MeshSegmenter: Zero-Shot Mesh Semantic Segmentation via Texture Synthesis

Ziming Zhong[1], Yanyu Xu[2], Jing Li[3], Jiale Xu[1], Zhengxin Li[1], Chaohui Yu[4], and Shenghua Gao[1,5]

[1] ShanghaiTech University [2] Shandong University [3] Xiaohongshu Inc [4] Alibaba Group
[5] The University of Hong Kong
{zhongzm, lijing1, xujl1, lizhx, gaoshh}@shanghaitech.edu.cn
xu_yanyu@sdu.edu.cn huakun.ych@alibaba-inc.com

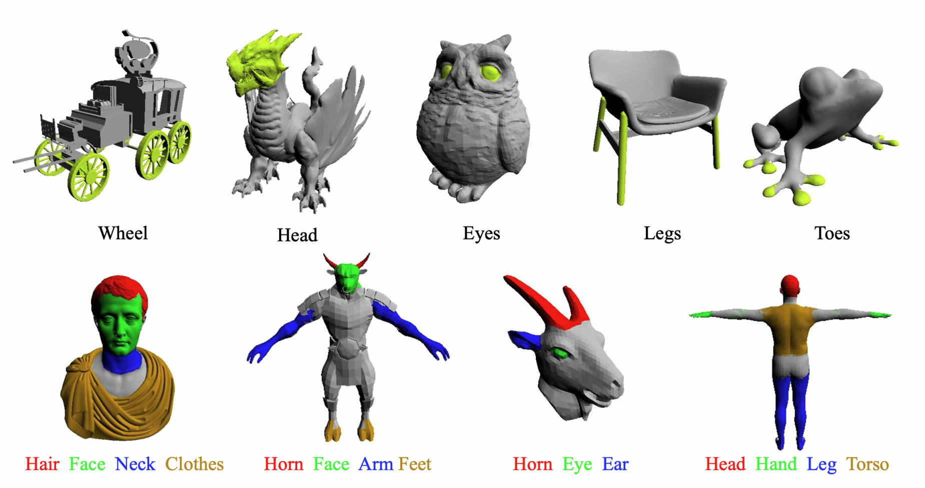

Fig. 1: Meshsegmenter performs zero-shot mesh semantic segmentation through texture synthesis. It can accurately segment the text-specified region by aggregating multi-view 2D segmentation results from single and multiple queries.

Abstract. We present MeshSegmenter, a simple yet effective framework designed for zero-shot 3D semantic segmentation. This model successfully extends the powerful capabilities of 2D segmentation models to 3D meshes, delivering accurate 3D segmentation across diverse meshes and segment descriptions. Specifically, our model leverages the Segment Anything Model (SAM) model to segment the target regions from images rendered from the 3D shape. In light of the importance of the texture for segmentation, we also leverage the pretrained stable diffusion model to generate images with textures from 3D shape, and leverage SAM to segment the target regions from images with textures. Textures supplement the shape for segmentation and facilitate accurate 3D segmentation even in geometrically non-prominent areas, such as segmenting a car door within a car mesh. To achieve the 3D segments, we render 2D images from different views and conduct segmentation for both textured and untextured images. Lastly, we develop a multi-view revoting scheme that integrates 2D segmentation results and confidence scores from various views onto the 3D mesh, ensuring the 3D consistency of segmentation results and eliminating inaccuracies from specific perspectives. Through these innovations, MeshSegmenter offers stable and reliable 3D segmentation results both quantitatively and qualitatively, highlighting its potential as a transformative tool in the field of 3D zero-shot segmentation. The code is available at `https://github.com/zimingzhong/MeshSegmenter`.

Keywords: Zero-Shot Learning · 3D Semantic Segmentation · Texture Synthesis

1 Introduction

Segmenting semantic regions within a 3D mesh is crucial in the fields of computer graphics and computer vision, as highlighted in works such as [8,9,29]. However, the availability of accurately annotated 3D data is limited, and obtaining such data is a costly work. As a result, existing models trained on these data often struggle to generalize well to unseen examples. To mitigate this challenge, using open vocabularies as inputs for segmenting regions has emerged as an efficient approach, showcasing the practical value of 3D zero-shot mesh segmentation [9]. This task requires a model that has a comprehensive understanding of both the overall object and its local components. However, understanding local regions in 3D meshes can introduce ambiguities, leading to erroneous segmentations. For example, black clothing can be mistaken for hair when viewed from certain perspectives. Consequently, 3D zero-shot mesh segmentation represents a highly challenging task. In this paper, we propose a network framework specifically designed for zero-shot 3D semantic segmentation. Our approach leverages object descriptions and segmentation region descriptions to guide the segmentation process. By incorporating both descriptions, our model can automatically segment fine-grained semantic regions. To demonstrate the effectiveness of our method, we showcase its application to mesh segmentation and editing tasks.

Recently, large-scale 2D segmentation models [4,18,24] have achieved remarkable success in various 2D segmentation tasks, showcasing exceptional zero-shot generalization capabilities. These achievements have been facilitated by extensive datasets of 2D images with corresponding annotations. However, acquiring equivalent annotated datasets for 3D data is a considerable challenge. Furthermore, the parameter size and computational costs associated with 2D segmentation models make them impractical for handling larger 3D datasets that demand higher quality segmentation. To address these limitations, we propose leveraging the powerful capabilities of 2D zero-shot models for 3D zero-shot segmentation, presenting a more efficient approach. Our proposed model initially performs zero-shot 2D segmentation on rendered mesh images. Subsequently, the segmentation results from different viewpoints are combined to generate a coherent 3D segmentation outcome. This approach ensures consistency in segmentation results across multiple viewpoints while integrating the results from neighboring viewpoints to enhance the robustness of the segmentation outcome.

