
Instruction-Guided Visual Masking

Jinliang Zheng^{*1,2}, Jianxiong Li^{*1}, Sijie Cheng¹, Yinan Zheng¹,
Jiaming Li¹, Jihao Liu^{3,2}, Yu Liu², Jingjing Liu^{†1}, Xianyuan Zhan^{†1,4}

¹ AIR, Tsinghua University, ² Sensetime Research

³ MMLab, CUHK, ⁴ Shanghai AI Lab

{zhengjl23, li-jx21}@mails.tsinghua.edu.cn

zhanxianyuan@mail.tsinghua.edu.cn

Abstract

Instruction following is crucial in contemporary LLM. However, when extended to multimodal setting, it often suffers from misalignment between specific textual instruction and targeted local region of an image. To achieve more accurate and nuanced multimodal instruction following, we introduce *Instruction-guided Visual Masking* (IVM), a new versatile visual grounding model that is compatible with diverse multimodal models, such as LMM and robot model. By constructing visual masks for instruction-irrelevant regions, IVM-enhanced multimodal models can effectively focus on task-relevant image regions to better align with complex instructions. Specifically, we design a visual masking data generation pipeline and create an IVM-Mix-1M dataset with 1 million image-instruction pairs. We further introduce a new learning technique, *Discriminator Weighted Supervised Learning* (DWSL) for preferential IVM training that prioritizes high-quality data samples. Experimental results on generic multimodal tasks such as VQA and embodied robotic control demonstrate the versatility of IVM, which as a plug-and-play tool, significantly boosts the performance of diverse multimodal models, yielding new state-of-the-art results across challenging multimodal benchmarks. Code, model and data are available at <https://github.com/2to1nf/IVM>.

1 Introduction

Multimodal instruction following is a fundamental multimodal task, powering a wide-range of applications such as visual question answering (VQA) [18], visual captioning [1, 41], and embodied robotic control [14]. To effectively solve this task, one critical capability required is nuanced image-language grounding, which current multimodal models grow implicitly and slowly through data-intensive end-to-end training without explicit grounding supervisions. Two challenges emerge in this indirect learning of image-instruction alignment: 1) How to accurately localize targeted image regions that corresponds to a specific textual instruction, as illustrated in Figure 1. 2) How to generalize to diverse visual representations (*e.g.*, same object with different colors, compositions, or backgrounds) that reflect similar textual instruction (*e.g.*, Q3 in Figure 1). Lacking an effective and direct solution to these challenges, the most advanced Large Multimodal Models (LMMs) [1, 6, 41, 14] still suffer from hallucinations even when trained with high-quality data in the magnitude of billions [34].

We introduce *Instruction-guided Visual Masking* (IVM), a versatile plug-and-play model designed to enhance multimodal instruction following via nuanced surgical visual grounding. To eliminate the distraction of instruction-irrelevant visual regions, IVM automatically masks out these regions to sharpen the focus of instruction following, and meticulously crops visual input to tailor for a specific

^{*}Equal contribution

[†]Corresponding author

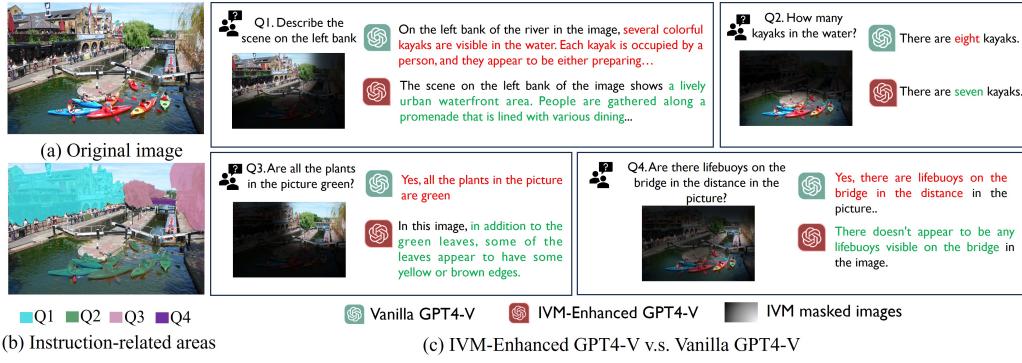


Figure 1: The most advanced LMMs (e.g. GPT4-V) still fail on complex instruction following tasks. With IVM assistance to simplify visual inputs, existing LMMs can gain significant improvement.

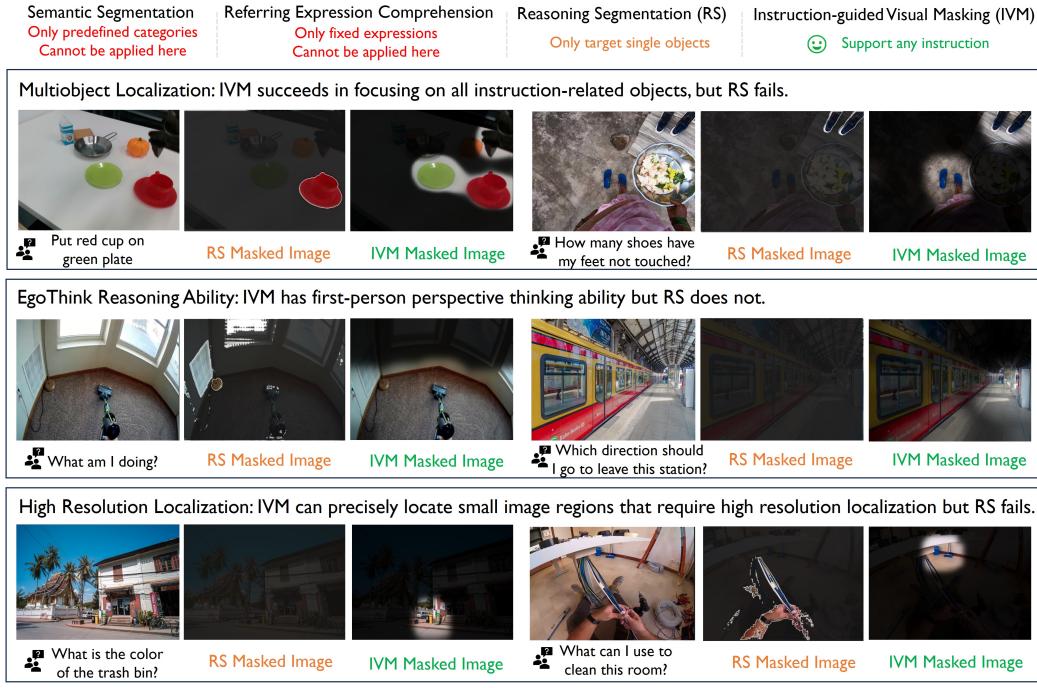


Figure 2: Comparison between IVM and Reasoning Segmentation (RS) [31]. Traditional methods such as semantic segmentation [68] and referring expression comprehension [64] are limited to fixed categories or fixed instruction formats, thus inapplicable to complex instruction following tasks. RS has reasoning ability, but only allows single object localization. IVM, instead, is universally applicable to any instruction.

instruction and enforce multimodal models to zoom in on task-related visual content. Existing visual grounding methods are limited either to predefined object categories, which cannot cover diverse instruction-related visual content; or they subscribe to a fixed instruction format, which restricts the expressiveness of instructions. As shown in Figure 2, such simplistic grounding techniques often fail to comprehend complex instruction-following tasks.

Learning an IVM model requires pixel-level, fine-grained, instruction-guided mask annotations that provide explicit grounding supervisions. To create such a dataset, we build a LLM-empowered Mixture of Expert pipeline with SOTA visual grounding models [52, 50, 31, 20] to efficiently create abundant reliable labels. To compensate the noises in auto-generated labels, we further manually label a smaller dataset with clean annotations, and integrate the two into an IVM-Mix-1M dataset that contains 1 million image-instruction pairs.

To reduce demand on costly human labels and ensure optimized utility of machine-generated labels, we employ a Discriminator-Weighted Supervised Learning (DWSL) framework for IVM training, inspired by recent advances in offline imitation learning [60]. Specifically, we introduce a discriminator to assign weights to masks, where high values are assigned to high-quality annotations and vice versa. Thus, these weights generated by the discriminator can naturally act as a weighting function for the IVM training objective, allowing for a preferential training process that prioritizes learning from reliable samples and discards misleading ones.

