

LISA: Reasoning Segmentation via Large Language Model

Xin Lai^{1*} Zhuotao Tian^{2*†} Yukang Chen¹ Yanwei Li¹ Yuhui Yuan⁴ Shu Liu³ Jiaya Jia^{1,3}
¹CUHK ²HIT (Shenzhen) ³SmartMore ⁴MSRA

Abstract

Although perception systems have made remarkable advancements in recent years, they still rely on explicit human instruction or pre-defined categories to identify the target objects before executing visual recognition tasks. Such systems cannot actively reason and comprehend implicit user intention. In this work, we propose a new segmentation task — reasoning segmentation. The task is designed to output a segmentation mask given a complex and implicit query text. Furthermore, we establish a benchmark comprising over one thousand image-instruction-mask data samples, incorporating intricate reasoning and world knowledge for evaluation purposes. Finally, we present LISA: large Language Instructed Segmentation Assistant, which inherits the language generation capabilities of multimodal Large Language Models (LLMs) while also possessing the ability to produce segmentation masks. We expand the original vocabulary with a <SEG> token and propose the embedding-as-mask paradigm to unlock the segmentation capability. Remarkably, LISA can handle cases involving complex reasoning and world knowledge. Also, it demonstrates robust zero-shot capability when trained exclusively on reasoning-free datasets. In addition, fine-tuning the model with merely 239 reasoning segmentation data samples results in further performance enhancement. Both quantitative and qualitative experiments show our method effectively unlocks new reasoning segmentation capabilities for multimodal LLMs. Code, models, and data are available at github.com/dylab-research/LISA.

1. Introduction

In daily life, users tend to issue direct commands like “Change the TV channel” to instruct a robot, rather than providing explicit step-by-step instructions such as “Go to the table first, find the TV remote, and then press the button to change the channel.” However, existing perception systems consistently rely on humans to explicitly indicate target objects or pre-define categories before executing visual recog-

nition tasks. These systems cannot actively reason and comprehend user intention based on implicit instruction. This reasoning ability is crucial in developing next-generation intelligent perception systems and holds substantial potential for industrial applications, particularly in robotics.

In this work, we introduce a new segmentation task — *reasoning segmentation*, which requires generating a binary segmentation mask based on an implicit query text involving *complex reasoning*. Notably, the query text is not limited to a straightforward reference (e.g., “the orange”), but a more complicated description involving *complex reasoning* or *world knowledge* (e.g., “the food with high Vitamin C”). To accomplish this task, the model must possess two key abilities: 1) reasoning *complex* and *implicit* text queries jointly with the image; 2) producing segmentation masks.

Inspired by the exceptional capacity of LLMs to reason and comprehend user intentions, we aim to leverage this capability of LLMs to address the aforementioned first challenge. However, while several studies [1, 23, 24, 28, 29, 55, 63] have integrated robust reasoning capabilities into multimodal LLMs to accommodate visual input, the majority of these models primarily concentrate on text generation tasks and still fall short in performing vision tasks that require fine-grained output formats, such as segmentation masks. This leads us to ask: can we enable multimodal LLMs with the capability to output segmentation masks?

To this end, we introduce LISA: a large Language Instructed Segmentation Assistant, a multimodal LLM capable of producing segmentation masks. Specifically, we incorporate an additional token, i.e., <SEG>, into the existing vocabulary. Upon generating the <SEG> token, its hidden embedding is further decoded into the corresponding segmentation mask. By representing the segmentation mask as an embedding, LISA acquires segmentation capabilities and benefits from end-to-end training. Remarkably, LISA demonstrates robust zero-shot abilities. Training the model solely on standard semantic segmentation and referring segmentation datasets yields surprisingly effective performance on the reasoning segmentation task. Furthermore, we find that LISA’s performance can be significantly enhanced by fine-tuning on just 239 reasoning segmentation data samples. As illustrated in Fig. 1, LISA can handle various scenarios

*Equal Contribution

†Corresponding Author.

