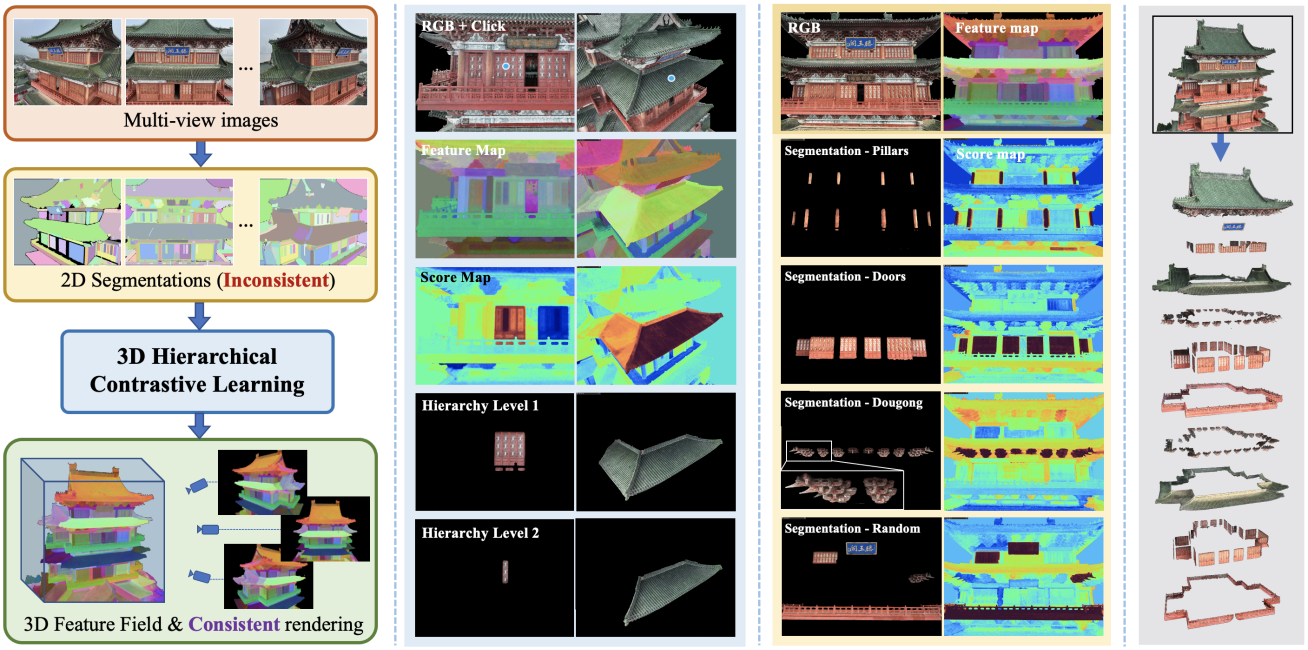
OmniSeg3D: Omniversal 3D Segmentation via Hierarchical Contrastive Learning

Haiyang Ying[1], Yixuan Yin[1], Jinzhi Zhang[1], Fan Wang[2], Tao Yu[1], Ruqi Huang[1], Lu Fang[1†]

[1]Tsinghua University, [2]Alibaba Group



(a) OmniSeg3D Pipeline (b) Hierarchical Segmentation (c) Multi-object Selection (d) 3D Discretization

Figure 1. We propose **OmniSeg3D**, a 3D segmentation framework that (a) takes multi-view inconsistent 2D segmentations as input, and outputs a consistent 3D feature field via a hierarchical contrastive learning method. This method supports (b) hierarchical segmentation, (c) multi-object selection, and (d) holistic discretization in an interactive manner. Our code and demo are available at project page.

Abstract

Towards holistic understanding of 3D scenes, a general 3D segmentation method is needed that can segment diverse objects without restrictions on object quantity or categories, while also reflecting the inherent hierarchical structure. To achieve this, we propose OmniSeg3D, an omniversal segmentation method aims for segmenting anything in 3D all at once. The key insight is to lift multi-view inconsistent 2D segmentations into a consistent 3D feature field through a hierarchical contrastive learning framework, which is accomplished by two steps. Firstly, we design a novel hierarchical representation based on category-agnostic 2D segmentations to model the multi-level relationship among pixels. Secondly, image features rendered from the 3D feature field are clustered at different levels, which can be further drawn closer or pushed apart according to the hierarchical relationship between different levels. In tackling the challenges posed by inconsistent 2D segmentations, this framework yields a global consistent 3D feature field, which further enables hierarchical segmentation, multi-object selection, and global discretization. Extensive experiments demonstrate the effectiveness of our method on high-quality 3D segmentation and accurate hierarchical structure understanding. A graphical user interface further facilitates flexible interaction for omniversal 3D segmentation.

[†]Corresponding author (fanglu@tsinghua.edu.cn, website).

1. Introduction

3D segmentation forms one of the cornerstones in 3D scene understanding, which is also the basis of 3D interaction, editing, and extensive applications in virtual reality, medical analysis, and robot navigation. To meet the requirement of complex world sensing, a general/omniversal category-agnostic 3D scene segmentation method is required, capable of segmenting any object in 3D without limitations on object quantity or categories. For instance, to accurately discretize a pavilion as shown in Fig. 1, the user needs to accurately segment each roof, column, eaves, and other intricate structures. Existing 3D-based segmentation methods based on 3D point clouds, meshes, or volumes fall short of these requirements. They are either restricted to limited categories due to the scarcity of large-scale 3D datasets, such as learning-based methods [18, 25, 30], or they could only identify local geometric similarity or smoothness without extracting semantic information, as typified by traditional algorithms [15, 20, 44].

An alternative approach involves lifting 2D image understanding to 3D space, leveraging the impressive class-agnostic 2D segmentation performance achieved by recent methods [8, 26, 28, 31, 46]. Current lifting-based methods either rely on annotated 2D masks [4, 53, 64], or are restricted to a limited set of pre-defined classes [3, 45]. Other methods propose distilling semantic-rich image features [28, 42] onto point clouds [41, 48] or NeRF [17, 24, 27]. However, due to the absence of boundary information, directly distilling these semantic feature into 3D space often leads to noisy segmentations [24, 41]. Further works use SAM [26] or video segmentation methods [37] to generate accurate 2D masks of targeted objects, and unproject them into 3D space [6]. However, these approaches are limited to single-object segmentation and exhibit unstable results in cases with severe occlusion because the 2D segmentation is performed on each image independently.

