

---

# Leveraging Hallucinations to Reduce Manual Prompt Dependency in Promptable Segmentation

---

Jian Hu<sup>1</sup>, Jiayi Lin<sup>1</sup>, Junchi Yan<sup>2</sup>, Shaogang Gong<sup>1</sup>

<sup>1</sup>Queen Mary University of London, <sup>2</sup>Shanghai Jiao Tong University

{jian.hu, jiayi.lin, s.gong}@qmul.ac.uk, yanjunchi@sjtu.edu.cn

<https://lwpjh.github.io/ProMaC/>

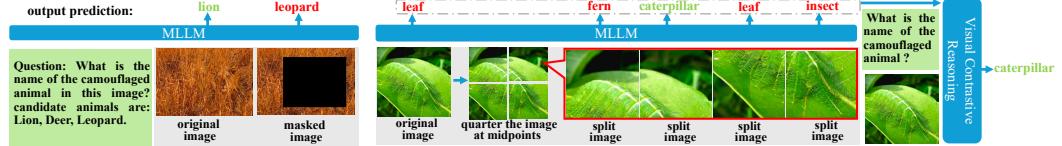
## Abstract

Promptable segmentation typically requires instance-specific manual prompts to guide the segmentation of each desired object. To minimize such a need, task-generic promptable segmentation has been introduced, which employs a single task-generic prompt to segment various images of different objects in the same task. Current methods use Multimodal Large Language Models (MLLMs) to reason detailed instance-specific prompts from a task-generic prompt for improving segmentation accuracy. The effectiveness of this segmentation heavily depends on the precision of these derived prompts. However, MLLMs often suffer hallucinations during reasoning, resulting in inaccurate prompting. While existing methods focus on eliminating hallucinations to improve a model, we argue that MLLM hallucinations can reveal valuable contextual insights when leveraged correctly, as they represent pre-trained large-scale knowledge beyond individual images. In this paper, we utilize hallucinations to mine task-related information from images and verify its accuracy for enhancing precision of the generated prompts. Specifically, we introduce an iterative **Prompt-Mask Cycle** generation framework (ProMaC) with a prompt generator and a mask generator. The prompt generator uses a multi-scale chain of thought prompting, initially exploring hallucinations for extracting extended contextual knowledge on a test image. These hallucinations are then reduced to formulate precise instance-specific prompts, directing the mask generator to produce masks that are consistent with task semantics by mask semantic alignment. The generated masks iteratively induce the prompt generator to focus more on task-relevant image areas and reduce irrelevant hallucinations, resulting jointly in better prompts and masks. Experiments on 5 benchmarks demonstrate the effectiveness of ProMaC. Code given in <https://lwpjh.github.io/ProMaC/>.

## 1 Introduction

Current promptable segmentation methods rely on instance-specific manual prompts to guide segmentation, greatly limiting its large-scale application. Recently, a manual-free task-generic promptable segmentation approach was introduced [19]: only a single task-generic prompt is needed for all samples under the same task, e.g., “camouflaged animal” is a task-generic prompt for all images in a camouflaged object detection task. The model segments task-relevant objects in various images based on this generic prompt, significantly reducing the annotation workload.

A task-generic prompt is both coarse and potentially ambiguous, can result in poor segmentation when directly applied. To address this problem, existing methods [19, 36] utilize the prior knowledge embedded in Multimodal Large Language Models (MLLMs) to infer more detailed, instance-specific prompts, such as bounding boxes or keywords, to guide the segmentation. However, these MLLMs often generate hallucinations due to object co-occurrence priors [66, 63], mistakenly predicting non-existent elements based on the environment as instance-specific prompts (Fig.1(a)). This can mislead segmentation and degrade model performance. While it is common to consider MLLM’s



(a) Hallucination by co-occurrence prior. (b) Using hallucinations can benefit accurate prompt generation.

Figure 1: (a) During MLLM pretraining, leopards often co-occur with grass. If the lion is masked, the model incorrectly identifies it as a leopard based on the grass. (b) Directly inputting the image into MLLM causes the hidden caterpillar being incorrectly predicted as a leaf. Splitting the image results in interested objects being incomplete or absent, prompting MLLM to induce hallucinations and utilize prior knowledge to predict potential task-related objects within the image. Our visual contrastive reasoning then eliminates the hallucinations and validates the gathered predictions, aiding in the accurate identification of the caterpillar.

hallucinations as detrimental and should be eradicated [50], this phenomenon actually demonstrates a MLLM’s significant capacity for contextual inference based on prior training. We want to explore MLLM hallucinations as a valuable untapped knowledge resource for scene understanding, critical in complex segmentation scenarios. In practice, when task-related objects are not prominently visible, hallucinations can fill in missing information with plausible predictions based on learned patterns of association. Moreover, they can also extend beyond these familiar patterns, exploring and identifying new relationships within the data that were not explicitly taught during training. This dual ability to replicate and innovate makes hallucinations a valuable asset for enhancing model performance in complex or new situations. This predictive reasoning capacity not only fills perceptual gaps but also enriches the model’s understanding, as hallucinations utilize prior knowledge to replicate and discover new patterns, enhancing insight into the target domain (see Fig.1(b)). Despite the potential benefits, using hallucinations to extract useful information from images to aid task remains unexplored.

In this work, instead of direct eliminating hallucinations, we utilize them as prior knowledge to mine extended task-related information from a given test image, performing scene understanding on the image before segmentation, then systematically reduce irrelevant hallucinations iteratively by visual masking verification, optimizing jointly instance-specific prompts and masks. To this end, we introduce an iterative, training-free Prompt-Mask Cycle Generation method (ProMaC) that refines segmentation through cyclic interactions between a prompt and mask generator (see Fig. 2). The prompt generator uses a multi-scale chain-of-thought prompting mechanism, which utilizes hallucinations to hypothesize and visual masking to verify, thereby creating more accurate instance-specific prompts. We trigger the hallucinatory tendencies of MLLMs, the process starts by dividing the image into patches at different scales and positions. Such partial visibilities of objects facilitate MLLMs to hypothesize potential object semantic labels and visual locations based on its prior knowledge. For validating the correctness of these hypotheses, we formulate a visual contrastive reasoning mechanism to generate contrastive images that contain only the background without any potential task-related objects. This helps identify all possible co-occurrence hallucinations caused by the background. By comparing these contrastive images with the original images, the MLLM effectively distinguishes between accurate hypotheses and those influenced by misleading prior knowledge, leading to more reliable prompts. Given the current promptable segmentation models’ strength at mask prediction but struggle with label prediction, the mask generator uses mask semantic alignment to ensure that the produced masks align with the task semantics. These aligned masks not only serve as outputs but also guide the prompt generator in subsequent cycles, enhancing both prompt and mask quality continuously. **Our contributions are three-folds:**

- 1). We introduce a training-free Prompt-Mask Cycle Generation (ProMaC) to perform two tasks: Explore MLLM hallucinations as prior knowledge to enhance contextual scene understanding on each test image; systematically reduce irrelevant hallucinations to verify iteratively and optimize jointly both generated prompts and visual masking in object segmentation.
- 2). We formulate an iterative optimization method including a prompt generator and a mask generator. To improve prompt relevance, the prompt generator utilizes a multi-scale chain of thought approach. It first leverages hallucinations to expand task-related plausible prompts, then applies visual contrastive reasoning to validate and reduce irrelevant prompts. ProMaC’s mask generator overcomes SAM’s shortcomings in label prediction by creating masks that align semantically with generated prompts.
- 3). Comprehensive comparative evaluations on 5 different segmentation tasks with 12 diverse datasets against 22 existing models demonstrate the effectiveness of ProMaC.

