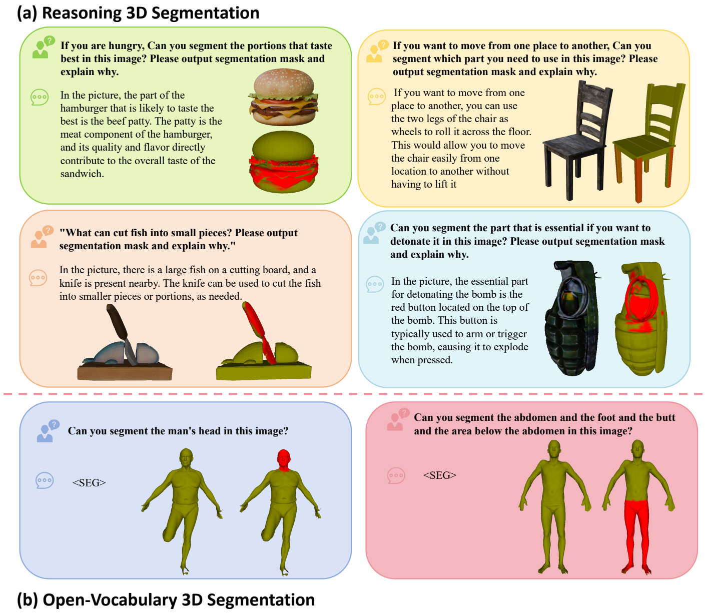
Reasoning3D - Grounding and Reasoning in 3D: Fine-Grained Zero-Shot Open-Vocabulary 3D Reasoning Part Segmentation via Large Vision-Language Models

Tianrun Chen[1,2+] and Chunan Yu[3+], Jing Li[3], Jianqi Zhang[3], Lanyun Zhu[4],
Deyi Ji[6], Yong Zhang[3], Ying Zang[3*], Zejian Li[5], Lingyun Sun[2]

[1] KOKONI, Moxin (Huzhou) Tech. Co., LTD.
[2] College of Computer Science and Technology, Zhejiang University.
[3] School of Information Engineering, Huzhou University.
[4] Information Systems Technology and Design Pillar, Singapore University of Technology and Design.
[5] School of Software Technology, Zhejiang University.
[6] School of Information Science and Technology, University of Science and Technology of China.
+ Equal Contribution * Corresponding Author
`tianrun.chen@zju.edu.cn; 02750@zjhu.edu.cn`
Project Page: http://tianrun-chen.github.io/Reason3D/


(a) Reasoning 3D Segmentation
(b) Open-Vocabulary 3D Segmentation

If you are hungry, Can you segment the portions that taste best in this image? Please output segmentation mask and explain why.

In the picture, the part of the hamburger that is likely to taste the best is the beef patty. The patty is the meat component of the hamburger, and its quality and flavor directly contribute to the overall taste of the sandwich.

If you want to move from one place to another, Can you segment which part you need to use in this image? Please output segmentation mask and explain why.

If you want to move from one place to another, you can use the two legs of the chair as wheels to roll it across the floor. This would allow you to move the chair easily from one location to another without having to lift it

"What can cut fish into small pieces? Please output segmentation mask and explain why."

In the picture, there is a large fish on a cutting board, and a knife is present nearby. The knife can be used to cut the fish into smaller pieces or portions, as needed.

Can you segment the part that is essential if you want to detonate it in this image? Please output segmentation mask and explain why.

In the picture, the essential part for detonating the bomb is the red button located on the top of the bomb. This button is typically used to arm or trigger the bomb, causing it to explode when pressed.

Can you segment the man's head in this image?

<SEG>

Can you segment the abdomen and the foot and the butt and the area below the abdomen in this image?

<SEG>

Fig. 1: In this work, we propose a new task: reasoning 3D segmentation. We also propose a method that can segment 3D object parts with explanations based on various criteria such as reasoning, shape, location, function, and conceptual instructions.

Abstract. In this paper, we introduce a new task: Zero-Shot 3D Reasoning Segmentation for parts searching and localization for objects, which is a new paradigm to 3D segmentation that transcends limitations for previous category-specific 3D semantic segmentation, 3D instance segmentation, and open-vocabulary 3D segmentation. We design a simple baseline method, Reasoning3D, with the capability to understand and execute complex commands for (fine-grained) segmenting specific parts for 3D meshes with contextual awareness and reasoned answers for interactive segmentation. Specifically, Reasoning3D leverages an off-the-shelf pre-trained 2D segmentation network, powered by Large Language Models (LLMs), to interpret user input queries in a zero-shot manner. Previous research have shown that extensive pre-training endows foundation models with prior world knowledge, enabling them to comprehend complex commands, a capability we can harness to "segment anything" in 3D with limited 3D datasets (source efficient). Experimentation reveals that our approach is generalizable and can effectively localize and highlight parts of 3D objects (in 3D mesh) based on implicit textual queries, including these articulated 3d objects and real-world scanned data. Our method can also generate natural language explanations corresponding to these 3D models and the decomposition. Moreover, our training-free approach allows rapid deployment and serves as a viable universal baseline for future research of part-level 3d (semantic) object understanding in various fields including robotics, object manipulation, part assembly, autonomous driving applications, augment reality and virtual reality (AR/VR), and medical applications. The code, the model weight, the deployment guide, and the evaluation protocol are: http://tianrunchen.github.io/Reason3D/.

Keywords: Reasoning Segmentation · 3D Segmentation · 3D Model Parsing · 3D Part Understanding · Large Language Model · Large Vision-Language Model · Computer-Human Interaction

1 Introduction

The importance of 3D segmentation cannot be overstated - it is foundational in fields like robotics, autonomous driving, and augmented reality [24, 63, 67]. Traditional approaches have often required extensive manual labeling or complex rule-based algorithms that struggle to generalize to diverse real-world scenarios [37, 39, 66, 68]. The sheer complexity of 3D data, combined with the inherent ambiguities and varying viewpoints, has posed significant challenges in developing robust and generalizable 3D segmentation techniques.

In this work, we introduce fine-grained **Zero-Shot 3D Reasoning Segmentation** for parts in 3D objects, which aims to bring 3D segmentation to a new level. Imagine instructing a system with words like "segment the part of the chair where you would sit" or "highlight the nutritious parts of this vegetable" and watching it magically understand and perform the task in the 3D world. You can have natural conversations with the system and see it output the segmentation mask along with explanations (See Fig. 1 for examples). It is a future where 3D systems can intuitively understand and respond to intricate queries – The possibilities are endless.

