shows the most effect on object detection and scene reconstruction, and the cooperative loss $\mathcal{L}_{co}$ provides more benefits than others on layout estimation. Besides, scene mesh loss decreases with the increasing of object detection performance. It is inline with the intuition that object alignment significantly affects mesh reconstruction. Fine-tuning MGN on SUN RGB-D can not improve single object reconstruction on Pix3D. It reflects that object reconstruction depends on clean mesh for supervision. All the facts above explain that the targets for full scene reconstruction actually are intertwined together, which makes joint reconstruction a feasible solution toward total scene understanding.

Version	Layout (IoU) (higher is better)	3D Objects (mAP) (higher is better)	Scene mesh ($\mathcal{L}_g$) (lower is better)
$\boldsymbol{C_0}$	57.63	20.19	2.10
$\boldsymbol{C_1}$	57.63	23.32	1.89
$\boldsymbol{C_2}$	58.21	21.77	1.73
$\boldsymbol{C_3}$	57.92	24.59	1.64
$\boldsymbol{C_4}$	58.87	25.62	1.52
Full	**59.25**	**26.38**	**1.43**

Table 5: Ablation analysis in layout estimation, 3d object detection and scene mesh reconstruction on SUN RGB-D. The $\mathcal{L}_g$ values are in units of 10^{-2}.

5. Conclusion

We develop an end-to-end indoor scene reconstruction approach from a single image. It embeds scene understanding and mesh reconstruction for joint training, and automatically generates the room layout, camera pose, object bounding boxes and meshes. Extensive experiments show that our joint learning approach significantly improves the performance on each subtask and advances the state-of-the-arts. It indicates that each individual scene parsing process has an implicit impact on the others, revealing the necessity of training them integratively toward total 3D reconstruction. One limitation of our method is the requirement for dense point cloud for learning object meshes, which is labor-consuming to obtain in real scenes. To tackle this problem, a self or weakly supervised scene reconstruction method would be a desirable solution in the future work.

Acknowledgment This work was partially supported by grants No.2018YFB1800800, No.2018B030338001, NSFC-61902334, NSFC-61629101, NSFC-61702433, NSFC-61661146002, No.ZDSYS201707251409055, No. 2017ZT07X152, VISTA AR project (funded by the Interreg France (Channel) England, ERDF), Innovate UK Smart Grants (39012), the Fundamental Research Funds for the Central Universities, the China Scholarship Council and Bournemouth University.

References

[1] 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. *arXiv preprint arXiv:1909.01507*, 2019.

[2] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5939–5948, 2019.

[3] Wongun Choi, Yu-Wei Chao, Caroline Pantofaru, and Silvio Savarese. Understanding indoor scenes using 3d geometric phrases. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 33–40, 2013.

[4] Christopher Bongsoo Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 628–644, 2016.

[5] Saumitro Dasgupta, Kuan Fang, Kevin Chen, and Silvio Savarese. Delay: Robust spatial layout estimation for cluttered indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 616–624, 2016.

[6] Theo Deprelle, Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry. Learning elementary structures for 3d shape generation and matching. *arXiv preprint arXiv:1908.04725*, 2019.

[7] Yilun Du, Zhijian Liu, Hector Basevi, Ales Leonardis, Bill Freeman, Josh Tenenbaum, and Jiajun Wu. Learning to exploit stability for 3d scene parsing. In *Advances in Neural Information Processing Systems*, pages 1726–1736, 2018.

[8] Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 605–613, 2017.

[9] Georgia Gkioxari, Jitendra Malik, and Justin Johnson. Mesh r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9785–9795, 2019.

[10] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan Russell, and Mathieu Aubry. AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[12] Varsha Hedau, Derek Hoiem, and David Forsyth. Recovering the spatial layout of cluttered rooms. In *2009 IEEE 12th international conference on computer vision*, pages 1849–1856. IEEE, 2009.

[13] Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, and Yichen Wei. Relation networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3588–3597, 2018.

[14] 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 *Advances in Neural Information Processing Systems*, pages 207–218, 2018.

[15] 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 *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 187–203, 2018.

[16] 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.

[17] Hamid Izadinia, Qi Shan, and Steven M Seitz. Im2cad. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5134–5143, 2017.

[18] Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada. Neural 3d mesh renderer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3907–3916, 2018.

[19] Nilesh Kulkarni, Ishan Misra, Shubham Tulsiani, and Abhinav Gupta. 3d-relnet: Joint object and relational network for 3d prediction. *International Conference on Computer Vision (ICCV)*.

[20] Andrey Kurenkov, Jingwei Ji, Animesh Garg, Viraj Mehta, JunYoung Gwak, Christopher Choy, and Silvio Savarese. Deformnet: Free-form deformation network for 3d shape reconstruction from a single image. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 858–866. IEEE, 2018.