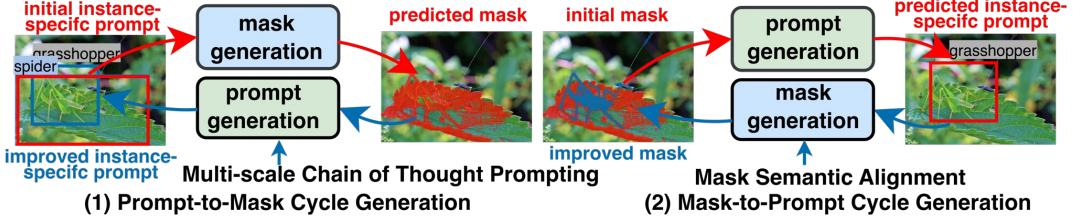


Figure 2: An overview of ProMac: Masks created iteratively by the mask generator guide the prompt generator to jointly improve instance-specific prompts and visual masking in segmentation.

## 2 Related Works

**Promptable Segmentation** refers to object segmentation with active interactions from user inputs. Interaction methods vary from points, boxes, to scribbles. SAM [27], AV-SAM[39], GroundingSAM [36] and SEEM [69] accept video, audio, and multimodal inputs. However, they often rely on manual prompts, which can be unclear and subjective. Even with these prompts, they typically excel only in specific tasks. To address this issue, GenSAM [19] introduces a manual-free promptable segmentation setting, where only one task-generic prompt is provided. This prompt can be applied to all images within the task for instance-specific segmentation without any additional manual prompting. GenSAM primarily utilizes MLLM to infer the names of task-related objects in the images and uses them as instance-specific prompts for SAM to guide segmentation. However, GenSAM lacks spatial information about objects and may lead to inaccurate prompt predictions in complex scenes.

**Hallucinations in MLLMs** refers to models generate content that does not exist in the input data [63]. This issue often arises from the models leveraging extensive prior training rather than just the immediate input, leading to false predictions on fine-grained details. There are some efforts to mitigate this problem, including refining training processes [53, 42] and improving model architectures [4]. Other efforts focus on aligning model outputs more closely with actual data, employing feedback mechanisms for real-time adjustments [52]. While current works focus on eliminating hallucinations to enhance performance [29, 62], our work explores how to utilize hallucinations to expand and reason plausible context and validate them iteratively to remove irrelevant generalizations.

**Visual Marking for MLLMs** has been explored in recent research to prompt MLLMs through manipulation of visual inputs: (i) adding learnable soft tokens to visual inputs for efficient parameter tuning [1, 26], (ii) using image sequences as demonstrations of a new task [2, 9], and (iii) overlaying visual markers like masks, boxes, and circles onto visual inputs to ground regions [59, 52]. Our work falls into the third category, employing visual guidance for reasoning. Yang et al. [57] propose set-of-mark (SoM) prompts, where images are segmented and numbered regions to improve GPT-4V [41] visual grounding. However, as detailed in Tab.5, we confirm previous findings [5] that this visual marker approach struggles with open-source MLLMs like LLaVA. Instead of proprietary models [52] or fine-tuning [5, 8, 21], our training-free ProMaC uses inpainting task-related regions and contrasting model output distributions to prompt MLLMs.

## 3 Methodology

We introduce ProMaC, a cycle-generation method for segmenting unknown multiple classes of objects training-free with only a single task-generic prompt. Specifically, given an image  $X \in \mathbb{R}^{H \times W \times 3}$  from a test set, ProMaC employs a task-generic prompt  $P_g$  across datasets in the same task to produce a final segmentation mask  $M \in \mathbb{R}^{H \times W}$ , thereby removing the need for individual supervision for each image. The prompt generator leverages prior knowledge gained to reason and deduce instance-specific prompts, which then guide a mask generator to create masks aligned with task semantics. These masks act both as the current segmentation outcome and as visual markers for generating subsequent prompts. Training-free ProMaC relies solely on test-time adaptation.

### 3.1 Prompt Generator

Prompt generator employs MLLMs to generate instance-specific prompts based on image content and prior knowledge. It transforms the general prompt  $P_g$ , into an instance-specific prompt for each individual instance, providing more detailed descriptions of task-relevant objects. MLLM with parameters  $\theta$  receives an image  $X$  and query  $P$  as inputs.  $X$  provides contextual visual information to assist the model in generating a relevant response  $y$  to the query  $P$ . The response  $y$  is sampled

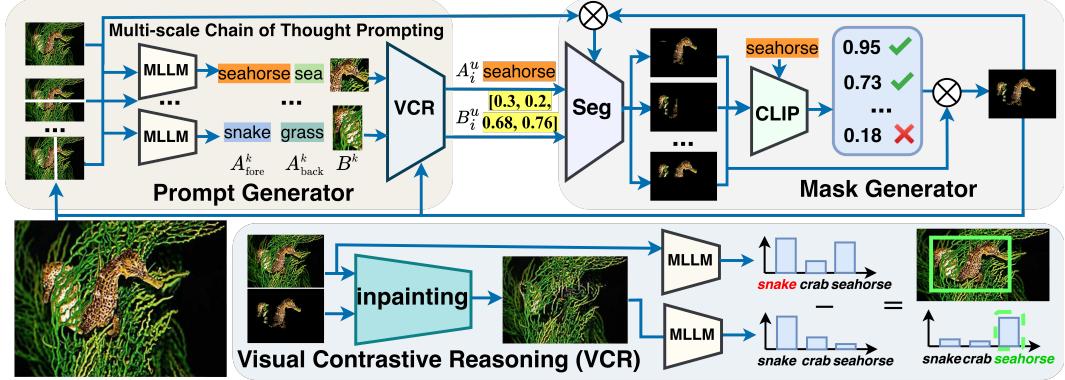


Figure 3: ProMaC consists of a prompt generator and a mask generator for cyclical optimization. The prompt generator employs multi-scale chain-of-thought prompting. It initially use hallucinations for exploring task-related information within image patches. It identifies task-relevant objects and their backgrounds ( $A_{\text{fore}}^k$ ,  $A_{\text{back}}^k$ ) along with their locations ( $B^k$ ). Subsequently, it uses visual contrastive reasoning to refine and finalize instance-specific prompts ( $A_i^u$ ,  $B_i^u$ ) by eliminating hallucinations. The mask generator then processes these prompts into the segmentation model ("Seg"), producing a mask aligned with task semantics. This mask further guides the visual contrastive reasoning process, which leverages an inpainting model to eliminate masked regions, creating contrastive images. These images enable the prompt generator to further refine its prompts, enhancing segmentation accuracy.