However, achieving this vision is no small feat. Traditional 3D segmentation approaches are typically confined to fixed object categories, severely limiting their flexibility. Recent endeavors in open-vocabulary segmentation can handle a broader range of labels but are still limited to dealing with straightforward tasks like labeling "the apple" and cannot handle complex, reasoning-based queries. Asking a system to perform more nuanced operations like "segment edible parts of a fruit" requires a level of contextual understanding and reasoning that current methods do not possess.

Thanks to the recent advancements in Multi-modal Large Language Models (LLMs) [33, 38, 51, 56, 74, 78], we can now bring our aforementioned vision to life. Recently, Large Vision-Language Models (LVLM) have shown remarkable capabilities in comprehending 2D images, excelling in tasks that require complex reasoning, multi-turn conversations, and explanatory answers [28,65,76]. We aim to extend their capabilities into the 3D realm, and we believe that this transition is promising with much practical value – never forget that we live in a 3D world!

However, extending the success of reasoning segmentation from 2D to 3D domains also presents substantial challenges. The scarcity of available 3D data and ground-truth Question-and-answer pairs stopped us from performing large-scale training. The added dimension also increases the computational demands of 3D architectural components. Here, inspired by research that has tackled similar challenges in 3D generation [16, 17, 52, 54, 60, 72] – using network models in 2D and then lifting some information to 3D, we introduce our approach to leverage off-the-shelf 2D models to perform the task in a zero-manner. This approach, which we named Reasoning3D, allows us to circumvent the limitations imposed by the scarcity of extensive 3D datasets and the high computational costs with its training-free property and 2D pre-training.

Specifically, our Reasoning3D approach involves rendering a 3D model from multiple viewpoints and applying a pre-trained reasoning segmentation network to each 2D view based on the given query input. By doing so, we generate segmentation masks and accompanying text explanations for each perspective. These individual masks and explanations are then fused to produce a comprehensive 3D segmentation mask (labels are assigned to the vertices of the 3D model). We have evaluated our approach in various models in the wild, both with and without textures. We have also tested our approach in existing open-vocabulary segmentation benchmarks, which validates the effectiveness of our approach.

While Reasoning3D is a straightforward baseline method, we believe it serves as a good starting point for researchers to explore and expand the future of 3D part segmentation. We will release the implementation code and the benchmark code publicly to facilitate future research, with the hope that our initial step sets the stage for further innovation and refinement, and eventually bring us closer to a future where 3D computer vision systems are as versatile and perceptive as human cognition, capable of revolutionizing a myriad of applications across various fields.

2 Related Work

2.1 3D Semantic Segmentation.

Segmentation in 2D scenes has achieved significant progress in recent years [11, 58, 70, 77, 79], yet understanding and reasoning in 3D environments is still a crucial research area that needs more attention. In the domain of 3D semantic segmentation, our objective is to predict the semantics of each point in a point cloud. Notable advancements in this field encompass point-based approaches [5, 23], leveraging sophisticated point convolution techniques [55, 61], and voxel-based approaches [12, 18]. Some techniques generate point-level segmentation results using 3D sparse convolutions [11], while others utilize transformer-based approaches [27]. Furthermore, multi-view semantic segmentation methods such as DeepViewAgg [47], Diffuser [26, 42], 3D-CG [20], and 3D-CLR [21] enhance representation learning by creating 2D projections of 3D scenes from different viewpoints. Studies have demonstrated that multi-view representations effectively improve the performance and robustness of various 3D tasks. Nevertheless, these methods usually depend on predefined semantic label sets, whereas our approach is tailored to address and interpret complex reasoning queries. We believe that, following the trend that researchers trying to use various inputs to improve the computer-human interaction in 3D models [7–10, 69, 71], this work can provide another avenue for manipulating 3D contents.

2.2 Large Multimodal Models.

Extensive research on large language models (LLMs) has highlighted their reasoning capabilities, leading to efforts to expand these abilities into the visual domain using large multimodal models (LMMs). LMMs are highly adaptable and versatile, capable of performing tasks that require both language and vision skills. Notable models like BLIP-2 [30], LLaVA [35], and MiniGPT-4 [75] typically utilize a two-phase training approach, which aligns visual representations with LLMs' linguistic embeddings through extensive image-text and video-text datasets [2, 4, 34, 43, 48–50]. Recently, the focus has been on merging multimodal LLMs with vision tasks. VisionLLM [57], for example, offers a versatile interface for various vision-centric tasks via instruction tuning. Nevertheless, this model does not fully leverage the sophisticated reasoning capabilities of LLMs. Kosmos-2 [45] seeks to bolster the foundational abilities of LLMs by creating large datasets of aligned image-text pairs. DetGPT [46] smoothly combines a fixed multimodal LLM framework with an open-vocabulary detector to enable instruction-based detection. LISA, LISA++ [28, 65] generates segmentation masks using embeddings from vision-language models, and LLaFS generates segmentation masks using coordinates exported from LLM. GPT4RoI [73] presents an innovative method by incorporating spatial boxes as inputs and training on region-text pairs.

Unlike previous approaches, our approach aims to integrate the vision-language capabilities of LMMs with the reasoning strengths of LLMs in new 3D perception tasks, taking advantage of these developments in the LMM field.

2.3 Language Instructed 3D Tasks.

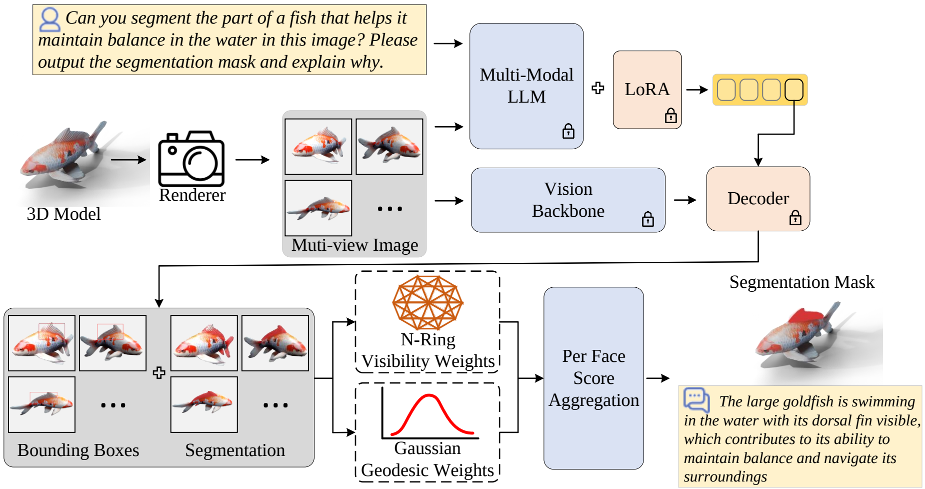
The fusion of point clouds with natural language processing has profound implications, generating significant interest in the field of 3D scene comprehension. This rapidly evolving domain holds promises for advancing human-robot interaction, metaverse development, robotics, and embodied intelligence. Two pivotal abilities crucial to 3D environment dialogue systems include spatial perception and logical reasoning. Recently, there has been a surge in tasks integrating 3D scenes and languages, such as 3D captioning, question answering, situated Q and A, embodied dialogue, planning, navigation, multi-turn dialogue assistance, object detection, and scene description. We categorize 3D perception task models into three groups (refer to Table 1, split by dashed lines). The first group encompasses models handling tasks like 3D captioning, situated question answering, and visual grounding [41,80]. These models can generate single words or phrases as textual outputs. The second category consists of 3D semantic segmentation models producing 3D segmentation masks, such as 3DOVS [36], Openmask3D [53], OpenScene [44], and PLA [15], which perform open-vocabulary semantic segmentation for 3D scenes. However, these methods do not offer conversational responses to user queries or provide reasoning for their tasks. The third category comprises models employing LLMs to conduct visual perception tasks like captioning, scene understanding, and visual grounding, offering conversational outputs [6, 19, 22, 32, 59, 62, 64]. Nonetheless, they lack fine-grained semantic segmentation or reasoning-based 3D vision tasks.