Therefore, significant challenges still persist. First, multi-view consistency remains an obstacle due to the substantial variations in 2D segmentations across different viewpoints. Second, ambiguity arises when distinguishing in-the-wild objects like eaves and roofs, which inherently possess a hierarchical semantic structure. To this end, we propose OmniSeg3D, an **Omniversal 3D Segmentation** method which enjoys multi-object, category-agnostic, and hierarchical segmentation in 3D all at once. We demonstrate that a global 3D feature field (which can be formulated on NeRF [36, 40, 57], point cloud [23], mesh [50, 59], etc) is inherently well-suited for integrating occlusion-free, boundary-clear, and hierarchical semantic information from 2D segmentations through hierarchical contrastive learning. The key lies in hierarchically clustering 2D image features rendered from the 3D feature field at different levels of segmentation blocks, where the multi-level segmentations

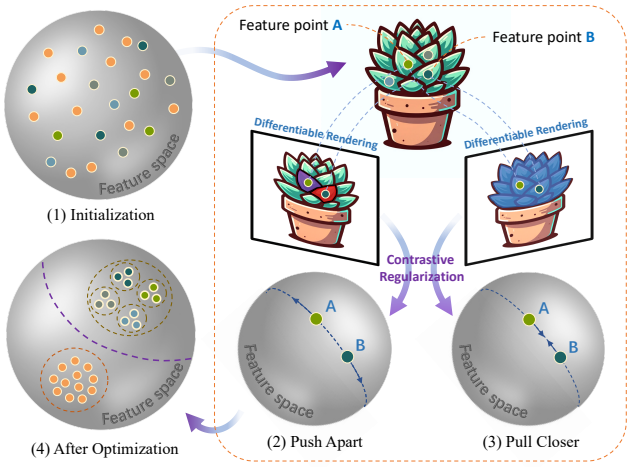


Figure 2. **Method Overview**. We utilize differentiable rendering on a 3D feature field to generate 2D feature points, which are then regularized by multi-view 2D segmentations through a hierarchical contrastive learning strategy, resulting in a hierarchical 3D feature field that supports versatile 3D segmentation tasks.

are specified by a proposed hierarchical 2D representation. Then the clustered features will be drawn closer or pushed apart via a hierarchical contrastive loss, which enables the learning of a feature field that encodes hierarchical information into feature distances, effectively eliminating semantic inconsistencies between different images. This unified framework facilitates multi-object selection, hierarchical segmentation, global discretization in 3D space.

We evaluate OmniSeg3D on segmentation tasks for single object selection and hierarchical inference. Extensive quantitative and qualitative results on real-world and synthetic datasets demonstrate our method enjoys high-quality 3D object segmentation and holistic comprehension of scene structure across various scales. An interactive interface is also provided for flexible 3D segmentation. Our contributions are summarized as follows:

- We propose a **hierarchical 2D representation** to reveal and store the part-level relationship within objects based on class-agnostic 2D segmentations and a voting strategy.
- We present a **hierarchical contrastive learning method** to optimize a globally consistent 3D hierarchical feature field given 2D observations.
- Extensive experiments demonstrate that our **omniversal 3D segmentation framework** can segment anything in 3D all at once, which enables hierarchical segmentation, multi-object selection, and 3D discretization.

2. Related Works

2.1. 2D Segmentation

2D segmentation has experienced a long history. Early works mainly rely on the clue of pixel similarity and con-

tinuity [1, 12, 16] to segment images. Since the introduction of FCN [33] and large-scale 2D datasets [14, 52], there has been a rapid expansion in research of different subfields of 2D segmentation [7, 19, 25, 62]. The involvement of transformer [49] in the segmentation domain has led to the proposal of several novel segmentation architectures [10, 11, 63]. However, most of these methods are limited to pre-defined class labels.

Prompt-based segmentation is a special task that enables segmenting unseen object categories [8, 31, 46]. One recent breakthrough is the Segment Anything Model (SAM) [26], aiming to unify the 2D segmentation task through the introduction of a prompt-based segmentation approach , is considered a promising innovation in the field of vision.

2.2. 3D Segmentation

Closed-set segmentation. The task of 3D segmentation has been explored with various types of 3D representation such as RGBD images [51, 54], pointcloud [21, 55, 56], and voxels [18, 22, 30, 32]. However, due to the insufficiency of annotated 3D datasets for training a unified 3D segmentation model, they are still limited to closed-set 3D understanding, which largely restricts the application scenarios.

Given the shortage of annotated 3D datasets [13, 60] for the development of foundational 3D models, recent works have proposed to lift 2D information into 3D for 3D segmentation and understanding. Some works rely on ground truth masks [4, 53, 64] or pre-trained 2D semantic/instance segmentation models for mask generation [3, 45]. However, obtaining ground truth annotation is often impractical for general scenarios, and model-based methods typically provide closed-set object masks only. ContrastiveLift [3] proposes to segment closed-set 3D objects via contrastive learning. However, it cannot handle unseen classes and reveal object hierarchy. In contrast, our method achieves panoptic, category-agnostic, and hierarchical segmentation based on a hierarchical contrastive learning framework, which can be interpreted as a sound combination of click-based segmentation methods and holistic 3D modeling.

Open-set segmentation. LERF [24] and subsequent works [17, 27, 48] propose to distill language feature [42] into 3D space for open-vocabulary interactive segmentation. Since the learned feature is trained on entire images without explicit boundary supervision, these methods prone to produce noisy segmentation boundaries. Besides, these methods are unable to distinguish different instances due to the lack of instance-level supervision. Alternatively, we take advantage of category-agnostic segmentation methods and distill the 2D results into 3D to get a consistent feature field and enable high-quality 3D segmentation.

SPInNeRF [37] utilizes video segmentation to initialize 2D masks and then lift them into 3D space with a NeRF. A followed multi-view refinement stage is utilized to achieve consistent 3D segmentation. SA3D [6] introduces an online interactive segmentation method that propagates one SAM [26] mask into 3D space and other views iteratively. However, these methods may heavily rely on a good choice of reference view and cannot handle complex cases such as severe occlusion. Instead, our method can segment anything in 3D all at once via a global consistent feature field, which is more robust to object occlusion.