auto-regressively from the probability distribution conditioned on  $P$  and  $X$  as follows:

$$y_t \sim p_\theta(y_t | X, P, y_{<t}) \propto \exp(\text{logit}_\theta(y_t | X, P, y_{<t})) \quad (1)$$

where  $y_t$  denotes the token at time step  $t$ , and  $y < t$  represents the sequence of generated tokens up to the time step  $(t - 1)$ . In practice, predicted task-relevant objects can often blend into the background due to texture, color, size, or position, leading to inaccuracies in instance-specific prompts. To address this problem, we explore MLLM hallucinations as contextual prior knowledge from pretraining, rather than eliminate them. These hallucinations are particularly useful when direct visual cues are absent or ambiguous, helping the model fill in information gaps and hypothesize potential task-related elements within the image that are not prominent. By revealing these often-overlooked subtle associations, hallucinations provide a more comprehensive scene understanding of the image content. This deeper contextual understanding provide a reasoning context for generating more accurate and relevant instance-specific prompts candidates. Thus, using hallucinations to uncover task-related knowledge helps overcome challenges from visual ambiguities and object camouflage in complex scenes. To this end, we propose a multi-scale chain-of-thought prompting strategy that stimulates hallucinations to leverage prior knowledge, fully extracts task-relevant information, and then uses this information to enhance the precision of the generated instance-specific prompts.

### 3.1.1 Multi-scale Chain of Thought Prompting

Multi-scale Chain of Thought Prompting consists of two processes: Gathering candidate knowledge and generating accurate instance-specific prompts. To efficiently collect task-relevant information from an image, as shown in Fig. 3, we divide the input image into patches at various scales by cutting horizontally, vertically, or by leaving it whole. These patches are then processed by the MLLM to gather preliminary instance-specific prompts. The differing levels of task-relevant object visibility in each patch prompt the MLLM to induce hallucinations. These hallucinations utilize prior knowledge to explore connections between the image data and the associated task, aiding in the detection of potential bounding boxes and object names. The process is computed by:

$$B^k = \text{MLLM}(X^k, C^k, P_B), \quad A_{\text{fore}}^k, A_{\text{back}}^k = \text{MLLM}(X^k, C^k, P_A), \quad (2)$$

where  $C^k$  is the caption generated by MLLM for the  $k$ -th image patch  $X^k$ .  $P_g$  is task-generic prompt. For bounding box prediction, the prompt  $P_B$ , which instructs “*This image is from the  $P_g$  detection task, output the bounding box of the  $P_g$ .*”. This guides the MLLM to predict the bounding box  $B^k$  of the task-related objects within the patch. For predicting name, the prompt  $P_A$ , stating “*Output the name of the  $P_g$  and its environment in one word.*” is used, guiding the MLLM to predict the names of the task-related objects  $A_{\text{fore}}^k$  and their backgrounds  $A_{\text{back}}^k$  from each patch. The preliminary data,

including object names  $A_i^{\text{fore}}$  and bounding boxes  $B_i^k$ , gathered from various patches, are compiled into candidate lists  $\mathcal{A}_i$  and  $\mathcal{B}_i$ . Here,  $i$  denotes the iteration in the iterative learning cycle. In this process, the hallucinations employed are essentially based on object co-occurrence priors, where objects commonly associated with background elements during pre-training are predicted to be task-relevant, even if they are not present in the current image. This prior knowledge is useful during the knowledge collection stage as it uncovers implicit relationships and details in the image. However, it can also reduce accuracy of the later fine-grained instance-specific prompts generation. Therefore, it is crucial to control these hallucinations in the latter stage to prevent incorrect predictions.

**Visual Contrastive Reasoning.** To mitigate hallucinations caused by object co-occurrence priors, recent research highlights particularly relevant regions of an image to direct MLLMs focus toward task-related elements, thereby minimizing background interference and enhancing model accuracy [57, 52]. To achieve this, visual markers are employed to steer MLLM attention on task-relevant visual regions, thereby reducing hallucinations. While closed-source MLLMs like GPT-4V [41] can interpret these markers effectively, they are costly and large. In contrast, models like LLaVA [35] are open-source, but cannot process visual markers such as points or bounding boxes, and employing these markers might disrupt the original pixel data, degrading performance on LLaVA (see Tab. 5). Moreover, accurate pixel-level visual markers are unavailable in our setting. To solve this problem, we aim to enable LLaVA to focus on task-related regions without altering the original pixel data, thereby effectively minimizing hallucinations and enhancing the precision of instance-specific prompts.

Despite the absence of instance-level annotations, promptable segmentation models produce masks with detailed textures, which provide rich positional and textural information about interested regions. We use these masks as visual markers to guide a MLLM to focus on task-related areas during the generation of instance-specific prompts. Inspired by classifier-free guidance [18, 45], we introduce visual contrastive reasoning (VCR), a training-free visual marking method to help MLLM focus on specific regions, reducing hallucinations. The relevance of a region is assessed by observing MLLM output changes when key areas are excluded. It guides the MLLM to focus on areas with notable changes (bottom of Fig.3). Based on Eq. (1), we derive a probability distribution by comparing original image  $X$  with a modified image,  $X' = \text{process}(X, \text{IM})$ , where region IM is excluded.

$$y_t \propto p_\theta(y_t | X, P, y_{<t}) \left( \frac{p_\theta(y_t | X, P, y_{<t})}{p_\theta(y_t | \text{process}(X, \text{IM}), P, y_{<t})} \right)^\alpha \\ \sim \text{softmax}[(1 + \alpha) \cdot \text{logit}_\theta(y_t | X, P, y_{<t}) - \alpha \cdot \text{logit}_\theta(y_t | \text{process}(X, \text{IM}), P, y_{<t})], \quad (3)$$

where  $\alpha$  is the level of focus on region IM. A higher  $\alpha$  increases emphasis on that region. Following [52], we set  $\alpha = 1$  in all tasks. It preserves the integrity of the original image pixels  $X$ , while constructing contrastive samples  $X'$  that encourage the model to focus on task-related regions. Ideally,  $X'$  should exclude task-related objects while maintaining a uniform appearance and overall context with the original image. But directly marking  $X'$  disrupts its pixels, making contrastive sample generation challenging.