Although current approaches to 3D zero-shot segmentation [1,2,9] primarily rely on untextured meshes, the absence of texture information exacerbates the ambiguity of meshes across different viewpoints. However, recent advancements in generative models have enabled significant progress in generating multi-view consistent mesh textures by merging results from multiple viewpoints. By leveraging the geometric information obtained from these generative models, we can incorporate realistic texture information that helps reduce ambiguity in segmentation regions. Additionally, existing 2D segmentation models are trained on real images, creating a substantial domain gap when detecting untextured meshes, as shown in Fig. 3. To address this issue, our model first generates high-quality textured meshes based on the untextured meshes and object mesh descriptions. This approach allows us to harness the potential of 2D segmentation models while benefiting from the purer geometric information provided by the untextured mesh model. Consequently, we perform 3D zero-shot segmentation on both untextured meshes and textured meshes, capitalizing on their respective strengths.

By adopting this novel approach, our proposed model addresses these challenges associated with untextured meshes. Through experimental evaluation, we demonstrate the efficacy and potential of our method for various 3D zero-shot segmentation tasks. Our work contributes to advancing 3D zero-shot segmentation and paves the way for further research in this domain.

Our contributions can be summarized as follows:

- We have introduced MeshSegmenter, a simple yet effective framework for 3D zero-shot semantic segmentation that efficiently elevates the capabilities of robust 2D segmentation models to 3D meshes. MeshSegmenter consistently presents precise 3D segmentation results across various meshes and a rich array of segment descriptions.
- We have proposed the generation of textures based on object descriptions to enhance 2D segmentation models by providing additional texture information, thereby assisting these models in achieving accurate results. Leveraging latent texture information excavated from the generative models based on 3D mesh, our model can accurately perform 3D segmentation in geometrically non-prominent areas, such as segmenting the door in a car mesh.
- We have developed a multi-view revoting module that integrates 2D detection results and confidence scores from various views onto the 3D mesh, effectively ensuring the 3D consistency of segmentation results and eliminating incorrect detection results from certain perspectives. This feature enables our model to provide stable and reliable 3D segmentation results.

2 Related Work

Zero-shot 2D Segmentation. Zero-shot 2D semantic segmentation is an important research area [5,27,36,48] and a challenging problem [41,43,51], which

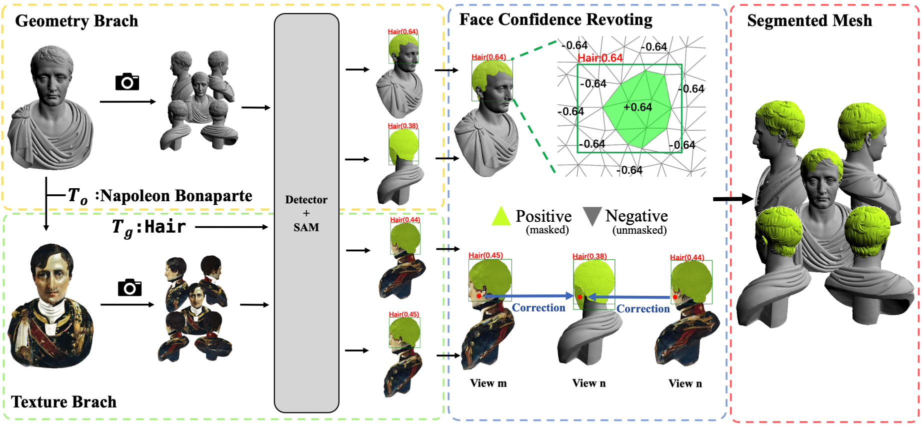

Fig. 2: Overview of the proposed pipeline. The Stable Diffusion (SD) model can generate high-quality textures under the guidance of textual prompts. Textured and untextured meshes are rendered from a fixed perspective. The rendered images are processed by GroundingDino [24] and SAM [18], which detect bounding boxes with corresponding confidence scores and segment the specific regions guided by these boxes, respectively. Ultimately, we employ face confidence revoting (FCR) to aggregate the detection and segmentation results of the textured and untextured meshes from multiple viewpoints to revote the 3D-awara scores to triangles.

could be divided into two groups. The first group [3, 6, 16, 33, 49] enhances the generalization capability to unseen categories by aligning the semantic features of images with the image recognition model. For instance, Bucher *et al.* [6] first introduced the task of zero-shot semantic segmentation, which aims to learn pixel-wise classifiers for never-seen object categories with zero training examples. In [3], Baek *et al.* propose to leverage visual and semantic encoders to learn a joint embedding space, where the semantic encoder transforms semantic features to semantic prototypes. The second group [10, 14, 28, 44, 45] focuses on adapting rich semantic spaces of large-scale multi-modal neural networks, such as [17, 32, 35] for segmentation tasks. These approaches enhance the generalizability to unseen categories through the combination of text, captioning, and self-supervision. Recent advancements [42, 47, 51] in zero-shot 2D semantic segmentation based on powerful large-scale multimodal models and natural language representations have shown significant improvement. These approaches have demonstrated remarkable performance in generalizing to unseen categories.

Zero-shot 3D Segmentation. In zero-shot 3D segmentation [11, 20, 25, 30], a notable focus has been placed on point cloud segmentation, as evidenced by pioneering works such as [7, 12, 23, 29] . The core idea is to enable models to semantically segment 3D shapes or point clouds without having seen examples of these specific categories during training. This research area leverages advancements in zero-shot learning from 2D image processing and applies them innovatively to 3D data. Recent advancements [15, 19, 30, 39]in zero-shot

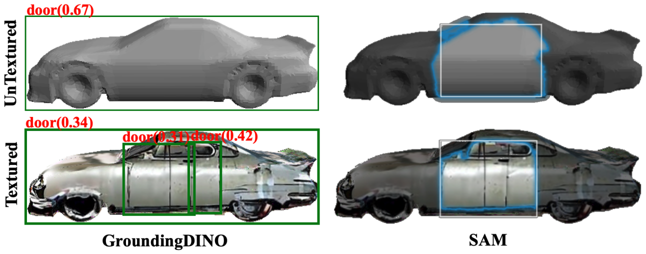

Fig. 3: Performance of GroundingDINO [24] and SAM [18] on M_{Texture} and $M_{\text{unTextured}}$. The large-scale pre-trained models are typically trained on images rich in texture. However, there is a domain gap when these models are applied to untextured meshes. GroundingDINO [24] and SAM [18] are respectively provided with textual prompts and bounding boxes for detection and segmentation tasks, demonstrating significantly superior performance on textured images compared to performance on untextured images.