Extensive experiments demonstrate great versatility of the IVM model when integrated into existing multimodal chatbots (commercial and open-sourced) without fine-tuning. Our IVM-enhanced LMMs gain significant performance improvement across new challenging benchmarks such as V*Bench [58], EgoThink [10] and POPE [34], achieving new state of the art. IVM model also proves valuable in vision-language robotic manipulation tasks, where data collection is notoriously challenging and generalization is a major concern [35]. With the integration of IVM, our enhanced robot model exhibits boosted performance and better generalization capabilities.

Our contributions are summarized as follows: 1) We propose Instruction-guided Visual Masking (IVM), a novel approach that serves as a versatile plug-and-play module to enhance multimodal models through visual grounding. 2) We introduce the IVM-Mix-1M dataset and propose an LLM-empowered Mixture of Expert pipeline to create visual grounding labels. 3) We present the DWSL algorithm for IVM training that automatically prioritizes high-quality training samples.

2 Related Work

Large Multimodal Models. LLaVA [41] first demonstrates promising capabilities in following complex instructions. Subsequent works such as LLaVA-1.5 [38], MiniGPT4 [69] Qwen-VL [6] and CogVLM [57], further enhance LMMs via refined model design and enriching the quality of training data, achieving state-of-the-art performance on diverse downstream tasks including visual grounding [36], visual reasoning [55], visual question and answering [18]. Moreover, by integrating the robotics action modality, LMMs perform versatile planning and manipulation in instruction-driven robotics tasks. Notable studies in this line of inquiry include PaLM-E [14], the series of RT models [8, 9, 54], and text-guided video planning diffusion models [15, 62, 7]. Despite the success, LMMs still struggle with complex visual grounding challenges, often misreading instruction-irrelevant visual contents (Figure 1). To address this, researchers have tried to adapt existing visual modules to higher-resolution images to obtain better perception [40], but with limited improvement.

Visual Grounding Tasks. Visual grounding requires precisely localizing image regions corresponding to a referring expression, among which the RefCOCO series [64] is the most well-known benchmark, and numerous public visual grounding data are available [36, 63, 19]. Recently, LMMs incorporate these visual grounding data via visual-instruction tuning [41, 38, 69], establishing new SOTA in this area [50]. To further broaden the reasoning ability of visual grounding, LISA [31] introduces a new task, reasoning segmentation, which demands higher capabilities in instruction comprehension. However, visual grounding is still limited to align simple instruction with specific objects, which cannot adapt to more complex instruction following tasks (e.g. Figure 2).

Visual Grounding Augmented LMMs. Recently, a series of visual grounding methods emerged to enhance the performance of LMMs in complex visual scenes. V* [58] employs a heuristic search strategy to search, locate, and crop image areas relevant to instructions through a multi-step iterative process. VisualCot [50] is trained end-to-end with a customized dataset to achieve target localization capabilities. These two methods allow LMMs to dynamically focus on visual inputs until the correct answer is derived. However, these complex inference pipelines lead to substantial computational overhead, and their heuristic designs further hinder the extension beyond VQA to other multimodal instruction following tasks such as robotic control.

Besides these explicit strategies incorporating additional visual grounding modules, other studies pursue refining data or introducing extra training targets to enhance the grounding capabilities of LMMs implicitly. ViGor [61] proposes a fine-grained reward modeling to enhance visual grounding of LMMs, and SynGround [23] introduces a pragmatic framework for image-text-box synthesis tailored for visual grounding. These methods, however, are primarily focused on the visual grounding task itself, overlooking its influence on downstream multimodal instruction following tasks.

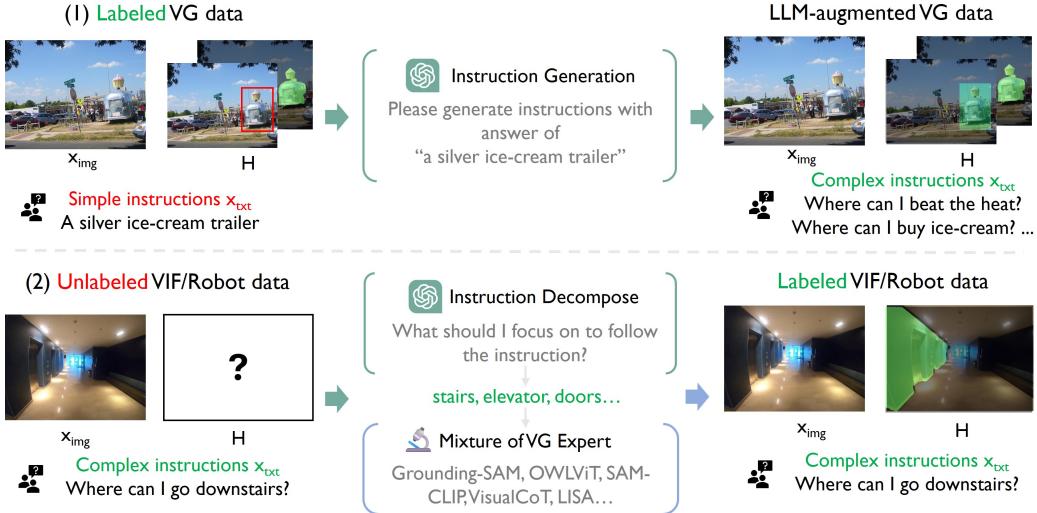


Figure 4: LLM-empowered Mixture-of-Expert pipeline for auto-annotation. (1) For labeled VG data, we utilize an LLM to generate complex instruction annotations. (2) For unlabeled VIF or robot data, we first use an LLM to simplify the instruction and then leverage a mixture of VG models to generate candidate labels.

Distinct from previous efforts, this paper introduces a generic visual grounding model that is adaptable to any multimodal instruction following tasks, and provides a systematic investigation into the advantages of integrating an additional visual grounding model into downstream applications.

3 Instruction-Guided Visual Masking

To help multimodal models focus on instruction-sensitive image regions without distractions from irrelevant visual elements, we introduce Instruction-guided Visual Masking (IVM), a versatile plug-and-play model that enhances multimodal instruction following via surgical targeted visual grounding.

3.1 Problem Definition

IVM aims to produce a heatmap H , given an image x_{img} and a textual instruction x_{txt} . The heatmap H identifies the critical image region to follow the instructions, as illustrated in Figure 3, allowing multimodal models to easily zoom in on targeted image regions while ignoring neighboring areas.

This formulation evokes the problem definition of Reasoning Segmentation (RS) [31]. There are two main differences: 1) IVM addresses a more challenging problem. RS tries to target single objects from simple instructions, *e.g.*, "what is..., where is..., who is...", while IVM aims to include all instruction-related visual regions within the image given any instruction, which demands advanced and nuanced image-language grounding ability (as illustrated in Figure 2). 2) RS has clear ground truths but IVM does not. The instructions in RS primarily correspond to simple and semantic-meaningful objects that are straightforward for human annotations. IVM, however, deals with broader and more ambiguous instruction-related regions (*e.g.*, the left bank regions in Figure 3), making the training and annotating much more challenging.

3.2 Data Preparation

To train an IVM model, the first main challenge is the scarcity of training data. Most existing Visual Grounding (VG) datasets [64, 31] typically feature simple instructions focused primarily on prominent objects within images, lacking both diversity and complexity required for IVM. To tackle this, we



Figure 3: Instruction-guided Visual Masking.

compiled one million data from various sources, including labeled visual grounding, unlabeled multi-modal instruction following, and robotics data. As outlined in Section 3.1, scaling human annotations is challenging due to the high complexity of such data. Therefore, we introduce an *LLM-empowered Mixture of Expert pipeline* that integrates SOTA visual grounding models to efficiently generate reliable annotations. We further manually annotate a smaller dataset to compensate inaccuracies in auto-generated labels. The resulted combined dataset, IVM-Mix-1M, comprises one million data samples ready for IVM training, which can be found in <https://github.com/2toinf/IVM>.