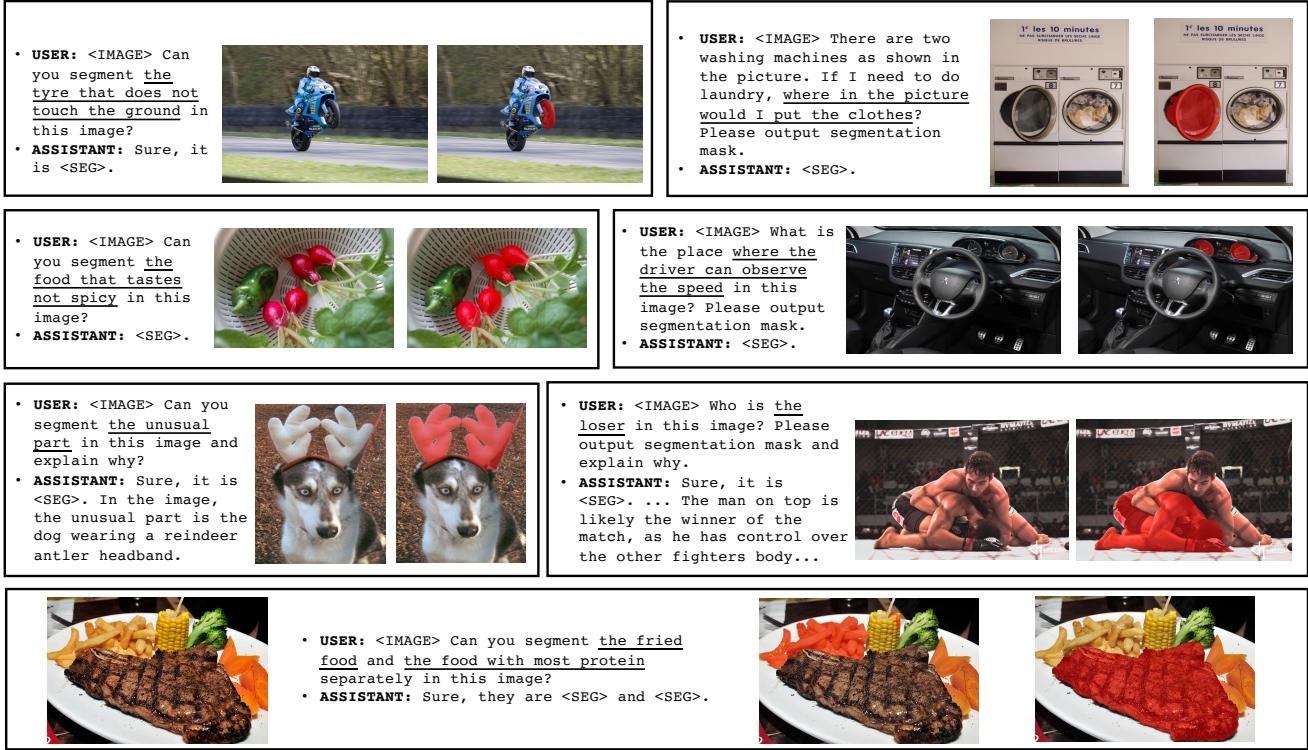


Figure 1. We unlock new segmentation capabilities for existing multimodal LLMs. Our model (i.e., LISA) can deal with cases involving complex reasoning and world knowledge. Also, we demonstrate the cases of explanatory answers in the 3rd row. Additionally, in the 4th row, our model can output multiple segmentation masks in a single answer. More illustrations can be found in the supplementary material.

involving complex reasoning and world knowledge.

In addition, to validate the effectiveness, we establish a benchmark for reasoning segmentation evaluation, called *ReasonSeg*. Comprising over one thousand image-instruction pairs, this benchmark offers persuasive evaluation metrics for the task. To align more closely with practical applications, we annotate the images from OpenImages [21] and ScanNetv2 [10] with implicit text queries that involve complex reasoning.

In summary, our contributions are as follows:

- We introduce the *reasoning segmentation* task, which necessitates reasoning based on implicit human instructions. Such reasoning capability is crucial for building a genuinely intelligent perception system.
- We present our model — LISA, which incorporates new segmentation capabilities. It demonstrates robust zero-shot ability on the reasoning segmentation task when trained solely on reasoning-free datasets, and achieves further performance boost by fine-tuning on just 239 data samples that involve reasoning.
- We establish a reasoning segmentation benchmark, *ReasonSeg*, containing over one thousand image-instruction-mask data samples. This benchmark is essential for evaluation and encourages the community to further explore the reasoning ability for vision tasks.

2. Related Work

2.1. Image Segmentation

Semantic segmentation aims to assign a class label to every pixel in an image. Numerous studies [2, 5, 8, 12, 16, 22, 31, 37, 42, 43, 45, 46, 51, 56, 59–61, 64] have proposed diverse designs (such as encoder-decoder, dilated convolution, pyramid pooling module, non-local operator, and more) to effectively encode semantic information. Research on instance segmentation [9, 14, 58] and panoptic segmentation [7, 18, 25, 50] has introduced various architectural innovations for instance-level segmentation, including DETR [4]-based structures, mask attention, and dynamic convolution. In recent years, typical segmentation tasks have made significant progress and become increasingly mature. Consequently, it is imperative to develop more intelligent interaction ways for image segmentation.

The referring segmentation task [17, 36] enables interaction with human language, aiming to segment the target object based on a given explicit text description. Recently, Kirillov et al. [19] introduced SAM, trained with billions of high-quality masks, supporting bounding boxes and points as prompts while demonstrating exceptional segmentation quality. X-Decoder [65] bridges vision and language, unifying multiple tasks within a single model. SEEM [66] further supports various human interaction methods, including text, audio, and scribble. However, these studies primarily focus



Figure 2. Examples of the annotated image-instruction-mask data samples. Left: short phrase query. Right: long sentence query. More examples are given in the supplementary material.

on addressing multi-task compatibility and unification, neglecting the injection of new capabilities. In this work, we present LISA and it possesses reasoning ability that has not been explored yet in existing segmentors.

2.2. Multimodal Large Language Model

Motivated by the remarkable reasoning abilities of LLMs, researchers are exploring ways to transfer these capabilities into the vision domain, developing multimodal LLMs. Flamingo [1] employs a cross-attention structure to attend to visual contexts, enabling visual in-context learning. Models such as BLIP-2 [24] and mPLUG-OWL [55] propose encoding image features with a visual encoder, which are then fed into the LLM alongside text embeddings. Otter [23] further incorporates robust few-shot capabilities through in-context instruction tuning on the proposed MIMIC-IT dataset. LLaVA [29] and MiniGPT-4 [63] first conduct image-text feature alignment followed by instruction tuning. Koh et al. [20] also investigates image retrieval for LLMs. Moreover, numerous works [32, 44, 49, 52, 54] utilize prompt engineering, connecting independent modules via API calls, but without the benefits of end-to-end training. Recently, there have been studies examining the intersection between multimodal LLMs and vision tasks. VisionLLM [47] offers a flexible interaction interface for multiple vision-centric tasks through instruction tuning but fails to fully exploit LLMs for complex reasoning. Kosmos-2 [38] constructs large-scale data of grounded image-text pairs, infusing grounding capabilities into LLMs. DetGPT [39] bridges the fixed multimodal LLM and open-vocabulary detector, enabling detection to be performed based on user instruction. GPT4RoI [57] introduces spatial boxes as input and trains the model on region-text pairs. In contrast, our work aims to efficiently inject segmentation capabilities into multimodal LLMs in the manner of end-to-end training.