3D segmentation have been largely influenced by the integration of 2D image recognition models into 3D shape analysis. Recent advancements from Neural Radiance Fields (NeRFs) [26, 31] focus on constructing semantic fields. These approachs [13, 21, 40, 50] enforce consistency between the rendered multi-view semantic labels and the ground truth or estimated semantic labels.

We categorize zero-shot segmentation based on its ability to segment a single or multiple queries into single query segmentation and multiple queries segmentation.

Single Query Segmentation. In recent work on 3D zero-shot segmentation [9, 26, 31], the method known as 3DHighlighter [9] employs a neural network to predict the probabilities of mesh vertices belonging to the text-specified region. By coloring the vertices based on these probabilities and rendering the mesh from random viewpoints, the approach constrains the distance between the rendered images and the given text prompt in CLIP [35] space to obtain the segmentation results.

Multiple Queries Segmentation. The recent advancements in visual-text pretrained models have enabled the possibility of zero-shot segmentation of 3D objects into multiple parts. SATR [2] employs the GLIP [22] approach to obtain semantic segmentation results of images rendered from various perspectives, subsequently aggregating these multi-view segmentations. For each facet, the prompt with the highest confidence level is selected as the semantic segmentation result. Benefiting from the correction of low-confidence erroneous semantic segmentation by high-confidence accurate ones, SATR [2] achieves commendable performance. ZSC [1] uses ChatGPT to get all segmented prompts about the object and focuses on 3D shape correspondence in different objects. However, their reliance on the error-correcting capability of multi-queries semantic segmentation falters under the conditions of single-query segmentation.

3 Method

3.1 Overview

Mesh segmentation aims to segment the semantic regions according to the corresponding text descriptions. The input is a untextured mesh $M_{\text{unTextured}}$, characterized by vertices $V \in \mathbb{R}^{n\times 3}$ and faces $F \in \{1, \cdots, n\}^{m\times 3}$, where m is the number of faces. Users provide an object class text T_{o} and a grounding text T_{g}. The objective of 3D zero-shot segmentation is to accurately segment the regions specified by the text within the given mesh.

Current 3D semantic segmentation models based on untextured meshes can only utilize shape information for segmentation, which is easily influenced by the lighting conditions and the viewpoints. We propose leveraging generated textures to incorporate texture information for mesh segmentation, rather than relying solely on shape information to address this issue. As shown in Fig. 2, we propose a novel 3D mesh semantic segmentation framework, consisting of three parts, including text-guided texture synthesis, 2D zero-shot semantic segmentation, and face confidence revoting strategy.

In text-guided texture synthesis, we generate textures for untextured mesh $M_{\text{unTextured}}$ based on object class text T_{o}. The texture information comes from the pre-trained generative model Stable Diffusion [38] which has great prior texture information. We render the textured meshes M_{Textured} and the untextured meshes $M_{\text{unTextured}}$ to 2D images in different viewpoints and segment the 2D images according to the object class T_{o} and the grounding description T_{g} by the 2D zero-shot segmentation model GroundingDINO [24] and SAM [18]. Finally, we fuse the segmentation results from different views and obtained the segmented faces. Considering the inconsistency among the predictions from different views, especially from the extreme hard-case views, we further propose to fuse the 2D segments by the face confidence voting module and achieve consistency among different views.

3.2 Text-Guided Texture Synthesis

Our method segments the textured mesh and text-guided texture synthesis first generates textures given the raw untextured meshes guided by the text descriptions. As shown in Fig. 2, the untextured meshes only provide the structure information. It is hard to localize the semantic regions without colors, for example, the car doors that are integrated with the car body structure. Therefore, we leverage the texture information generated from the pre-trained generative model such as Stable Diffusion [38] to help the mesh segmentation. The generative model is trained on large-scale of data therefore is good at generating textures.

It should be noted that the descriptions used in this module are different from the grounding descriptions. For example, our model employs “a photo of $\{T_o\}$ and $\{\}$ view” as the prompt for the diffusion model, while the grounding descriptions are like “hair”, “door” or “tail”. TEXTure [37] introduces different

