Uni-3D: A Universal Model for Panoptic 3D Scene Reconstruction

Xiang Zhang* [1] Zeyuan Chen* [1] Fangyin Wei [2] Zhuowen Tu [1]
[1]UC San Diego [2] Princeton University
{xiz102, zec016}@ucsd.edu, fwei@princeton.edu, ztu@ucsd.edu

Abstract

Performing holistic 3D scene understanding from a single-view observation, involving generating instance shapes and 3D scene segmentation, is a long-standing challenge. Prevailing works either focus only on geometry or segmentation, or model the task in two folds by separate modules, whose results are merged later to form the final prediction. Inspired by recent advances in 2D vision that unify image segmentation and detection by Transformer-based models, we present Uni-3D, a holistic 3D scene parsing/reconstruction system for a single RGB image. Uni-3D features a universal model with query-based representations for predicting segments of both object instances and scene layout. In Uni-3D, we also introduce a single Transformer for 2D depth-aware panoptic segmentation, which offers queries that serve as strong shape priors in 3D. Uni-3D seamlessly integrates 2D and 3D in its architecture and it outperforms previous methods significantly.

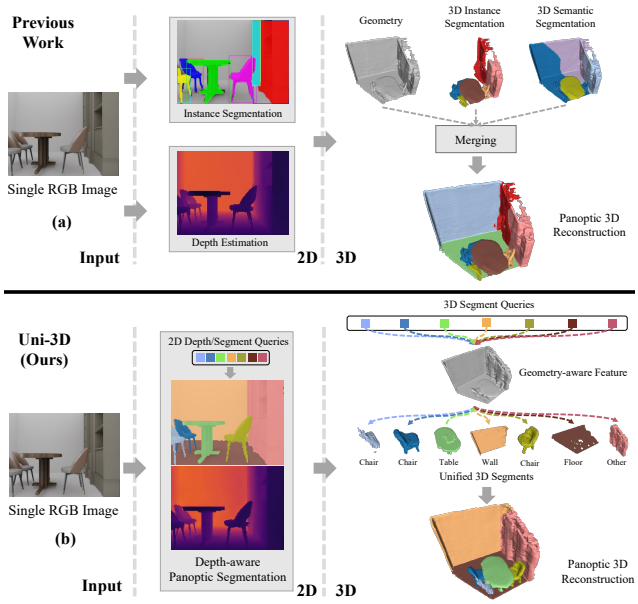

Figure 1. Pipeline comparison between (a) previous work (*e.g.* [8]) and (b) our proposed universal 3D reconstruction framework (Uni-3D). Uni-3D integrates panoptic segmentation and depth estimation in 2D for shared knowledge and unifies thing and stuff instances in 3D, which is a universal pipeline for panoptic 3D scene reconstruction.

1. Introduction

Humans have a remarkable ability to infer 3D shapes and scene layouts accurately from limited or single-view observations, which is attributed to the efficient representations that encode the 3D world for 2D projection. In computer vision and graphics, understanding the 3D world from a single-view 2D observation is a longstanding task, which plays an essential role in multiple downstream applications such as autonomous driving, augmented reality, and robotic systems.

Notable breakthroughs have been made recently in addressing this challenge, thanks to the advancements in neural networks and the rapid growth of data quantity. 3D shape reconstruction methods [45, 48, 24, 17] aim to predict 3D models of instances in images and have exhibited impressive reconstruction quality. Another category of methods, including [41, 10], performs scene reconstruction by directly recovering the geometric structure of the 3D scene captured by 2D images. The aforementioned approaches demonstrate the capability of deep neural networks to learn shape priors from training data and recover the 3D world from 2D observations during inference. However, they focus solely on 3D shapes and silhouettes of instances or entire scenes and do not extract 3D semantic information necessary for scene understanding.

Factored3D [42] is a pioneering work that not only recovers 3D shapes but also performs scene understanding by predicting the layout of instances in scenes. Huang *et al*. [21] propose a holistic scene understanding system that unifies the estimation of 3D object poses, camera pose, and scene layout, providing a comprehensive understanding of the input scene. Total3D [37] further incorporates mesh reconstruction into the 3D scene understanding system. In summary, these models have approached 3D scene under-

* indicates equal contribution.
Code: https://github.com/mlpc-ucsd/Uni-3D.

standing from three aspects: 1) instance reconstruction; 2) scene geometric structure prediction; and 3) 3D layout. We are interested in an all-in-one system that integrates all three aspects and provides all estimations in a single run. Liu *et al*. [31] developed a two-stage system that can perform 3D scene parsing and reconstruction for a single in-the-wild image, as well as an end-to-end system for 3D scene reconstruction. The work most relevant to our target is PanoRe [8]. It tackles the tasks of single-image geometric reconstruction, 3D semantic segmentation, and 3D instance segmentation simultaneously, which is named by them as panoptic 3D scene reconstruction. However, it lifts 2D instance features back to 3D and embeds 2D instance segmentation maps as priors to link the two dimensions, which only handles instances in 2D without stuff information so that the 3D part of its network is only informed by instance priors.

Taking inspiration from the trend of unifying 2D semantic and instance segmentation via Transformers such as Mask2Former [4], we introduce a universal system called Uni-3D that addresses the task of 3D scene understanding in a holistic manner. Specifically, we first introduce a depth-aware panoptic segmentation architecture based on Transformers, which integrates all 2D predictions in a unified paradigm. Benefiting from this, the query embeddings of our 2D Transformers are informative, effectively learning multiple properties of the input scene, such as layout, instance silhouettes, category labels, and depth. We then build a query-based architecture for the 3D part, where 3D segments of both object instances and stuff layout are predicted individually with the guidance of corresponding 2D queries. This query-based design seamlessly integrates learned 2D features and priors into 3D, offering more flexibility and robustness.

Our contribution is summarized as follows:

- We introduce Uni-3D, a universal model that integrates 3D instance and semantic segmentation as a panoptic one by learning segment representations for both instances and stuff layout. Each 3D segment is learned with the guidance from its corresponding 2D query which serves as a strong 2D prior.

- In Uni-3D, we develop a 2D Transformer for unifying 2D panoptic segmentation and depth estimation. This approach facilitates interactions between the two tasks, offering shared knowledge from both tasks to features and learnable queries that serve as shape priors for producing 3D segments.

Uni-3D outperforms previous methods by a large margin both in quantitative and qualitative evaluations.