3 Method

As illustrated in Fig. 2, Reasoning3D begins with a mesh input fed into the renderer for viewpoint rendering, generating the face ID for each corresponding viewpoint. Next, the rendered viewpoints and the user-input prompt are processed by the pre-trained 2D reasoning segmentation network, which segments the image to extract the desired parts and output explanations. Finally, using the mapping relationship between each viewpoint and its corresponding mesh face ID, the segmented parts are reconstructed back onto the mesh with a specially designed multi-view fusion mechanism.

3.1 Multi-View Image Rendering and Face ID Generation.

Human interaction with the 3D environment often involves dynamic exploration, amalgamating viewpoints from various angles to construct a cohesive 3D comprehension, rather than assimilating a 3D setting instantaneously. Our methodology advocates for 3D reasoning cultivated from multi-perspective imagery. This strategy also leverages the extensive 2D pretraining accessible in vision-language models, akin to prior methodologies capitalizing on pre-trained vision-language models for 3D visual tasks. The input for this process is mesh $F = \{f_n\}_{n=1}^{N}$, which is composed of N sets of faces f_n. During this rendering process, the 3D



Can you segment the part of a fish that helps it maintain balance in the water in this image? Please output the segmentation mask and explain why.

The large goldfish is swimming in the water with its dorsal fin visible, which contributes to its ability to maintain balance and navigate its surroundings

Fig. 2: The overview of Reasoning3D. First, a 3D model represented by 3D meshes is fed into a renderer to obtain multi-view images. Then, each image goes through a vision backbone and a multi-modal LLM along with user input queries. The decoder decodes the final layer embedding which contains the extra token, thus producing K segmentation masks. We also extract the bounding boxes in this stage. Finally, a specially designed mask-to-3D segmentation algorithm elevates the projections back into the 3D space.

model is converted into multiple 2D images $X_{img} = \{x_i\}_{i=1}^{11}$ from various perspectives. In addition to generating these 2D images, the rendering process also produces corresponding face IDs for each image. These face IDs serve as a crucial link between the 2D images and the original 3D mesh. Specifically, they form a mapping matrix W_{pf} that connects each pixel $P = \{p_i\}_{i=k}^{Mm}$ in the 2D images to a specific face f_n on the 3D mesh, ensuring ensures that the 2D and 3D data remain accurately aligned. The process is formulated as follows:

$$P = W_{pf} \sum_{n=1}^{N} f_n \tag{1}$$

where P represents the pixels in the rendered images, and f_n represents the faces of the 3D mesh from the viewpoint. Denoted by k, the face ID W_{pf} connects the pixels P in the rendered image and the faces of the 3D mesh f_n visible in the current view.

3.2 Reasoning and Segmenting with User Input Prompt

Unlike previous methods (e.g. CLIPSeg [40], LSeg [29], and GLIP [31]) which can handle open-vocabulary explicit prompt, our method aims to handle the implicit prompt such as "Can you segment the appropriate parts of the image containing a 'caged bird'?" Here, we leverage the recent advances of large foundation models to perform this multi-modal reasoning task.

Following Lai et al., [28], we extend the original LLM vocabulary with a new token, <SEG>, which denotes a request for segmentation output. Given the user-input prompt question $X_{question}$ and the input image X_{img}, these are input into the multimodal large language model (LLM)F_{MM}, which subsequently outputs the textual response Y_{answer}. The process is formulated as:

$$Y_{answer} = F_{MM}(X_{img}, X_{question}) \tag{2}$$

Next, the generation of segmentation masks corresponding to the input images involves a series of steps. Firstly, we extract the embedding $\hat{E}_{answer}$ corresponding to the <SEG> token from the output answer textual Y_{answer}. This step enables us to capture information relevant to the segmentation task from the language prompts. Subsequently, we process $\hat{E}_{answer}$ through the MLP γ projection layer to obtain the feature vector E_{answer}. Concurrently, utilizing a visual backbone network F_{vb}, we extract visual embeddings E_{img} from the visual inputs X_{img}. Finally, we feed both the feature vector E_{answer} and the visual embeddings E_{img} into the decoder F_{dec}. The decoder F_{dec} utilizes these features to generate the final segmentation mask M and confidence scores S_M for each mask. This yields segmentation results based on both language prompts and visual information, where each segmentation mask is accompanied by its respective confidence score and corresponding answer textual. The detailed structure of the decoder follows Segment Anything [25]. The process is formulated as follows:

$$E_{answer} = \gamma \hat{E}_{answer} \tag{3}$$

$$E_{img} = F_{vb}(X_{img}) \tag{4}$$

$$M, S_M = F_{dec}(E_{answer}, E_{img}) \tag{5}$$

3.3 Mask Fusion and Refinement in 3D

The obtained 2D segmentation mask is eventually needed to be fused in 3D space to obtain the desired 3D segmentation result. We find that the result from directly merging the multi-view segmentation may not be coherent and high-quality due to the accumulated error and lack of comprehensive multi-view 3D information. Therefore, we designed a multi-stage fusion and refinement mechanism to fully exploit the semantic information and viewpoint information to obtain better 3D segmentation results.

First, we use the top-k method to filter the generated masks to reduce errors in 2D segmentation. Specifically, if the area difference between two masks is greater than a certain threshold T, we select k=1, indicating that is the mask (the most salient part) we want, we generate a bounding box that fits with the mask; otherwise, we select multiple masks and generate multiple bounding boxes. The filtered top-k masks S_M, the corresponding confidence scores S_M, and the face ID is then used as the input to the fusion algorithm. We use the mapping relationship W_{pf} that maps the 2D image mask regions onto the faces of the 3D mesh, resulting in an initial segmented mesh. Note that only the masks within the generated bounding box are involved in the fusion process.