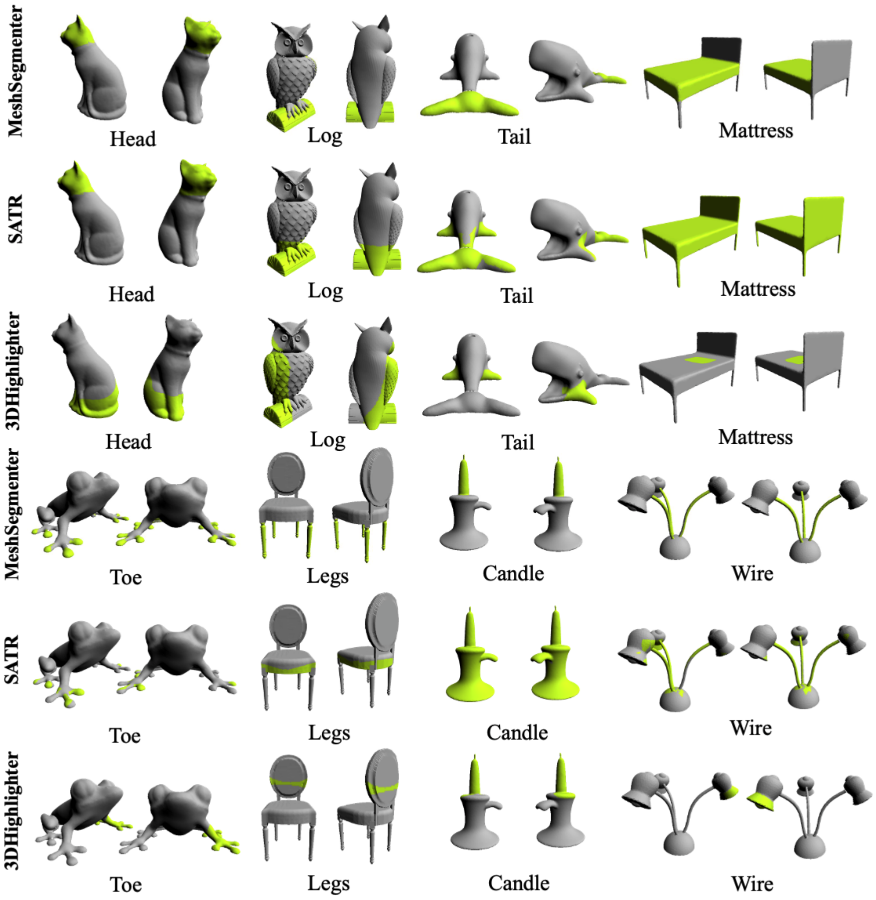

Fig. 4: Qualitative results in the single query: MeshSegmenter performs efficient zero-shot mesh semantic segmentation across diverse meshes.

view prompts to facilitate better generation conditioned on viewpoints, which include “front”, “left”, “back”, “right”. Further details on the prompt settings for other orientations are available in the supplementary material.

3.3 2D zero-shot semantic segmentation

Our work utilizes both $M_{\text{unTextured}}$ and M_{Textured} meshes to provide geometric and texture information, respectively. The processing workflow is consistent for both types of meshes. For simplicity, we will explicitly enumerate only the segmentation steps for $M_{\text{unTextured}}$.

Get the rendered K images, x_k is the rendered image, and VisFace_k is visible face set in the k-th viewpoint. Randomly generating viewpoints during rendering can have a detrimental effect on the efficiency of object semantic segmentation.

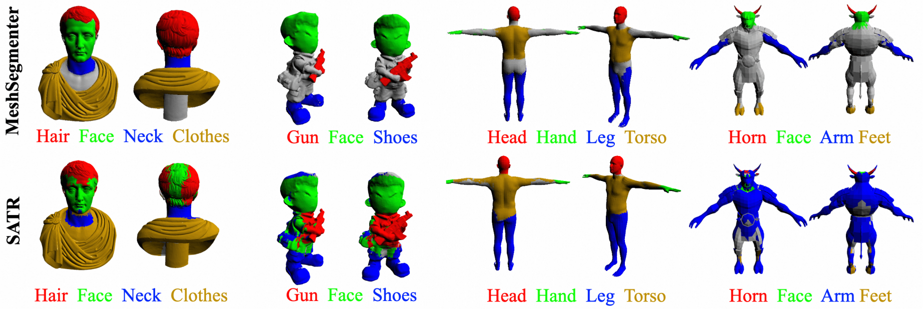

Fig. 5: Qualitative results in the multiple queries: MeshSegmenter performs accurate zero-shot mesh semantic segmentation, which does not rely on competition between multi queries segmentations. For instance, in the first column of MeshSegmenter, "chest" is not mistakenly segmented as "clothes". In the third column of MeshSegmenter, the "buttocks" are not mistakenly segmented as "leg" or "torso".

This can result in the introduction of extreme perspectives, which can cause inaccuracies in semantic segmentation. To address this concern, we define a camera rendering trajectory that aims to strike a balance between the semantic segmentation results from rendered images and the coverage of all object viewpoints. In particular, we first render $M_{\text{unTextured}}$ from K viewpoints $V \in \{v_1, \ldots, v_l\}$. We set radius r as 2, and for the polar angle θ being 75° and 115° respectively, we have the azimuthal angle ϕ uniformly sampled $\frac{K}{2}$ times in a range of 0° to 360°. The rendering process is formulated as:

$$\text{Render}(M_{\text{unTextured}}, v_k) = (x_k, \text{VisFace}_k). \tag{1}$$

GroundingDINO [24] is a novel approach to open-set object detection that combines the strengths of DINO architecture and grounded pre-training model. It takes an image x_k and a 2D region-level description T_{g} as input and accurately segments the region specified by the text into bounding boxes b_k and corresponding 2D confidence scores c_k. The text-guided detection process is formulated as:

$$D(x_k, T_g) = (b_k, c_k). \tag{2}$$

Segment Anything Model (SAM) [18] is a foundational approach to image segmentation that enables promotable segmentation. This means the model can be trained on a pre-training task and then used to respond appropriately to any prompt at inference time, making it highly adaptable to a range of downstream tasks. An inference approach is to use the bounding box as input and SAM [18] outputs the mask Mask_k for the region of interest within the bounding box b_k. The process is formulated as:

$$\text{SAM}(b_k) = (\text{Mask}_k). \tag{3}$$

Our model is specifically developed for performing zero-shot semantic segmentation of 3D objects, where the task involves delineating a region specified

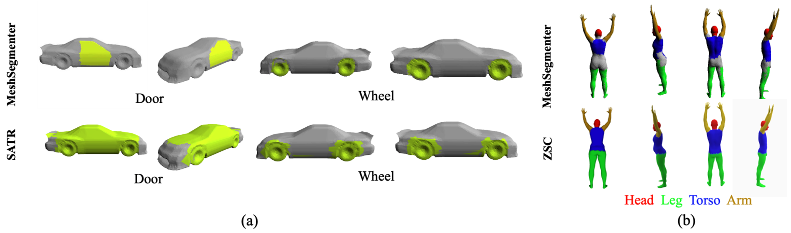

Fig. 6: (a): Semantic Segmentation on Regions with Weakly Geometric Signal.(b) Qualitative results with ZSC.

by text from an object. These regions are required to be partial regions of the object. Therefore, when detection produces bounding boxes that cover the entire object, we classify these as evident segmentation errors. These detection results are promptly deleted.

3.4 Face Confidence Revoting (FCR)

Based on the results of a powerful 2D detection and segmentation model, for each rendered image, we can obtain a set of bounding boxes b_k, 2D confidence scores c_k, and accurate masks Mask_k for the regions described in text within that viewpoint. Additionally, using Nvidia Kaolin, we can obtain a face index map for that particular viewpoint. This allows us to identify visible faces VisFace_k and determine whether they are within the mask region.