3. Reasoning Segmentation

3.1. Problem Definition

The reasoning segmentation task is to output a binary segmentation mask \mathbf{M} , given an input image \mathbf{x}_{img} and an im-

plicit query text instruction \mathbf{x}_{txt} . The task shares a similar formulation with the referring segmentation task [17], but is far more challenging. The key distinction lies in the complexity of the query text in reasoning segmentation. Instead of a straightforward phrase (e.g., “the trash can”), the query text includes more intricate expressions (e.g., “something that the garbage should be put into”) or longer sentences (e.g., “After cooking, consuming food, and preparing for food, where can we throw away the rest of the food and scraps?”) that involve complex reasoning or world knowledge.

3.2. Benchmark

Given the lack of quantitative evaluation, it is imperative to establish a benchmark for the reasoning segmentation task. To ensure reliable assessment, we have collected a diverse set of images from OpenImages [21] and ScanNetv2 [10], annotating them with implicit text instructions and high-quality target masks. To cover different scenarios, our text instructions consist of two types: 1) short phrases; 2) long sentences; as illustrated in Figure 2. The resulting *ReasonSeg* benchmark comprises a total of 1218 image-instruction-mask data samples. This dataset is further partitioned into three splits: *train*, *val*, and *test*, containing 239, 200, and 779 data samples, respectively. As the primary purpose of the benchmark is evaluation, the validation and testing sets include a larger number of data samples. The details of data annotation are given in the supplementary material.

4. Our Method

In this section, we first introduce the model architecture in Sec. 4.1. After that, we elaborate on the training data preparation and training parameters in Sec. 4.2.

4.1. Architecture

Embedding as Mask. Most current multimodal LLMs (such as LLaVA [29], Flamingo [1], BLIP-2 [24], Otter [23], etc.) support image and text as input, but they can only output text and cannot directly output fine-grained segmentation masks. VisionLLM [47] offers a solution by parsing

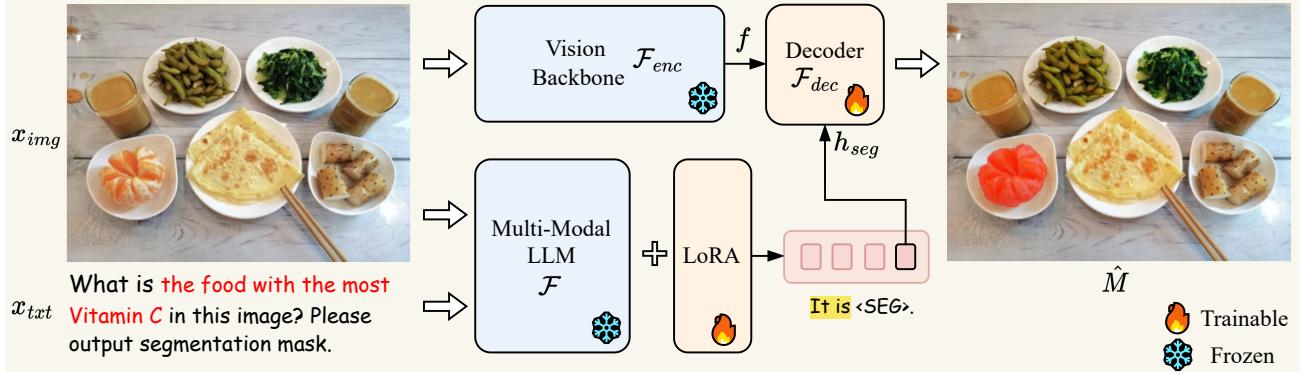


Figure 3. The pipeline of LISA. Given the input image and text query, the multimodal LLM (e.g., LLaVA [29]) generates text output. The last-layer embedding for the $\langle \text{SEG} \rangle$ token is then decoded into the segmentation mask via the decoder. We use LoRA [15] for efficient fine-tuning. The choice of vision backbone can be flexible (e.g., SAM [66], Mask2Former [9]).

segmentation masks as sequences of polygons, enabling the representation of segmentation masks as plain text and allowing end-to-end training within the framework of existing multimodal LLMs. However, end-to-end training with the polygon sequences introduces optimization challenges and may compromise generalization ability unless a massive amount of data and computational resources are employed. For instance, training a 7B model, VisionLLM requires 4×8 NVIDIA 80G A100 GPUs and 50 epochs, which is computationally prohibitive. In contrast, it takes less than 3 days to train LISA-7B on 8 NVIDIA 24G 3090 GPUs.