2. Related Work

2D panoptic segmentation. The panoptic segmentation task is first proposed in [25]. The task is to assign each image pixel a semantic label and instance id, where the instance id is ignored for stuff classes, which unifies instance ("thing" classes) segmentation and semantic ("stuff" classes) segmentation, allowing a holistic understanding of the image. Earlier literature [9, 27, 28, 47, 25, 29] typically addresses instance and semantic segmentation in separate branches, and merge the results via either heuristics-guided or learnable approaches. Recently, DEtection TRansformers (DETR) [2] further pushes the boundary for detection and segmentation tasks by formulating such tasks as set-prediction problems. Subsequent DETR family models [5, 4, 30] benefit from this formulation and treat thing and stuff classes as image segments to achieve unified and end-to-end learning for both instance and semantic segmentation, enabling universal image segmentation that performs well on both types of classes.

Other work explores depth-aware panoptic segmentation, which accomplishes monocular depth estimation besides panoptic segmentation. While a straightforward solution is to add a depth regression head sharing the backbone feature with panoptic segmentation, as in [38, 40], PanopticDepth [15] mutually enhances the two tasks through a unified model. As both segmentation and depth provide strong 2D priors necessary for adequate 3D understanding, we propose a depth-aware panoptic segmentation model in the 2D part, where the two tasks have knowledge of and benefit from each other.

Single-view 3D reconstruction. 3D shape reconstruction from a single-view observation input is a long-standing problem in computer vision. Traditional methods extract multi-modal information from 2D image observations for reconstructing shapes, including shading [20, 1, 39], texture [44], and silhouettes [6]. Recent learning-based approaches have demonstrated impressive performance boosts in reconstruction quality. To reconstruct a single object from monocular observation, methods have been investigated employing different representations including voxel grids [7, 46], point clouds [12], mesh [43], and implicit functions [35]. Other methods have advanced to predicting multiple shapes [17] and even with layout [23, 26]. More recently, Gkioxari *et al*. [18] proposes a method that uses only 2D supervision during training for single-view 3D reconstruction during inference.

3D scene understanding and panoptic reconstruction. The primary goal of 3D Scene Understanding is to predict 3D shape instances, as well as estimate instance properties including their layouts and category labels. IM2CAD [23] is one of the pioneering works in this field. It leverages information from 2D object detection information to generate

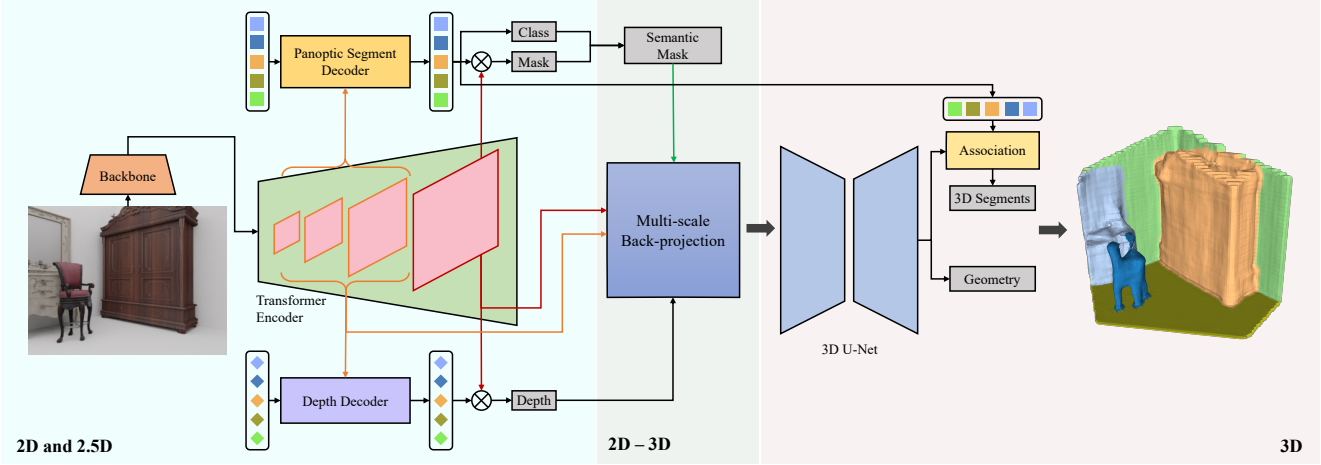

Figure 2. Overview of the proposed framework. In the 2D/2.5D parts, a Transformer encoder generates multi-scale features from the backbone, which are fed into the panoptic segment decoder and depth decoder for panoptic segmentation and depth estimation simultaneously. The features utilized to generate segmentation masks and depth maps, along with estimated depth, semantic masks, and multi-scale features from the Transformer encoder are back-projected into 3D volumes. A 3D U-Net processes multi-scale feature volumes, with a geometry head predicting the truncated distance field for the entire scene. Query embeddings from the panoptic segment decoder are projected as convolution kernels that associate features from the U-Net, yielding unified 3D segments for object instances (things) and layouts (stuff).

3D layouts and optimize 3D CAD models. Holistic3D [22] introduced the Holistic Scene Grammar module, which represents the 3D scene structure by capturing image contexts and geometric constraints. Mask2CAD [26] achieved a CAD-based 3D representation by learning a joint embedding space between 2D detection results and CAD models. These methods have produced impressive reconstruction quality. However, their retrieval-based pipelines can sometimes generate 3D models that are not consistent with 2D inputs. Total3D is an end-to-end learning approach that simultaneously detects object categories, poses, 3D models, and room layouts.

The panoptic 3D scene reconstruction task, introduced by Dahnert *et al*. [8] and Liu *et al*. [31], aims to predict 3D scene models along with the semantic labels of all instances and stuff in the scene. Liu *et al*. [31] proposed a stage-wise panoptic 3D parsing system that can perform reconstruction on images in the wild. Dahnert *et al*. [8] presented a voxel-based sparse neural network that predicts scene geometry, semantic, and instance segmentation using parallel network heads.

3. Method

We present a novel system that performs 2D panoptic segmentation, 2.5D depth estimation, and 3D scene understanding and reconstruction concurrently. The overall architecture is shown in Figure 2. Our approach comprises two components responsible for 2D and 3D respectively. We first propose a Transformer-based architecture that simultaneously addresses panoptic segmentation and depth estimation, and elaborate on our 3D network for panoptic reconstruction, which takes in back-projected multi-scale features and transforms them into 3D segments with query embeddings for the reconstruction output.

3.1. Depth-aware Panoptic Segmentation

Our 2D panoptic segmentation is primarily based on Mask2Former [4], where panoptic segmentation is formulated as a mask classification task. Given an image, we need to partition it into N regions represented by binary masks (segments) $\{m_i | m_i \in [0,1]^{H \times W}\}$. Each mask is associated with a distribution p_i over $(K+1)$ categories, where the extra category is for no-object class $\varnothing$. Such formulation is appropriate for both instance and semantic segmentation due to its unified representation for thing and stuff classes.