**Contrastive Sample Generation.** To address it, we employ inpainting, where the mask  $M_{i-1}$  obtained from the previous iteration segmentation is treated as the inpainting mask  $\text{IM}_i$  to guide the creation of  $X'$ . We use a negative prompt  $P_n$ : " $A_i^{\text{fore}}$ , is a  $P_g$ ", to ensure that the inpainted  $X'$  does not contain potentially task-related objects  $A_i^{\text{fore}}$ . Additionally, we use a positive prompt  $P_p$ : " $A_i^{\text{back}}$ , high quality, detailed, blended to the original image.", to ensure consistency between the generated portion and the surrounding background  $A_i^{\text{back}}$ . The corresponding inpainting is defined as:

$$X' = F_{in}(X, \text{IM}_i, P_p, P_n), \quad (4)$$

where  $F_{in}$  represents the inpainting module, and we choose Stable Diffusion to perform this operation. This method ensures the generated  $X'$  excludes task-related objects without disrupting the pixel continuity. In the first iteration, since  $\text{IM}_i$  does not yet exist, we use bounding box predictions from various patches  $\mathcal{B}_i$  as an alternative. As  $X'$  contains only the background, comparing it with  $X$  eliminates co-occurrence hallucination caused by the background and highlights differences in task-related regions, subtly guiding the model to focus on these areas. Finally, we use visual contrastive reasoning to identify accurate instance-specific prompt to guide segmentation as follows,

$$B_i^u = \text{VCR}(X, X', C, P_B), \quad A_i^u = \text{VCR}(X, X', C, P_A), \quad (5)$$

where VCR represents our visual contrastive reasoning, and  $C$  is the caption of the image. The collected knowledge,  $\mathcal{A}_i$  and  $\mathcal{B}_i$ , is integrated into the prompt  $P_A$  and  $P_B$ . This process aids in identify the ultimate instance-specific names  $A_i^u$  and bounding boxes  $B_i^u$  of the objects.

### 3.2 Mask Generator

Until now, we described how to use SAM-generated masks as a visual marker to guide the model to focus on task-relevant areas for generating accurate instance-specific prompts. But this method relies on an assumption that the mask accurately delineates task-related regions. However, SAM is trained on large-scale prompt-mask pairs without category labels, it excels at identifying masks based on image textures but lacks label prediction capabilities. Consequently, the SAM-generated mask may not always align with task semantics, yet such alignment is crucial for our method.

#### 3.2.1 Mask Semantic Alignment

We need to utilize texture generalization capabilities of SAM to describe possible task-related objects within the prompt-targeted areas, while also ensuring that the generated masks align with the task semantics. To achieve this, we divide the input image into patches of varying scales using horizontal, vertical, and uncut divisions as outlined in the last section. These processed patches are then reintegrated onto the original image with surrounding areas blacked out, and fed into SAM to ensure it focuses exclusively on the patch. Finally, masks generated from different patches are aggregated based on their relevance to task semantics, providing an accurate representation of task-related objects. The masks for each patch is generated as follows,

$$m_i^k = \text{SAM}(\text{Spatial CLIP}(A_i^u, X_i), B_i^u, X_i^k), \quad (6)$$

where mask  $m_i^k$  is obtained by inputting the corresponding image patch  $X_i^k$  and associated prompts into SAM during the  $i$ -th iteration. Following [19], Spatial CLIP maps the text prompt  $A_i^u$  to regions in the image  $X_i$  that correspond to the content of the prompt. The processed images, along with the generated instance-specific text prompts  $A_i^u$ , are then input into CLIP to assess semantic similarity.

$$s(m_i^k) = \text{CLIP}(m_i^k \odot X_i, A_i^u), \quad (7)$$

the operation  $\odot$  results in retaining only those parts of  $X_i$  that are covered by the predicted mask.  $s(m_i^k)$  represents the similarity between masked image and  $A_i^u$ , calculated using CLIP. The similarity scores obtained from different patches are denoted as  $S_i = [s(m_i^1), s(m_i^2), \dots, s(m_i^K)]$ . After normalizing the elements within  $S_i$ , the closer the normalized  $s(m_i^k)$  is to 1, the more semantically aligned  $m_i^k$  is with the instance-specific text prompt  $A_i^u$ . Finally, we compute the weighted sum of the normalized  $s(m_i^k)$  and  $m_i^k$  as follows.

$$M_i = \sum_{k=1}^K (s(m_i^k) * m_i^k), \quad (8)$$

$M_i$  is the output mask of the  $i$ -th iteration of  $X$ . The generated  $M_i$  leverages SAM’s mask prediction capabilities to create highly detailed masks. Simultaneously, through the mask semantic alignment process, it ensures that the output mask aligns with the task’s semantics, thereby overcoming the limitation of SAM’s mask prediction lacking semantic understanding. The mask is applied to the original image as a weight, to generate the next iteration image  $X_i$  for segmentation. This excludes irrelevant regions to reduce interference during segmentation.

$$X_{i+1} = w \cdot (X_i \odot M_i) + (1 - w) \cdot X_i, \quad (9)$$

where  $w$  is a hyperparameter, which we have assigned a value of 0.3.

### 3.3 Mask Prompt Cycle Generation

The mask generated from the last iteration will guide the prompt generator in the next iteration to focus on potential task-related regions, eliminating the erroneous effects of irrelevant hallucinations and generating more accurate instance-specific prompts. These prompts, in turn, help the mask generator produce better masks. Through iterative prompt generation and mask generation jointly, we yield both better instance-specific prompts and visual masks. Finally, the masks from different iterations are averaged, and the mask closest to the mean is considered the final output.

$$i^* = \arg \min_i \left( \left| M_i - \frac{\sum_i (M_1, \dots, M_I)}{i_{\text{result}}} \right| \right). \quad (10)$$

Here,  $I$  is the number of adaptation epoches and  $M_{i^*}$  is the corresponding final mask for image  $X$ .

Table 1: Results on Camouflaged Object Detection (COD) under different settings. Best are in **bold**.