To this end, we propose the embedding-as-mask paradigm to infuse new segmentation capabilities into the multimodal LLM. The pipeline of our method is illustrated in Fig. 3. Specifically, we first expand the original LLM vocabulary with a new token, i.e., $\langle \text{SEG} \rangle$, which signifies the request for the segmentation output. Given a text instruction x_{txt} along with the input image x_{img} , we feed them into the multimodal LLM \mathcal{F} , which in turn outputs a text response \hat{y}_{txt} . It can be formulated as

$$\hat{y}_{txt} = \mathcal{F}(x_{img}, x_{txt}). \quad (1)$$

When the LLM intends to generate a binary segmentation mask, the output \hat{y}_{txt} would include a $\langle \text{SEG} \rangle$ token. We then extract the LLM last-layer embedding \tilde{h}_{seg} corresponding to the $\langle \text{SEG} \rangle$ token and apply an MLP projection layer γ to obtain h_{seg} . Simultaneously, the vision backbone \mathcal{F}_{enc} extracts the dense visual features f from the visual input x_{img} . Finally, h_{seg} and f are fed to the decoder \mathcal{F}_{dec} to produce the final segmentation mask \hat{M} . The detailed structure of the decoder \mathcal{F}_{dec} follows [19]. The process can be formulated as

$$\begin{aligned} h_{seg} &= \gamma(\tilde{h}_{seg}), \quad f = \mathcal{F}_{enc}(x_{img}), \\ \hat{M} &= \mathcal{F}_{dec}(h_{seg}, f). \end{aligned} \quad (2)$$

Training Objectives. The model is trained end-to-end using the text generation loss \mathcal{L}_{txt} and the segmentation

mask loss \mathcal{L}_{mask} . The overall objective \mathcal{L} is the weighted sum of these losses, determined by λ_{txt} and λ_{mask} :

$$\mathcal{L} = \lambda_{txt} \mathcal{L}_{txt} + \lambda_{mask} \mathcal{L}_{mask}. \quad (3)$$

Specifically, \mathcal{L}_{txt} is the auto-regressive cross-entropy loss for text generation, and \mathcal{L}_{mask} is the mask loss, which encourages the model to produce high-quality segmentation results. To compute \mathcal{L}_{mask} , we employ a combination of per-pixel binary cross-entropy (BCE) loss and DICE loss, with corresponding loss weights λ_{bce} and λ_{dice} . Given the ground-truth targets y_{txt} and M , these losses can be formulated as

$$\begin{aligned} \mathcal{L}_{txt} &= \text{CE}(\hat{y}_{txt}, y_{txt}), \\ \mathcal{L}_{mask} &= \lambda_{bce} \text{BCE}(\hat{M}, M) + \lambda_{dice} \text{DICE}(\hat{M}, M). \end{aligned} \quad (4)$$

It is noteworthy that the proposed method endows existing multimodal LLMs with new segmentation capabilities, such that they can generate not only text but also fine-grained output formats. Also, our method is based on an end-to-end training pipeline and connects the LLM and vision modules with hidden embedding representation, which proves significantly more effective than the decoupled two-stage method as discussed in Sec. 5.2.

4.2. Training

Training Data Formulation. As illustrated in Fig. 4, our training data comprises mainly three parts, all of which are derived from widely-used public datasets. The details are as follows:

- **Semantic Segmentation Dataset.** Semantic segmentation datasets typically consist of images and the corresponding multi-class labels. During training, we randomly choose several categories for each image. To generate data that matches the format of visual question answering, we employ a question-answer template like “**USER:** <IMAGE> Can you segment the

Raw Data



Processed Data

- USER:** <IMAGE> Can you segment the **table** in this image?
- ASSISTANT:** It is <SEG>.



- USER:** <IMAGE> Can you segment **the lady with the blue shirt** in this image?
- ASSISTANT:** Sure, it is <SEG>.



- USER:** What type of sign and traffic device can be seen in the image? <IMAGE>
- ASSISTANT:** In the image, there is a street sign and a traffic light above a city road.
- USER:** How many traffic lights are visible in the image?
- ...

No Binary Segmentation Mask

VQA Data

Figure 4. The illustration of training data formulation from different types of data, including semantic segmentation data, referring segmentation data, and visual question answering (VQA) data.

{class_name} in this image? **ASSISTANT:** It is <SEG>.", where {class_name} is the chosen category, and <IMAGE> denotes the placeholder for tokens of image patches. The corresponding binary segmentation mask is used as the ground truth to provide mask loss supervision. During training, we also use other templates to generate the QA data to ensure data diversity, as shown in the supplementary material. We adopt ADE20K, COCO-Stuff, and LVIS-PACO part segmentation datasets.

- Vanilla Referring Segmentation Dataset.** Referring segmentation datasets provide an input image and an explicit short description of the target object. Thus, it is easy to convert them into question-answer pairs using a template like “**USER:** <IMAGE> Can you segment {description} in this image? **ASSISTANT:** Sure, it is <SEG>.”, where {description} is the given explicit description. For this part, we adopt refCOCO, refCOCO+, refCOCOg, and refCLEF datasets.
- Visual Question Answering Dataset.** To preserve the original Visual Question Answering (VQA) ability of the multimodal LLM, we also include the VQA dataset during training. We use LLaVA-Instruct-150k [29] for LLaVA v1

and LLaVA-v1.5-mix665k for LLaVA v1.5 [28].

Notably, the above datasets do not include any reasoning segmentation data sample. Instead, it only contains samples where the target objects are explicitly indicated in the query texts. Surprisingly, even without complex reasoning training data, LISA demonstrates impressive zero-shot ability on the *ReasonSeg* benchmark, as shown in Table 1. Moreover, we find that further performance boost could be yielded by finetuning the model on only 239 data samples that involve complex reasoning.

Trainable Parameters. To preserve the learned knowledge of the pre-trained multimodal LLM \mathcal{F} (i.e., LLaVA in our experiments), we leverage LoRA [15] to perform efficient fine-tuning, and completely freeze the vision backbone \mathcal{F}_{enc} . The decoder \mathcal{F}_{dec} is fully fine-tuned. Additionally, the LLM token embeddings (`embed_tokens`), the LLM head (`lm_head`), and the projection layer γ are also trainable.