To enable monocular depth estimation with the panoptic segmentation framework, instead of adding another separate network after the backbone, we argue that the two tasks can benefit from the shared knowledge of each other, leading to a unified solution to 2D and providing better features for 3D reconstruction. Inspired by architectures such as PanopticDepth [15], we introduce a depth decoder, parallel to the original panoptic segment decoder in Mask2Former. The details are illustrated in Figure 3. Both the decoders have their own N learnable query embeddings, denoted as q_i^S, q_i^D respectively, where i is the index for each segment. The four-level feature maps produced by the Transformer encoder, with scale $1/32, 1/16, 1/8, 1/4$ of the input image, are denoted as F_1, F_2, F_3, F_4, respectively. The output segmentation mask $\hat{m}_i$ and the depth map within the mask $\hat{d}_i$ are simply

$$\begin{aligned}\hat{m}_i &= \text{Sigmoid}\left(P^S(F_4)\cdot q_i^S\right)\\ \hat{d}_i &= D_{\max}\,\text{Sigmoid}\left(P^D(F_4)\cdot q_i^D\right)\end{aligned} \tag{1}$$

where $D_{\max}$ is the depth scale. P^S and P^D are learnable projection functions that maps the largest encoder feature F_4 into different spaces. In decoder layer l, masks from layer $l-1$ are resized and binarized with threshold 0.5, where cross-attention is restricted within features where masks are valid. F_1, F_2, F_3 from the encoder are fed into the 9 decoder layers in a round-robin fashion to leverage multi-level features.

To further enhance the interaction between the two decoders and facilitate knowledge sharing, we introduce **cross-decoder query association**. In this module, both the segment query and depth query belonging to the same segment i are updated as (with skip connection omitted)

$$q_i^j = \text{Self-Attention}\left(q=q_i^j, k=[q_i^S, q_i^D], v=[q_i^S, q_i^D]\right) \tag{2}$$

where $j = S$ or D.

During training, bipartite matching is performed between G ground truth labels and N predicted segments ($G \le N$), such that an injective function $\sigma : [G] \to [N]$ can be found. The matching cost between j-th ground truth and i-th prediction is

$$\mathcal{C}(j,i) = -\hat{p}_i(c_j) + \mathcal{L}_{\text{mask}}(\hat{m}_i, m_j) \tag{3}$$

where $\mathcal{L}_{\text{mask}}$ is the mask loss, and c_j is the class label for j-th ground truth. Note for the simplicity of the notation, we ignore the weights before each loss term unless otherwise mentioned. The loss for the panoptic segment decoder is

$$\mathcal{L}_{\text{segment}} = \mathcal{L}_{\text{cls}} + \mathbb{1}_{\{i\in\text{Im}(\sigma)\}}\mathcal{L}_{\text{mask}}\left(\hat{m}_i, m_{\sigma^{-1}(i)}\right) \tag{4}$$

where classification loss $\mathcal{L}_{\text{cls}} = -\log \hat{p}_i(c_{\sigma^{-1}(i)})$.

For the depth decoder, we use the same matching σ obtained from the panoptic segment decoder. Following [38], the loss between j-th ground truth depth map and i-th ($i=\sigma(j)$) matched prediction comprises scale-invariant logarithm error [11] and relative square error [16], specifically

$$\begin{aligned}\mathcal{L}_{\text{depth}} = \mathbb{1}_{\{i\in\text{Im}(\sigma)\}}\Bigg[&\frac{1}{n}\sum_k\left(\log d_j^{(k)} - \log \hat{d}_i^{(k)}\right)^2 - \frac{1}{n^2}\\ &\left(\sum_k \log d_j^{(k)} - \log \hat{d}_i^{(k)}\right)^2 + \sqrt{\frac{1}{n}\sum_k\left(1-\frac{\hat{d}_i^{(k)}}{d_j^{(k)}}\right)^2}\Bigg]\end{aligned} \tag{5}$$

where k is the index for each pixel. We only calculate the depth loss within the region for each segment, as demarcated by the ground truth mask m_j.

Panoptic segmentation and depth estimation are trained jointly with the loss defined in Equations (4) and (5). The same inference strategy in MaskFormer [5] is utilized for panoptic segmentation, where each pixel $[h, w]$ is assigned one of the N segments via $\arg\max_{i,c_i\neq\varnothing} p_i(c_i)\cdot m_i[h,w]$. c_i is the most likely class label $c_i = \arg\max_c p_i(c)$. The segments are filtered out if most of the mask confidence is lower than 0.5. Multiple segments belonging to the same stuff class are merged. The depth for each pixel is extracted from the depth map $\hat{d}_i$ accordingly, if it belongs to the i-th segment $\hat{m}_i$ in panoptic segmentation.

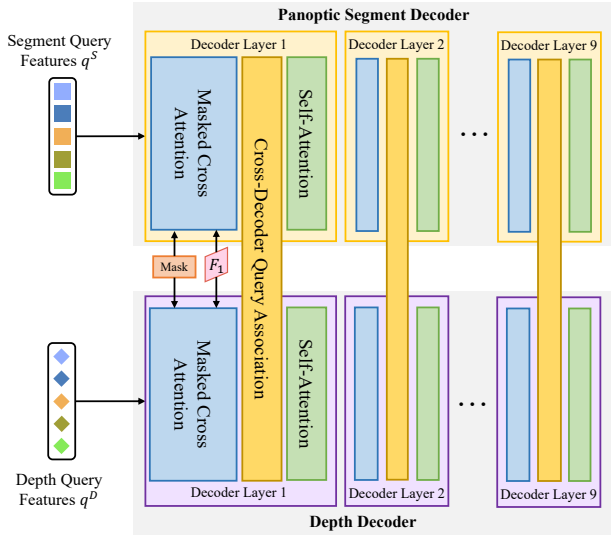

Figure 3. Illustration of the dual decoders for panoptic segmentation and depth estimation. Each decoder takes in its own query embeddings and performs masked cross-attention with image features (F_1, F_2, or F_3) from the Transformer encoder, where masks are generated from the previous decoder output. A subsequent cross-decoder query association layer enables the communication between segment and depth queries belonging to the same segment, followed by a typical self-attention layer. Multiple decoder layers are stacked for gradual refinement of the outputs.

3.2. Panoptic 3D Scene Reconstruction

3.2.1 Multi-scale Feature Lifting

Following [8], we back-project 2D feature maps $P^S(F_4)$, $P^D(F_4)$, along with 2D depth and semantic segmentation map into a 256^3 3D volumetric grid containing the camera frustum using the estimated depth. The features are encoded as a truncated distance field (TDF) along the view direction with truncation $\tau = 3$. A 3D generative U-Net, as in [8], takes in the 256^3 volumetric feature and generates representations of smaller scales. In order to leverage the rich semantics from multi-scale features, we also back-project features F_3, F_2, F_1 taken from the Transformer encoder into the same camera frustum, but with voxel sizes doubled between levels, thus yielding $128^3, 64^3, 32^3$ feature volumes. These features are then fused into encoded features from the U-Net encoders of the respective scale.