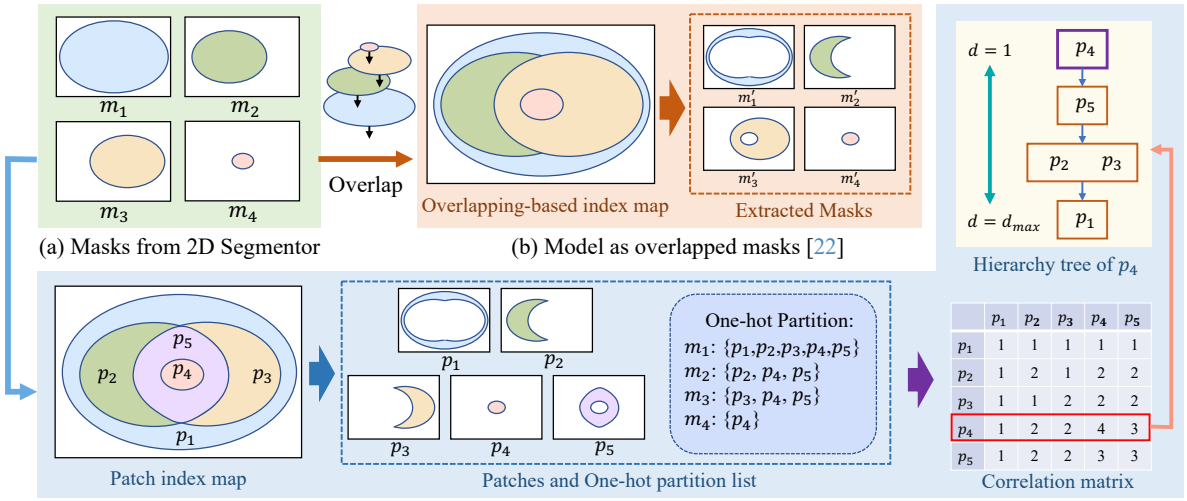
Hierarchical segmentation. For hierarchical segmentation, existing methods mainly rely on the paradigm of geometric analysis of single-class objects [9, 38, 39, 56, 58], which can only be applied to specific categories. Instead, we focus on general scenarios and achieve category-agnostic hierarchical segmentation in 3D.

3. Methods

Given a set of 2D images with poses [36] as input, our goal is to learn a 3D feature field that supports multi-object, category-agnostic, and hierarchical segmentation all at once. We first use a pretrained 2D segmentation model to segment each image into a set of masks M_{segs}. The masks are then organized into smaller units P_{segs} accompanied with a correlation indicator C_{hi}. During training, a pre-defined 3D feature field can be rendered to features $\mathbf{f} \in \mathbb{R}^D$ on 2D image plane. With our proposed hierarchical contrastive clustering strategy, the rendered features will be forced to establish precise feature distance with right order that corresponds to the patch relationship depicted in C_{hi}. In this section, we first introduce our hierarchical 2D representation (Sec. 3.1) which models the hierarchical relationship among pixels. Then a hierarchical contrastive learning framework will be discussed (Sec. 3.2), including both basic and hierarchical implementations for lifting 2D masks into 3D space. Implementation details are shown in Sec. 3.3. Finally, we show how to use the optimized 3D field to achieve various 3D segmentation tasks interactively (Sec. 3.4).

3.1. Hierarchical Representation

2D Label Map Creation. We borrow the idea from [64] that multi-view 2D label maps can be lifted into a 3D feature field via differentiable rendering. The key difference is that for omniversal segmentation, a 2D segmentation method should be able to handle unseen categories. We seek solution from click-based method like SAM [26], which exhibits a class-agnostic property. Given an input image I, we sample a grid of points (typically 32×32) as the prompts and send them into a pretrained SAM [26] to get a set of 2D binary masks $M_{segs} = \{m_i \in \mathbb{R}^{H \times W} | i = 1, ..., |M_{segs}|\}$ (see Fig. 3(a)). To create a label map as training data for 3D field optimization (as in [64]), one way is to overlap masks in M_{segs} one-by-one based on their pixel counts (see Fig. 3(b)). Unfortunately, this method (as done in



	p_1	p_2	p_3	p_4	p_5
p_1	1	1	1	1	1
p_2	1	2	1	2	2
p_3	1	1	2	2	2
p_4	1	2	2	4	3
p_5	1	2	2	3	3

Figure 3. **2D Hierarchical Representation**. (a) For each image, click-based 2D segmentors provide a set of masks $\{m_i\}$. (b) Directly overlapping masks implemented by conventional methods [26] lead to the loss of hierarchical information. (c) Patch-based modeling effectively preserves inclusion information. The hierarchical representation of each image includes a patch index map I_p and a correlation matrix C_{hi}, where the relevance between p_i and other patches is evaluated via a voting strategy.

SAM [26]), may destroy the hierarchical information embedded inside M_{segs}, since each pixel in image I may belong to more than one masks in M_{segs} (consider the fact that a pixel belonging to the mask of a chair may also belong to the mask of the chair's leg). Alternatively, storing each M_{segs} directly may result in high memory consumption, as $|M_{segs}|$ usually exceeds 500, using memory equivalent to 20x input images.

Hierarchical Modeling. To overcome the aforementioned problem, we design a novel representation that preserves the hierarchical information within each image and largely reduces the memory consumption. Specifically, we divide the entire 2D image into disjoint patches. As shown in Fig. 3(a), let $m_i \in M_{segs}, (i = 1, ..., 4)$ represent masks in M_{segs}. For each pixel, we create a one-hot vector to indicate which masks the pixel belongs to. Then we define a patch set P_{segs}, where each patch includes pixels that share the same one-hot vector as shown in Fig. 3(c). P_{segs} also results in a patch index map I_p, on which each pixel contains an index of the patch.

Next, we proceed to model the hierarchical structure with patches P_{segs} (as the unit) and the original masks M_{segs} (as the correlation binding). The core idea is that, if two patches fall into the same mask, then these two patches has some degree of correlation. To model the degree of the correlation, we introduce a voting-based strategy. Specifically, for each pair of patches p_i and p_j, we count the number of masks that contain both p_i and p_j. By traversing all the patch pairs, we get a matrix $C_{hi} \in \mathbb{R}^{N_p \times N_p}$:

$$C_{hi}(p_i, p_j) = \sum_{k=1}^{N_m} \mathbb{1}(p_i \subseteq m_k) \cdot \mathbb{1}(p_j \subseteq m_k), \quad (1)$$

where $N_m = |M_{segs}|$ is the number of masks and $N_p = |P_{segs}|$ is the number of patches. This process can be interpreted as utilizing masks to vote for the relationship between patches. To inference the hierarchical relationship among patches, we select a patch p_i as the anchor and take the i-th row of matrix $C_{hi}(p_i, \cdot) = v_i$. We then sort the patches according to the vote counts in vector v_i and construct a hierarchical tree for anchor patch p_i, as illustrated in Fig. 3(c). Patches located at higher level (smaller d) in the tree has stronger correlation to p_i, which can serve as the guidance of the hierarchical contrastive learning introduced in the subsequent section. As a summary, we construct the hierarchical representation for each image, which consists of a patch index map I_p and a correlation matrix C_{hi}.