LLM-empowered Mixture of Expert Annotation Pipeline. Leveraging the power of LLM, this pipeline can efficiently generate high-quality annotation, which consists of two components (Figure 4): 1) *Labeled visual grounding data*. We collect 250K labeled VG data from multiple sources including VG caption [36], Flickr30K [63], VSR [3], OpenImage [30], and RefCoCo [64, 37], which provide bounding boxes with simple instructions for each image. To increase the diversity and complexity of instructions, we utilize GPT-4 [1], known for its robust language understanding and generation capabilities, to create diverse instruction-answer pairs based on existing language instructions. 2) *Unlabeled Visual-Instruction-Following (VIF) and robotics data*. We sample a 700K subset from LLaVA-Instruction-tuning [41] for VQA-type data, and a 50K subset from OpenX [54] for robotics data. Given that these data lack grounded labels but contain complex instructions, we use GPT-4 to simplify the language instructions by prompting it to infer the names of targeted objects necessary for following the instructions. These simplified instructions then guide existing VG models to generate candidate labels. To ensure the quality of these labels and compensate for the ambiguous nature of the IVM task, we integrate proposals from several VG experts, such as Grounding-Sam [49], LISA [31], AlphaClip [52], and OwlViT [20], via an ensemble approach.

Manual Annotation. Despite integrating the most advanced models, the auto-generation design still faces challenges that can lead to data inaccuracies. First, employing LLM to simplify or complicate language annotations without considering image content can introduce uncontrollable biases. Second, as the task exceeds the capabilities of existing models, it becomes impossible to totally exclude low-quality annotations that contain irrelevant visuals or mistakenly filter out critical contents. Thus, to enhance the overall quality of the dataset, we further manually annotate a 10K subset of the constructed dataset to inject human expert knowledge.

Data Analysis. Here, we provide quantitative analysis on the IVM-Mix-1M dataset. Figure 5 depicts the data quantities w.r.t the percentage of annotated instruction-related image area. Here, each ratio range is further categorized by different data sources, where manually annotations are treated as a separate category (Human), while all others are machine-generated. Our analysis reveals that the instruction-related image regions only occupy a small fraction of the total image area (*e.g.* most data have less than 40% instruction-relevant image regions), indicating that most visual contents may cause distraction and corroborating the necessity of visual masking for instruction following tasks.

3.3 Discriminator-Weighted Supervised Learning Framework

The challenge now is to train the IVM model with a small high-quality human-annotated dataset (\mathcal{D}_e) as well as a large but mixed-quality auto-generated dataset (\mathcal{D}_o). Training naively on the combined dataset may yield suboptimal results due to inaccuracies in auto-generated labels, while solely using limited human-annotated data is insufficient. Inspired by recent advances in imitation learning using mixed-quality data [60, 65], we employ a Discriminator-Weighted Supervised Learning (DWSL) framework to effectively leverage the strengths of both auto- and human-annotated data.

Discriminator Training. Specifically, we introduce a discriminator d optimized by Eq. (1) to assign high weights to high-quality annotations and vice versa:

$$\min_d \mathbb{E}_{(\mathbf{x}_{\text{img}}, \mathbf{x}_{\text{txt}}, \mathbf{H}) \sim \mathcal{D}_e} [-\log d(\mathbf{x}_{\text{img}}, \mathbf{x}_{\text{txt}}, \mathbf{H})] + \mathbb{E}_{(\mathbf{x}_{\text{img}}, \mathbf{x}_{\text{txt}}, \mathbf{H}) \sim \mathcal{D}_o} [-\log(1 - d(\mathbf{x}_{\text{img}}, \mathbf{x}_{\text{txt}}, \mathbf{H}))], \quad (1)$$

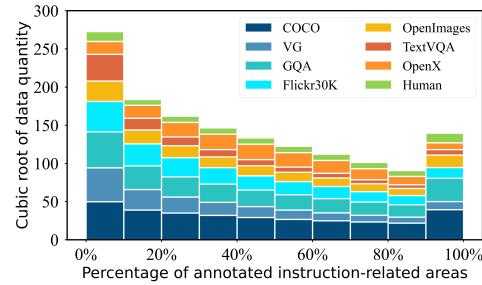


Figure 5: Data analysis on the IVM-Mix-1M dataset: data quantity v.s percentage of instruction-related areas.

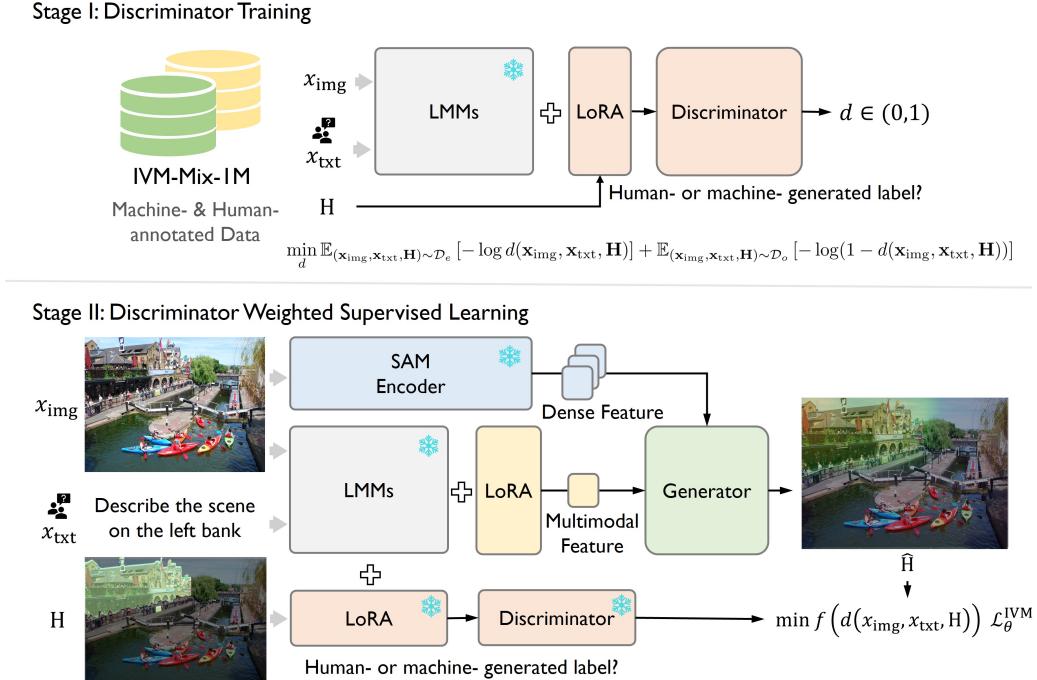


Figure 6: **IVM model architecture and training pipeline.** Stage I: A LoRA-tuned LMMs is trained to discriminate human- and machine-annotated data. Stage II: A frozen SAM vision backbone and a LoRA-tuned LMMs are utilized to extract dense image features and multimodal representations, respectively. These features are then fed into a generator for dense prediction and is trained via DWSL. Same color represents the same model. See Appendix C.1 for more details.

where (x_{img}, x_{txt}, H) are image-instruction-heatmap pairs sampled from \mathcal{D}_o and \mathcal{D}_e datasets. Eq. (1) is similar to the one in GAN [17], but the "fake" data in [17] is replaced by machine-generated data from \mathcal{D}_o . After training with Eq. (1), the discriminator d assigns high weights to high-quality human-annotated data from \mathcal{D}_e and relatively high values to similarly high-quality data from \mathcal{D}_o that aligns with human preferences, acting as a judge for annotation quality.

Discriminator-weighted IVM Training. Then, we apply the trained discriminator as a weighting function for the IVM training objective:

$$\min_\theta \mathbb{E}_{(x_{img}, x_{txt}, H) \sim \mathcal{D}_o \cup \mathcal{D}_e} [f(d(x_{img}, x_{txt}, H)) \mathcal{L}_\theta^{\text{IVM}}(x_{img}, x_{txt}, H)], \quad (2)$$

$$\mathcal{L}_\theta^{\text{IVM}}(x_{img}, x_{txt}, H) = \lambda_{\text{bce}} \mathbf{BCE}(\hat{H}_\theta, H) + \lambda_{\text{dice}} \mathbf{DICE}(\hat{H}_\theta, H), \quad (3)$$

where λ_{bce} and λ_{dice} are set to 1.0 and 1.0 to balance the binary cross-entropy loss (**BCE**) and the DICE loss for segmentation (**DICE**) [28], respectively. $f(x) \geq 0$ can be any non-negative, non-decreasing function. For simplicity, we set $f(x) := \min(\max(0.1, x), 1)$. This allows the weighting function $f(d(\cdot))$ in Eq. (2) to dynamically prioritize training with high-quality data determined by the discriminator d . This approach maximizes the usage of reliable annotations in \mathcal{D}_o to compensate for the small \mathcal{D}_e , while minimizing the impact of low-quality data in \mathcal{D}_o , thus optimizing performance.