Following [1], we smooth and refine the segmentation boundaries, reducing noise and errors with Gaussian Geodesic Reweighting. Subsequently, we apply the Visibility Smoothing technique to eliminate discontinuities caused by changes in viewpoints, ensuring that the segmented mesh appears natural and coherent from all angles. Finally, we use a Global Filtering Strategy that filters out the masked regions with low confidence scores.

Specifically, for each 2D mask M, we estimate its central face G_i^j, where i denotes the view and j denotes the mask within the view. For the 3D mesh under the current view, we retrieve the vertices of all faces corresponding to the current mask and compute their area-weighted average. This average point is then projected onto point F, and the face F_i^j containing this projection is identified as the central face for the current view and mask. Subsequently, we calculate the geodesic distance vector $d_i^j \in R^N$ from the central face G_i^j to F_i^j for all faces in $f \in F_i^j$. Here, N represents the number of faces in the mesh for the current mask.

$$f(x) = \begin{cases} gdist(G_i^j, f), & \text{if } f \in F_i^j \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

Where $gdist(,)$ represents the geodesic length between two faces computed using a heat method [13] on mesh F. The geodesic distance between mesh faces measures the path length along the surface from one face to another.

Next, we fit a Gaussian distribution on the distances and calculate the corresponding probability density values given the geodesic distances between each face and the uppercase face.

$$r_i^j = \xi[(d; \mu_i^j, (\sigma_i^j)^2), d \in d_i^j] \tag{7}$$

Where μ_i^j and σ_i^j represent the mean and standard deviation of the distances to d_i^j, respectively. Subsequently, we tally the number of times n each face in the mesh is segmented in each view. Finally, we multiply the frequency of each face by the corresponding probability density, and then by the corresponding confidence score S_M, to obtain the final confidence for each mesh face.

However, using only the above method may result in insufficient weighting around the central face G_i^j, especially in regions where the average distances between faces are large. To address this issue, we use computes its local neighborhood, where neighbors are determined by mesh connectivity: if two faces share at least one vertex, then face m is considered a neighbor of face n. To achieve this, we construct a q-rank neighborhood $N_q(n)$ $(q = 5)$ as follows. For face $m \in F$, if there exists a path on the graph connecting m and n with at most q other vertices along the path, then we include face $n \in F$ in the neighborhood.

Finally, we adopt a global filtering using the calculated threshold. We filter out masked regions with low confidence scores. The threshold is the mean confidence score calculated for every face.

4 Experiment

4.1 Experimental Setup

Dataset and Evaluation Metric: Since there are no existing zero-shot reasoning 3D segmentation benchmarks, we first evaluated the zero-shot open-vocabulary segmentation performance on the FAUST [3] benchmark (an open-vocabulary 3D segmentation benchmark) proposed in SATR [1]. We also validated the effectiveness of our method on reasoning 3D segmentation by our collected in-the-wild data from SketchFab. The FAUST dataset consists of manually annotated registered meshes of human body scans, re-meshed independently for each scan to contain approximately 20K triangular faces. We randomly collected samples from the 3D modeling website SketchFab and asked human volunteers to give implicit segmentation commands. For the evaluation metric, we employ the mean Intersection over Union (mIoU) for semantic segmentation as described in [67] for qualitative evaluation for each semantic category across all test shapes in open-vocabulary 3D segmentation. For the reasoning 3D segmentation, the result is visualized and rated by the user.

Implementation Details: We utilized a single NVIDIA A100 GPU for each set of experiments. During the rendering process, we centered the input mesh at the origin and normalized it within a unit sphere. We evenly sample 8 images horizontally around all 360 degrees, maintaining consistency in viewpoints across all experiments. During the rendering process, we used a resolution of 1024×1024 and set a uniform black background color. Multiple reasons (or explanations) will be generated in each view to give a comprehensive understanding for the object, and users can choose one as the desired answer.

Comparison Experiments for Open-Vocabulary Segmentation Since there is no existing reasoning 3D segmentation approach that can be compared, we first compared our method with existing open-vocabulary 3D segmentation models such as SATR [1] and 3DHighlighter [14] following the protocol in [1] but use the same rendering protocol in our method. As illustrated in Table. 1 and Table. 2 We show that though not designed for open-vocabulary segmentation tasks and without fine-tuning or specially designed structure, our method achieves competitive performance in the open-vocabulary segmentation benchmark.

Table 1: Performance on the coarse-grained semantic segmentation on FAUST dataset

Model	Backbone	Arm	Head	Leg	Torso
3DHighlighter	CLIP	28.60	14.20	14.90	8.20
SATR	GLIP	61.54	**76.89**	**87.41**	**52.32**
Ours	**LISA**	**64.65**	72.60	83.58	50.39

Table 2: Performance on the fine-grained semantic segmentation on FAUST dataset

Model	Arm	Belly button	Chin	Ear	Elbow	Eye	Foot	Forehead	Hand	Head	Knee	Leg	Mouth	Neck	Nose	Shoulder	Torso
3DHighlighter	18.39	1.99	0.46	0.72	0.08	0	20.81	0.70	0.02	3.49	6.17	3.91	0.05	1.94	0.07	0.04	7.28
SATR	24.23	**22.00**	**26.53**	**34.55**	**33.67**	**22.55**	**75.20**	**30.35**	**75.11**	**40.31**	**46.96**	56.5	**20.46**	**22.01**	**37.41**	**24.41**	**50.52**
Ours	**26.47**	1.87	3.36	10.61	18.18	2.77	71.85	6.56	43.15	39.81	13.95	**62.23**	4.12	11.88	5.5	9.6	48.78

4.2 Performance in Reasoning 3D Segmentation

A better property that our method has compared to existing open-vocabulary segmentation is that our method can use natural language as the input information. The LLM parses the natural language and gives the segmentation result directly, which enables a more natural and convenient computer-human interaction experience. An example is shown in Fig. 4. The models are from the FAUST dataset.

In the open-vocabulary segmentation, only explicit segmentation command is given, in which Reasoning3D's potential has not been fully exploited. We randomly collect 3D models from the 3D modeling website SketchFab perform the assessment with these in-the-wild 3D models and let human volunteers give "implicit" segmentation commands. Figure. 6 and fig. 1 shows some examples. The examples show that Reasoning3D has the capabilities to offer in-depth reasoning, 3D understanding, part segmentation, and conversational abilities. The model can output the segmentation masks and the explanation as we desire.

To better allow users to interact with our system, we also designed a User Interface (UI) so that users can input arbitrary 3D models and their desired prompt to segment the desired region. (Fig. 6) This UI will also be open-sourced.