Methods	Venue	Camouflaged Object Detection											
		CHAMELEON [48]				CAMO [28]				COD10K [14]			
		$M \downarrow$	$F_\beta \uparrow$	$E_\phi \uparrow$	$S_\alpha \uparrow$	$M \downarrow$	$F_\beta \uparrow$	$E_\phi \uparrow$	$S_\alpha \uparrow$	$M \downarrow$	$F_\beta \uparrow$	$E_\phi \uparrow$	$S_\alpha \uparrow$
Scribble Supervision Setting													
WSSA[60]	CVPR20	0.067	0.692	0.860	0.782	0.118	0.615	0.786	0.696	0.071	0.536	0.770	0.684
SCWS[58]	AAAI21	0.053	0.758	0.881	0.792	0.102	0.658	0.795	0.713	0.055	0.602	0.805	0.710
TEL[60]	CVPR22	0.073	0.708	0.827	0.785	0.104	0.681	0.797	0.717	0.057	0.633	0.826	0.724
SCOD[17]	AAAI23	<b>0.046</b>	<b>0.791</b>	<b>0.897</b>	0.818	<b>0.092</b>	0.709	0.815	0.735	0.049	0.637	0.832	0.733
SAM-S[27]	ICCV23	0.076	0.729	0.820	0.650	0.105	0.682	0.774	0.731	0.046	0.695	0.828	0.772
WS-SAM[16]	NeurIPS23	<b>0.046</b>	0.777	<b>0.897</b>	<b>0.824</b>	<b>0.092</b>	<b>0.742</b>	<b>0.818</b>	<b>0.759</b>	<b>0.038</b>	<b>0.719</b>	<b>0.878</b>	<b>0.803</b>
Point Supervision Setting													
WSSA[60]	CVPR20	0.105	0.660	0.712	0.711	0.148	0.607	0.652	0.649	0.087	0.509	0.733	0.642
SCWS[58]	AAAI21	0.097	0.684	0.739	0.714	0.142	0.624	0.672	0.687	0.082	0.593	0.777	0.738
TEL[60]	CVPR22	0.094	0.712	0.751	0.746	0.133	0.662	0.674	0.645	0.063	0.623	0.803	0.727
SCOD[17]	AAAI23	0.092	0.688	0.746	0.725	0.137	0.629	0.688	0.663	0.060	0.607	0.802	0.711
SAM[27]	ICCV23	0.207	0.595	0.647	0.635	0.160	0.597	0.639	0.643	0.093	0.673	0.737	0.730
SAM-P[27]	ICCV23	0.101	0.696	0.745	0.697	0.123	0.649	0.693	0.677	0.069	0.694	0.796	0.765
WS-SAM[16]	NeurIPS23	<b>0.056</b>	<b>0.767</b>	<b>0.868</b>	<b>0.805</b>	<b>0.102</b>	<b>0.703</b>	<b>0.757</b>	<b>0.718</b>	<b>0.039</b>	<b>0.698</b>	<b>0.856</b>	<b>0.790</b>
Task-Generic Prompt Setting													
GPT4V+SAM [41, 27]	Arxiv23	0.180	0.557	0.710	0.637	0.206	0.466	0.666	0.573	0.187	0.448	0.672	0.601
LLaVA1.5+SAM [35, 27]	NeurIPS23	0.168	0.561	0.718	0.666	0.314	0.401	0.585	0.501	0.170	0.530	0.728	0.662
X-Decoder [67]	CVPR23	0.124	0.654	0.748	0.716	0.104	0.628	0.745	0.709	0.171	0.556	0.705	0.652
SEEM [69]	NeurIPS23	0.094	0.011	0.307	0.454	0.192	0.023	0.315	0.404	0.143	0.001	0.280	0.425
GroundingSAM [27, 36]	ICCV23	0.122	0.662	0.776	0.744	0.157	0.656	0.753	0.707	0.085	0.670	0.813	0.764
GenSAM [19]	AAAI24	0.073	0.696	0.806	0.774	0.106	0.669	0.798	0.729	0.058	0.695	0.843	0.783
ProMaC	Ours	<b>0.044</b>	<b>0.790</b>	<b>0.899</b>	<b>0.833</b>	<b>0.090</b>	<b>0.725</b>	<b>0.846</b>	<b>0.767</b>	<b>0.042</b>	<b>0.716</b>	<b>0.876</b>	<b>0.805</b>

Table 2: Results for Medical Image Segmentation (MIS) under task-generic prompt setting.

Methods	Venue	Polyp Image Segmentation								Skin Lesion Segmentation			
		CVC-ColonDB [49]				Kvasir [23]				ISIC [10]			
		$M \downarrow$	$F_\beta \uparrow$	$E_\phi \uparrow$	$S_\alpha \uparrow$	$M \downarrow$	$F_\beta \uparrow$	$E_\phi \uparrow$	$S_\alpha \uparrow$	$M \downarrow$	$F_\beta \uparrow$	$E_\phi \uparrow$	$S_\alpha \uparrow$
Task-Generic Prompt Setting													
GPT4V+SAM [41, 27]	Arxiv23	0.578	0.051	0.246	0.242	0.614	0.128	0.236	0.253	0.514	0.387	0.366	0.334
LLaVA1.5+SAM [35, 27]	NeurIPS23	0.491	0.194	0.355	0.357	0.479	0.293	0.400	0.403	0.369	0.473	0.497	0.477
X-Decoder [67]	CVPR23	0.462	0.095	0.327	0.331	0.449	0.202	0.371	0.384	0.338	0.315	0.127	0.407
SEEM [69]	NeurIPS23	0.570	0.085	0.280	0.284	0.520	0.215	0.339	0.367	0.362	0.250	0.002	0.280
GroundingSAM [27, 36]	ICCV23	0.711	0.071	0.195	0.206	0.387	0.353	0.521	0.468	0.301	0.348	0.247	0.533
GenSAM [19]	AAAI24	0.244	0.059	0.494	0.379	0.172	0.210	0.619	0.487	0.171	0.699	0.744	0.678
ProMaC	Ours	<b>0.176</b>	<b>0.243</b>	<b>0.583</b>	<b>0.530</b>	<b>0.166</b>	<b>0.394</b>	<b>0.726</b>	<b>0.573</b>	<b>0.168</b>	<b>0.717</b>	<b>0.755</b>	<b>0.689</b>

## 4 Experimental Setup

**Baselines.** To evaluate our approach across various scenarios, we first assess the performance of ProMaC on challenging segmentation tasks, including Camouflaged Object Detection (COD), Medical Image Segmentation (MIS), and Transparent Object Detection (TOD). These tasks are areas where SAM struggle [24]. In the COD task, we compare ProMaC with weakly supervised segmentation methods [27, 60, 58, 60, 17, 17, 20, 22]. Two supervision levels are used for comparison: scribble supervision, where main structures for the foreground and background are sketched during training, and point supervision, where separate points are provided for both foreground and background. In the task-generic prompt setting, we introduce a challenging scenario by only providing a task description as a generic prompt for segmentation. ProMaC integrates LLaVA1.5 [35] with SAM [27]. We also experiment on the MIS and PIS tasks to demonstrate the effectiveness of our method using task-generic prompts compared to previous methods. We assess GPT4V+SAM and LLaVA1.5+SAM in this setting to demonstrate that current MLLM models cannot address it well. We also compare ProMaC against current SOTA promptable segmentation methods to showcase its effectiveness. Next, we evaluate ProMaC on Open-Vocabulary Segmentation (OVS), and compare with leading methods [31, 68, 69, 36, 61, 64, 30]. Our Visual Contrastive Reasoning (VCR) strategy is applicable to other tasks, especially those requiring complex spatial understanding. We used the What’s Up spatial reasoning dataset [25] to evaluate how well VCR guides models to focus on task-relevant regions in Spatial Reasoning (SR) task. Our results are the average of three trials.

**Metric.** For evaluating the first three tasks, metrics Mean Absolute Error (M), adaptive F-measure ( $F_\beta$ ) [38], mean E-measure ( $E_\phi$ ) [15], and structure measure ( $S_\alpha$ ) [13] are used. Lower M or higher values for  $F_\beta$ ,  $E_\phi$ , and  $S_\alpha$  reflect better performance. Mean Intersection over Union (mIoU) and accuracy measure OVS and SR performance respectively, with higher values indicating better results.

**PyTorch Implementation Details.** For the MLLM models, we utilize LLaVA-1.5-13B for evaluation purposes. For CLIP, we adopt the CS-ViT-B/16 pretrained model. The inpainting model is stable-diffusion-2-inpainting. The task-generic prompts for the COD task is "camouflaged animal". The MIS task consists of two sub-tasks: polyp image segmentation and skin lesion segmentation, each with its own task-generic prompts, "polyp" and "skin lesion" respectively. For TOD task, prompt is fixed as "glass". All tasks are optimized using training-free test-time adaptation, with each task iterating for four epochs, except for the polyp image segmentation task, which undergoes six epochs.

Table 3: Result on Transparent Object Segmentation and Open-Vocabulary Segmentation Tasks.