3.4 Model Architecture

The overall model framework is illustrated in Figure 6. Due to its complexity, IVM requires both reasoning and precise localization of the target object, closely paralleling reasoning segmentation [31]. Consequently, for the heatmap generator, we adopt a model design similar to that of LISA [31]. Specifically, we first extract dense image features using an isolated vision backbone and multimodal representation from an LMM, which processes image-instruction pairs. These two types of features are then fed into a lightweight generator that integrates them to produce a dense prediction.

For the discriminator, we deploy a lightweight discriminator that encodes the segmentation map using a two-layer convolution network. This discriminator interacts with the outputs of the LMMs through multiple cross-attention operators and finally outputs a quality score for each sample.

Trainable Parameters. To enhance training efficiency, we freeze the pre-trained large foundation models and perform LoRA finetuning [25]. The vision backbone, inherited from *Segment Anything Model* [29], is completely frozen, while the lightweight generator and discriminator are fully finetuned. Notably, we utilize a shared LMM for both the generator and discriminator branches but employing separate LoRA parameters to avoid interference between the two tasks.

4 Experiments

In this work, we employ *LLaVA-7B* [42] as the LMM and *SAM-H* [29] as the vision backbone for our IVM model (Figure 6), which is trained on the IVM-Mix-1M dataset using the proposed DWSL algorithm. More details on the architecture and training can be found in Appendix C. We conduct extensive experiments to assess the effectiveness of the IVM model. Specifically, we utilize the heatmap generated by the IVM for image post-processing. These processed images can then be seamlessly fed into downstream multimodal models for diverse tasks, as shown in Figure 7. Unless otherwise specified, we use the image post-processing method of overlaying and cropping to discard instruction-irrelevant image content. A detailed discussion on post-processing methods is presented in Section 4.2. We also provide more evaluation results and analysis in Appendix E.

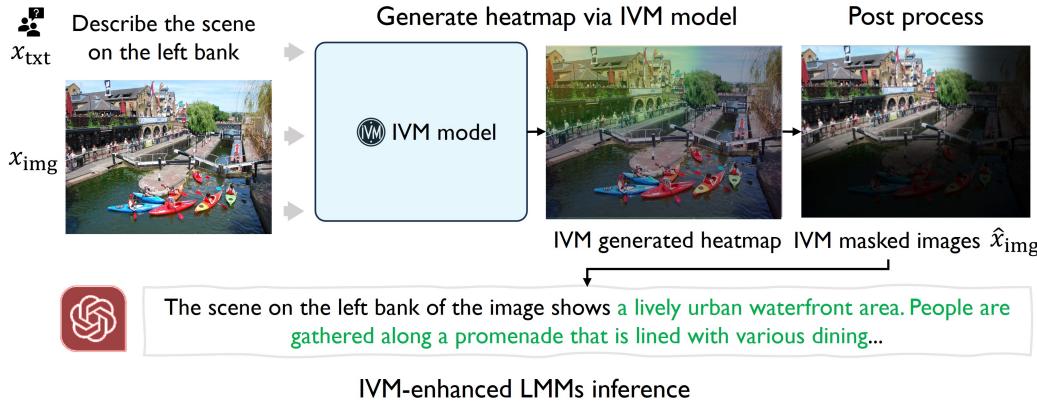


Figure 7: **IVM inference pipeline.** IVM generates heatmap given a pair of image and instruction. Then, instruction-irrelevant visual areas are masked out via post process methods. LMMs can correctly follow the instruction given the masked images.

Table 1: V* bench results.

LMMs	Attribute(%)	Spatial(%)	Overall(%)
<i>Open-Sourced LMMs</i>			
BLIP2 [33]	27.0	53.9	37.7
MiniGPT-4 [69]	30.4	50.0	38.2
InstructBLIP [12]	25.2	47.4	34.0
Otter [32]	27.0	56.6	38.7
LLaVA-1.5 [38]	43.5	56.6	48.7
<i>Commercial Chatbots</i>			
Bard [45]	31.3	46.1	37.2
Gemini-Pro [53]	40.9	59.2	48.2
GPT4-V [1]	51.3	60.5	55.0
<i>Specific Visual Search Models</i>			
SEAL [58]	74.8 (+23.5)	76.3 (+15.8)	75.4 (+20.4)
IVM-Enhanced GPT4-V	87.0 (+35.7)	72.4 (+11.9)	81.2 (+26.2)

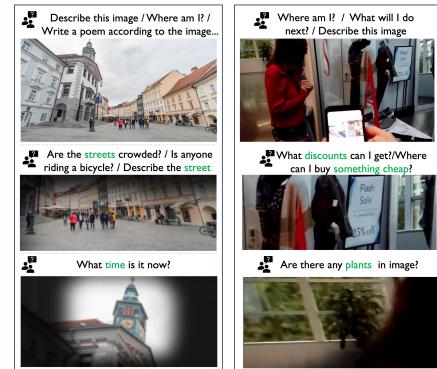


Figure 8: IVM can handle various instructions, ranging from retaining entire images for captioning (row 1) to localizing unique objects (row 2 and 3).

Table 2: Results on other multimodal benchmarks. MME* denotes the aggregate of scores from -p and -c.

LMMs	#Param	EgoThink	POPE	MME*	GQA	SQA	VQAv2
InstructBLIP [12]	13B	-	78.9	1212.8	49.5	60.5	-
Qwen-7B [6]	7B	-	-	-	58.3	67.1	78.8
SEAL-7B [58]	7B	-	82.4	1129	-	-	-
LLaVA-7B [38]	7B	51.1	85.9	1748	62.0	70.2	78.5
LLaVA-13B [38]	13B	55.2	85.9	1834	67.1	71.6	80.0
LISA [31]-Enhanced LLaVA-7B	20B	47.9 (-3.2)	80.0 (-5.9)	1560 (-188)	56.6 (-5.4)	69.3 (-0.9)	78.2 (-0.3)
IVM-Enhanced LLaVA-7B	14B	54.5 (+3.4)	87.2 (+1.3)	1806 (+58)	62.2 (+0.2)	70.2 (-)	79.0 (+0.5)

4.1 Main Results

Integration with Commercial Chatbot. We use GPT4-V [1] as the base model. Considering the superior perception and reasoning capability of GPT4-V, we evaluate IVM-enhanced GPT4-V on V*bench [58], a recently proposed challenging VQA-type benchmark characterized by images with abundant redundancies. Results are presented in Table 1. The accuracy of the vanilla GPT4-V is mediocre (55.0%). Our IVM model, however, can significantly improve the performance (+26.2%) and establish a new state of the art on this benchmark, even surpassing the task-specialized SEAL [58] that requires a complex heuristic visual search pipeline.

Integration with Open-sourced LMMs. To demonstrate the versatility of our IVM model, we further integrate it into an open-sourced LMM, LLaVA-7B [38]. We conduct extensive experiments across various benchmarks, including EgoThink [10], POPE [34], MME [16], GQA [27], SQA [44], and VQAv2 [18]. As shown in Table 2, our IVM-enhanced LLaVA-7B gains consistent performance improvements, achieving comparable performance to (even surpassing) LLaVA-13B on EgoThink, POPE and MME. Although IVM-enhanced LLaVA-7B and LLaVA-13B [38] have roughly the same number of parameters, the latter integrates more powerful pretrained foundation models. In contrast, our IVM model allows the 7B model to outperform the 13B model by merely simplifying visual input, further validating the power of visual masking.

Meanwhile, IVM-enhanced LLaVA-7B does not show significant gains on GQA, SQA and VQAv2, which is expected, as these benchmarks do not heavily rely on grounding capabilities: VQAv2 and GQA contain relatively simple visual input where most regions of the images are instruction-relevant, while SQA primarily focuses on assessing model reasoning capability.