3.2. Hierarchical Contrastive Learning

In this section, we show how to lift the hierarchical relationship of 2D patches into the 3D space through a hierarchical contrastive learning framework.

3D feature field. Our 3D representation is built upon NeRFs [36, 40], which uses an MLP F_Θ to model the density σ_i and color $\mathbf{c}_i$ of each 3D point $\mathbf{x}_i \in \mathbb{R}^3$ under view direction $\mathbf{d}_i \in \mathbb{R}^2$. Additionally, we define a segmentation identity feature $\mathbf{f}_i \in \mathbb{R}^D$ to model semantic information of each 3D point. The formulation is shown below:

$$(\sigma_i, \mathbf{f}_i) = F_\Theta(\gamma_1(\mathbf{x}_i)), \quad \mathbf{c}_i = F_\Theta(\gamma_1(\mathbf{x}_i), \gamma_2(\mathbf{d}_i)), \quad (2)$$

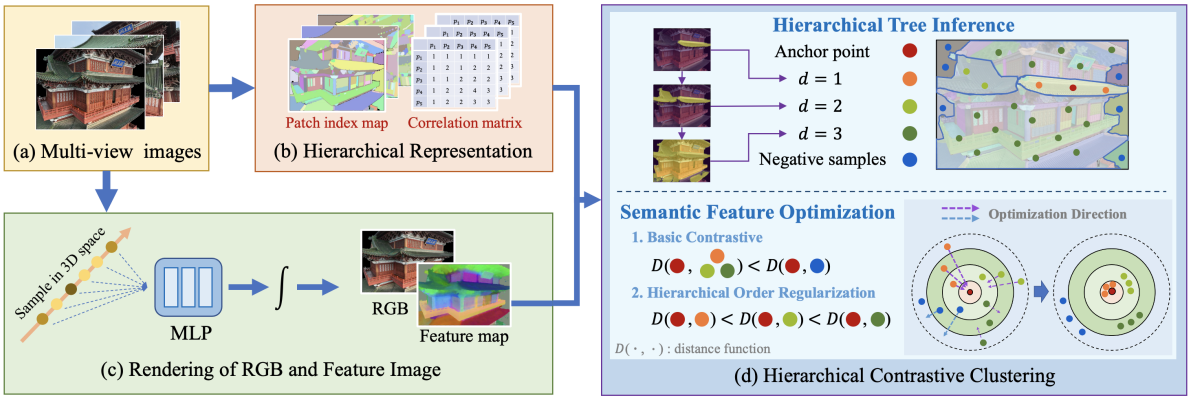


Figure 4. **Hierarchical Contrastive Learning Framework**. (a) For each input RGB image, we apply (b) 2D hierarchical modeling to get a patch index map and a correlation matrix. During training, we utilize (c) NeRF-based rendering pipeline to render features from 3D space and apply hierarchical contrastive learning (d) to the rendered features to optimize the feature field for segmentation.

where γ_1 and γ_2 are positional encoding functions in [40].

Subsequently, by integrating the sampled attributes $\mathbf{c}_i$, σ_i, and $\mathbf{f}_i$ along the ray, we can get rendered $\mathbf{c}(\mathbf{r}) = \sum_{i=1}^{n} T_i \alpha_i \mathbf{c}_i$ and $\mathbf{f}(\mathbf{r}) = \sum_{i=1}^{n} T_i \alpha_i \mathbf{f}_i$ on 2D image plane [24, 36], where $\alpha_i = 1 - exp(-\sigma_i \delta_i)$ is the opacity, $T_i = \prod_{j=1}^{i-1}(1-\alpha_j)$ is the accumulated opacity, and $\delta_i = r_{i+1} - r_i$ is the distance between adjacent samples.

Basic implementation. In this section, we present a basic implementation that lifts 2D segmentations into 3D space via differentiable rendering and contrastive learning. No hierarchical information is considered in this section.

For each image, we randomly sample N points on it and identify the patch id of each point according to the patch index map I_p. Then we render features $\mathbf{f}_i (i \in [1, N])$ of these points via differentiable rendering from the 3D feature field. For each sampled point, we designate the points with the same patch id as positive samples, and all the other sampled points as negative ones. The correlation between two feature points is modelled as cosine distance $\mathbf{f}_i \cdot \mathbf{f}_j$.

We apply a contrastive clustering method [29] to supervised the feature distance between rendered feature points. Specifically, cluster $\mathbf{F}^i (i \in [1, N_p])$ is defined as the collection of rendered features that share the same patch id i and $\mathbf{f}_j^i$ is the j-th feature in $\mathbf{F}^i$. The center of each cluster is defined as the mean value $\bar{\mathbf{f}}^i$ of features in $\mathbf{F}^i$. Then for each chosen feature point $\mathbf{f}_j^i$, we take $\bar{\mathbf{f}}^i$ and $\bar{\mathbf{f}}^k$ as positive and negative samples respectively. The contrastive loss is shown below, which favors high similarity among samples within the same patch p_i and low similarity between samples located in different patches (p_i and p_k):

$$\mathcal{L}_{CC} = -\frac{1}{N_p} \sum_{i=1}^{N_p} \sum_{j=1}^{|\mathbf{F}^i|} \log \frac{\exp(\mathbf{f}_j^i \cdot \bar{\mathbf{f}}^i / \phi_i)}{\sum_{k=1}^{N_p} \exp(\mathbf{f}_j^i \cdot \bar{\mathbf{f}}^k / \phi_k)}, \quad (3)$$

where N_p is the number of patch ids, ϕ_i is the temperature of cluster $\mathbf{F}^i$ to balance the cluster size and variance: $\phi_i = \sum_{j=1}^{n_i} ||\mathbf{f}_j^i - \bar{\mathbf{f}}^i||_2 / (n_i \log(n_i + \alpha))$, $n_i = |\mathbf{F}^i|$, $\alpha = 10$ is a smooth term to prevent small clusters from exhibiting an excessively large ϕ_i.