[21] David C Lee, Martial Hebert, and Takeo Kanade. Geometric reasoning for single image structure recovery. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2136–2143. IEEE, 2009.

[22] Lin Li, Salman Khan, and Nick Barnes. Silhouette-assisted 3d object instance reconstruction from a cluttered scene. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[23] Yiyi Liao, Simon Donne, and Andreas Geiger. Deep marching cubes: Learning explicit surface representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2916–2925, 2018.

[24] 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 *European conference on computer vision*, pages 740–755. Springer, 2014.

[25] Arun Mallya and Svetlana Lazebnik. Learning informative edge maps for indoor scene layout prediction. In *Proceedings of the IEEE international conference on computer vision*, pages 936–944, 2015.

[26] Priyanka Mandikal, Navaneet KL, and R Venkatesh Babu. 3d-psrnet: Part segmented 3d point cloud reconstruction from a single image. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018.

[27] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks:

Learning 3d reconstruction in function space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4460–4470, 2019.

[28] Mateusz Michalkiewicz, Jhony K Pontes, Dominic Jack, Mahsa Baktashmotlagh, and Anders Eriksson. Deep level sets: Implicit surface representations for 3d shape inference. *arXiv preprint arXiv:1901.06802*, 2019.

[29] KL Navaneet, Priyanka Mandikal, Mayank Agarwal, and R Venkatesh Babu. Capnet: Continuous approximation projection for 3d point cloud reconstruction using 2d supervision. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8819–8826, 2019.

[30] Junyi Pan, Xiaoguang Han, Weikai Chen, Jiapeng Tang, and Kui Jia. Deep mesh reconstruction from single rgb images via topology modification networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9964–9973, 2019.

[31] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. *arXiv preprint arXiv:1901.05103*, 2019.

[32] Despoina Paschalidou, Ali Osman Ulusoy, and Andreas Geiger. Superquadrics revisited: Learning 3d shape parsing beyond cuboids. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10344–10353, 2019.

[33] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 918–927, 2018.

[34] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[35] Yuzhuo Ren, Shangwen Li, Chen Chen, and C-C Jay Kuo. A coarse-to-fine indoor layout estimation (cfile) method. In *Asian Conference on Computer Vision*, pages 36–51. Springer, 2016.

[36] Gernot Riegler, Ali Osman Ulusoy, and Andreas Geiger. Octnet: Learning deep 3d representations at high resolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3577–3586, 2017.

[37] Lawrence G Roberts. *Machine perception of three-dimensional solids*. PhD thesis, Massachusetts Institute of Technology, 1963.

[38] Alexander G Schwing, Sanja Fidler, Marc Pollefeys, and Raquel Urtasun. Box in the box: Joint 3d layout and object reasoning from single images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 353–360, 2013.

[39] Daeyun Shin, Zhile Ren, Erik B Sudderth, and Charless C Fowlkes. 3d scene reconstruction with multi-layer depth and epipolar transformers. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019.

[40] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision*, pages 746–760. Springer, 2012.

[41] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 567–576, 2015.

[42] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2974–2983, 2018.

[43] Jiapeng Tang, Xiaoguang Han, Junyi Pan, Kui Jia, and Xin Tong. A skeleton-bridged deep learning approach for generating meshes of complex topologies from single rgb images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4541–4550, 2019.

[44] Maxim Tatarchenko, Alexey Dosovitskiy, and Thomas Brox. Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2088–2096, 2017.

[45] Yonglong Tian, Andrew Luo, Xingyuan Sun, Kevin Ellis, William T Freeman, Joshua B Tenenbaum, and Jiajun Wu. Learning to infer and execute 3d shape programs. *arXiv preprint arXiv:1901.02875*, 2019.

[46] Shubham Tulsiani, Saurabh Gupta, David F Fouhey, Alexei A Efros, and Jitendra Malik. Factoring shape, pose, and layout from the 2d image of a 3d scene. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 302–310, 2018.

[47] Shubham Tulsiani, Hao Su, Leonidas J Guibas, Alexei A Efros, and Jitendra Malik. Learning shape abstractions by assembling volumetric primitives. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2635–2643, 2017.

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[49] Bram Wallace and Bharath Hariharan. Few-shot generalization for single-image 3d reconstruction via priors. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3818–3827, 2019.

[50] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. Pixel2mesh: Generating 3d mesh models from single rgb images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 52–67, 2018.

[51] Peng-Shuai Wang, Chun-Yu Sun, Yang Liu, and Xin Tong. Adaptive o-cnn: a patch-based deep representation of 3d shapes. In *SIGGRAPH Asia 2018 Technical Papers*, page 217. ACM, 2018.

[52] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. *arXiv preprint arXiv:1905.10711*, 2019.