5 Discussion and Limitations

This research represents preliminary findings in the task of Reasoning 3D segmentation, and several areas require further exploration and validation. One major aspect is the need for comprehensive benchmarking to rigorously evaluate our method's performance. Additionally, conducting user studies will provide valuable insights into the practical applicability and usability of our approach.

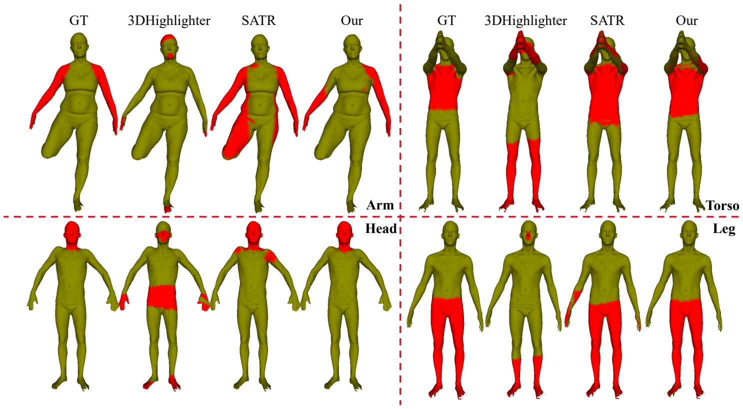
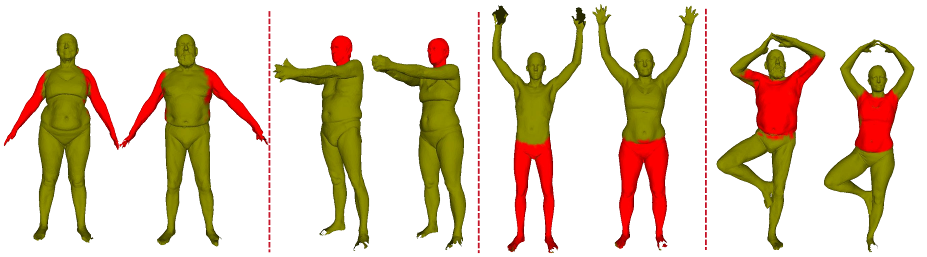


Fig. 3: Qualitative results and comparison between our method and baseline method in FAUST benchmark. The segmented regions are shown in red.



Can you segment the man's arms and hand in this image?

Can you segment the man's head in this image?

Can you segment the abdomen and the foot and the butt and the area below the abdomen in this image?

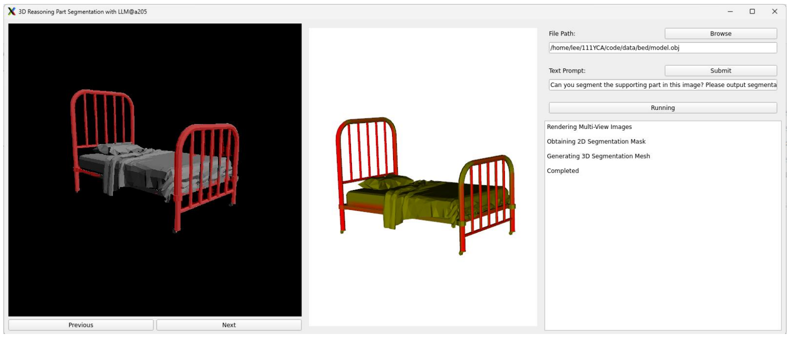
Can you segment the man's torso area and the abdomen area and not the neck and not the arms in this image?

Fig. 4: A natural language command can make the model segment the desired regions. The segmented regions are shown in red.

Our findings indicate that view information plays a critical role in the performance of 3D segmentation tasks. Optimizing view selection to align with the pre-trained vision encoder could significantly enhance outcomes. This suggests that a strategic approach to view selection is essential for leveraging the full potential of the pre-trained models.

The flexibility of our method is noteworthy, as the LVLM can perform zero-shot inference without the need for additional training. While fine-tuning with data could potentially improve performance, we observed that fine-tuning with a very small dataset might negatively impact the network's generalization ability, sometimes resulting in poorer performance compared to fine-tuning. It is also worth noting that our multi-view 2D segmentation and 3D projection method can be applied to scenes, which will be beneficial for more real-world applications.

To foster further advancements and collaborative innovation in 3D reasoning and segmentation, we are releasing our code. We encourage the community to build upon our work and develop improved methods.



File Path: /home/lee/111YCA/code/data/bed/model.obj

Text Prompt: Can you segment the supporting part in this image? Please output segmenta

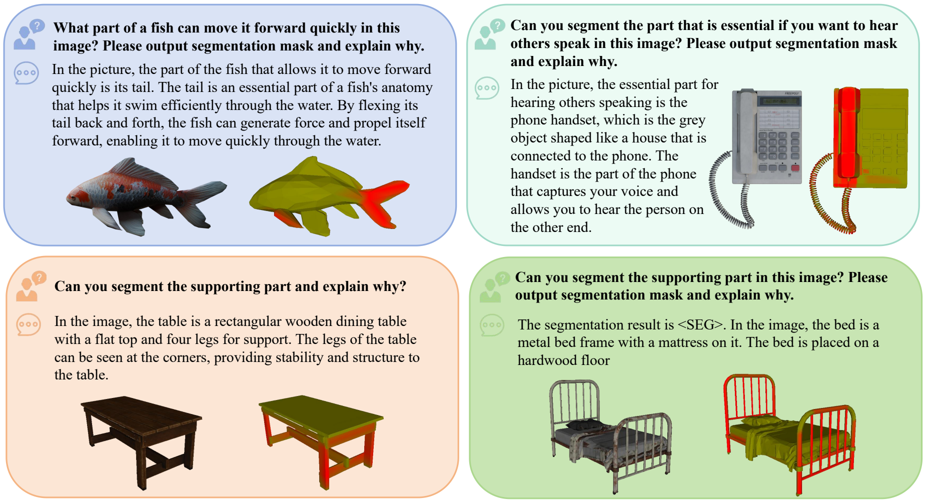
Rendering Multi-View Images

Obtaining 2D Segmentation Mask

Generating 3D Segmentation Mesh

Completed

Fig. 5: We offer a user-friendly interface designed for performance assessment, facilitating the easy upload of 3D models and prompts by users. It enables swift acquisition of 3D segmentation outcomes. This tailored software is available as open-source.



What part of a fish can move it forward quickly in this image? Please output segmentation mask and explain why.

In the picture, the part of the fish that allows it to move forward quickly is its tail. The tail is an essential part of a fish's anatomy that helps it swim efficiently through the water. By flexing its tail back and forth, the fish can generate force and propel itself forward, enabling it to move quickly through the water.