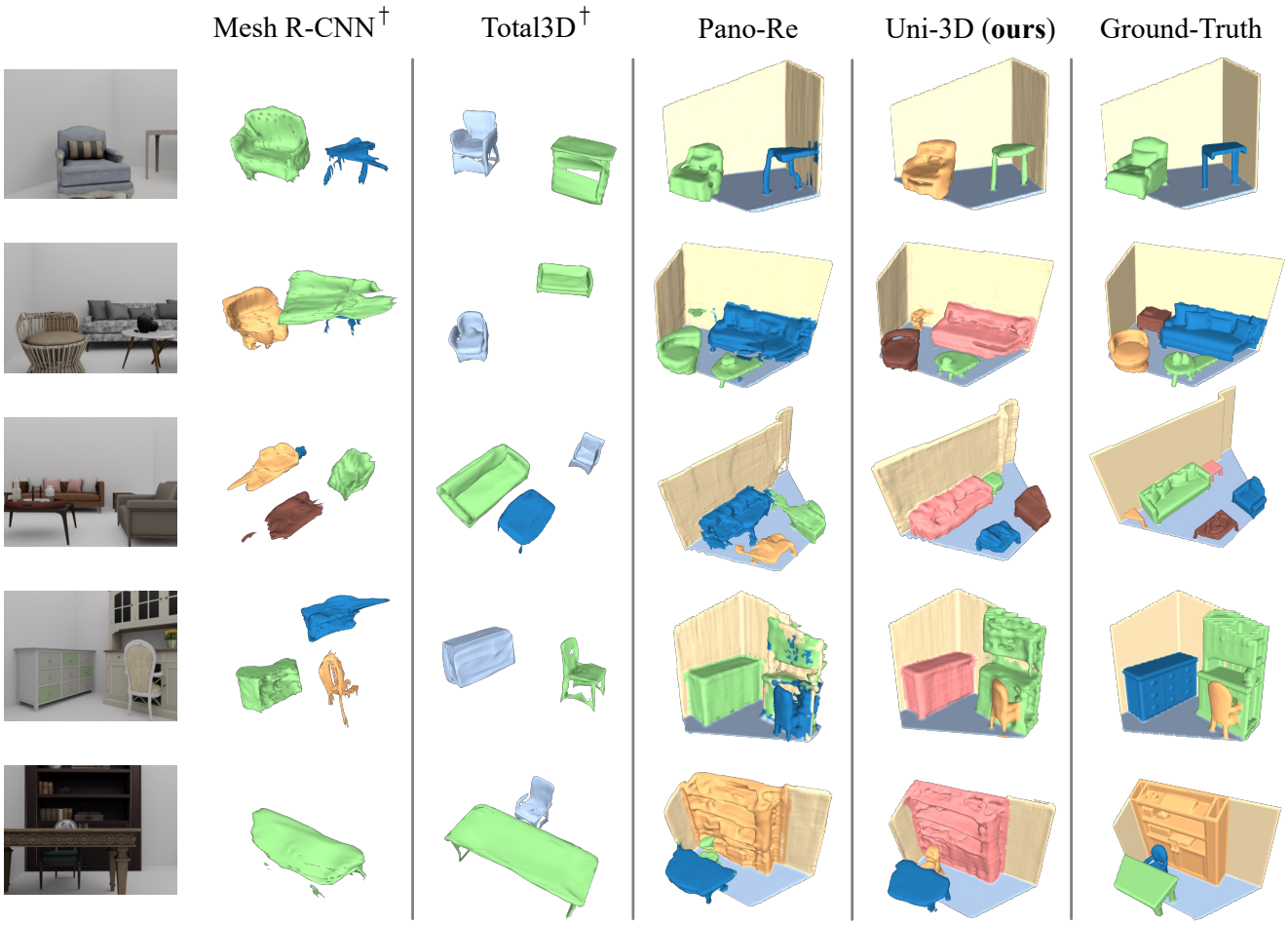

Figure 4. Qualitative comparison of panoptic 3D reconstruction on 3D-Front [14]. Different colors represent separate instances. † denotes that the models adopt official weights without fine-tuning on 3D-Front.

3.2.2 2D Queries as Strong Priors

Analogous to the 2D case, we formulate the panoptic 3D segmentation as dividing the camera frustum into N regions represented by 3D binary masks (segments) $\{s_i | s_i \in [0,1]^{H \times W \times D}\}$, as opposed to the per-voxel classification paradigm in [8]. This bridges the gap between 2D and 3D, where 2D and 3D segments have one-to-one correspondence. And therefore the 3D network can leverage the strong 2D priors via segment query embeddings q_i^S, which are learned in the 2D panoptic segment decoder and contain abundant information about stuff and thing segments. Each segment s_i can be obtained from query embeddings and decoded feature F' from U-Net, as follows

$$s_i = \text{Sigmoid}\left(K(q_i^S) * F'\right) \tag{6}$$

where K is a learnable projection function. The projected queries $K(q_i^S)$ serve as 1×1 kernels that convolve feature F', which is geometry-aware, as a separate geometry head is attached to F' and its output is supervised by the truncated distance field for the entire scene. Here query embeddings from the last Transformer decoder layer are utilized.

3.2.3 3D Reconstruction Outputs

We utilize the coarse-to-fine prediction strategy, which contains three levels of volumes, namely $64^3, 128^3, 256^3$. Each level outputs N 3D segments, with a separate head predicting the occupancy for the entire scene. The occupancy grid is thresholded at 0.5 and used to prune the current-level features before passing them onto the next level. A geometry head operates on the last feature level to predict the truncated distance field. The training loss for the 3D network is

$$\mathcal{L} = \mathcal{L}_{\text{geometry}} + \sum_l \mathcal{L}^l_{\text{seg}}(\hat{s}_i, s_{\sigma^{-1}(i)}) + \mathcal{L}^l_{\text{occupancy}} \tag{7}$$

where $\mathcal{L}_{\text{geometry}}$ denotes the L-1 loss between predicted TDF and ground truth, $\mathcal{L}^l_{\text{seg}}$ and $\mathcal{L}^l_{\text{occupancy}}$ are binary cross-entropy loss for 3D segments and occupancy respectively.

The latter two are summed over scale l. The matching function σ remains the same as 2D, which is obtained from the 2D predictions of the last decoder layer.