(a) Transparent Object Segmentation.

(b) Open-vocabulary Segmentation.

Methods	GSD [32]			Trans10K-hard [54]			Methods	Venue	Seg. Anno.	Image-Text pairs	VOC			
	$M \downarrow$	$F_\beta \uparrow$	$E_\phi \uparrow$	$S_\alpha \uparrow$	$M \downarrow$	$F_\beta \uparrow$	$E_\phi \uparrow$	$S_\alpha \uparrow$						
mIoU					mIoU				mIoU	Context	Object	mIoU	mIoU	
GPT4V+SAM [41, 27]	0.312	0.104	0.392	0.363	0.288	0.199	0.607	0.512	MaskCLIP [65]	CCCV22	-	38.8	23.6	20.6
LLaVA1.5+SAM [35, 27]	0.197	0.202	0.545	0.433	0.272	0.167	0.621	0.555	TCL [6]	CVPR23	-	51.2	24.3	<b>30.4</b>
X-Decoder [67]	0.191	0.240	0.643	0.480	0.568	<b>0.611</b>	0.218	0.280	GroupViT [55]	CVPR22	-	52.3	22.4	-
SEEM [69]	0.184	0.224	0.573	0.479	0.557	0.501	0.013	0.256	ViewCo [44]	ICLR23	-	52.4	23.0	23.5
GroundingSAM [27, 36]	0.168	0.230	0.572	0.483	0.436	0.415	0.047	0.424	SegCLIP [37]	ICML23	COCO [33]	52.6	24.7	26.5
GenSAM [19]	0.155	0.394	0.700	0.559	0.263	0.489	0.612	0.536	OVSegmentor [56]	CVPR23	-	53.8	20.4	25.1
ProMaC	<b>0.147</b>	<b>0.409</b>	<b>0.723</b>	<b>0.569</b>	<b>0.251</b>	0.509	<b>0.654</b>	<b>0.557</b>	ProMaC	Ours	-	<b>59.3</b>	<b>30.7</b>	25.2

Table 4: Ablation Study on COD and MIS Tasks

Method's Variants	CHAMELEON [48]				CVC-ColobNB [49]			
	$M \downarrow$	$F_\beta \uparrow$	$E_\phi \uparrow$	$S_\alpha \uparrow$	$M \downarrow$	$F_\beta \uparrow$	$E_\phi \uparrow$	$S_\alpha \uparrow$
✓	✓	✓	✓	✓	0.052	0.764	0.885	0.816
✓	✓	✓	✓	✓	0.080	0.720	0.833	0.757
✓	✓	✓	✓	✓	0.089	0.683	0.823	0.756
✓	✓	✓	✓	✓	0.061	0.769	0.893	0.815
✓	✓	✓	✓	✓	0.054	0.740	0.884	0.798
✓	✓	✓	✓	✓	<b>0.044</b>	<b>0.790</b>	<b>0.899</b>	<b>0.833</b>
					0.176	<b>0.243</b>	<b>0.583</b>	<b>0.530</b>

Table 6: Parameter ablation study on COD10K [14].

(a) Number of iteration I.	(b) Image preprocess strategy.						(c) Visual marker strategy.								
	Scale	$M \downarrow$	$F_\beta \uparrow$	$E_\phi \uparrow$	$S_\alpha \uparrow$	strategy	$M \downarrow$	$F_\beta \uparrow$	$E_\phi \uparrow$	$S_\alpha \uparrow$	strategy	$M \downarrow$	$F_\beta \uparrow$	$E_\phi \uparrow$	$S_\alpha \uparrow$
1	Original	0.075	0.535	0.750	0.662	None	0.058	0.690	0.855	0.789					
2	Havel	0.069	0.579	0.775	0.689	Bbox	0.065	0.682	0.836	0.766					
3	Quarters	0.087	0.423	0.673	0.586	VCD	0.047	0.705	0.863	0.793					
4	Original+Havel	<b>0.042</b>	<b>0.714</b>	<b>0.875</b>	<b>0.804</b>	Original + Havel + Quarters	0.049	0.702	0.867	0.796	Ours	<b>0.042</b>	<b>0.714</b>	<b>0.875</b>	<b>0.804</b>
5															
6															

Since the second epoch, VCR also operates on different patches, ensuring that while the non-inpainted parts still gather information through hallucinations, the inpainted parts eliminate hallucinations to generate accurate candidate prompts. The promptable segmentation methods is the ViT-H/16 version of SAM. Our experiment is conducted on a single NVIDIA A100 GPU. More details are in appendix.

## 5 Results and Analysis

**Results on COD Task.** The COD tasks focus on finding animals that blend into their complex surroundings. We evaluated ProMaC on three representative datasets: CHAMELEON [48], CAMO [28], and COD10K [14]. As shown in Tab. 1, we compared ProMaC with others that utilize varying levels of supervision. Overall, methods with scribble supervision generally perform better than those with point supervision. Importantly, ProMaC only uses a single generic task prompt for the entire task and it stil outperforms all point-supervised methods. It also surpasses methods with scribble supervision on the CHAMELEON and CAMO datasets, and matches the top-performing scribble-supervised methods on COD10K. It demonstrates the superiority of ProMaC.

**Results on MIS and TOD Task.** The MIS task identifies pathological tissues in medical images. We used three datasets: ColonDB [49] and Kvasir [23] for polyp image segmentation, and ISIC [10] for skin lesion segmentation. We compared our approach with others using task-generic prompt settings (see Tab. 2). While other models underperform in medical imaging due to limited generalization, ProMaC improves significantly over the baseline by iteratively mining task-related knowledge. For the TOD task, we evaluated ProMaC on the GSD [32] and Trans10K-hard [54] datasets (See Tab. 3(a)). Using the task-generic prompt setting, our method achieves the best results despite challenging scenarios. This demonstrates ProMac’s versatility and adaptability across complex visual tasks.

**Results on OVS and SR Task.** We evaluated ProMaC’s effectiveness on the OVS task for multi-class segmentation based on a list of candidate classes. Specifically, we tested it on the validation splits of PASCAL VOC (21 classes) [12, 11], Pascal Context (59 classes) [40], and COCO-Object (80 classes) [3], using LLaVA to identify and confirm the presence of candidate classes. After obtaining masks, we resolved overlaps using the argmax operation based on SAM probabilities. Tab. 3(b) shows how ProMaC compares to other state-of-the-art OVS methods. Unlike some methods trained specifically on these datasets (risking knowledge leaking), ProMaC is not. Yet, ProMaC still outperforms all others on PASCAL VOC and Pascal Context and is competitive on COCO-Object. Additionally, as shown in Tab. 5, we integrated our VCR into LLaVA1.5 for enhanced spatial reasoning. This integration allows LLaVA to better focus on critical areas, thereby boosting performance.