It is notable that the resulting model avoids the catastrophic forgetting of the original text generation capability and preserves the conversation ability, as verified in the supplementary material. The potential reasons are: we 1) employ LoRA fine-tuning to reduce the trainable parameters and 2) incorporate the VQA dataset during fine-tuning.

Table 1. Reasoning segmentation results among LISA (ours) and previous related works. ‘ft’ denotes using 239 reasoning segmentation data samples to fine-tune the model. Unless otherwise specified, we use LLaVA v1 [29] as the base model. LLaVA1.5 denotes LLaVA v1.5 [28].

Method	val		test					
	overall		short query		long query		overall	
	gIoU	cIoU	gIoU	cIoU	gIoU	cIoU	gIoU	cIoU
OVSeg [26]	28.5	18.6	18.0	15.5	28.7	22.5	26.1	20.8
GRES [27]	22.4	19.9	17.6	15.0	22.6	23.8	21.3	22.0
X-Decoder [65]	22.6	17.9	20.4	11.6	22.2	17.5	21.7	16.3
SEEM [66]	25.5	21.2	20.1	11.5	25.6	20.8	24.3	18.7
Grounded-SAM [30]	26.0	14.5	17.8	10.8	22.4	18.6	21.3	16.4
LISA-7B	44.4	46.0	37.6	34.4	36.6	34.7	36.8	34.1
LISA-7B (ft)	52.9	54.0	40.6	40.6	49.4	51.0	47.3	48.4
LISA-13B	48.9	46.9	39.9	43.3	46.4	46.5	44.8	45.8
LISA-13B (ft)	56.2	62.9	44.3	42.0	54.0	54.3	51.7	51.1
LLaVA1.5-7B + OVSeg	38.2	23.5	24.2	18.7	44.6	37.1	39.7	31.8
LISA-7B-LLaVA1.5	53.6	52.3	47.1	48.5	49.2	48.9	48.7	48.8
LISA-7B-LLaVA1.5 (ft)	61.3	62.9	48.3	46.3	57.9	59.7	55.6	56.9
LLaVA1.5-13B + OVSeg	37.9	26.4	27.1	19.4	46.1	40.6	41.5	34.1
LISA-13B-LLaVA1.5	57.7	60.3	50.8	50.0	54.7	50.9	53.8	50.8
LISA-13B-LLaVA1.5 (ft)	65.0	72.9	55.4	50.6	63.2	65.3	61.3	62.2

5. Experiment

5.1. Experimental Setting

Network Architecture. Unless otherwise specified, we use LLaVA-7B-v1-1 or LLaVA-13B-v1-1 [29] as the base multimodal LLM \mathcal{F} , and adopt the ViT-H SAM [19] backbone as the vision backbone \mathcal{F}_{enc} . The projection layer of γ is an MLP with channels of [256, 4096, 4096].

Implementation Details. We adopt 8 NVIDIA 24G 3090 GPUs for training. The training scripts are based on deep-speed [41] engine. We use AdamW [33] optimizer with the learning rate and weight decay set to 0.0003 and 0, respectively. We also adopt WarmupDecayLR as the learning rate scheduler, where the warmup iterations are set to 100. The weights of the text generation loss λ_{txt} and the mask loss λ_{mask} are set to 1.0 and 1.0, respectively, and those of the bce loss λ_{bce} and the dice loss λ_{dice} are set to 2.0 and 0.5, respectively. Besides, the batch size per device is set to 2, and the gradient accumulation step is set to 10. During training, we select at most 3 categories for each image in semantic segmentation datasets.

Datasets. As mentioned in Sec. 4.2, our training data is mainly composed of three types of datasets: (1) For the semantic segmentation dataset, we use ADE20K [62] and COCO-Stuff [3]. Besides, to enhance the segmentation result for some part of an object, we also use part semantic segmentation datasets, including PACO-LVIS [40], PartImageNet [13], and PASCAL-Part [6]; (2) For the referring segmentation dataset, we use refCLEF, refCOCO, refCOCO+ [17], and refCOCOg [35]; (3) For the visual ques-

tion answering (VQA) dataset, we use the datasets of LLaVA-Instruct-150k for LLaVA v1 [29] and LLaVA-v1.5-mix665k for LLaVA v1.5 [28]. In order to avoid data leakage, we exclude the COCO samples whose images are present in the refCOCO(+g) validation sets during training. Furthermore, we surprisingly find that by fine-tuning the model on only 239 ReasonSeg data samples, the model’s performance can be further boosted.

Evaluation Metrics. We follow most previous works on referring segmentation [17, 35] to adopt two metrics: gIoU and cIoU. gIoU is defined by the average of all per-image Intersection-over-Unions (IoUs), while cIoU is defined by the cumulative intersection over the cumulative union. Since cIoU is highly biased toward large-area objects and it fluctuates too much, gIoU is preferred.

5.2. Reasoning Segmentation Results

The reasoning segmentation results are shown in Table 1. It is worth noting that existing works fail to handle the task, but our model can accomplish the task involving complex reasoning with more than 20% gIoU performance boost. As mentioned before, the reasoning segmentation task is essentially different from the referring segmentation task in that it requires the model to possess *reasoning ability* or access *world knowledge*. Only by truly understanding the query, can the model do well in the task. The existing works have no proper way to understand an implicit query, but our model exploits multimodal LLMs to reach the goal.

Notably, we also make a comparison with the vanilla two-stage method (LLaVA1.5 + OVSeg). Specifically, the

Table 2. Referring segmentation results (cIoU) among LISA (ours) and existing methods.