In the inference phase, we adopt a similar approach to the 2D case. Specifically, we assign each voxel $[x, y, z]$ a segment via $\arg\max_{i, c_i \neq \varnothing} p_i(c_i) \cdot s_i[x, y, z]$, with low confidence segments filtered. Afterward, for each voxel within the distance threshold τ_s in the TDF prediction but not belonging to any segment, we assign it the most likely segment regardless of the mask confidence for that segment.

4. Experiments

4.1. Datasets

For main results, we use the 3D-Front dataset built in Pano-Re [8], which is a synthetic dataset containing 18,797 indoor scenes with randomized 3D shapes from the 3D-Future [14] dataset. 3D-Front includes wall and floor as two stuff classes and 9 different instance (thing) classes. 2D ground-truth information, *i.e.* RGB image, depth, instance segmentation, and semantic segmentation, is also provided. We adopt the same train/val/test split, with 4,389/489/1,206 images respectively. Following Pano-Re [8], we also evaluate the single-scale Uni-3D on the real-world dataset Matterport3D [3], which contains 34,737/4,898/8,631 train/val/test images for 61/11/18 scenes respectively.

4.2. Implementation Details

Depth-aware panoptic segmentation Transformer. For the panoptic segment decoder and Transformer encoder, we adopt the same settings as [4], with ResNet-50 [19] backbone, 9 decoder layers, and 100 queries. The mask loss $\mathcal{L}_{\text{mask}}$ comprises binary cross-entropy loss $\mathcal{L}_{\text{ce}}$ and dice loss [36] $\mathcal{L}_{\text{dice}}$. We set loss weights as $\lambda_{\text{ce}} = 5.0$, $\lambda_{\text{dice}} = 5.0, \lambda_{\text{cls}} = 2.0$. For efficiency, $K = 12,544$ points are randomly sampled during bipartite matching to calculate the matching cost, while the identical number of points are importance sampled for the mask loss following the practice in [4].

In terms of the depth decoder, we set $\lambda_{\text{depth}} = 2.0$, and no extra weight is used to balance the scale-invariant logarithm error and relative square error. As the 2D depth estimation is a regression problem, the sampling strategy used in the panoptic segment decoder is not applicable. To maintain pixel-wise accuracy while capping the computational cost, We down-sample the ground truth via nearest neighbor interpolation to the size of predicted depth maps, which is 1/4 of the original image size.

The 2D network is optimized with AdamW [34] optimizer, with initial learning rate 10^{-4}. A polynomial-based learning rate scheduler multiplies the base learning rate with $(1 - i/N)^{0.9}$, where i and N denote the current iteration and the total number of iterations respectively. On Front-3D [13] dataset, the model is trained for 160k iterations, with batch size 16. In terms of data augmentation, the input image is randomly scaled from the range 0.5 to 2.0 of the original size 320×240 followed by a fixed-size crop to size 240×240. Color augmentations introduced in [32] are also employed. For the evaluation of real-world performance, the model is then fine-tuned on Matterport3D [3] for another 120k iterations, with the same polynomial-based learning rate scheduling.

3D panoptic reconstruction network. We load and freeze the weights of the 2D network obtained in the previous step. The input size is fixed as 320×240, without any cropping. Adopting the empirical loss weights in [8], we set them as $\lambda_{\text{geometry}} = 5.0$, $\lambda_{\text{seg}} = 25.0$ for three-levels of 3D segment losses, and $\lambda_{\text{occupancy}} = 50.0, 25.0, 10.0$ for occupancy at level $64^3, 128^3, 256^3$, respectively. The network is trained for another 110k iterations with a step learning rate schedule and batch size 8. The initial learning rate remains 10^{-4} and decreases by a factor of 10 at the 80k iteration. On Matterport3D [3], we combine 3D weights trained on 3D-Front [13] with 2D weights already fine-tuned on Matterport3D, and train the network for another 100k iterations with learning rate 10^{-4}, which is decayed at the 80k iteration.

The inference procedure to evaluate PRQ is described in Section 3.2, where outputs are voxel-based representations as required by the PRQ metric. For visualization, we apply the marching cubes algorithm [33] to extract the isosurface from the predicted TDF as meshes and assign each vertex a color based on its semantic class and instance id, which is determined by the label of its nearest neighbor in the predicted panoptic segments.

4.3. Metrics

Depth-aware panoptic segmentation. Besides the standard Panoptic Quality (PQ) metric introduced in [25] to evaluate panoptic segmentation, we also adopt Depth-aware Panoptic Quality (DPQ) [38], which evaluates segmentation and depth estimation jointly. Specifically, given prediction p and ground truth g, and depth threshold λ, DPQ^{λ} is computed as

$$\text{DPQ}^{\lambda}(p, g) = \text{PQ}(p^{\lambda}, g) \tag{8}$$

where p^{λ} is obtained by filtering out pixels in p with relative depth errors higher than λ. The overall DPQ is computed by averaging over $\lambda = \{0.1, 0.25, 0.5\}$. We also report the root mean square error (RMSE) of the estimated depth for reference.

3D panoptic reconstruction. We use 3D panoptic reconstruction quality (3D PRQ) defined in [8] to evaluate the model performance on 3D scene understanding. 3D PRQ is

Table 1. 2D quantitative results on 3D-Front [13]. Each cell contains values averaged over all / thing / stuff classes. Note that Pano-Re [8] has only instance segmentation, and thus metrics including stuff classes are unavailable.

Method	PQ ↑	$\text{DPQ}^{0.1}$ ↑	DPQ ↑	RMSE ↓
Pano-Re [8]	– / 68.39 / –	– / 60.18 / –	– / 64.99 / –	**0.10 / 0.14 / 0.08**
Uni-3D	73.80 / **80.19** / 54.63	66.85 / **71.02** / 54.32	71.62 / **77.26** / 54.71	0.12 / 0.15 / 0.09

Table 2. 3D quantitative results on 3D-Front [13]. Values for Mesh R-CNN and Total3D are taken from [13], which are finetuned by the authors on 3D-Front.