**Module Analysis.** As shown in Tab. 4, we perform an ablation study on the COD and MIS tasks to assess the effects of different modules. "MCoT" is multi-scale chain of thought prompting. "ITP" and "IVP" refer to using only instance-specific text prompts or visual prompts. "VCR" is visual contrastive reasoning, and "MSA" is mask semantic alignment. The first row shows replacing MCoT



Figure 4: Visualization of various segmentation methods among various segmentation tasks.

with just one original image results in reduced performance, highlighting the importance of using hallucinations to extract task-relevant information. The second and third rows show that single modal prompts perform worse than multimodal prompts, highlighting the significance of multimodal prompting. Removing VCR causes a significant drop in performance, indicating that visual prompts are crucial for directing LLaVA’s focus on relevant areas during inference. The comparison between the fifth and final rows emphasizes the importance of mask alignment with task semantics. The consistent positive results across tasks confirm the robustness and effectiveness of our approach.

**Parameter Analysis.** Tab. 6(a) examines how iterations influence performance. "cos" measures the cosine similarity between the predicted text prompt and the ground truth class through CLIP. "IoU" assesses the overlap between the predicted bounding box and the ground truth, comparing it against a rectangular outline of the mask. Mask predictions improve and stabilize after the fourth epoch. Tab. 6(b) investigates the effects of various image processing techniques. "Original" uses no modifications, "Halve" divides the image horizontally or vertically into halves, and "Quarters" divides it into four quarter-sized patches. Testing shows that combining "Original" and "Halve" yields the best results by balancing global and local information without excessive fragmentation.

**Visual Marker Strategy.** Tab. 6(c) assesses the impact of different visual marker strategies. "None" uses no visual prompts, while "Bbox" places bounding boxes directly on the image. "VCD" employs previous methods that introduce Gaussian noise into comparison images for contrastive reasoning. Results indicate that bounding boxes decrease performance, suggesting LLaVA struggles with this type of markers. Although VCD methods improve performance, they distort pixel data, making them less effective than our approach. Our VCR generates contrastive samples that focus on task-relevant areas without altering the image, reducing hallucinations and enhancing performance.

**Visualization.** Fig. 4 visually compares our ProMaC with other methods across 3 tasks and also shows the contrastive images we generated. GenSAM handles clear objects well but struggles with complex background. Although GenSAM performs well in complex backgrounds, but struggles with challenging tasks. ProMaC delivers solid segmentation results across different tasks, and our contrastive images remove task-related regions while maintaining semantic and pixel consistency.

## 6 Conclusion

In this work, we introduce an iterative ProMaC that uses MLLM hallucinations to guide automatic prompt generation, significantly improving segmentation without training. This iterative approach aligns masks with task semantics, enhancing model performance. Testing on multiple benchmarks has demonstrated ProMaC’s effectiveness in a wide range of complex segmentation tasks.

## References

- [1] Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. Exploring visual prompts for adapting large-scale models. *arXiv preprint arXiv:2203.17274*, 2022.
- [2] Yutong Bai, Xinyang Geng, Karttikeya Mangalam, Amir Bar, Alan Yuille, Trevor Darrell, Jitendra Malik, and Alexei A Efros. Sequential modeling enables scalable learning for large vision models. *arXiv preprint arXiv:2312.00785*, 2023.
- [3] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1209–1218, 2018.
- [4] Deng Cai, Yan Wang, Huayang Li, Wai Lam, and Lemao Liu. Neural machine translation with monolingual translation memory. *arXiv preprint arXiv:2105.11269*, 2021.
- [5] Mu Cai, Haotian Liu, Siva Karthik Mustikovela, Gregory P Meyer, Yuning Chai, Dennis Park, and Yong Jae Lee. Making large multimodal models understand arbitrary visual prompts. *arXiv preprint arXiv:2312.00784*, 2023.
- [6] Junbum Cha, Jonghwan Mun, and Byungseok Roh. Learning to generate text-grounded mask for open-world semantic segmentation from only image-text pairs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11165–11174, 2023.
- [7] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3558–3568, 2021.
- [8] Junjie Chen, Li Niu, Liu Liu, and Liqing Zhang. Weak-shot fine-grained classification via similarity transfer. *Advances in Neural Information Processing Systems*, 34:7306–7318, 2021.
- [9] Junjie Chen, Li Niu, Jianfu Zhang, Jianlou Si, Chen Qian, and Liqing Zhang. Amodal instance segmentation via prior-guided expansion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 313–321, 2023.
- [10] Noel Codella, Veronica Rotemberg, Philipp Tschandl, M Emre Celebi, Stephen Dusza, David Gutman, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael Marchetti, et al. Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). *arXiv preprint arXiv:1902.03368*, 2019.
- [11] M Everingham, L Van Gool, CKI Williams, J Winn, and A Zisserman. The pascal visual object classes challenge 2012 (voc2012) results. 2012 <http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html>, 2012.
- [12] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88:303–338, 2010.
- [13] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A new way to evaluate foreground maps. In *Proceedings of the IEEE international conference on computer vision*, pages 4548–4557, 2017.
- [14] Deng-Ping Fan, Ge-Peng Ji, Ming-Ming Cheng, and Ling Shao. Concealed object detection. *IEEE transactions on pattern analysis and machine intelligence*, 44(10):6024–6042, 2021.
- [15] Deng-Ping Fan, Ge-Peng Ji, Xuebin Qin, and Ming-Ming Cheng. Cognitive vision inspired object segmentation metric and loss function. *Scientia Sinica Informationis*, 6(6), 2021.
- [16] Chunming He, Kai Li, Yachao Zhang, Guoxia Xu, Longxiang Tang, Yulun Zhang, Zhenhua Guo, and Xiu Li. Weakly-supervised concealed object segmentation with sam-based pseudo labeling and multi-scale feature grouping. *arXiv preprint arXiv:2305.11003*, 2023.
- [17] Ruozhen He, Qihua Dong, Jiaying Lin, and Rynson WH Lau. Weakly-supervised camouflaged object detection with scribble annotations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 781–789, 2023.
- [18] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.
- [19] Jian Hu, Jiayi Lin, Shaogang Gong, and Weitong Cai. Relax image-specific prompt requirement in sam: A single generic prompt for segmenting camouflaged objects. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 12511–12518, 2024.