Can you segment the part that is essential if you want to hear others speak in this image? Please output segmentation mask and explain why.

In the picture, the essential part for hearing others speaking is the phone handset, which is the grey object shaped like a house that is connected to the phone. The handset is the part of the phone that captures your voice and allows you to hear the person on the other end.

Can you segment the supporting part and explain why?

In the image, the table is a rectangular wooden dining table with a flat top and four legs for support. The legs of the table can be seen at the corners, providing stability and structure to the table.

Can you segment the supporting part in this image? Please output segmentation mask and explain why.

The segmentation result is <SEG>. In the image, the bed is a metal bed frame with a mattress on it. The bed is placed on a hardwood floor

Fig. 6: This figure shows Reasoning3D's ability to segment 3D object parts (in a fine-grained manner) from in-the-wild samples, including real-world scanned data (samples are randomly collected from SketchFab). These examples highlight Reasoning3D's advanced capabilities in in-depth reasoning, comprehensive 3D understanding, precise part segmentation, and robust conversational abilities. The original mesh and the segmentation result are visualized, and the segmented region is highlighted in Red.

6 Conclusion

This paper introduces a new task: Zero-Shot 3D Reasoning Segmentation for part searching and localization within objects. This new approach moves beyond the constraints of traditional category-specific 3D semantic segmentation, 3D instance segmentation, and open-vocabulary 3D segmentation. We have developed Reasoning3D, a simple yet effective baseline method that can understand and perform complex commands to segment specific parts of 3D meshes with contextual understanding and reasoned outputs for interactive segmentation.

Reasoning3D leverages pre-trained 2D segmentation networks in conjunction with Large Language Models (LLMs) to interpret user queries in a zero-shot manner. Previous studies have shown that extensive pre-training equips foundational models with a broad understanding of the world, enabling them to process complex commands. Our method harnesses this capability, allowing for effective 3D segmentation with limited 3D datasets, making it a resource-efficient solution.

Our experiments demonstrate that Reasoning3D is generalizable and capable of accurately localizing and identifying parts of 3D objects based on implicit textual queries. This includes both articulated 3D objects and real-world scanned data. Additionally, our method can produce natural language explanations for the segmented 3D models and their components. The training-free nature of our approach facilitates rapid deployment and provides a robust baseline for future research in part-level 3D object understanding. This has potential applications across various domains, such as robotics, object manipulation, part assembly, autonomous driving, augmented and virtual reality (AR/VR), and medical fields.

We are releasing the code, model weights, deployment guide, and evaluation protocol to encourage further innovation and collaboration. These resources are available at: http://tianrun-chen.github.io/Reason3D/.

References

1. Abdelreheem, A., Skorokhodov, I., Ovsjanikov, M., Wonka, P.: Satr: Zero-shot semantic segmentation of 3d shapes (2023)
2. Bain, M., Nagrani, A., Varol, G., Zisserman, A.: Frozen in time: A joint video and image encoder for end-to-end retrieval. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV) (Oct 2021). `https://doi.org/10.1109/iccv48922.2021.00175`, `http://dx.doi.org/10.1109/iccv48922.2021.00175`
3. Bogo, F., Romero, J., Loper, M., Black, M.J.: Faust: Dataset and evaluation for 3d mesh registration. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition (Jun 2014). `https://doi.org/10.1109/cvpr.2014.491`, `http://dx.doi.org/10.1109/cvpr.2014.491`
4. Changpinyo, S., Sharma, P., Ding, N., Soricut, R.: Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (Jun 2021). `https://doi.org/10.1109/cvpr46437.2021.00356`, `http://dx.doi.org/10.1109/cvpr46437.2021.00356`

5. Charles, R.Q., Su, H., Kaichun, M., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (Jul 2017). `https://doi.org/10.1109/cvpr.2017.16`, `http://dx.doi.org/10.1109/cvpr.2017.16`
6. Chen, S., Chen, X., Zhang, C., Li, M., Yu, G., Fei, H., Zhu, H., Fan, J., Chen, T.: Ll3da: Visual interactive instruction tuning for omni-3d understanding, reasoning, and planning (2023)
7. Chen, T., Cao, R., Li, Z., Zang, Y., Sun, L.: Deep3dsketch-im: rapid high-fidelity ai 3d model generation by single freehand sketches. Frontiers of Information Technology & Electronic Engineering **25**(1), 149–159 (2024)
8. Chen, T., Ding, C., Zhu, L., Zang, Y., Liao, Y., Li, Z., Sun, L.: Reality3dsketch: Rapid 3d modeling of objects from single freehand sketches. arXiv preprint arXiv:2310.18148 (2023)
9. Chen, T., Fu, C., Zang, Y., Zhu, L., Zhang, J., Mao, P., Sun, L.: Deep3dsketch+: Rapid 3d modeling from single free-hand sketches. In: International Conference on Multimedia Modeling. pp. 16–28. Springer (2023)
10. Chen, T., Fu, C., Zhu, L., Mao, P., Zhang, J., Zang, Y., Sun, L.: Deep3dsketch: 3d modeling from free-hand sketches with view-and structural-aware adversarial training. In: ICASSP. pp. 1–5. IEEE (2023)
11. Chen, T., Zhu, L., Deng, C., Cao, R., Wang, Y., Zhang, S., Li, Z., Sun, L., Zang, Y., Mao, P.: Sam-adapter: Adapting segment anything in underperformed scenes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3367–3375 (2023)
12. Choy, C., Gwak, J., Savarese, S.: 4d spatio-temporal convnets: Minkowski convolutional neural networks. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (Jun 2019). `https://doi.org/10.1109/cvpr.2019.00319`, `http://dx.doi.org/10.1109/cvpr.2019.00319`
13. Crane, K., Weischedel, C., Wardetzky, M.: The heat method for distance computation. Communications of the ACM p. 90âĂŞ99 (Oct 2017). `https://doi.org/10.1145/3131280`, `http://dx.doi.org/10.1145/3131280`
14. Decatur, D., Lang, I., Hanocka, R.: 3d highlighter: Localizing regions on 3d shapes via text descriptions (2022)
15. Ding, R., Yang, J., Xue, C., Zhang, W., Bai, S., Qi, X.: Language-driven open-vocabulary 3d scene understanding (Nov 2022)
16. Fu, X., Zhang, S., Chen, T., Lu, Y., Zhou, X., Geiger, A., Liao, Y.: Panopticnerf-360: Panoramic 3d-to-2d label transfer in urban scenes (2023)
17. Fu, X., Zhang, S., Chen, T., Lu, Y., Zhu, L., Zhou, X., Geiger, A., Liao, Y.: Panoptic nerf: 3d-to-2d label transfer for panoptic urban scene segmentation (2022)
18. Graham, B., Engelcke, M., Maaten, L.v.d.: 3d semantic segmentation with submanifold sparse convolutional networks. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (Jun 2018). `https://doi.org/10.1109/cvpr.2018.00961`, `http://dx.doi.org/10.1109/cvpr.2018.00961`
19. Guo, Z., Zhang, R., Zhu, X., Tang, Y., Ma, X., Han, J., Chen, K., Gao, P., Li, X., Li, H., Heng, P.A.: Point-bind & point-llm: Aligning point cloud with multi-modality for 3d understanding, generation, and instruction following
20. Hong, Y., Du, Y., Lin, C., Tenenbaum, J., Gan, C.: 3d concept grounding on neural fields (Jul 2022)
21. Hong, Y., Lin, C., Du, Y., Chen, Z., Tenenbaum, J., Gan, C., Ucla, U.: 3d concept learning and reasoning from multi-view images
22. Hong, Y., Zhen, H., Chen, P., Zheng, S., Du, Y., Chen, Z., Gan, C.: 3d-llm: Injecting the 3d world into large language models