Method	PRQ	RSQ	RRQ	PRQ	RSQ	RRQ	PRQ	RSQ	RRQ
					Things			*Stuff*	
Mesh R-CNN [17]	–	–	–	20.90	38.00	53.20	–	–	–
Total3D [37]	15.08	36.63	40.15	13.77	34.88	38.89	20.94	44.49	45.85
Pano-Re [8]	42.60	53.71	70.85	36.79	49.57	65.67	68.73	72.36	94.19
Uni-3D (Single-scale)	52.48	60.91	83.90	47.22	56.60	81.56	76.17	80.28	94.39
Uni-3D (Multi-scale)	**53.54**	**61.67**	**84.71**	**48.33**	**57.44**	**82.43**	**77.00**	**80.72**	**94.97**

a simple extension of PQ in 3D space. For a specific category c, the corresponding PRQ value is defined as

$$\text{PRQ}^c = \frac{\sum_{(i,j)\in \text{TP}^c} \text{IoU}(i,j)}{|\text{TP}^c| + 0.5|\text{FP}^c| + 0.5|\text{FN}^c|} \tag{9}$$

where TP, FP, and FN is true positives, false positives, and false negatives for the category c, respectively. PRQ^c can also be divided into two terms: RSQ^c representing segmentation and RRQ^c representing recognition accuracy. RSQ^c and RRQ^c can be computed as

$$\text{RSQ}^c = \frac{\sum_{(i,j)\in \text{TP}} \text{IoU}(i,j)}{|\text{TP}^c|} \tag{10}$$

$$\text{RRQ}^c = \frac{|\text{TP}^c|}{|\text{TP}^c| + 0.5|\text{FP}^c| + 0.5|\text{FN}^c|} \tag{11}$$

4.4. Baselines

We compare the proposed method with state-of-the-art approaches that can perform the 3D panoptic scene reconstruction task, including Mesh R-CNN [17], Total3D [37], and Pano-Re [8]. Mesh R-CNN reconstructs instances in input images by instance segmentation features, so it could not predict 3D stuff shapes and we only evaluate its performance on thing reconstruction. For Total3D, we follow [8] and set its layout prediction target as wall and floor for evaluating stuff reconstruction quality.

4.5. Results

2D quantitative results. We provide the results of quantitative evaluation on 3D-Front [13] with Pano-Re [8] in Table 1. Here only DPQ with the strictest $\lambda = 0.1$ and the average over all the thresholds are shown. Our framework outperforms Pano-Re in terms of 2D segmentation, while the depth estimation is on par, with RMSE lower than 5% of the mean depth (2.34) in the dataset. For both frameworks, the depth estimation errors in thing classes tend to be higher than in stuff classes. We will discuss the impact of depth accuracy on the final 3D reconstruction results in Section 4.6.5.

3D quantitative results. We show quantitative comparisons in Table 2, evaluated with the PRQ metric proposed in [8]. Our method, Uni-3D, either with single- or multi-scale feature lifting, is able to surpass Pano-Re by ∼10 PRQ. The improvement is consistent through thing and stuff classes.

3D qualitative results. We present qualitative results in Figure 4. As shown in the figure, Mesh R-CNN (column 2) tends to generate noisy reconstructed shapes. Total3D predicts shape-based instance category priors, which leads to inconsistency between shapes and 2D observations (column 3). These two methods predict 3D instance shapes and layouts separately, resulting in poor global structure and arrangement. Pano-Re and the proposed Uni-3D system are able to accurately generate instance geometry and scene layouts. Compared to Pano-Re (column 4), Uni-3D performs better in 3D segmentation (column 5) due to its universal 3D scene reconstruction pipeline, which correctly distinguishes separate instances and provides precise semantic label predictions. Furthermore, Uni-3D demonstrates better reconstruction quality and alignment with the ground truth in all five samples.

Evaluation on real-world data. We show 3D quantitative evaluations on Matterport3D [3] in Table 3 and some qualitative examples in Figure 5. Compared with synthetic dataset 3D-Front, real-world images are significantly more challenging due to the complexity of scene arrangement, noises in input data, and quality of labels. Thanks to the unified paradigm, Uni-3D is able to effectively reconstruct the shape and provide semantics for many 3D instances.

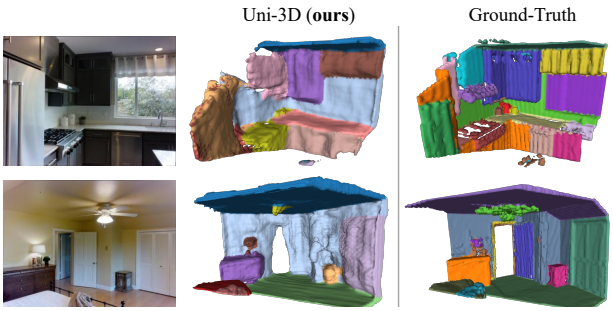

Figure 5. Qualitative results of panoptic 3D reconstruction on Matterport3D [3]. Different colors represent separate instances.

Table 3. 3D quantitative results on Matterport3D [3].

Method	PRQ	RSQ	RRQ	PRQ	RSQ *Things*	RRQ	PRQ	RSQ *Stuff*	RRQ
Mesh R-CNN [17]	–	–	–	6.29	31.12	15.60	–	–	–
Pano-Re [8]	7.01	28.57	17.65	6.34	26.06	16.06	10.78	40.03	26.77
Uni-3D	**8.21**	**37.18**	**21.53**	**7.28**	**36.83**	**19.39**	**11.00**	**38.23**	**27.96**

4.6. Ablation Studies

We provide various ablation studies to demonstrate the efficacy of our system designs. Except for comparisons regarding multi-scale features, we only utilize the single-scale version of our framework in this section.

Table 4. Ablation studies on 3D-Front [13] with the 3D reconstruction network in Pano-Re [8].

Method	PRQ	RSQ	RRQ
Pano-Re [8]	42.60	53.71	70.85
Pano-Re w/ our seg	43.10	54.34	71.06
Pano-Re w/ our depth	32.64	45.58	63.56
Pano-Re w/ our seg + depth	33.34	45.57	65.19
Pano-Re w/ our 2D network	46.00	53.81	83.04
Uni-3D	**52.48**	**60.91**	**83.90**

4.6.1 Importance of Unified Architecture

In Table 4, we demonstrate the effectiveness of our unified design for 3D reconstruction, where 3D stuff and things are unified as 3D segments, guided by 2D queries. As our 2D segmentation outperforms that of Pano-Re [8], we first replace their 2D segmentation and/or depth estimation results with ours while retaining features from their ResNet-18 backbone (row 2 – 4 in the table). Our 2D segmentation results improve the 3D reconstruction quality of Pano-Re, while our depth decreases PRQ regardless of segmentation, which is in line with 2D qualitative results in Table 1.

We then substitute the 2D network of Pano-Re with our depth-aware panoptic segmentation framework. Thanks to the stronger feature backbone, further boost to PRQ (+2.9) is achieved on top of replacing 2D segmentation results. However, even with the same 2D network, Uni-3D still surpasses Pano-Re by a large margin (+6.48 in PRQ). This illustrates the importance of our unified design of 3D segments and 2D queries as priors in reconstruction quality.