Note that ConstrastiveLift [3] uses a slow-fast learning strategy for stable training. We refer to contrastive clustering [29] to realize faster training and stable convergence.

Hierarchical implementation. Here we show how to incorporate hierarchical information into the pipeline of contrastive learning. Still we cluster the sampled feature points into N_p feature sets $\mathbf{F}^i (i \in [1, N_p])$ based on the patch index map I_p. Then for each anchor patch p_i, we find all related patches according to the correlation matrix C_{hi} and construct a set $\{S_d^i\}$ where S_d^i is the patch index set at level $d \in [1, d_{max}^i]$ of anchor patch p_i (e.g., $S_{d=3}^{i=4} = \{2, 3\}$ in Fig. 3). Note that all the samples in the related patches are potential positive samples in this formulation.

To achieve hierarchical contrastive clustering in 3D, we employ the hierarchical regularization proposed in [61]. Firstly, we add a regularization term λ^{d-1} to Eq. 3 with a per-level decay factor $\lambda \leq 1$, which means higher penalty is applied to the patches with stronger correlation to the anchor patch p_i. Secondly, a strategy for regularizing the optimization order is implemented to ensure that a patch higher in the hierarchy tree (smaller d) exhibits a higher feature similarity with the anchor patch than patches at lower levels (as shown in Fig. 4(d)). The final loss is shown below:

$$\mathcal{L}_H = \sum_{i=1}^{N_p} \sum_{d=1}^{d_{max}^i} \frac{\lambda^{d-1}}{NL} \sum_{j=1}^{|\mathbf{F}^i|} \sum_{s \in S_d^i} \max(\mathcal{L}^{i,j}(s), \mathcal{L}_{max}^{i,j}(d-1)), \quad (4)$$

where S_d^i is the patch index set at level d of anchor patch p_i, $\mathcal{L}^{i,j}(s)$ is the contrastive loss that favors high similarity

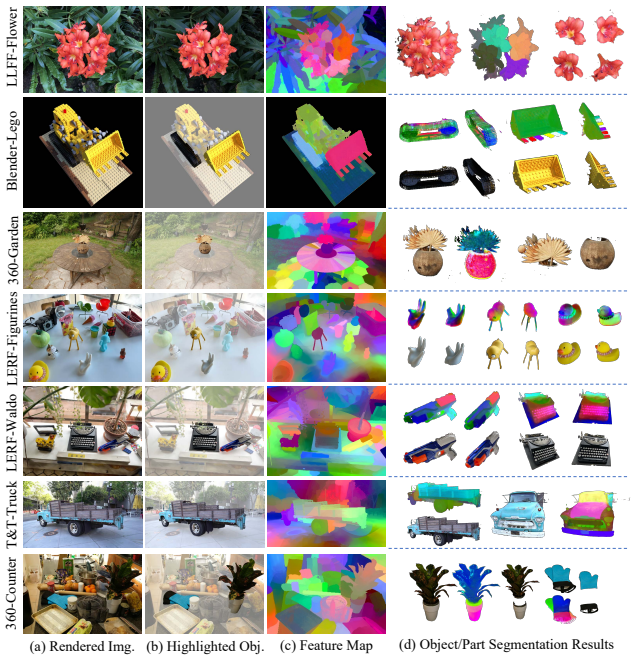


Figure 5. Qualitative performance visualization. We visualize both scene-level and object-level feature maps (with UMAP [34]) to reveal the hierarchical structure learned by OmniSeg3D.

between $\mathbf{f}_j^i$ and feature center of patch p_s ($s \in S_d^i$):

$$\mathcal{L}^{i,j}(s) = -\log \frac{\exp(\mathbf{f}_j^i \cdot \bar{\mathbf{f}}^s/\phi_s)}{\sum_{k=1}^{N_p} \exp(\mathbf{f}_j^i \cdot \bar{\mathbf{f}}^k/\phi_k)}, \tag{5}$$

and $\mathcal{L}_{max}^{i,j}(d)$ is the maximum loss at level d:

$$\mathcal{L}_{max}^{i,j}(d) = \max_{s \in S_d^i} \mathcal{L}^{i,j}(s). \tag{6}$$

Since the volumetric rendering may introduce ambiguity in the calculation of the integration, we apply normalization loss to ensure feature vectors distributed on the sphere surface: $\mathcal{L}_{norm} = \frac{1}{N}\sum_{i=1}^{N}(||\mathbf{f}_i|| - 1)^2$.

3.3. Implementation details

The method is built on top of InstantNGP [40] and we follows the same parameter settings for positional encoding and MLP $F_{\mathbf{\Theta}}$. We adopt a two-stage training paradigm. For each scene, we first train our model with $\mathcal{L}_{geo} = \mathcal{L}_c + w_1\mathcal{L}_{reg}$ to construct right geometry, where $\mathcal{L}_c = \sum_{\mathbf{r}} \|\mathbf{c}(\mathbf{r}) - \mathbf{c}_{\text{gt}}(\mathbf{r})\|_2^2$, and $\mathcal{L}_{reg} = \sum_{\mathbf{r}} -o(\mathbf{r})\log(o(\mathbf{r}))$, $o(\mathbf{r}) = \sum_{i=1}^{N} T_i\alpha_i$. $\mathcal{L}_{reg}$ is used to regularize each ray to be completely saturated or empty. Then the feature field will be supervised via $\mathcal{L}_{sem} = w_2\mathcal{L}_H + w_3\mathcal{L}_{norm}$. The per-level decay factor is set to $\lambda = 0.5$. The hyper-parameters are set to $w_1 = 1e-3$, $w_2 = 5e-4$, $w_3 = 5e2$ for all the experiments. The ray number of each batch is 8192 and both stages are trained for 50k iterations, which takes $30 \sim 40min$ in total on a single RTX 3090 GPU.

Although we choose InstantNGP [40] as our backbone, we demonstrate that OmniSeg3D is a lightweight plug-in which can be easily adapted to 3D representations like mesh, point cloud, and gaussian splatting [23]. For 2D backbones, beside of SAM [26], other click-based segmentation methods like [8, 31, 46] can be used as a substitute. Please refer to our supplementary material for more details.

3.4. Interactive Segmentation

To realize flexible and interactive 3D segmentation, we develop a graphical user interface (GUI). The GUI can serve as a novel 3D annotation tool, which may largely improve the efficiency of 3D data annotation and help solve the 3D data shortage problem. One typical case based on NeRF is shown in Fig. 1. With a single click on the object of interest, our model generates a score field based on feature similarities. By adjusting the binarization threshold, the segmentation can seamlessly traverse the scene hierarchy from atomic components to entire objects, and holistic portions of the scene. Besides, users can select and segment multiple objects simultaneously through multiple clicks.