Comparison with Reasoning Segmentation Model. We also compare against LISA [31], which is most analogous to IVM. We provide carefully tailored prompts like *"what should we focus on the image to follow the given instruction? Give me the seg"* to extend LISA into visual masking task. However, even with larger 13B model and extensive tuning of input prompt, masks generated by LISA consistently result in severe performance degradation on all tasks.

Evaluation on Real Robotic Control. We also plug the IVM model into robot control tasks to help robot model improve generalization. Specifically, we evaluate a language-conditioned behavior cloning (LCBC) robot agent trained with or without IVM masked images. Figure 9 clearly demonstrates that without IVM assistance, the LCBC robot agent suffers from severe performance drop when noticeable distractions are applied. With IVM assistance, however, the agent consistently pays close attention to correct instruction-related image regions, performing robustly against diverse distractions such as human disturbances and numerous task-irrelevant objects of various colors and shapes. This demonstrates promising potentials of using IVM to enhance embodied agents to follow complex instructions in unseen scenes with plenty distractions.

4.2 Ablation

We ablate the key components of IVM and report overall accuracy improvement(%) of IVM-Enhanced GPT4-V, evaluated on V* bench [58] due to its high demand on precise visual grounding abilities.

Training Data. We investigate the impact of IVM-Mix data characteristics on IVM performance from two key perspectives: 1) Large machine-annotated data volume clearly enhances IVM model performance, as illustrated by the progressive improvement in Figure 10 (a) with increased machine-annotated data volume (red, blue and yellow line). This demonstrates the effectiveness of our proposed LLM-empowered Mixture-of-Expert pipeline in generating reliable data for IVM training. 2) Figure 10 (a) also reveals that incorporating human annotations significantly boosts training

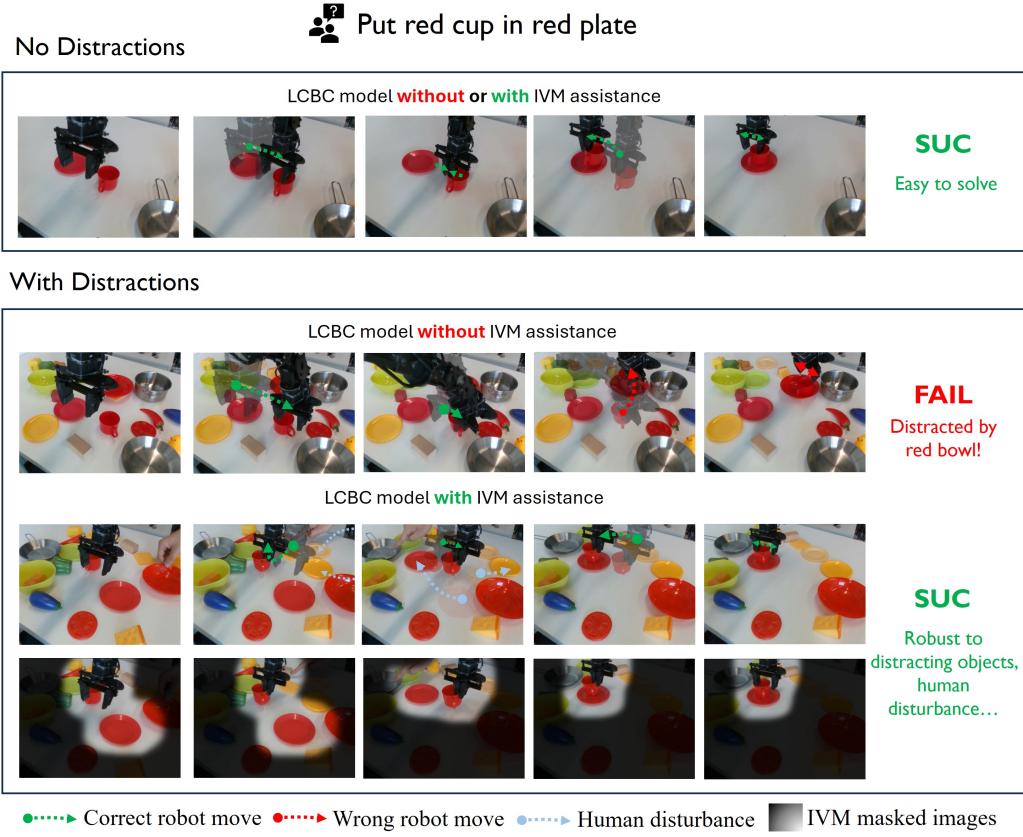
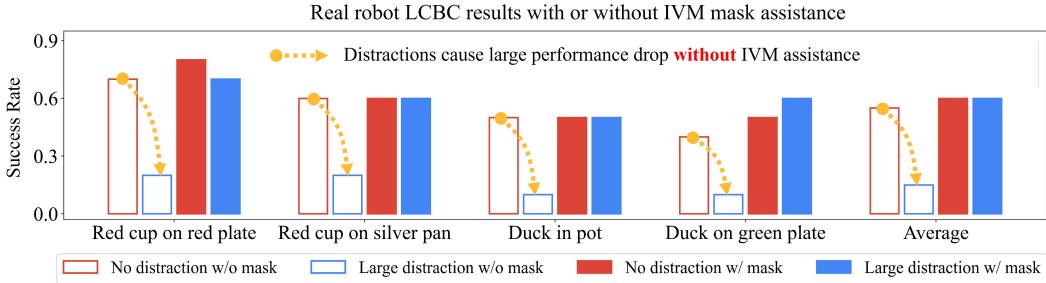
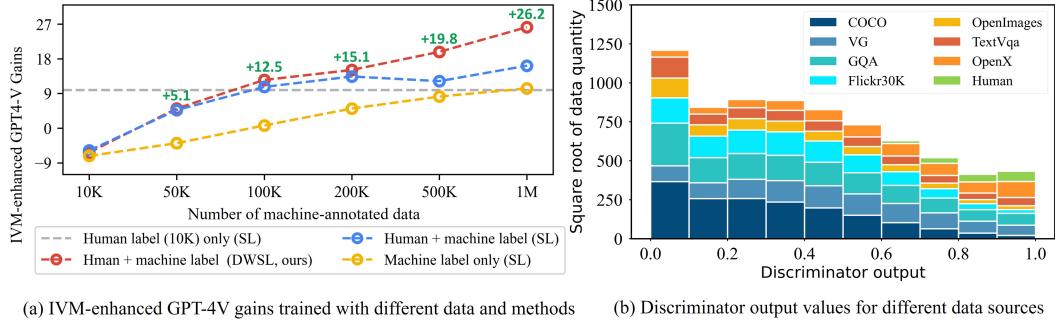


Figure 9: Real robot results with or without IVM assistance. IVM greatly helps LCBC agent to overcome major distractions, enjoying better robustness and generalization. See Appendix C.4 for experiment setups.

efficiency (red and blue *v.s* yellow line), highlighting the critical role of introducing human preferences in IVM-Mix-1M dataset, despite its relatively small volume compared to machine-annotated data (only 1:100).

DWSL Framework. We also explore the efficacy of the DWSL framework in Figure 10 (a) by comparing IVM training using: 1) DWSL (red line), 2) traditional Supervised Learning (SL) without DWSL (blue line), and 3) SL on limited human data (gray line). The results demonstrate that DWSL effectively leverages both human- and auto-annotated data, particularly as the volume of machine-annotated data increases, enjoying higher asymptotic performances. This is expected as machine-annotated data often contain inaccuracies and training naively using all these data can lead to suboptimal results. Meanwhile, the limited human data alone cannot provide satisfactory outcomes. DWSL, however, addresses these challenges by dynamically prioritizing good samples and discarding misleading ones, resulting in stable and improved results. This is further illustrated in Figure 10 (b) which visualizes the outputs of the discriminator for each sample, where the discriminator can correctly retain good samples (e.g. Human) and filter out low-quality data with lower weights.



(a) IVM-enhanced GPT-4V gains trained with different data and methods (b) Discriminator output values for different data sources

Figure 10: Ablations on training data and the proposed DWSL framework.