4.6.2 Multi-scale Feature Lifting

In Section 3.2, we introduce back-projection of multi-scale 2D features and fuse them into the U-Net encoders to leverage the richer semantics in these features. Table 2 provide comparisons between Uni-3D with single- and multi-scale feature lifting. Multi-scale features bring a prominent PRQ boost 1.06, with the increase in thing classes (+1.11) larger than in stuff classes (+0.83). Figure 6 visualizes two samples where multi-scale features help the 3D network reconstruct better fine details, *i.e*. seatbacks in the top row, and table-top vases in the bottom row. The wall in the bottom row also looks more complete for the multi-scale variant due to a more accurate prediction of the TDF.

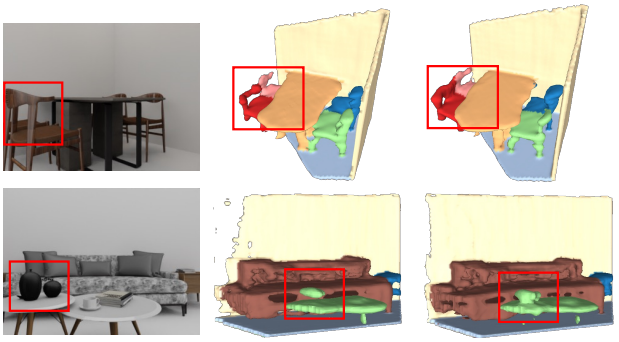

Figure 6. Side-by-side comparison of reconstruction results between single- (column 2) and multi-scale (column 3) feature lifting. Multi-scale features enhance the reconstruction quality of finer details.

4.6.3 Features Used in Back-projection

Following [8], multiple types of inputs, *i.e*. 2D features, depth, and 2D semantic segmentation maps are back-projected and encoded in the 3D network. We ablate the use of these features by removing one at a time and the results are provided in Table 5.

Efficacy of 2D segmentation masks as priors. The use of semantic segmentation maps is equivalent to the instance propagation in [8]. The inclusion of 2D segmentation maps does improve 3D results, especially in thing classes. However, the performance gain is minor (0.33 PRQ) on Uni-3D compared with Pano-Re [8]. The latter suffers a large PRQ drop (22.84) without instance propagation, mostly in thing classes. As we generate 3D convolution kernels from 2D query embeddings, along with the 3D features lifted from 2D Transformer encoders, the strong affinity between queries and features after multiple layers of depth or segment decoders are maintained, which provides significantly more robust 2D priors for both thing and stuff segments in 3D panoptic reconstruction.

Table 5. Ablation studies on 3D-Front [13] dataset with single-scale features in back-projection. Our model remains relatively robust with different feature inputs.

Method	PRQ		
	all	*Things*	*Stuff*
Pano-Re [8]	42.60	36.79	68.73
− Instance propagation	19.76 (−22.84)	9.60 (−27.19)	65.47 (−3.26)
+ GT depth	44.58 (+1.98)	38.17 (+1.38)	73.43 (+4.70)
Uni-3D	52.48	47.22	76.17
− Semantic segm. map	52.15 (−0.33)	46.81 (−0.41)	76.20 (+0.03)
− Depth DF embedding	52.44 (−0.04)	47.22	75.90 (−0.27)
− Mask feature	52.03 (−0.45)	46.78 (−0.44)	75.65 (−0.52)
− Depth feature	52.32 (−0.16)	47.14 (−0.08)	75.66 (−0.51)
+ GT depth	56.13 (+3.65)	51.55 (+4.33)	76.71 (+0.54)

Mask vs depth feature. For mask feature $P^S(F_4)$ and depth feature $P^D(F_4)$, they are derived from the same feature F_4 in the Transformer encoder but projected into different spaces that are optimized for segmentation and depth estimation respectively. Removing mask features $P^S(F_4)$ results in a 0.45 PRQ drop, which is much larger than the drop (0.16) when removing depth features $P^D(F_4)$, signifying the closer affinity between query embeddings $K(q_i^S)$ and mask features. Also, despite the dip in performance when only the depth feature is used, it alone can still yield a relatively high PRQ of 52.03, demonstrating the benefits of interactions between depth estimation and panoptic segmentation in the 2D Transformer, where the knowledge is shared across tasks. Hence, even though the two features are just different "views" of the feature F_4, we incorporate both into our 3D network.

Use of depth information. Regarding the depth as a distance field embedding, we only find it yields a minor improvement (0.27) in stuff PRQ. In terms of features, the depth feature has a greater influence on stuff PRQ (0.51) than things (0.08), consistent with the functionality exhibited by depth DF embedding.

4.6.4 Effect of 2D Queries and Features in 3D Network

In Table 6, we further ablate the model by either removing 2D features (both mask feature $P^S(F_4)$ and depth feature $P^D(F_4)$), or replacing learned 2D queries q_i^S with randomly initialized learnable query embeddings, or doing both. When 2D features are used, without learned 2D queries, PRQ only has a minor dip (0.15) on stuff classes, while it suffers a larger drop (0.96) on thing classes. Similar trends can be observed when 2D features are not present, where stuff PRQ remains the same while thing PRQ drops 1.66. These results demonstrate that the learned 2D queries mainly affect object instances instead of stuff layouts in 3D, where the latter can be straightforwardly solved by semantic segmentation.

In terms of the 2D features, removing them brings about a consistent decrease in PRQ for both thing and stuff classes. Without 2D features, the PRQ drop on thing classes is smaller when 2D queries are used (1.68 vs 2.38), while for stuff classes the difference is minor (2.45 vs 2.30). It suggests that the 2D queries serve as strong priors for 3D that majorly encode instance information, and the 2D features provide rich semantics crucial for both instance and stuff layouts.

4.6.5 Effect of Depth Estimation

The efficacy of back-projection relies on the depth accuracy, and therefore we investigate its effects by providing ground truth depth to the model. The results in Table 5 demonstrate that for Uni-3D, the depth impacts more significantly the reconstruction quality for thing classes, which often contain finer structures compared with stuff classes such as wall or floor. For stuff classes, the performance is close to saturation as only 0.54 PRQ gain is observed with ground truth depth. This is in contrast to [8], where depth inaccuracies penalize more stuff instances.

Table 6. Ablation studies of Uni-3D (single-scale) on 3D-Front [13] dataset regarding the effects of 2D queries and features. 2D features pertain to both mask feature $P^S(F_4)$ and depth feature $P^D(F_4)$, while 2D queries are q_i^S from the last decoder layer in the 2D network.

2D Queries	2D Features	PRQ		
		all	*Things*	*Stuff*
✓	✓	52.48	47.22	76.17
	✓	51.67 (−0.81)	46.26 (−0.96)	76.02 (−0.15)
✓		50.67 (−1.81)	45.54 (−1.68)	73.72 (−2.45)
		49.30 (−3.18)	43.88 (−3.34)	73.72 (−2.45)

5. Conclusion

In this paper, we present Uni-3D, a universal approach that unifies instance and layout representation by leveraging a query-based network design. It also incorporates depth-aware panoptic segmentation to improve 2D depth estimation and segmentation qualities, thereby enhancing the robustness of 3D predictions. The proposed method significantly outperforms related approaches both qualitatively and quantitatively, demonstrating the superiority of our universal model design.