4. Experiments

Our experiments encompass various datasets including indoor [36, 47] and outdoor [2, 24, 35] scenes. Qualitative results can be found in Fig. 5. For quantitative performance, we evaluate OmniSeg3D on both hierarchical (Sec. 4.1) and instance (Sec. 4.2) 3D segmentation tasks.

4.1. Hierarchical 3D Segmentation

Dataset. To quantitatively evaluate OmniSeg3D, we create a scene-scale dataset with hierarchical semantic annotations. We utilize the Replica dataset [47] processed by Semantic-NeRF [64], which comprises 8 realistic indoor scenes. We uniformly sample a total of 281 images and manually annotate each image with a query pixel $\mathbf{q}$ and two corresponding masks, the smaller one M_{L_1} properly included by the larger one $M_{L_2} \supset M_{L_1}$. M_{L_1} and M_{L_2} typically correspond to object parts and complete instances respectively, as shown in Fig. 6. In case multiple levels of reasonable segmentations $M_a \subset M_b \subset M_c$ exist, we choose different pairs as the ground truth (M_{L_1}, M_{L_2}) in different images, so that the selected masks exhibit diverse scales and represent the full range of possible hierarchical relationships present in the scene.

Benchmark. We benchmark our algorithm as follows. The model receives as input a 2D query point $\mathbf{q}$ in the given frame $\mathbf{I}$, and is expected to output a dense 2D score map $\{\text{score}(\mathbf{p}) \,|\, \mathbf{p} \in \mathbf{I}\}$. Ideally, there exist thresholds

Figure 6. Comparison of hierarchical segmentation results on the Replica dataset. Prompts are shown as black dots. Colored pixels denote TP: True-Positive, FP: False-Positive and FN: False-Negative respectively.

Method	mIoU (%)		
	Level 1	Level 2	Average
DINO [5]	67.9	64.2	66.1
LSeg [28]	51.7	82.1	66.9
SAM [26]	92.8	80.2	86.5
Ours, w/o hierarchy	**93.1**	80.4	86.7
OmniSeg3D (ours)	91.3	**88.9**	**90.1**

Table 1. Comparison of hierarchical segmentation on Replica [47].

$th_1 > th_2$ that, when applied to the score map, yields $M_{L_1} \subset M_{L_2}$ respectively:

$$\exists th_i \; s.t. \; M_{L_i} = \{\mathbf{p} \in \mathbf{I} \,|\, \text{score}\,(\mathbf{p}) > th_i\}. \quad (7)$$

For evaluation, we choose the thresholds (th_1, th_2) that maximize the IoU between the predicted masks and the ground truth (M_{L_1}, M_{L_2}), and define the metrics as:

$$\text{IoU}_{L_i} = \max_{th_i} \text{IoU}\left(\{\mathbf{p} \in \mathbf{I} \,|\, \text{score}\,(\mathbf{p}) > th_i\}, M_{L_i}\right), \quad (8)$$

and we have $\text{IoU}_{Avg} = (\text{IoU}_{L_1} + \text{IoU}_{L_2})/2$.

Baseline methods. We first compare OmniSeg3D with state-of-the-art 2D segmentation models and semantic feature extractors. SAM [26] predicts three hierarchical masks from the point query. We compare each to the ground truth masks (M_{L_1}, M_{L_2}) and report the highest IoU. DINO [5] and LSeg [28] (based on CLIP [42]) predict a feature image, which is converted to a score map based on cosine similarities and then binarized using Eq. 8 to compute the IoU. In addition, we compare our full method with the basic implementation in Sec. 3.2, i.e., 3D contrastive learning without hierarchical modelling.

Results. The quantitative and qualitative results of hierarchical segmentation on the Replica [47] dataset are demonstrated in Tab. 1 and Fig. 6 respectively. Our OmniSeg3D achieves the highest average mIoU, while substantially leading in level-2 segmentation, which shows the advantage on high-level semantic understanding.

As shown in Fig. 6, the self-supervised DINO method struggles to delineate clear object boundaries. LSeg captures overall semantics better but fails to discriminate between instances. SAM performs well at fine-grained segmentation, but occasionally fails to group together multiple objects or large regions, resulting in lower level-2 mIoU. Our basic implementation without hierarchical modeling inherits these characteristics of SAM, with slightly better metrics. Our full method achieves large improvements in high-level segmentation while maintains comparable performance in level-1 segmentation, which implies that the hierarchical modelling effectively aggregates fragmented part-whole correlations from multiple views. Moreover, in

Dataset	Method	mIoU (%)	Acc (%)
NVOS	NVOS [43]	70.1	92.0
	ISRF [17]	83.8	96.4
	SA3D [6]	90.3	98.2
	OmniSeg3D (ours)	**91.7**	**98.4**
MVSeg	MVSeg [37]	90.9	98.9
	SA3D [6]	92.4	98.9
	OmniSeg3D (ours)	**94.3**	**99.3**
Replica	MVSeg [37]	32.4	-
	SA3D [6]	83.0	-
	OmniSeg3D (ours)	**84.4**	-

Table 2. Quantitative comparison of instance segmentation.

contrast to the instability in performance that 2D models may exhibit across different resolutions, our OmniSeg3D implicitly integrates voting-based correlations from multi-view inputs, which distills a stable hierarchical semantic order into the 3D representation, thereby enhancing global-scale semantic clustering.

4.2. 3D Instance Segmentation

While designed for omniversal 3D segmentation, our method is able to handle 3D instance segmentation as a subtask. Different from existing methods [6, 37], we do not require instance-specific training. The 3D feature field of OmniSeg3D is trained *only once* for each scene and reused for different instances, while still performing competitively on datasets proposed by previous work.

We follow NVOS [43], SPIn-NeRF [37] and SA3D [6] to benchmark 3D instance segmentation as prompt propagation. For each scene, given prompts (scribbles or masks) in the reference view, the algorithm is supposed to segment the instance in the target view. The predicted mask is compared with the ground truth segmentation. As shown in Tab. 2, OmniSeg3D outperforms the baseline methods in terms of mIoU and pixel-wise accuracy, while alleviating the need to retrain different segmentation fields for the same scene.