23. Huang, Q., Wang, W., Neumann, U.: Recurrent slice networks for 3d segmentation of point clouds. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (Jun 2018). `https://doi.org/10.1109/cvpr.2018.00278`, `http://dx.doi.org/10.1109/cvpr.2018.00278`
24. Jones, R.K., Habib, A., Ritchie, D.: Shred: 3d shape region decomposition with learned local operations (2022)
25. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., DollÃąr, P., Girshick, R.: Segment anything (2023)
26. Kundu, A., Yin, X., Fathi, A., Ross, D., Brewington, B., Funkhouser, T., Pantofaru, C.: Virtual multi-view fusion for 3d semantic segmentation. Cornell University - arXiv,Cornell University - arXiv (Jul 2020)
27. Lai, X., Liu, J., Jiang, L., Wang, L., Zhao, H., Liu, S., Qi, X., Jia, J.: Stratified transformer for 3d point cloud segmentation
28. Lai, X., Tian, Z., Chen, Y., Li, Y., Yuan, Y., Liu, S., Jia, J., Kong, H., Research, M.: Lisa: Reasoning segmentation via large language model
29. Li, B., Weinberger, K.Q., Belongie, S., Koltun, V., Ranftl, R.: Language-driven semantic segmentation (2022)
30. Li, J., Li, D., Savarese, S., Hoi, S.: Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models
31. Li, L.H., Zhang, P., Zhang, H., Yang, J., Li, C., Zhong, Y., Wang, L., Yuan, L., Zhang, L., Hwang, J.N., Chang, K.W., Gao, J.: Grounded language-image pre-training (2022)
32. Li, M., Chen, X., Zhang, C., Chen, S., Zhu, H., Yin, F., Yu, G., Chen, T.: M3dbench: Let's instruct large models with multi-modal 3d prompts (2023)
33. Lin, J., Yang, A., Bai, J., Zhou, C., Jiang, L., Jia, X., Wang, A., Zhang, J., Li, Y., Lin, W., Zhou, J., Yang, H.: M6-10t: A sharing-delinking paradigm for efficient multi-trillion parameter pretraining (2021)
34. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., DollÃąr, P., Zitnick, C.L.: Microsoft COCO: Common Objects in Context, p. 740âĂŞ755 (Jan 2014). `https://doi.org/10.1007/978-3-319-10602-1_48`, `http://dx.doi.org/10.1007/978-3-319-10602-1_48`
35. Liu, H., Li, C., Wu, Q., Lee, Y., Madison, M., Research, M.: Visual instruction tuning
36. Liu, K., Zhan, F., Zhang, J., Xu, M., Yu, Y., Saddik, A., Theobalt, C., Xing, E., Lu, S.: Weakly supervised 3d open-vocabulary segmentation (Sep 2023)
37. Liu, M., Zhu, Y., Cai, H., Han, S., Ling, Z., Porikli, F., Su, H.: Partslip: Low-shot part segmentation for 3d point clouds via pretrained image-language models (2023)
38. Liu, Z., He, Y., Wang, W., Wang, W., Wang, Y., Chen, S., Zhang, Q., Lai, Z., Yang, Y., Li, Q., Yu, J., Li, K., Chen, Z., Yang, X., Zhu, X., Wang, Y., Wang, L., Luo, P., Dai, J., Qiao, Y.: Interngpt: Solving vision-centric tasks by interacting with chatgpt beyond language (2023)
39. Lyu, Y., Huang, X., Zhang, Z.: Learning to segment 3d point clouds in 2d image space (2020)
40. LÃijddecke, T., Ecker, A.S.: Image segmentation using text and image prompts (2022)
41. Ma, X., Yong, S., Zheng, Z., Li, Q., Liang, Y., Zhu, S.C., Huang, S.: Sqa3d: Situated question answering in 3d scenes (Oct 2022)
42. Mascaro, R., Teixeira, L., Chli, M.: Diffuser: Multi-view 2d-to-3d label diffusion for semantic scene segmentation. In: 2021 IEEE International Conference on Robotics

and Automation (ICRA) (May 2021). `https://doi.org/10.1109/icra48506.2021.9561801`, `http://dx.doi.org/10.1109/icra48506.2021.9561801`