Method	refCOCO			refCOCO+			refCOCOg	
	val	testA	testB	val	testA	testB	val(U)	test(U)
MCN [34]	62.4	64.2	59.7	50.6	55.0	44.7	49.2	49.4
VLT [11]	67.5	70.5	65.2	56.3	61.0	50.1	55.0	57.7
CRIS [48]	70.5	73.2	66.1	62.3	68.1	53.7	59.9	60.4
LAVT [53]	72.7	75.8	68.8	62.1	68.4	55.1	61.2	62.1
ReLA [27]	73.8	76.5	70.2	66.0	71.0	57.7	65.0	66.0
X-Decoder [65]	-	-	-	-	-	-	64.6	-
SEEM [66]	-	-	-	-	-	-	65.7	-
LISA-7B	74.1	76.5	71.1	62.4	67.4	56.5	66.4	68.5
LISA-7B (fine-tuned on ReferSeg)	74.9	79.1	72.3	65.1	70.8	58.1	67.9	70.6

two-stage method refers to first using a multimodal LLM (e.g., LLaVA v1.5) to generate a text output for the input query, and then adopting a referring or open-vocabulary segmentation model (e.g., OVSeg) to generate the segmentation mask. If the intermediate text output remains too long and exceeds the input token length limit of OVSeg, we use GPT-3.5 to further summarize. More details can be found in the supplementary material. The results in Table 1 show that our model outperforms the two-stage method significantly. We explain that the potential reasons are: 1) Our model is trained end-to-end, while the two-stage method is completely decoupled; 2) The two-stage method relies on text as an intermediary to transmit information, while our model utilizes the hidden embedding that is more expressive.

Another finding is that LISA-13B outperforms the 7B counterpart substantially, especially on the long-query scenarios, which indicates that the current performance bottleneck may still lie in understanding the query text, and a stronger multimodal LLM (e.g., LLaVA v1.5 [28]) leads to even better results.

5.3. Vanilla Referring Segmentation Results

To show that our model is also competent in the vanilla referring segmentation task, we make a comparison with existing state-of-the-art methods in Table 2. We evaluate the methods on refCOCO, refCOCO+, refCOCOg validation and testing sets. Our model achieves state-of-the-art results across various referring segmentation benchmarks.

5.4. Ablation Study

In this section, we conduct an extensive ablation study to reveal the contribution of each component. Unless otherwise specified, we report the metrics of gIoU and cIoU of LISA-7B on the validation set.

Design Choices of Vision Backbone. We emphasize that vision backbones other than SAM are also applicable in our framework. In Table 3, we notice that SAM performs the best, potentially because of the massive high-quality

Table 3. Ablation study on the design choice of vision backbone. ‘ft’ denotes finetuning on ReasonSeg training set.

Vision Backbone	gIoU	cIoU
Mask2Former-Swin-L	42.4	38.8
SAM (w/ LoRA)	41.5	37.3
SAM	44.4	46.0
Mask2Former-Swin-L (ft)	50.7	52.3
SAM w/ LORA (ft)	51.8	51.9
SAM (ft)	52.9	54.0

Table 4. Ablation study on SAM pre-trained weight and rephrasing.

Exp. ID	Pre-train _{SAM} γ	rephrasing	gIoU	cIoU
1		✓	35.9	44.6
2	✓		50.7	51.1
3	✓	✓	52.9	54.0

data used in its pre-training phase. Further, we also find that with the Mask2Former backbone, our framework still achieves a decent performance on the reasoning segmentation task, significantly outperforming previous works such as X-Decoder [65]. This reveals the fact that the design choice of vision backbone is flexible and not limited to SAM.

SAM LoRA Fintuning. We also investigate the effectiveness of applying LoRA on the SAM backbone. In Table 3, we note that the performance of LoRA fine-tuned SAM backbone is inferior to that of the frozen one. A potential reason is that fine-tuning impairs the generalization ability of the original SAM model.

SAM Pre-trained Weight. To demonstrate the contribution of SAM pre-trained weight, we make a comparison between Experiments 1 and 3 in Table 4. Without being initialized with SAM pre-trained weight, the vision backbone is trained from scratch. This causes the performance to fall substantially behind that of the baseline model.



Figure 5. Visual comparison among LISA (ours) and existing related methods. More illustrations are given in the supplementary material.

Table 5. Ablation study on training data.

ID	SemanticSeg			ReferSeg	VQA	ReasonSeg	gIoU	cIoU
	ADE20K	COCO-Stuff	PartSeg					
1	✓			✓	✓	✓	48.9	53.5
2	✓			✓	✓	✓	48.5	50.8
3	✓	✓		✓	✓	✓	46.7	50.9
4		✓		✓	✓	✓	46.6	46.7
5			✓	✓	✓	✓	30.4	20.4
6	✓	✓	✓		✓	✓	47.7	51.1
7	✓	✓	✓	✓	✓		44.4	46.0
8	✓	✓	✓	✓	✓	✓	52.9	54.0

Table 6. Results on the ReasonSeg test set.

Training splits	# data samples	gIoU	cIoU
train	239	51.7	51.1
train + val	439	54.0	54.9

Instruction Rephrasing by GPT-3.5. When fine-tuning the model on the reasoning segmentation data samples, we rephrase the text instruction by GPT-3.5 (the details are shown in the supplementary material), and randomly choose one. The comparison between Experiments 2 and 3 in Table 4 shows that the performance is increased by 2.2% gIoU and 2.9% cIoU. This result verifies the effectiveness of such data augmentation.