A simple approach would be to aggregate the faces of the mask region across different viewpoints to obtain the final segmentation result. However, this approach would introduce errors from incorrectly detected and segmented faces in a particular viewpoint, thus contaminating the overall segmentation result. Therefore, we propose a technique called Face Confidence Revoting.

In Face Confidence Revoting, we selectively consider the faces within the mask region with their corresponding confidence scores, while penalizing visible faces outside the mask region. This approach allows the correct segmentation results of neighboring faces to compensate for the errors in detection and segmentation within the current viewpoint. We denote the global confidence scores from $M_{\text{unTextured}}$ or M_{Textured} as g_m For the visable faces in k-th view v_k, we set the local confidence l_k of the masked faces to c_k and the unmasked faces to $-c_k$. The global confidence of the m-th face is the sum of multi-view local confidence:

$$g_m = \sum_{i=1}^{k} l_i \tag{4}$$

MeshSegmenter leverages the $M_{\text{unTextured}}$ and M_{Textured} generated from the generative models to obtain the overall semantic segmentations. Consequently, our strategy involves setting the overall confidence scores o_m as the average of the g_m values of the untextured and textured meshes. When the value of f_m exceeds $o_{\text{threshold}}$. It is considered to belong to the region determined as a text.

$$\begin{cases} f_m \in R_{\text{text}}, & \text{if } o_m > o_{\text{threshold}} \\ f_m \notin R_{\text{text}}, & \text{if } o_m \leq o_{\text{threshold}} \end{cases} \tag{5}$$

R_{text} is a collection of faces for a text-specified region. We set the threshold as 0 for all cases to ensure a fair comparison. Finally, we averages the confidence of each face with their neighbor faces to obtain a smooth segmentation.

4 Experiment

4.1 Implementation Details

We leveraged a single Nvidia A40 GPU for each experiment, involving a single object, its corresponding description, and a grounding text. We use Nvidia Kaolin to render multi-view 2D images from M_{Textured} and $M_{\text{unTextured}}$. We compare MeshSegmenter with 3DHighlighter [9], SATR [2], and ZSC [1]. Since ZSC [1] is not open-source, we use its official results for comparison. We utilize existing 2D detection model GroundingDINO [24] and segmentation model SAM [18], selecting the segmentation results corresponding to the bounding box with the highest confidence score from different perspectives as our baseline.

4.2 Zero-shot Mesh Semantic Segmentation

Qualitative Comparisons. MeshSegmenter demonstrates superior performance in zero-shot mesh semantic segmentation compared with 3DHighlighter [9] and SATR [2], as shown in Fig. 4. While 3DHighlighter [9] is limited to segmenting based on a single text query, MeshSegmenter can accurately segment a mesh in single and multiple queries. When dealing with multiple queries, MeshSegmenter does not rely on the competition of semantic segmentation for neighboring text queries and can independently and accurately segment each semantic region. We show the comparisons with SATR [2] in Fig. 5 and the comparisons with ZSC [1] in Fig. 6 (b). Texture systhesis can effectively enhance the performance of zero-shot semantic segmentation, particularly for regions with minimal geometric signals, as shown in Fig. 6 (a).

Quantitative Comparisons. We evaluate the performance of 3D zero-shot segmentation using our proposed MeshSegmenter on the ShapeNetPart [46] dataset. The ShapeNetPart [46] dataset contains 31,963 objects and 50 annotated parts. Due to the computational resource-intensive nature of our model, which relies on multi-view texture generation, we randomly selected 20 objects from each category for evaluation. We compared our results with 3DHighlighter [9] and SATR [2], and our approach clearly outperforms them, as shown in Tab. 1.

User Study. We conducted a user study on our dataset of 20 3D meshes, with 30 respondents rating segmentation results on a scale of 1 to 5, with higher values indicating better performance. MeshSegmenter outperforms both 3DHighlighter [9] and SATR [2] in single query and multiple queries tasks, as shown in Tab. 2.

Table 1: Quantitative comparisons on PartNet subset

Method	mIou (%)	Bag	Knife	Earphone	Chair
3DHighlighter [9]	5.8	2.4	3.2	7.8	9.7
SATR [2]	43.4	50.2	53.7	32.5	37.2
MeshSegment-Baseline(Ours)	42.3	47.1	38.2	32.2	51.7
MeshSegmenter(Ours)	**69.0**	**78.3**	**68.2**	**57.2**	**72.2**

Table 2: User Study. MeshSegmenter demonstrates superior performance in both single query and multiple queries tasks.

Method	Single	Multiply
3DHighlighter [9]	5.8	2.4
SATR [2]	43.4	50.2
MeshSegmenter(Ours)	**69.0**	**78.3**

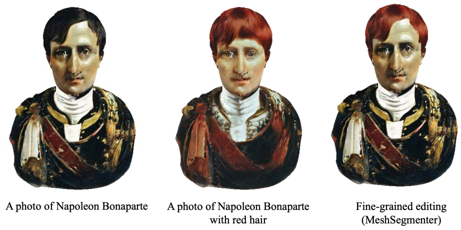
A photo of Napoleon Bonaparte
A photo of Napoleon Bonaparte with red hair
Fine-grained editing (MeshSegmenter)

Fig. 7: Controlled Mesh Stylization. Meshsegmenter can accurately segment regions specified by text, significantly enhancing the capability for fine-grained editing. For example, it precisely segments editable region, allowing for detailed modifications such as changing black hair to red.

4.3 Application of Meshsegmenter

Controlled Mesh Stylization Previous mesh stylization work focuses solely on the stylization of the entire object. Local descriptive prompts could easily corrupt the global stylization appearance. It is challenging to accurately stylize the text-specified region while preserving other areas from being corrupted. MeshSegmenter can effectively locate text-specified local regions to achieve controllable local editing, as shown in Fig. 7. We begin with a human mesh and stylize it using the prompt “a photo of Napoleon Bonaparte”. For a fair comparison, we use the same seed to stylize the same mesh with the prompt “a photo of Napoleon Bonaparte with red hair”. TEXTure [37] noticeably corrupts the “ribbon” region with red information, which is not specified by the grounding text “hair”. In contrast, MeshSegmenter effectively segments the “hair” region, enabling precise mesh editing.