- [20] Jian Hu, Hongya Tuo, Chao Wang, Lingfeng Qiao, Haowen Zhong, and Zhongliang Jing. Multi-weight partial domain adaptation. In *BMVC*, page 5, 2019.
- [21] Jian Hu, Hongya Tuo, Chao Wang, Lingfeng Qiao, Haowen Zhong, Junchi Yan, Zhongliang Jing, and Henry Leung. Discriminative partial domain adversarial network. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*, pages 632–648. Springer, 2020.
- [22] Jian Hu, Haowen Zhong, Fei Yang, Shaogang Gong, Guile Wu, and Junchi Yan. Learning unbiased transferability for domain adaptation by uncertainty modeling. In *European Conference on Computer Vision*, pages 223–241. Springer, 2022.
- [23] Debesh Jha, Pia H Smedsrød, Michael A Riegler, Pål Halvorsen, Thomas de Lange, Dag Johansen, and Håvard D Johansen. Kvasir-seg: A segmented polyp dataset. In *MultiMedia Modeling: 26th International Conference, MMM 2020, Daejeon, South Korea, January 5–8, 2020, Proceedings, Part II 26*, pages 451–462. Springer, 2020.
- [24] Wei Ji, Jingjing Li, Qi Bi, Wenbo Li, and Li Cheng. Segment anything is not always perfect: An investigation of sam on different real-world applications. *arXiv preprint arXiv:2304.05750*, 2023.
- [25] Amita Kamath, Jack Hessel, and Kai-Wei Chang. What's " up" with vision-language models? investigating their struggle with spatial reasoning. *arXiv preprint arXiv:2310.19785*, 2023.
- [26] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19113–19122, 2023.
- [27] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023.
- [28] Trung-Nghia Le, Tam V Nguyen, Zhongliang Nie, Minh-Triet Tran, and Akihiro Sugimoto. Anabanch network for camouflaged object segmentation. *Computer vision and image understanding*, 184:45–56, 2019.
- [29] Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. *arXiv preprint arXiv:2311.16922*, 2023.
- [30] Ang Li, Jian Hu, Ke Ding, Xiaolu Zhang, Jun Zhou, Yong He, and Xu Min. Uncertainty-based heterogeneous privileged knowledge distillation for recommendation system. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2471–2475, 2023.
- [31] Yi Li, Hualiang Wang, Yiqun Duan, and Xiaomeng Li. Clip surgery for better explainability with enhancement in open-vocabulary tasks. *arXiv preprint arXiv:2304.05653*, 2023.
- [32] Jiaying Lin, Zebang He, and Rynson WH Lau. Rich context aggregation with reflection prior for glass surface detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13415–13424, 2021.
- [33] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.
- [34] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2023.
- [35] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023.
- [36] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023.
- [37] Huaishao Luo, Junwei Bao, Youzheng Wu, Xiaodong He, and Tianrui Li. Segclip: Patch aggregation with learnable centers for open-vocabulary semantic segmentation. In *International Conference on Machine Learning*, pages 23033–23044. PMLR, 2023.

- [38] Ran Margolin, Lih Zelnik-Manor, and Ayellet Tal. How to evaluate foreground maps? In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 248–255, 2014.
- [39] Shentong Mo and Yapeng Tian. Av-sam: Segment anything model meets audio-visual localization and segmentation. *arXiv:2305.01836*, 2023.
- [40] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *Proceedings of IEEE conference on computer vision and pattern recognition*, pages 891–898, 2014.
- [41] OpenAI. Gpt-4v: Enhancing gpt-4 for visual processing. 2024. Accessed: 2024-05-20.
- [42] Ankur P Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. Totto: A controlled table-to-text generation dataset. *arXiv preprint arXiv:2004.14373*, 2020.
- [43] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.
- [44] Pengzhen Ren, Changlin Li, Hang Xu, Yi Zhu, Guangrun Wang, Jianzhuang Liu, Xiaojun Chang, and Xiaodan Liang. Viewco: Discovering text-supervised segmentation masks via multi-view semantic consistency. *arXiv preprint arXiv:2302.10307*, 2023.
- [45] Guillaume Sanchez, Honglu Fan, Alexander Spangher, Elad Levi, Pawan Sasanka Ammanamchi, and Stella Biderman. Stay on topic with classifier-free guidance. *arXiv preprint arXiv:2306.17806*, 2023.
- [46] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 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*, pages 2556–2565, 2018.
- [47] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. Flava: A foundational language and vision alignment model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15638–15650, 2022.
- [48] Przemysław Skurowski, Hassan Abdulameer, J Błaszczyk, Tomasz Depta, Adam Kornacki, and P Kozięł. Animal camouflage analysis: Chameleon database. *Unpublished manuscript*, 2(6):7, 2018.
- [49] Nima Tajbakhsh, Suryakanth R Gurudu, and Jianming Liang. Automated polyp detection in colonoscopy videos using shape and context information. *IEEE transactions on medical imaging*, 35(2):630–644, 2015.
- [50] Lv Tang, Peng-Tao Jiang, Zhihao Shen, Hao Zhang, Jinwei Chen, and Bo Li. Generalization and hallucination of large vision-language models through a camouflaged lens. *arXiv preprint arXiv:2311.11273*, 2023.
- [51] Bart Thomée, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016.
- [52] David Wan, Jaemin Cho, Elias Stengel-Eskin, and Mohit Bansal. Contrastive region guidance: Improving grounding in vision-language models without training. *arXiv preprint arXiv:2403.02325*, 2024.
- [53] Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*, 2022.
- [54] Enze Xie, Wenjia Wang, Wenhui Wang, Mingyu Ding, Chunhua Shen, and Ping Luo. Segmenting transparent objects in the wild. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pages 696–711. Springer, 2020.

- [55] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18134–18144, 2022.
- [56] Jilan Xu, Junlin Hou, Yuejie Zhang, Rui Feng, Yi Wang, Yu Qiao, and Weidi Xie. Learning open-vocabulary semantic segmentation models from natural language supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2935–2944, 2023.
- [57] Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*, 2023.
- [58] Siyue Yu, Bingfeng Zhang, Jimin Xiao, and Eng Gee Lim. Structure-consistent weakly supervised salient object detection with local saliency coherence. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 3234–3242, 2021.
- [59] Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. Merlot: Multimodal neural script knowledge models. *Advances in Neural Information Processing Systems*, 34:23634–23651, 2021.
- [60] Jing Zhang, Xin Yu, Aixuan Li, Peipei Song, Bowen Liu, and Yuchao Dai. Weakly-supervised salient object detection via scribble annotations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12546–12555, 2020.
- [61] Shizhao Zhang, Hongya Tuo, Jian Hu, and Zhongliang Jing. Domain adaptive yolo for one-stage cross-domain detection. In *Asian conference on machine learning*, pages 785–797. PMLR, 2021.
- [62] Yi-Fan Zhang, Weichen Yu, Qingsong Wen, Xue Wang, Zhang Zhang, Liang Wang, Rong Jin, and Tieniu Tan. Debiasing large visual language models. *arXiv preprint arXiv:2403.05262*, 2024.
- [63] Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, et al. Siren’s song in the ai ocean: a survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219*, 2023.
- [64] Haowen Zhong, Chao Wang, Hongya Tuo, Jian Hu, Lingfeng Qiao, and Zhongliang Jing. Transfer learning based on joint feature matching and adversarial networks. *Journal of Shanghai Jiaotong University (Science)*, 24:699–705, 2019.
- [65] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *European Conference on Computer Vision*, pages 696–712. Springer, 2022.
- [66] Yiyang Zhou, Chenhang Cui, Jaehong Yoon, Linjun Zhang, Zhun Deng, Chelsea Finn, Mohit Bansal, and Huaxiu Yao. Analyzing and mitigating object hallucination in large vision-language models. *arXiv preprint arXiv:2310.00754*, 2023.
- [67] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15116–15127, 2023.
- [68] Xueyan Zou\*, Zi-Yi Dou\*, Jianwei Yang\*, Zhe Gan, Linjie Li, Chunyuan Li, Xiyang Dai, Jianfeng Wang, Lu Yuan, Nanyun Peng, Lijuan Wang, Yong Jae Lee\*, and Jianfeng Gao\*. Generalized decoding for pixel, image and language. 2022.
- [69] 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.