Contribution of All Types of Training Data. In Table 5, we show the contribution of each type of data to the performance. We find that in Exp. 5, we do not use any semantic segmentation dataset, and the performance drops a lot. We conjecture that semantic segmentation datasets provide a large amount of ground-truth binary masks for training, since a multi-class label can induce multiple binary masks.

We also notice that adding more reasoning segmentation data samples during training leads to better results. In Table 6, we also add the ReasonSeg val set (200 data samples) during fine-tuning, and it yields better performance in both gIoU and cIoU metrics. This indicates that more reasoning segmentation training samples are beneficial at this moment.

5.5. Qualitative Results

As depicted in Fig. 5, we provide a visual comparison with existing related works, including the model for open-vocabulary semantic segmentation (OVSeg), referring segmentation (GRES), and the generalist models for segmentation (X-Decoder and SEEM). These models fail to handle the displayed cases with various errors, while our approach produces accurate and high-quality segmentation results. More illustrations are given in the supplementary material.

6. Conclusion

In this work, we have proposed a new segmentation task—*reasoning segmentation*. Also, we have introduced an evaluation benchmark *ReasonSeg*, which comprises over one thousand data samples. Finally, we have presented our model—LISA. It injects segmentation capabilities into current multimodal LLMs and performs surprisingly effectively on the reasoning segmentation task. We hope our work can shed new light on the direction of combining LLMs and vision tasks in the future.

Acknowledgements

This work is supported in part by the Research Grants Council under the Areas of Excellence scheme grant AoE/E-601/22-R and the Shenzhen Science and Technology Program under No. KQTD20210811090149095.

References

- [1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *NeurIPS*, 2022. 1, 3
- [2] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *TPAMI*, 2017. 2
- [3] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *CVPR*, 2018. 6
- [4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 2
- [5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI*, 2018. 2
- [6] Xianjie Chen, Roozbeh Mottaghi, Xiaobai Liu, Sanja Fidler, Raquel Urtasun, and Alan Yuille. Detect what you can: Detecting and representing objects using holistic models and body parts. In *CVPR*, 2014. 6
- [7] Bowen Cheng, Maxwell D Collins, Yukun Zhu, Ting Liu, Thomas S Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *CVPR*, 2020. 2
- [8] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *NeurIPS*, 2021. 2
- [9] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022. 2, 4
- [10] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 2, 3
- [11] Henghui Ding, Chang Liu, Suchen Wang, and Xudong Jiang. Vision-language transformer and query generation for referring segmentation. In *ICCV*, 2021. 7
- [12] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *CVPR*, 2019. 2
- [13] Ju He, Shuo Yang, Shaokang Yang, Adam Kortylewski, Xiaodong Yuan, Jie-Neng Chen, Shuai Liu, Cheng Yang, Qihang Yu, and Alan Yuille. Partimagenet: A large, high-quality dataset of parts. In *ECCV*, 2022. 6
- [14] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 2
- [15] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv:2106.09685*, 2021. 4, 5
- [16] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *ICCV*, 2019. 2
- [17] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *EMNLP*, 2014. 2, 3, 6
- [18] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *CVPR*, 2019. 2
- [19] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv:2304.02643*, 2023. 2, 4, 6
- [20] Jing Yu Koh, Ruslan Salakhutdinov, and Daniel Fried. Grounding language models to images for multimodal inputs and outputs. 2023. 3
- [21] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Mallocci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020. 2, 3
- [22] Xin Lai, Zhuotao Tian, Li Jiang, Shu Liu, Hengshuang Zhao, Liwei Wang, and Jiaya Jia. Semi-supervised semantic segmentation with directional context-aware consistency. In *CVPR*, 2021. 2
- [23] Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Jingkang Yang, and Ziwei Liu. Otter: A multi-modal model with in-context instruction tuning. *arXiv:2305.03726*, 2023. 1, 3
- [24] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv:2301.12597*, 2023. 1, 3
- [25] Yanwei Li, Hengshuang Zhao, Xiaojuan Qi, Liwei Wang, Zeming Li, Jian Sun, and Jiaya Jia. Fully convolutional networks for panoptic segmentation. In *CVPR*, 2021. 2
- [26] Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. In *CVPR*, 2023. 6
- [27] Chang Liu, Henghui Ding, and Xudong Jiang. Gres: Generalized referring expression segmentation. In *CVPR*, 2023. 6, 7
- [28] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint*, 2023. 1, 5, 6, 7
- [29] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv:2304.08485*, 2023. 1, 3, 4, 5, 6
- [30] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint*, 2023. 6
- [31] Wei Liu, Andrew Rabinovich, and Alexander C. Berg. Parsenet: Looking wider to see better. *arXiv preprint*, 2015. 2