Mesh Editing Currently, mesh editing mainly focuses on object-level editing, such as adding, deleting, and scaling. Precisely segmenting the text-specified region can bring about fine and powerful object script editing capabilities. Our proposed model has demonstrated the robust semantic segmentation capability

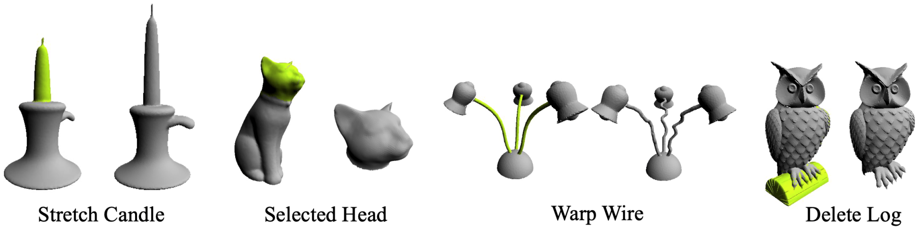

Fig. 8: Mesh Editing: Utilizing the accurate zero-shot mesh semantic segmentation results provided by MeshSegmenter, we can perform precise mesh editing tasks such as stretching, selecting, warping, and deleting text-specified regions.

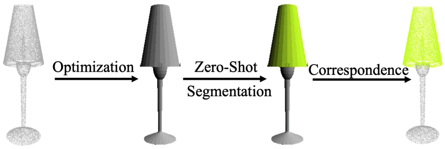

Fig. 9: Point Cloud Semantic Segmentation: We optimize a mesh result to obtain the occlusion relationship of 3D data using [34] and finally correspond the mesh segmentation result to the point cloud. Our model can also segment point cloud data effectively.

of 3D objects through a combination of object description and grounding text. This approach can be readily integrated into 3D rendering applications. We can input diverse 3D object meshes and leverage object grounding to select the segmentation region based on the text input automatically. Coupled with the mesh editing functionalities of 3D rendering applications, we can precisely manipulate the region corresponding to the grounding text, thus providing direct control over the editing process without requiring challenging manual selection, particularly for non-uniform regions, as shown in Fig. 8.

More 3D Representations MeshSegmenter not only performs precise and effective semantic segmentation on mesh structures, but this capability can also be extended to other 3D representations. MeshSegmenter relies on the detection and segmentation results of images rendered from various viewpoints of a 3D object, as well as the mapping relationship between 2D segmentation regions and the 3D object. Point clouds can accurately represent a variety of 3D objects but fail to convey their topological structure, making it challenging to establish the mapping relationship between rendered images and point clouds, and hence difficult to determine the visibility of point clouds from specific viewpoints. Therefore, we optimize point clouds into mesh structures to capture the object's topological structure, and then align the zero-shot 3D semantic segmentation results on the mesh to the point cloud structure, as shown in Fig. 9. MeshSegmenter can be extended to more 3D representations when the mapping relationship between 2D rendered images and 3D representations is obtained.

Table 3: Ablation studies of different modules on PartNet subset

Shape-branch	Texture-branch	Face Revoting	mIoU (%)
✓			42.3
✓	✓		49.7
✓		✓	52.5
	✓	✓	62.2
✓	✓	✓	**69.0**

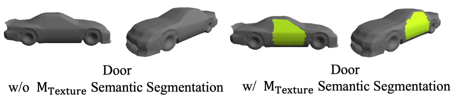

Fig. 10: Ablation Study of Segmentation via Texture Synthesis.

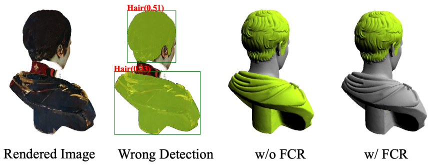

Fig. 11: Ablation Studies of Face Confidence Revoting.

4.4 Ablation Studies

Effectiveness of the Segmentation via Texture Synthesis Both the 2D zero-shot detection model and segmentation model are trained on textured images, and the application of texture synthesis for segmentation effectively reduces the domain gap between inference images and training images. Additionally, regions with insignificant geometric information are difficult to segment without textured meshes. Extracting texture information from the generative model allows for effectively segmenting the text-specified region, as shown in Fig. 10.

Effectiveness of the Face Confidence Revoting. We ablate the effectiveness of our proposed Face Confidence Revoting, as shown in Fig. 11. The most direct approach to aggregating multi-view detection and segmentation results is to assign each detected face to the text-specified region with the highest confidence. However, this approach will introduce incorrect local detection and segmentation results into the global results. To address this issue, MeshSegmenter assigns negative local confidence scores to faces that are not within the segmented regions and aggregates local confidence scores into a global confidence score, using neighboring viewpoints to correct erroneous segmentation results.

Robustness of Object Description The 3D zero-shot segmentation task should maintain robustness for diverse object descriptions. MeshSegmenter effectively achieves 3D zero-shot segmentation by leveraging geometric and color information of object meshes through text. Accurate description of an object's class can lead to the most appropriate texture generation, however, this poses a challenge for users, particularly when confronted with complex objects. Nonetheless, MeshSegmenter is still able to obtain effective texture information through inaccurate object descriptions and achieve accurate and robust segmentation, as shown in Fig. 12.

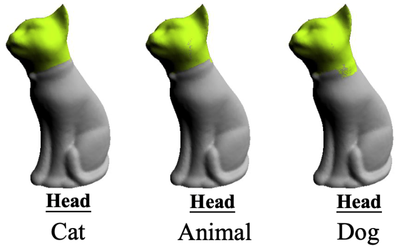

Fig. 12: Robustness of Object Description: MeshSegmenter exhibits strong robustness towards diverse object descriptions. It not only achieves accurate results for the correct description of objects "cat" with the grounding description "head", but it also achieves precise segmentation for homogeneous description of objects, such as animals and dogs.