4.3. Ablation Studies

Hierarchical decay. As illustrated in Eq. 4, we apply a decay $\lambda \in [0, 1]$ to downweight the contrastive loss for patches of lower correlation with the anchor. Setting $\lambda = 0$ resembles the basic implementation without hierarchical modeling, while setting $\lambda = 1$ puts equal emphasis on samples from all hierarchies, enhancing high-level semantics. Tab. 3 demonstrates hierarchical segmentation results on the Replica dataset. With the increase of λ, IoU_{L_1} decreases while IoU_{L_2} increases, reaching $\mathrm{IoU}_{L_1} \approx \mathrm{IoU}_{L_2}$ at $\lambda = 1$. We choose $\lambda = 0.5$ with the highest average mIoU, implying a balance between local and global semantic clustering.

Hierar. model	Per-level decay λ	Hierar. mIoU (%) Lv.1	Lv.2	Avg.	Instance mIoU (%)
×	-	**93.1**	80.4	86.7	83.6
✓	0.1	92.5	84.7	88.6	84.3
✓	0.2	92.1	86.5	89.4	**84.6**
✓	0.5	91.3	88.9	**90.1**	84.4
✓	1	89.2	**89.2**	89.2	83.3

Table 3. Ablation of hierarchical modelling on Replica.

Feat. dim.	4	8	16	32	64	128
Avg. mIoU	89.8	91.8	93.0	93.0	93.1	93.2

Table 4. Ablation of feature dimensions on `room-0` of Replica.

Feature dimension. We study how the dimension D of semantic features affects the performance of hierarchical contrastive clustering. The Tab. 4 indicates that the average mIoU rises with D and levels off after $D = 16$, suggesting that a D of 16 is sufficient for our algorithm.

5. Limitations

Due to the absence of a clear definition for hierarchy levels, there is no assurance that the objects will be segmented at the same level by simply clustering features with one threshold. To address this issue, text-aligned hierarchical segmentation may be a future direction. Besides, since the contrastive learning is applied on single images, two objects that have never appeared in the same image may have similar semantic feature. This problem can be alleviated by introducing local geometric continuity, but global contrastive learning across images is also a topic worth exploring.

6. Conclusion

In this paper, we propose OmniSeg3D, an omniversal segmentation method that facilitates holistic understanding of 3D scenes. Leveraging a hierarchical representation and a hierarchical contrastive learning framework, OmniSeg3D effectively transforms inconsistent 2D segmentations into a globally consistent 3D feature field while retaining hierarchical information, which enables correct hierarchical 3D sensing and high-quality object segmentation performance. Besides, variant interactive functionalities including hierarchical inference, multi-object selection, and global discretization are realized, which may further enable downstream applications in the field of 3D data annotation, robotics and virtual reality.

Acknowledgements This work is supported in part by Natural Science Foundation of China (NSFC) under contract No. 62125106 and 62088102, in part by Tsinghua-Zhijiang joint research center.

References

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012. 3

[2] Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. *CVPR*, 2022. 6

[3] Yash Bhalgat, Iro Laina, João F Henriques, Andrew Zisserman, and Andrea Vedaldi. Contrastive lift: 3d object instance segmentation by slow-fast contrastive fusion. *arXiv preprint arXiv:2306.04633*, 2023. 2, 3, 5

[4] WANG Bing, Lu Chen, and Bo Yang. Dm-nerf: 3d scene geometry decomposition and manipulation from 2d images. In *The Eleventh International Conference on Learning Representations*, 2022. 2, 3

[5] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021. 7

[6] Jiazhong Cen, Zanwei Zhou, Jiemin Fang, Wei Shen, Lingxi Xie, Xiaopeng Zhang, and Qi Tian. Segment anything in 3d with nerfs. *arXiv preprint arXiv:2304.12308*, 2023. 2, 3, 8

[7] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 3

[8] Xi Chen, Zhiyan Zhao, Yilei Zhang, Manni Duan, Donglian Qi, and Hengshuang Zhao. Focalclick: Towards practical interactive image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1300–1309, 2022. 2, 3, 6

[9] Zhiqin Chen, Andrea Tagliasacchi, and Hao Zhang. Bsp-net: Generating compact meshes via binary space partitioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 45–54, 2020. 3

[10] 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. 3

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

[12] Guy Barrett Coleman and Harry C Andrews. Image segmentation by clustering. *Proceedings of the IEEE*, 67(5): 773–785, 1979. 3

[13] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 3

[14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 3

[15] Peter Dorninger and Clemens Nothegger. 3d segmentation of unstructured point clouds for building modelling. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 35(3/W49A):191–196, 2007. 2

[16] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *International journal of computer vision*, 59:167–181, 2004. 3

[17] Rahul Goel, Dhawal Sirikonda, Saurabh Saini, and PJ Narayanan. Interactive segmentation of radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4201–4211, 2023. 2, 3, 8

[18] Lei Han, Tian Zheng, Lan Xu, and Lu Fang. Occuseg: Occupancy-aware 3d instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2940–2949, 2020. 2, 3

[19] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 3

[20] Karl Heinz Höhne and William A Hanson. Interactive 3d segmentation of mri and ct volumes using morphological operations. *Journal of computer assisted tomography*, 16(2): 285–294, 1992. 2

[21] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11108–11117, 2020. 3

[22] Jing Huang and Suya You. Point cloud labeling using 3d convolutional neural network. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 2670–2675. IEEE, 2016. 3

[23] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics (ToG)*, 42(4):1–14, 2023. 2, 6

[24] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. Lerf: Language embedded radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19729–19739, 2023. 2, 3, 5, 6

[25] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9404–9413, 2019. 2, 3

[26] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 2, 3, 4, 6, 7

[27] Sosuke Kobayashi, Eiichi Matsumoto, and Vincent Sitzmann. Decomposing nerf for editing via feature field distillation. *Advances in Neural Information Processing Systems*, 35:23311–23330, 2022. 2, 3

[28] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and Rene Ranftl. Language-driven semantic segmentation. In *International Conference on Learning Representations*, 2022. 2, 7

[29] Junnan Li, Pan Zhou, Caiming Xiong, and Steven Hoi. Prototypical contrastive learning of unsupervised representations. In *International Conference on Learning Representations*, 2020. 5

[30] Leyao Liu, Tian Zheng, Yun-Jou Lin, Kai Ni, and Lu Fang. Ins-conv: Incremental sparse convolution for online 3d segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18975–18984, 2022. 2, 3

[31] Qin Liu, Zhenlin Xu, Gedas Bertasius, and Marc Niethammer. Simpleclick: Interactive image segmentation with simple vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22290–22300, 2023. 2, 3, 6