- [32] Zhaoyang Liu, Yinan He, Wenhai Wang, Weiyun Wang, Yi Wang, Shoufa Chen, Qinglong Zhang, Yang Yang, Qingyun Li, Jiashuo Yu, et al. Internchat: Solving vision-centric tasks by interacting with chatbots beyond language. [arXiv:2305.05662](#), 2023. 3
- [33] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. [arXiv:1711.05101](#), 2017. 6
- [34] Gen Luo, Yiyi Zhou, Xiaoshuai Sun, Liujuan Cao, Chenglin Wu, Cheng Deng, and Rongrong Ji. Multi-task collaborative network for joint referring expression comprehension and segmentation. In [CVPR](#), 2020. 7
- [35] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In [CVPR](#), 2016. 6
- [36] Varun K Nagaraja, Vlad I Morariu, and Larry S Davis. Modeling context between objects for referring expression understanding. In [ECCV](#), 2016. 2
- [37] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In [ICCV](#), 2015. 2
- [38] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shao-han Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. [arXiv:2306.14824](#), 2023. 3
- [39] Renjie Pi, Jiahui Gao, Shizhe Diao, Rui Pan, Hanze Dong, Jipeng Zhang, Lewei Yao, Jianhua Han, Hang Xu, Lingpeng Kong, and Tong Zhang. Detgpt: Detect what you need via reasoning. [arXiv:2305.14167](#), 2023. 3
- [40] Vignesh Ramanathan, Anmol Kalia, Vladan Petrovic, Yi Wen, Baixue Zheng, Baishan Guo, Rui Wang, Aaron Marquez, Rama Kovvuri, Abhishek Kadian, et al. Paco: Parts and attributes of common objects. In [CVPR](#), 2023. 6
- [41] Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In [SIGKDD](#), 2020. 6
- [42] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In [MICCAI](#), 2015. 2
- [43] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. [TPAMI](#), 2017. 2
- [44] Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueling Zhuang. Hugginggpt: Solving ai tasks with chatgpt and its friends in huggingface. [arXiv:2303.17580](#), 2023. 3
- [45] Zhuotao Tian, Pengguang Chen, Xin Lai, Li Jiang, Shu Liu, Hengshuang Zhao, Bei Yu, Ming-Chang Yang, and Jiaya Jia. Adaptive perspective distillation for semantic segmentation. [TPAMI](#), 2022. 2
- [46] Zhuotao Tian, Jiequan Cui, Li Jiang, Xiaojuan Qi, Xin Lai, Yixin Chen, Shu Liu, and Jiaya Jia. Learning context-aware classifier for semantic segmentation. [AAAI](#), 2023. 2
- [47] Wenhai Wang, Zhe Chen, Xiaokang Chen, Jiannan Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu Qiao, et al. Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. [arXiv:2305.11175](#), 2023. 3
- [48] Zhaoqing Wang, Yu Lu, Qiang Li, Xunqiang Tao, Yandong Guo, Mingming Gong, and Tongliang Liu. Cris: Clip-driven referring image segmentation. In [CVPR](#), 2022. 7
- [49] Chenfei Wu, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. Visual chatgpt: Talking, drawing and editing with visual foundation models. [arXiv:2303.04671](#), 2023. 3
- [50] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun. Upsnet: A unified panoptic segmentation network. In [CVPR](#), 2019. 2
- [51] Maoke Yang, Kun Yu, Chi Zhang, Zhiwei Li, and Kuiyuan Yang. Denseaspp for semantic segmentation in street scenes. In [CVPR](#), 2018. 2
- [52] Rui Yang, Lin Song, Yanwei Li, Sijie Zhao, Yixiao Ge, Xiu Li, and Ying Shan. Gpt4tools: Teaching large language model to use tools via self-instruction. [arXiv:2305.18752](#), 2023. 3
- [53] Zhao Yang, Jiaqi Wang, Yansong Tang, Kai Chen, Hengshuang Zhao, and Philip HS Torr. Lavt: Language-aware vision transformer for referring image segmentation. In [CVPR](#), 2022. 7
- [54] Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Ehsan Azarnasab, Faisal Ahmed, Zicheng Liu, Ce Liu, Michael Zeng, and Lijuan Wang. Mm-react: Prompting chatgpt for multimodal reasoning and action. [arXiv:2303.11381](#), 2023. 3
- [55] Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. mplug-owl: Modularization empowers large language models with multimodality. [arXiv:2304.14178](#), 2023. 1, 3
- [56] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In [ICLR](#), 2016. 2
- [57] Shilong Zhang, Peize Sun, Shoufa Chen, Min Xiao, Wenqi Shao, Wenwei Zhang, Kai Chen, and Ping Luo. Gpt4roi: Instruction tuning large language model on region-of-interest. [arXiv:2307.03601](#), 2023. 3
- [58] Wenwei Zhang, Jiangmiao Pang, Kai Chen, and Chen Change Loy. K-net: Towards unified image segmentation. [NeurIPS](#), 2021. 2
- [59] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In [CVPR](#), 2017. 2
- [60] Hengshuang Zhao, Xiaojuan Qi, Xiaoyong Shen, Jianping Shi, and Jiaya Jia. Icnet for real-time semantic segmentation on high-resolution images. In [ECCV](#), 2018.
- [61] Hengshuang Zhao, Yi Zhang, Shu Liu, Jianping Shi, Chen Change Loy, Dahua Lin, and Jiaya Jia. Psanet: Pointwise spatial attention network for scene parsing. In [ECCV](#), 2018. 2
- [62] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In [CVPR](#), 2017. 6
- [63] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-

- language understanding with advanced large language models.
[arXiv:2304.10592](#), 2023. [1](#), [3](#)
- [64] Zhen Zhu, Mengde Xu, Song Bai, Tengteng Huang, and Xiang Bai. Asymmetric non-local neural networks for semantic segmentation. In [ICCV](#), 2019. [2](#)
- [65] Xueyan Zou, Zi-Yi Dou, Jianwei Yang, Zhe Gan, Linjie Li, Chunyuan Li, Xiyang Dai, Harkirat Behl, Jianfeng Wang, Lu Yuan, et al. Generalized decoding for pixel, image, and language. In [CVPR](#), 2023. [2](#), [6](#), [7](#)
- [66] Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere all at once. [arXiv:2304.06718](#), 2023. [2](#), [4](#), [6](#), [7](#)