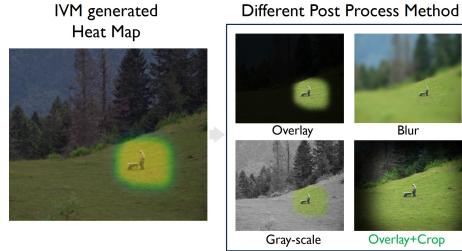


Figure 11: Different mask deployment methods.

Table 3: Ablations on different mask deployment methods on the V*bench.

	Overlay	Blur	Gray-scale
w/ crop	+26.2	+24.4	+22.1
w/o	+19.1	+17.2	+10.2

Mask Deployment Strategy. We investigate the impact of mask deployment strategy on downstream applications. While more complex solutions such as visual search algorithms [58] can be employed, our investigation focuses solely on simpler approaches to understand the intrinsic capabilities of IVM model. Specifically, we examine four basic masking methods: overlay, blur, grayscale, and cropping, as illustrated in Figure 11. In particular, for the crop method, we find the smallest area that retains all the activated (>0) values in the heatmap and crop it. Table 3 demonstrates that IVM maintains robustness across all simple post-processing methods, where overlay+crop enjoys the most performance enhancement and thus is used as our default mask deployment method.

5 Conclusion

We introduce Instruction-guided Visual Masking (IVM), a generic and powerful visual grounding method that enhances broad multimodal instruction following tasks in a plug-and-play way. By masking out all instruction-irrelevant image regions, IVM effectively injects superior visual grounding ability to downstream LMMs non-intrusively, significantly boosting both commercial and open-sourced LMMs and achieving state-of-the-art results across numerous challenging multimodal benchmarks. Real robot experiments further demonstrate the versatility of IVM, showcasing the potential to deploy IVM to embodied robotic tasks where failures caused by distractions are long-standing challenges. For further improvement, one promising direction is to finetune LMMs using IVM-generated heatmap as an additional input channel to reduce suboptimal heuristics caused by mask deployment methods. Due to resource limitation, we leave this for future work. We open source the IVM checkpoint and the IVM-Mix-1M dataset to help the community further explore relevant directions*. More discussion on limitations and future directions can be found in Appendix A.

6 Acknowledgements

The paper is supported by funding from Wuxi Research Institute of Applied Technologies, Tsinghua University under Grant 20242001120. The authors would like to thank the anonymous reviewers for their feedback on the manuscripts.

*<https://github.com/2toinf/IVM>

References

- [1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.
- [2] Anurag Ajay, Yilun Du, Abhi Gupta, Joshua Tenenbaum, Tommi Jaakkola, and Pulkit Agrawal. Is conditional generative modeling all you need for decision-making? *arXiv preprint arXiv:2211.15657*, 2022.
- [3] Jean-Baptiste Alayrac, Joao Carreira, and Andrew Zisserman. The visual centrifuge: Model-free layered video representations. In *CVPR*, 2019.
- [4] Reza Yazdani Aminabadi, Samyam Rajbhandari, Minjia Zhang, Ammar Ahmad Awan, Cheng Li, Du Li, Elton Zheng, Jeff Rasley, Shaden Smith, Olatunji Ruwase, and Yuxiong He. Deepspeed inference: Enabling efficient inference of transformer models at unprecedented scale, 2022.
- [5] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.
- [6] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023.
- [7] Kevin Black, Mitsuhiro Nakamoto, Pranav Atreya, Homer Rich Walke, Chelsea Finn, Aviral Kumar, and Sergey Levine. Zero-shot robotic manipulation with pre-trained image-editing diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024.
- [8] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.
- [9] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.
- [10] Sijie Cheng, Zhicheng Guo, Jingwen Wu, Kechen Fang, Peng Li, Huaping Liu, and Yang Liu. Egothink: Evaluating first-person perspective thinking capability of vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14291–14302, June 2024.
- [11] Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *arXiv preprint arXiv:2303.04137*, 2023.
- [12] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Albert Li, Pascale Fung, and Steven C. H. Hoi. Instructclip: Towards general-purpose vision-language models with instruction tuning. *ArXiv*, abs/2305.06500, 2023.
- [13] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
- [14] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023.
- [15] Yilun Du, Sherry Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Joshua B. Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[16] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui Yang, Xiawu Zheng, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023.

[17] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[18] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913, 2017.

[19] Agrim Gupta, Piotr Dollár, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation, 2019.

[20] Akshita Gupta, Sanath Narayan, KJ Joseph, Salman Khan, Fahad Shahbaz Khan, and Mubarak Shah. Ow-detr: Open-world detection transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9235–9244, 2022.

[21] Philippe Hansen-Estruch, Ilya Kostrikov, Michael Janner, Jakub Grudzien Kuba, and Sergey Levine. Idql: Implicit q-learning as an actor-critic method with diffusion policies. *arXiv preprint arXiv:2304.10573*, 2023.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[23] Ruozhen He, Paola Cascante-Bonilla, Ziyuan Yang, Alexander C. Berg, and Vicente Ordonez. Learning from models and data for visual grounding, 2024.

[24] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[25] 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. In *International Conference on Learning Representations*, 2022.

[26] Xiao Hu, Jianxiong Li, Xianyuan Zhan, Qing-Shan Jia, and Ya-Qin Zhang. Query-policy misalignment in preference-based reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024.

[27] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*, 2019.

[28] Shruti Jadon. A survey of loss functions for semantic segmentation. In *2020 IEEE conference on computational intelligence in bioinformatics and computational biology (CIBCB)*, pages 1–7. IEEE, 2020.

[29] 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.

[30] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*, 128(7):1956–1981, March 2020.

[31] Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model, 2024.

[32] Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Jingkang Yang, and Ziwei Liu. Otter: A multi-modal model with in-context instruction tuning. *arXiv preprint arXiv:2305.03726*, 2023.

[33] 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 preprint arXiv:2301.12597*, 2023.

[34] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023.

[35] Jianxiong Li, Jinliang Zheng, Yinan Zheng, Liyuan Mao, Xiao Hu, Sijie Cheng, Haoyi Niu, Jihao Liu, Yu Liu, Jingjing Liu, Ya-Qin Zhang, and Xianyuan Zhan. DecisionNCE: Embodied multimodal representations via implicit preference learning. In *Forty-first International Conference on Machine Learning*, 2024.

[36] Yuanzhi Liang, Yalong Bai, Wei Zhang, Xueming Qian, Li Zhu, and Tao Mei. Vrr-vg: Refocusing visually-relevant relationships, 2019.

[37] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[38] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023.

[39] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding dino: Marrying dino with grounded pre-training for open-set object detection, 2023.

[40] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, January 2024.

[41] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024.

[42] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024.

[43] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2017.

[44] Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering, 2022.

[45] James Manyika and Sissie Hsiao. An overview of bard: an early experiment with generative ai. *AI. Google Static Documents*, 2, 2023.

[46] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

[47] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[48] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

[49] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. Grounded sam: Assembling open-world models for diverse visual tasks, 2024.

[50] Hao Shao, Shengju Qian, Han Xiao, Guanglu Song, Zhuofan Zong, Letian Wang, Yu Liu, and Hongsheng Li. Visual cot: Unleashing chain-of-thought reasoning in multi-modal language models, 2024.

[51] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

[52] Zeyi Sun, Ye Fang, Tong Wu, Pan Zhang, Yuhang Zang, Shu Kong, Yuanjun Xiong, Dahua Lin, and Jiaqi Wang. Alpha-clip: A clip model focusing on wherever you want, 2023.

[53] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[54] Open X Team. Open x-embodiment: Robotic learning datasets and RT-x models. In *Towards Generalist Robots: Learning Paradigms for Scalable Skill Acquisition @ CoRL2023*, 2023.

[55] Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality, 2022.

[56] Homer Rich Walke, Kevin Black, Tony Z Zhao, Quan Vuong, Chongyi Zheng, Philippe Hansen-Estruch, Andre Wang He, Vivek Myers, Moo Jin Kim, Max Du, et al. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning*, pages 1723–1736. PMLR, 2023.

[57] Weihan Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang, Lei Zhao, Xixuan Song, et al. Cogvlm: Visual expert for pretrained language models. *arXiv preprint arXiv:2311.03079*, 2023.

[58] Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal llms, 2023.