43. Miech, A., Zhukov, D., Alayrac, J.B., Tapaswi, M., Laptev, I., Sivic, J.: Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV) (Oct 2019). `https://doi.org/10.1109/iccv.2019.00272`, `http://dx.doi.org/10.1109/iccv.2019.00272`
44. Peng, S., Genova, K., Jiang, C., Tagliasacchi, A., Pollefeys, M., Funkhouser, T.: Openscene: 3d scene understanding with open vocabularies (Nov 2022)
45. Peng, Z., Wang, W., Dong, L., Hao, Y., Huang, S., Ma, S.: Kosmos-2: Grounding multimodal large language models to the world
46. Pi, R., Gao, J., Diao, S., Pan, R., Dong, H., Zhang, J., Yao, L., Han, J., Xu, H., Kong, L., Zhang, T.: Detgpt: Detect what you need via reasoning
47. Robert, D., Vallet, B., Landrieu, L.: Learning multi-view aggregation in the wild for large-scale 3d semantic segmentation (Apr 2022)
48. Schuhmann, C., Vencu, R., Beaumont, R., Kaczmarczyk, R., Mullis, C., Aarush, K., Theo, C., Jitsev, J., Komatsuzaki, A.: Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. Cornell University - arXiv,Cornell University - arXiv (Nov 2021)
49. Schuhmann, C., Beaumont, Â., Vencu, V., Gordon, A., Wightman, W., Cherti, M., Coombes, T., Katta, A., Mullis, C., Schramowski, P., Kundurthy, S., Crowson, K., Schmidt, L., Kaczmarczyk, R., Jitsev, Â., Berkeley, U., Data, G.: Laion-5b: An open large-scale dataset for training next generation image-text models
50. Sharma, P., Ding, N., Goodman, S., Soricut, R.: Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (Jan 2018). `https://doi.org/10.18653/v1/p18-1238`, `http://dx.doi.org/10.18653/v1/p18-1238`
51. Shen, Y., Song, K., Tan, X., Li, D., Lu, W., Zhuang, Y.: Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face (2023)
52. Shi, Y., Wang, P., Ye, J., Long, M., Li, K., Yang, X.: Mvdream: Multi-view diffusion for 3d generation (2023)
53. Takmaz, A., Fedele, E., Sumner, R., Pollefeys, M., Tombari, F., Engelmann, F.: Openmask3d: Open-vocabulary 3d instance segmentation (Jun 2023)
54. Tang, J., Wang, T., Zhang, B., Zhang, T., Yi, R., Ma, L., Chen, D.: Make-it-3d: High-fidelity 3d creation from a single image with diffusion prior (2023)
55. Thomas, H., Qi, C.R., Deschaud, J.E., Marcotegui, B., Goulette, F., Guibas, L.: Kpconv: Flexible and deformable convolution for point clouds. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV) (Oct 2019). `https://doi.org/10.1109/iccv.2019.00651`, `http://dx.doi.org/10.1109/iccv.2019.00651`
56. Wang, W., Lv, Q., Yu, W., Hong, W., Qi, J., Wang, Y., Ji, J., Yang, Z., Zhao, L., Song, X., Xu, J., Xu, B., Li, J., Dong, Y., Ding, M., Tang, J.: Cogvlm: Visual expert for pretrained language models (2024)
57. Wang, W., Chen, Z., Chen, X., Wu, J., Zhu, X., Zeng, G., Luo, P., Lu, T., Zhou, J., Qiao, Y., Dai, J.: Visionllm: Large language model is also an open-ended decoder for vision-centric tasks
58. Wang, Y., Cheng, J., Chen, Y., Shao, S., Zhu, L., Wu, Z., Liu, T., Zhu, H.: Fvp: Fourier visual prompting for source-free unsupervised domain adaptation of medical image segmentation. IEEE Transactions on Medical Imaging (2023)
59. Wang, Z., Huang, H., Zhao, Y., Zhang, Z., Zhao, Z.: Chat-3d: Data-efficiently tuning large language model for universal dialogue of 3d scenes (2023)

60. Xiang, J., Yang, J., Huang, B., Tong, X.: 3d-aware image generation using 2d diffusion models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 2383–2393 (October 2023)
61. Xu, M., Ding, R., Zhao, H., Qi, X.: Paconv: Position adaptive convolution with dynamic kernel assembling on point clouds. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (Jun 2021). `https://doi.org/10.1109/cvpr46437.2021.00319`, `http://dx.doi.org/10.1109/cvpr46437.2021.00319`
62. Xu, R., Wang, X., Wang, T., Chen, Y., Pang, J., Lin, D.: Pointllm: Empowering large language models to understand point clouds (2023)
63. Xue, Y., Chen, N., Liu, J., Sun, W.: Zerops: High-quality cross-modal knowledge transfer for zero-shot 3d part segmentation (2023)
64. Yang, J., Chen, X., Qian, S., Madaan, N., Iyengar, M., Fouhey, D., Chai, J.: Llm-grounder: Open-vocabulary 3d visual grounding with large language model as an agent (Sep 2023)
65. Yang, S., Qu, T., Lai, X., Tian, Z., Peng, B., Liu, S., Jia, J.: Lisa++: An improved baseline for reasoning segmentation with large language model (2024)
66. Ying, H., Yin, Y., Zhang, J., Wang, F., Yu, T., Huang, R., Fang, L.: Omniseg3d: Omniversal 3d segmentation via hierarchical contrastive learning (2023)
67. Yu, F., Liu, K., Zhang, Y., Zhu, C., Xu, K.: Partnet: A recursive part decomposition network for fine-grained and hierarchical shape segmentation (2022)
68. Yu, F., Qian, Y., Gil-Ureta, F., Jackson, B., Bennett, E., Zhang, H.: Hal3d: Hierarchical active learning for fine-grained 3d part labeling (2024)
69. Zang, Y., Ding, C., Chen, T., Mao, P., Hu, W.: Deep3dsketch+\+: High-fidelity 3d modeling from single free-hand sketches. arXiv preprint arXiv:2310.18178 (2023)
70. Zang, Y., Fu, C., Cao, R., Zhu, D., Zhang, M., Hu, W., Zhu, L., Chen, T.: Resmatch: Referring expression segmentation in a semi-supervised manner. arXiv preprint arXiv:2402.05589 (2024)
71. Zang, Y., Fu, C., Chen, T., Hu, Y., Liu, Q., Hu, W.: Deep3dsketch+: Obtaining customized 3d model by single free-hand sketch through deep learning. arXiv preprint arXiv:2310.18609 (2023)
72. Zhang, S., Peng, S., Chen, T., Mou, L., Lin, H., Yu, K., Liao, Y., Zhou, X.: Painting 3d nature in 2d: View synthesis of natural scenes from a single semantic mask (2023)
73. Zhang, S., Sun, P., Chen, S., Xiao, M., Shao, W., Zhang, W., Chen, K., Luo, P.: Gpt4roi: Instruction tuning large language model on region-of-interest
74. Zheng, K., He, X., Wang, X.E.: Minigpt-5: Interleaved vision-and-language generation via generative vokens (2024)
75. Zhu, D., Chen, J., Shen, X., Li, X., Elhoseiny, M.: Minigpt-4: Enhancing vision-language understanding with advanced large language models
76. Zhu, L., Chen, T., Ji, D., Ye, J., Liu, J.: Llafs: When large language models meet few-shot segmentation (2024)
77. Zhu, L., Chen, T., Yin, J., See, S., Liu, J.: Continual semantic segmentation with automatic memory sample selection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3082–3092 (2023)
78. Zhu, L., Ji, D., Chen, T., Xu, P., Ye, J., Liu, J.: Ibd: Alleviating hallucinations in large vision-language models via image-biased decoding. arXiv preprint arXiv:2402.18476 (2024)
79. Zhu, L., Ji, D., Zhu, S., Gan, W., Wu, W., Yan, J.: Learning statistical texture for semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12537–12546 (2021)

80. Zhu, Z., Ma, X., Chen, Y., Deng, Z., Huang, S., Li, Q.: 3d-vista: Pre-trained transformer for 3d vision and text alignment (Aug 2023)