5 Limitation

MeshSegmenter effectively aggregates detection and segmentation results from both textured and untextured meshes across multiple viewpoints, achieving accurate semantic segmentation for text-specified regions. However, large-scale pretrained 2D models are not designed for local-level detection and segmentation, which can lead to inaccuracies, including the erroneous detection of entire objects as target regions. Our model addresses this challenge by filtering out detections that encompass whole objects and introducing the face confidence revoting strategy. This approach effectively corrects erroneous detections based on correct detections from adjacent viewpoints. Another issue to consider is the visibility of faces. Our method infers 3D semantic segmentation from multi-view 2D detection and segmentation results. However, no rendering strategy can ensure the visibility of every face. Our approach to sampling rendering viewpoints involves uniform sampling rendered viewpoints in fixed rotation, which maximizes the likelihood that more faces will be visible.

6 Conclusion

In this work, we have proposed MeshSegmenter, a pioneering framework for 3D zero-shot semantic segmentation. Our model extends the capabilities of 2D detection and segmentation models to accurately segment diverse 3D meshes by text. We propose generating textures based on object descriptions to enhance the accuracy of 2D segmentation models. We also develop a multi-view revoting module that ensures 3D consistency in segmentation results. MeshSegmenter provides a transformative tool for 3D zero-shot segmentation, with potential applications in computer graphics and computer vision. By leveraging 2D segmentation models, incorporating texture information, and integrating multiple views, our framework provides stable and reliable segmentation results.

Acknowledge

The work was supported by NSFC #61932020, #62172279 and Program of Shanghai Academic Research Leader.

References

1. Abdelreheem, A., Eldesokey, A., Ovsjanikov, M., Wonka, P.: Zero-shot 3d shape correspondence. arXiv preprint arXiv:2306.03253 (2023) 3, 5, 10
2. Abdelreheem, A., Skorokhodov, I., Ovsjanikov, M., Wonka, P.: Satr: Zero-shot semantic segmentation of 3d shapes. arXiv preprint arXiv:2304.04909 (2023) 3, 5, 10, 11
3. Baek, D., Oh, Y., Ham, B.: Exploiting a joint embedding space for generalized zero-shot semantic segmentation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9536–9545 (2021) 4
4. Bansal, A., Sikka, K., Sharma, G., Chellappa, R., Divakaran, A.: ZERO-SHOT OBJECT DETECTION, p. 397–414 (Jan 2018). https://doi.org/10.1007/978-3-030-01246-5_24, http://dx.doi.org/10.1007/978-3-030-01246-5_24 2
5. Bansal, A., Sikka, K., Sharma, G., Chellappa, R., Divakaran, A.: Zero-shot object detection. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 384–400 (2018) 3
6. Bucher, M., Vu, T.H., Cord, M., Pérez, P.: Zero-shot semantic segmentation. Advances in Neural Information Processing Systems **32** (2019) 4
7. Chen, R., Zhu, X., Chen, N., Li, W., Ma, Y., Yang, R., Wang, W.: Zero-shot point cloud segmentation by transferring geometric primitives. arXiv preprint arXiv:2210.09923 (2022) 4
8. Chen, X., Golovinskiy, A., Funkhouser, T.: A benchmark for 3d mesh segmentation. Acm transactions on graphics (tog) **28**(3), 1–12 (2009) 2
9. Decatur, D., Lang, I., Hanocka, R.: 3d highlighter: Localizing regions on 3d shapes via text descriptions. arXiv preprint arXiv:2212.11263 (2022) 2, 3, 5, 10, 11
10. Ding, J., Xue, N., Xia, G.S., Dai, D.: Decoupling zero-shot semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11583–11592 (2022) 4
11. Ding, R., Yang, J., Xue, C., Zhang, W., Bai, S., Qi, X.: Language-driven open-vocabulary 3d scene understanding (Nov 2022) 4
12. Ding, R., Yang, J., Xue, C., Zhang, W., Bai, S., Qi, X.: Language-driven open-vocabulary 3d scene understanding. arXiv preprint arXiv:2211.16312 (2022) 4
13. Fan, Z., Wang, P., Jiang, Y., Gong, X., Xu, D., Wang, Z.: Nerf-sos: Any-view self-supervised object segmentation on complex scenes. arXiv preprint arXiv:2209.08776 (2022) 5
14. Ghiasi, G., Gu, X., Cui, Y., Lin, T.Y.: Scaling open-vocabulary image segmentation with image-level labels. In: Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVI. pp. 540–557. Springer (2022) 4
15. Goel, R., Sirikonda, D., Saini, S., Narayanan, P.: Interactive segmentation of radiance fields. arXiv preprint arXiv:2212.13545 (2022) 4
16. Gu, Z., Zhou, S., Niu, L., Zhao, Z., Zhang, L.: Context-aware feature generation for zero-shot semantic segmentation. In: Proceedings of the 28th ACM International Conference on Multimedia. pp. 1921–1929 (2020) 4
17. Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., Le, Q., Sung, Y.H., Li, Z., Duerig, T.: Scaling up visual and vision-language representation learning with noisy text supervision. In: International Conference on Machine Learning. pp. 4904–4916. PMLR (2021) 4
18. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. arXiv preprint arXiv:2304.02643 (2023) 2, 4, 5, 6, 8, 10