[59] Tsung-Han Wu, Giscard Biamby, David Chan, Lisa Dunlap, Ritwik Gupta, Xudong Wang, Joseph E. Gonzalez, and Trevor Darrell. See, say, and segment: Teaching llms to overcome false premises, 2023.

[60] Haoran Xu, Xianyuan Zhan, Honglei Yin, and Huiling Qin. Discriminator-weighted offline imitation learning from suboptimal demonstrations. In *International Conference on Machine Learning*, pages 24725–24742. PMLR, 2022.

[61] Siming Yan, Min Bai, Weifeng Chen, Xiong Zhou, Qixing Huang, and Li Erran Li. Vigor: Improving visual grounding of large vision language models with fine-grained reward modeling, 2024.

[62] Sherry Yang, Yilun Du, Seyed Kamyar Seyed Ghasemipour, Jonathan Tompson, Leslie Pack Kaelbling, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. In *The Twelfth International Conference on Learning Representations*, 2024.

[63] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014.

[64] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part II 14*, pages 69–85. Springer, 2016.

[65] Wenjia Zhang, Haoran Xu, Haoyi Niu, Peng Cheng, Ming Li, Heming Zhang, Guyue Zhou, and Xianyuan Zhan. Discriminator-guided model-based offline imitation learning. In *Conference on Robot Learning*, pages 1266–1276. PMLR, 2023.

[66] Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.

[67] Yinan Zheng, Jianxiong Li, Dongjie Yu, Yujie Yang, Shengbo Eben Li, Xianyuan Zhan, and Jingjing Liu. Safe offline reinforcement learning with feasibility-guided diffusion model. *arXiv preprint arXiv:2401.10700*, 2024.

[68] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017.

[69] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.

A Limitation and Future Work

Here, we discuss our limitations, potential solutions and interesting future works.

1. **Computational Overhead.** Note that IVM introduces additional parameters and computational overhead to directly enhance visual grounding ability of LMMs, which in turn indirectly improve the VQA performances. However, more VQA performance gains can be obtained if the same amount of additional parameters are end-to-end trained directly on VQA data (LLaVA-13B v.s IVM-Enhanced LLaVa-7B in Table 2).

Solution and future work: Nevertheless, this is quite reasonable because IVM primarily focuses on improving the visual grounding ability, but accurate VQA also requires other abilities which can be learned through end-to-end training. End-to-end training, however, requires tremendous VQA data to implicitly and slowly improve the visual grounding ability, which is quite data-intensive. Both Table 1 and Figure 1 can show that even trained on billions of data, GPT4-V still performs subpar on tasks that require strong visual grounding ability. IVM, instead, can significantly boost the visual grounding ability of GPT4-V using just 7B parameters and less computations. One promising and interesting future direction is to include some auxiliary tasks to directly absorb the strong visual grounding ability in the IVM-Mix-1M dataset through end-to-end training like [59].

2. **Data Quality.** Due to task complexity, the machine-annotated data in IVM-Mix-1M inevitably includes wrong labels that mistakenly exclude instruction-sensitive image regions or suboptimal labels that not fully mask out all instruction-irrelevant areas. These inaccuracies may lead to suboptimal IVM model. We propose a DWSL framework to tackle this. However, the DWSL framework relies on a learned discriminator and a human-designed $f(x)$ function, which may not exclude all inaccuracies.

Solution and future work: We have clearly demonstrated in Figure 10 that with a simple $f(x)$ and a lightweight discriminator, DWSL consistently outperforms the naive Supervised Learning (SL), doing pretty well on prioritizing good samples and meanwhile identifying inaccurate labels. To further enhance this, one can use other advanced techniques such as Reinforcement Learning from Human Feedback (RLHF) [46, 5, 26] to provide more fine-grained judgement on annotation qualities, or resort a theoretical-soundness $f(x)$ [60] to achieve better results. In addition, one can also use our pretrained IVM model to directly generate high-quality heatmaps to enhance the machine annotations.

3. **Mask Deployment Methods.** In this paper, we directly use the simple post-processing method to apply the IVM generated heatmaps on images, which then are fed into LMMs to perform downstream tasks. However, these post-processing methods introduce some heuristics, which may be suboptimal for downstream LMMs. In addition, LMMs may not see many masked images during pretraining, thus some distributional shift may occur.

Solution and future work: Although these limitations exist, IVM still obtain consistent improvements using diverse mask deployment methods, as shown in Table 3, which showcases the great versatility of IVM to inject visual grounding abilities. To further improve this, one strategy is employ some task-specialized visual search method [58], but will bring many computational load during inference and limit the versatility on embodied agents. Another promising direction is directly using the IVM generated heatmaps as an additional input channel to finetune the LMMs like [52], which can fully eliminate the heuristics of post-process methods, may bring larger performance gains. Due to resources limits, we leave this for a future work.

4. **Fine-grained Heatmaps.** Note that the IVM generated heatmaps cannot provide exact semantic object segmentation with clear contours like reasoning segmentation [31] offers.

Discussions: We want to clarify that this is an advantage of the IVM model rather than a limitation. This is because of the ambiguous nature of the visual masking task. For this task, the ground truth heatmaps are mostly less semantic-meaningful for annotations as discussed in 3.1. So, we ensemble the annotation proposals from different visual grounding methods for data annotation, which will make the trained IVM model robust to include instruction-relevant image areas, rather than being aggressive to exclude some instruction-sensitive pixels like reasoning segmentation [31] does illustrated in Figure 2.

Overall, although some limitation exist, we have thoroughly discussed potential solutions to these limitations. Moreover, in this paper, we have demonstrated the superior effectiveness and versatility of IVM to directly inject strong visual grounding ability to downstream LMMs or embodied agents, representing a pioneer effort to extend traditional visual grounding methods towards a more complex and generic setting that covers diverse multimodal instruction following tasks.

B Broader Impact

This paper aims to advance the field of artificial intelligence, where no significant negative social impact is observed in this paper. The IVM-Mix-1M may contain some potential privacy issues and biases. However, in this paper, nearly all data are collected from open-sourced data, which have been well peer-reviewed, thus resolved this ethical concern.

C Training and Evaluation Details

C.1 Architecture Details

In this section, we primarily focus on the architectural design of the lightweight generator and discriminator, as both the Language Model Multitask (LMM) and the vision backbone are derived from the powerful foundation models (LLaVA & SAM). Both the generator and discriminator utilize the same transformer-based decoder block, as depicted in Figure 12. We employ two such blocks for both the generator and discriminator. Specifically, the generator produces dense predictions by upscaling the output features of the decoder block through a straightforward upsampling operation. In contrast, the discriminator first employs a two-layer convolutional downsampling network to encode segmentation labels. This network, in conjunction with the decoder block and a simple MLP (multi-layer perceptron) head, outputs the weights.

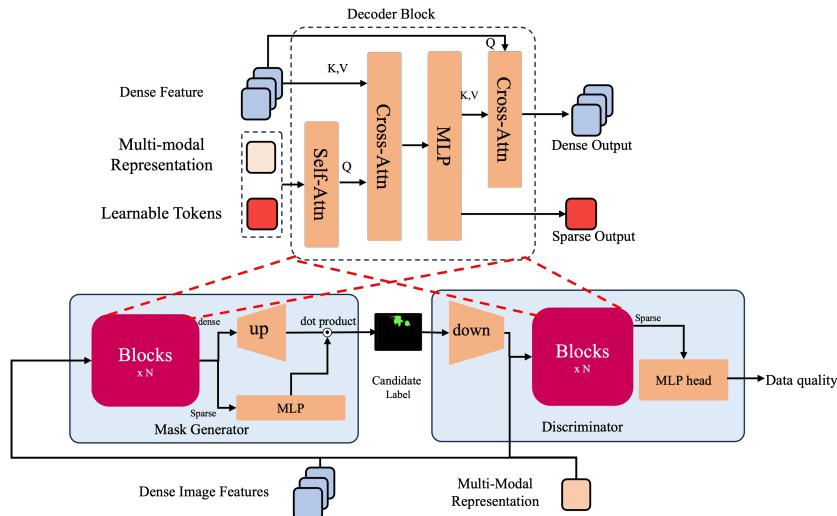


Figure 12: Generator/Discriminator Architecture Details

C.2 Training Details

We adopt 8 NVIDIA 80G A100 GPUs and take 4 days to train our IVM model. The training scripts are based on deepspeed [4] engine and the training hyperparameters can be found in Table 4.