[32] Zhijian Liu, Haotian Tang, Yujun Lin, and Song Han. Point-voxel cnn for efficient 3d deep learning. *Advances in Neural Information Processing Systems*, 32, 2019. 3

[33] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 3

[34] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018. 6

[35] Ben Mildenhall, Pratul P. Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*, 2019. 6

[36] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2, 3, 4, 5, 6

[37] Ashkan Mirzaei, Tristan Aumentado-Armstrong, Konstantinos G Derpanis, Jonathan Kelly, Marcus A Brubaker, Igor Gilitschenski, and Alex Levinshtein. Spin-nerf: Multiview segmentation and perceptual inpainting with neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20669–20679, 2023. 2, 3, 8

[38] Kaichun Mo, Paul Guerrero, Li Yi, Hao Su, Peter Wonka, Niloy Mitra, and Leonidas J Guibas. Structurenet: Hierarchical graph networks for 3d shape generation. *arXiv preprint arXiv:1908.00575*, 2019. 3

[39] Kaichun Mo, Shilin Zhu, Angel X Chang, Li Yi, Subarna Tripathi, Leonidas J Guibas, and Hao Su. Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 909–918, 2019. 3

[40] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)*, 41(4):1–15, 2022. 2, 4, 5, 6

[41] Songyou Peng, Kyle Genova, Chiyu Jiang, Andrea Tagliasacchi, Marc Pollefeys, Thomas Funkhouser, et al. Openscene: 3d scene understanding with open vocabularies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 815–824, 2023. 2

[42] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2, 3, 7

[43] Zhongzheng Ren, Aseem Agarwala, Bryan Russell, Alexander G Schwing, and Oliver Wang. Neural volumetric object selection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6133–6142, 2022. 8

[44] Ruwen Schnabel, Roland Wahl, and Reinhard Klein. Efficient ransac for point-cloud shape detection. In *Computer graphics forum*, pages 214–226. Wiley Online Library, 2007. 2

[45] Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulò, Norman Müller, Matthias Nießner, Angela Dai, and Peter Kontschieder. Panoptic lifting for 3d scene understanding with neural fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9043–9052, 2023. 2, 3

[46] Konstantin Sofiiuk, Ilya A Petrov, and Anton Konushin. Reviving iterative training with mask guidance for interactive segmentation. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 3141–3145. IEEE, 2022. 2, 3, 6

[47] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J. Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, Anton Clarkson, Mingfei Yan, Brian Budge, Yajie Yan, Xiaqing Pan, June Yon, Yuyang Zou, Kimberly Leon, Nigel Carter, Jesus Briales, Tyler Gillingham, Elias Mueggler, Luis Pesqueira, Manolis Savva, Dhruv Batra, Hauke M. Strasdat, Renzo De Nardi, Michael Goesele, Steven Lovegrove, and Richard Newcombe. The Replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 6, 7

[48] Ayça Takmaz, Elisabetta Fedele, Robert W Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann. Openmask3d: Open-vocabulary 3d instance segmentation. *arXiv preprint arXiv:2306.13631*, 2023. 2, 3

[49] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3

[50] Guangyu Wang, Jinzhi Zhang, Kai Zhang, Ruqi Huang, and Lu Fang. Giganticnvs: Gigapixel large-scale neural render-

ing with implicit meta-deformed manifold. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 2

[51] Weiyue Wang and Ulrich Neumann. Depth-aware cnn for rgb-d segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 135–150, 2018. 3

[52] Xueyang Wang, Xiya Zhang, Yinheng Zhu, Yuchen Guo, Xiaoyun Yuan, Liuyu Xiang, Zerun Wang, Guiguang Ding, David Brady, Qionghai Dai, et al. Panda: A gigapixel-level human-centric video dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3268–3278, 2020. 3

[53] Qianyi Wu, Xian Liu, Yuedong Chen, Kejie Li, Chuanxia Zheng, Jianfei Cai, and Jianmin Zheng. Object-compositional neural implicit surfaces. In *European Conference on Computer Vision*, pages 197–213. Springer, 2022. 2, 3

[54] Yajie Xing, Jingbo Wang, and Gang Zeng. Malleable 2.5 d convolution: Learning receptive fields along the depth-axis for rgb-d scene parsing. In *European Conference on Computer Vision*, pages 555–571. Springer, 2020. 3

[55] Bo Yang, Jianan Wang, Ronald Clark, Qingyong Hu, Sen Wang, Andrew Markham, and Niki Trigoni. Learning object bounding boxes for 3d instance segmentation on point clouds. *Advances in neural information processing systems*, 32, 2019. 3

[56] Li Yi, Wang Zhao, He Wang, Minhyuk Sung, and Leonidas J Guibas. Gspn: Generative shape proposal network for 3d instance segmentation in point cloud. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3947–3956, 2019. 3

[57] Haiyang Ying, Baowei Jiang, Jinzhi Zhang, Di Xu, Tao Yu, Qionghai Dai, and Lu Fang. Parf: Primitive-aware radiance fusion for indoor scene novel view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17706–17716, 2023. 2

[58] Fenggen Yu, Zhiqin Chen, Manyi Li, Aditya Sanghi, Hooman Shayani, Ali Mahdavi-Amiri, and Hao Zhang. Capri-net: Learning compact cad shapes with adaptive primitive assembly. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11768–11778, 2022. 3

[59] Jinzhi Zhang, Mengqi Ji, Guangyu Wang, Zhiwei Xue, Shengjin Wang, and Lu Fang. Surrf: Unsupervised multi-view stereopsis by learning surface radiance field. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7912–7927, 2021. 2

[60] Jianing Zhang, Jinzhi Zhang, Shi Mao, Mengqi Ji, Guangyu Wang, Zequn Chen, Tian Zhang, Xiaoyun Yuan, Qionghai Dai, and Lu Fang. Gigamvs: a benchmark for ultra-large-scale gigapixel-level 3d reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7534–7550, 2021. 3

[61] Shu Zhang, Ran Xu, Caiming Xiong, and Chetan Ramaiah. Use all the labels: A hierarchical multi-label contrastive learning framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16660–16669, 2022. 5

[62] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. 3

[63] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6881–6890, 2021. 3

[64] Shuaifeng Zhi, Tristan Laidlow, Stefan Leutenegger, and Andrew J Davison. In-place scene labelling and understanding with implicit scene representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15838–15847, 2021. 2, 3, 6