19. Kobayashi, S., Matsumoto, E., Sitzmann, V.: Decomposing nerf for editing via feature field distillation. arXiv preprint arXiv:2205.15585 (2022) 4
20. Koo, J., Huang, I., Achlioptas, P., Guibas, L., Sung, M.: Partglot: Learning shape part segmentation from language reference games 4
21. Kundu, A., Genova, K., Yin, X., Fathi, A., Pantofaru, C., Guibas, L.J., Tagliasacchi, A., Dellaert, F., Funkhouser, T.: Panoptic neural fields: A semantic object-aware neural scene representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12871–12881 (2022) 5
22. 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. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10965–10975 (June 2022) 5
23. 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. arXiv preprint arXiv:2212.01558 (2022) 4
24. Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., et al.: Grounding dino: Marrying dino with grounded pre-training for open-set object detection. arXiv preprint arXiv:2303.05499 (2023) 2, 4, 5, 6, 8, 10
25. Lombardi, S., Simon, T., Saragih, J., Schwartz, G., Lehrmann, A., Sheikh, Y.: Neural volumes: Learning dynamic renderable volumes from images. ACM Transactions on Graphics p. 1–14 (Aug 2019). https://doi.org/10.1145/3306346.3323020, http://dx.doi.org/10.1145/3306346.3323020 4
26. Lombardi, S., Simon, T., Saragih, J., Schwartz, G., Lehrmann, A., Sheikh, Y.: Neural volumes: Learning dynamic renderable volumes from images. arXiv preprint arXiv:1906.07751 (2019) 5
27. Luddecke, T., Ecker, A.: Image segmentation using text and image prompts. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (Jun 2022). https://doi.org/10.1109/cvpr52688.2022.00695, http://dx.doi.org/10.1109/cvpr52688.2022.00695 3
28. Lüddecke, T., Ecker, A.: Image segmentation using text and image prompts. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7086–7096 (2022) 4
29. Michele, B., Boulch, A., Puy, G., Bucher, M., Marlet, R.: Generative zero-shot learning for semantic segmentation of 3d point clouds. In: 2021 International Conference on 3D Vision (3DV). pp. 992–1002. IEEE (2021) 2, 4
30. Michele, B., Boulch, A., Puy, G., Bucher, M., Marlet, R.: Generative zero-shot learning for semantic segmentation of 3d point clouds. In: 2021 International Conference on 3D Vision (3DV) (Dec 2021). https://doi.org/10.1109/3dv53792.2021.00107, http://dx.doi.org/10.1109/3dv53792.2021.00107 4
31. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021) 5
32. Mu, N., Kirillov, A., Wagner, D., Xie, S.: Slip: Self-supervision meets language-image pre-training. In: Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVI. pp. 529–544. Springer (2022) 4
33. Pastore, G., Cermelli, F., Xian, Y., Mancini, M., Akata, Z., Caputo, B.: A closer look at self-training for zero-label semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2693–2702 (2021) 4

34. Peng, S., Jiang, C., Liao, Y., Niemeyer, M., Pollefeys, M., Geiger, A.: Shape as points: A differentiable poisson solver. Advances in Neural Information Processing Systems **34**, 13032–13044 (2021) 12
35. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021) 4, 5
36. Rahman, S., Khan, S., Porikli, F.: Zero-shot object detection: Learning to simultaneously recognize and localize novel concepts. Cornell University - arXiv,Cornell University - arXiv (Mar 2018) 3
37. Richardson, E., Metzer, G., Alaluf, Y., Giryes, R., Cohen-Or, D.: Texture: Text-guided texturing of 3d shapes. arXiv preprint arXiv:2302.01721 (2023) 6, 11
38. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10684–10695 (2022) 6
39. Shafiullah, N.M.M., Paxton, C., Pinto, L., Chintala, S., Szlam, A.: Clip-fields: Weakly supervised semantic fields for robotic memory. arXiv preprint arXiv:2210.05663 (2022) 4
40. Vora, S., Radwan, N., Greff, K., Meyer, H., Genova, K., Sajjadi, M.S., Pot, E., Tagliasacchi, A., Duckworth, D.: Nesf: Neural semantic fields for generalizable semantic segmentation of 3d scenes. arXiv preprint arXiv:2111.13260 (2021) 5
41. Xu, J., Mello, S., Liu, S., Byeon, W., Breuel, T., Kautz, J., Wang, X.: Groupvit: Semantic segmentation emerges from text supervision 3
42. Xu, J., Hou, J., Zhang, Y., Feng, R., Wang, Y., Qiao, Y., Xie, W.: Learning open-vocabulary semantic segmentation models from natural language supervision. arXiv preprint arXiv:2301.09121 (2023) 4
43. Xu, M., Zhang, Z., Wei, F., Hu, H., Bai, X.: Side adapter network for open-vocabulary semantic segmentation (Feb 2023) 3
44. Xu, M., Zhang, Z., Wei, F., Hu, H., Bai, X.: Side adapter network for open-vocabulary semantic segmentation. arXiv preprint arXiv:2302.12242 (2023) 4
45. Xu, M., Zhang, Z., Wei, F., Lin, Y., Cao, Y., Hu, H., Bai, X.: A simple baseline for zero-shot semantic segmentation with pre-trained vision-language model. arXiv preprint arXiv:2112.14757 (2021) 4
46. Yi, L., Kim, V.G., Ceylan, D., Shen, I.C., Yan, M., Su, H., Lu, C., Huang, Q., Sheffer, A., Guibas, L.: A scalable active framework for region annotation in 3d shape collections. ACM Transactions on Graphics p. 1–12 (Nov 2016). https://doi.org/10.1145/2980179.2980238, http://dx.doi.org/10.1145/2980179.2980238 10
47. Zhang, H., Zhang, P., Hu, X., Chen, Y.C., Li, L., Dai, X., Wang, L., Yuan, L., Hwang, J.N., Gao, J.: Glipv2: Unifying localization and vision-language understanding. Advances in Neural Information Processing Systems **35**, 36067–36080 (2022) 4
48. Zhang, H., Ding, H.: Prototypical matching and open set rejection for zero-shot semantic segmentation. International Conference on Computer Vision,International Conference on Computer Vision (Jan 2021) 3
49. Zhang, H., Ding, H.: Prototypical matching and open set rejection for zero-shot semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6974–6983 (2021) 4

50. Zhi, S., Laidlow, T., Leutenegger, S., Davison, A.J.: In-place scene labelling and understanding with implicit scene representation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15838–15847 (2021) 5
51. Zou, X., Dou, Z.Y., Yang, J., Gan, Z., Li, L., Li, C., Dai, X., Behl, H., Wang, J., Yuan, L., et al.: Generalized decoding for pixel, image, and language. arXiv preprint arXiv:2212.11270 (2022) 3, 4