Limitations. Uni-3D is capable of hallucinating occluded parts in the 2D input image. Yet we still observe that its performance may degrade in certain cases where large areas are occluded or only a limited portion of an instance is observable for reconstruction.

Acknowledgement This work is supported by NSF Award IIS-2127544.

References

[1] Joseph J Atick, Paul A Griffin, and A Norman Redlich. Statistical approach to shape from shading: Reconstruction of three-dimensional face surfaces from single two-dimensional images. *Neural computation*, pages 1321–1340, 1996. 2

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 213–229. Springer, 2020. 2

[3] 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 *2017 International Conference on 3D Vision (3DV)*, pages 667–676. IEEE, 2017. 6, 7, 8

[4] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1290–1299, June 2022. 2, 3, 6

[5] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *Advances in Neural Information Processing Systems*, 34:17864–17875, 2021. 2, 4

[6] German KM Cheung, Simon Baker, and Takeo Kanade. Visual hull alignment and refinement across time: A 3d reconstruction algorithm combining shape-from-silhouette with stereo. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 2, pages II–375. IEEE, 2003. 2

[7] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14*, pages 628–644. Springer, 2016. 2

[8] Manuel Dahnert, Ji Hou, Matthias Nießner, and Angela Dai. Panoptic 3d scene reconstruction from a single rgb image. *Advances in Neural Information Processing Systems*, 34:8282–8293, 2021. 1, 2, 3, 4, 5, 6, 7, 8, 9

[9] Daan De Geus, Panagiotis Meletis, and Gijs Dubbelman. Panoptic segmentation with a joint semantic and instance segmentation network. *arXiv preprint arXiv:1809.02110*, 2018. 2

[10] Maximilian Denninger and Rudolph Triebel. 3d scene reconstruction from a single viewport. In *European Conference on Computer Vision*, pages 51–67. Springer, 2020. 1

[11] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems*, 27, 2014. 4

[12] 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. 2

[13] Huan Fu, Bowen Cai, Lin Gao, Ling-Xiao Zhang, Jiaming Wang, Cao Li, Qixun Zeng, Chengyue Sun, Rongfei Jia, Binqiang Zhao, et al. 3d-front: 3d furnished rooms with layouts and semantics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10933–10942, 2021. 6, 7, 8, 9

[14] Huan Fu, Rongfei Jia, Lin Gao, Mingming Gong, Binqiang Zhao, Steve Maybank, and Dacheng Tao. 3d-future: 3d furniture shape with texture. *International Journal of Computer Vision*, 129:3313–3337, 2021. 5, 6

[15] Naiyu Gao, Fei He, Jian Jia, Yanhu Shan, Haoyang Zhang, Xin Zhao, and Kaiqi Huang. Panopticdepth: A unified framework for depth-aware panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1632–1642, 2022. 2, 3

[16] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3354–3361. IEEE, 2012. 4

[17] Georgia Gkioxari, Jitendra Malik, and Justin Johnson. Mesh r-cnn. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9785–9795, 2019. 1, 2, 7, 8

[18] Georgia Gkioxari, Nikhila Ravi, and Justin Johnson. Learning 3d object shape and layout without 3d supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1695–1704, 2022. 2

[19] 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. 6

[20] Berthold KP Horn. Shape from shading: A method for obtaining the shape of a smooth opaque object from one view. 1970. 2

[21] 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. *Advances in Neural Information Processing Systems*, 31, 2018. 1

[22] 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. 3

[23] 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. 2

[24] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7122–7131, 2018. 1

[25] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6399–6408, 2019. 2, 6

[26] Weicheng Kuo, Anelia Angelova, Tsung-Yi Lin, and Angela Dai. Mask2cad: 3d shape prediction by learning to segment and retrieve. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 260–277. Springer, 2020. 2, 3

[27] Qizhu Li, Anurag Arnab, and Philip HS Torr. Weakly-and semi-supervised panoptic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 102–118, 2018. 2

[28] Yanwei Li, Xinze Chen, Zheng Zhu, Lingxi Xie, Guan Huang, Dalong Du, and Xingang Wang. Attention-guided unified network for panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7026–7035, 2019. 2

[29] Yanwei Li, Hengshuang Zhao, Xiaojuan Qi, Liwei Wang, Zeming Li, Jian Sun, and Jiaya Jia. Fully convolutional networks for panoptic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 214–223, 2021. 2

[30] Zhiqi Li, Wenhai Wang, Enze Xie, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, Ping Luo, and Tong Lu. Panoptic segformer: Delving deeper into panoptic segmentation with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1280–1289, 2022. 2

[31] Sainan Liu, Vincent Nguyen, Yuan Gao, Subarna Tripathi, and Zhuowen Tu. Towards panoptic 3d parsing for single image in the wild. *arXiv preprint arXiv:2111.03039*, 2021. 2, 3

[32] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 21–37. Springer, 2016. 6

[33] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169, 1987. 6

[34] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2017. 6

[35] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4460–4470, 2019. 2

[36] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*, pages 565–571. Ieee, 2016. 6

[37] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 55–64, 2020. 1, 7

[38] Siyuan Qiao, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Vip-deeplab: Learning visual perception with depth-aware video panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3997–4008, 2021. 2, 4, 6

[39] Stephan R Richter and Stefan Roth. Discriminative shape from shading in uncalibrated illumination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1128–1136, 2015. 2

[40] Markus Schön, Michael Buchholz, and Klaus Dietmayer. Mgnet: Monocular geometric scene understanding for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15804–15815, 2021. 2

[41] 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/CVF international conference on computer vision*, pages 2172–2182, 2019. 1

[42] 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. 1

[43] 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. 2

[44] Andrew P Witkin. Recovering surface shape and orientation from texture. *Artificial intelligence*, 17(1-3):17–45, 1981. 2

[45] Jiajun Wu, Yifan Wang, Tianfan Xue, Xingyuan Sun, Bill Freeman, and Josh Tenenbaum. Marrnet: 3d shape reconstruction via 2.5 d sketches. *Advances in neural information processing systems*, 30, 2017. 1

[46] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. *Advances in neural information processing systems*, 29, 2016. 2

[47] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun. Upsnet: A unified panoptic segmentation network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8818–8826, 2019. 2

[48] Xiuming Zhang, Zhoutong Zhang, Chengkai Zhang, Josh Tenenbaum, Bill Freeman, and Jiajun Wu. Learning to reconstruct shapes from unseen classes. *Advances in neural information processing systems*, 31, 2018. 1