C.3 Multimodal Benchmarks Evaluations

We evaluate our IVM on diverse multimodal benchmarks, including general VQA (VQAv2 [18], GQA [27], MME [16]), first-person perspective QA (EgoThink [10]), scientific QA (SQA [44]),

Table 4: Hyper-parameters for pretraining.

config	value
training iteration	200K
optimizer	AdamW [43]
learning rate	1×10^{-5}
batch size	32
weight decay	0
optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.95$
data augmentation	<i>RandomCropResize</i>

hallucination adversarial QA (POPE [34]) and V* [58], a recently proposed challenging benchmark with high-resolution and complex visual input.

Our evaluation employs a two-stage inference pipeline: the image is firstly simplified by IVM-generated heatmap and mask deployment methods; Subsequently, the simplified image is fed into downstream LMMs(GPT4-V [1], LLaVA [38]) without finetuning. We adhere to the official procedures of each benchmark to evaluate the output of LMMs and report the results.

C.4 Real Robot Evaluations

Task descriptions. The real robot experiments evaluate several pick and place manipulation tasks that require strong visual grounding abilities. Specifically, we evaluate on 4 tasks as shown in Table 5, following the task definitions in DecisionNCE [35]. For each task, we collect around 100 demonstrations using the demonstration collection system in BridgedataV2 [56]. We take both a side camera view and a wrist camera view as the vision inputs, as shown in Figure 13. For each demonstration, the environmental steps are around 50 steps. During data collection, the object and robot locations are randomly initialized, and the scene also has lots of randomly located distractors with varied shape and color.

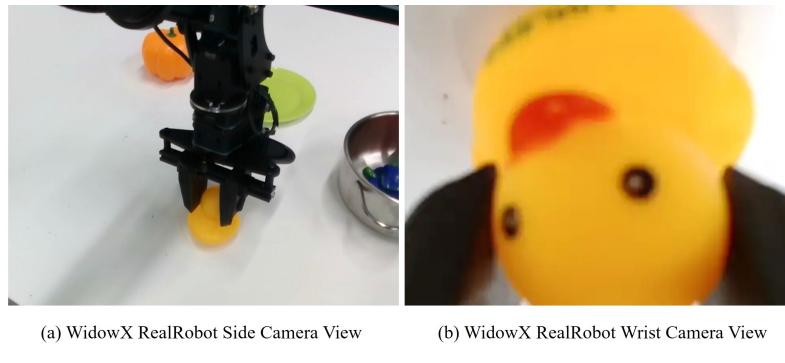


Figure 13: Visual input view for LCBC policy.

Table 5: Real Robot Tasks

Environment ID	Language Instruction
Red cup on silver pan	Pick up the red cup and place it on the silver pan
Red cup on red plate	Pick up the red cup and place it on the red plate
Duck on green plate	Pick up the duck and place it on the green plate
Duck in pot	Pick up the duck and place it in the pot

Training details. Here, we train Language-Conditioned Behavior Cloning (LCBC) policies using DDPM [24] loss since diffusion policies are good at fitting complex data distributions [67, 56, 2], especially human demonstrations [11]. For model architecture, the side and wrist images are augmented and then passed through a shared ResNet50 [22] image encoder and get an image

embedding for each camera view, following [56]. As the downstream data is quite limited, we load the ImageNet [13]-pretrained ResNet50 image encoder and further train it on the small robot data. Meanwhile, the language instruction is passed through a frozen T5 text-encoder [48], which is fused into the image encoder via Film conditioning layers [47]. Then, this language-conditioned image embedding is passed through a MLPs with residual connections similar to IDQL [21], which then outputs the predicted noise in DDPM [24]. To obtain smoothed policy rollouts, we adopt Action Chunking and Temporal Ensemble from ACT [66] with a chunking size 4 rather than 100 in [66] because the episode horizons in this paper are only around 50. The LCBC policies are trained either on the original side camera view (without IVM assistance) or on the IVM-masked side camera view (with IVM assistance) for 200K steps with a batch size of 64. The training can be completed on 2 NVIDIA RTX4090 GPU in 17h. All hyperparameters are summarized in Table 6.

Table 6: Real robot LCBC training details

Backbones	
Visual encoder	Resnet50 [22] (ImageNet [13] pretrained)
text encoder	T5 [48] (frozen)
DDPM hyperparameters	
noise schedule	VP [51]
denoising time steps	25
Other hyperparameters	
Chunking size	4
Optimizer	AdamW [43]
Learning rate	1e-4
Lr schedule	cosine annealing
Warm up steps	2000
Batch size	64
Gradient Steps	200K
Augmentation	Yes [56]

Evaluation details. We first evaluate the trained LCBC policies without strong distractions, where no or only small distractors appear in the image. Then, we add lots of distracting objects with varied shapes and colors, and even introduce strong human disturbance to attack the LCBC policies. For each score reported in Figure 9, we evaluate 10 episodes and report the success rates.

D Mixture of Expert Annotation Pipeline

D.1 Labeled Visual Grounding data

For labeled visual grounding data, We provide the following prompt to drive GPT-4 [1] to generate more complex instructions based on given language annotations.

[Image Description] %s

[System] You are an AI visual assistant, and you are seeing a single image. What you see are part of the image and are provided with a simple phrase. Please generate any instructions that can be executed based on the content of the picture described, including simple queries about the content of the picture, such as the object types, counting the objects, object actions, relative positions between objects, etc. Also consider more complex questions that require reasoning. For example, you can ask what time it is now for a clock and what can I use to clean the room for a broom. Ensure that the questions you ask can be clearly answered only based on what you see. Please generate as many five questions as possible and return them in a single line separated by ';' and avoid any other output.

D.2 Unlabeled Visual Instruction Following Data

For unlabeled visual instruction following data, we first try to simplify complex instructions. Specifically, we employ GPT-4 to infer the necessary object for executing the given instructions based on these instructions and a simple image caption. If the dataset lacks captions, they can be generated using an existing caption model like BLIP-Caption [33]. Below, we outline the prompts specifically designed for GPT-4.

[Image Caption] %s

[Instruction] %s

[System] You are an helpful AI assistant. I need to reply to the previous instruction based on an image, and I have a simple caption for the image. Please note that there may be objects in the image that I did not detect. Since you cannot view the image, please list any potential objects that might influence my responses, separated by semicolons, in a single line without any additional output. If you believe that the number of objects could be too extensive and might hinder my judgment, print 'None'.

With the simplified instruction, we can adopt existing visual grounding models to generate the candidate label. Specifically, we utilize four models: AlphaCLIP [52], LISA [31], OwVIT [20] and Grounding-SAM [49] and the inference pipelines are provided in the official implementation of these models.

E More result

E.1 Referring Expression Comprehension

As IVM is an extension of traditional visual grounding task, we also evaluate our IVM on RefCoCo, RefCoCo+ and RefCoCog [64]. We reported the accuracy (IOU-50%) on the validation split in Table 7. As a generalist model capable of handling complex instructions, our IVM achieves performance comparable to that of state-of-the-art (SOTA) specialist models.

Table 7: result in REC

Methods	RefCoCo	RefCoCo+	RefCoCog
<i>Specialist models</i>			
G-DINO-L [39]	90.56	82.75	86.13
<i>Generalist models</i>			
LLaVA-7B [41]	76.29	66.76	70.4
IVM(Ours)	90.1	83.3	82.9

E.2 Visualization Result

In this section, we provide more visualization result in VQA-type data as shown in Figure 14.

Failure Case. Although we observe numerous successful instances, our IVM still faces significant challenges, as illustrated in Figure 15. We summarize these challenges into three categories: missing target, misguided target, and insufficient reasoning.

(a) **Missing Target:** Challenges arise when target objects are relatively small and scattered around many separate image corners. In this case, accurately detecting all of the targeted objects is quite difficult. Even specialized open vocabulary detection models struggle with this task. For example, the cup on the right in the image is masked by the IVM mistakenly. However, we still observe that the IVM-generated heatmap for the right cup is partially activated, meaning that IVM have partially focus this regions. We believe by providing more training data, IVM can handle this better.



Figure 14: Visualization results of IVM generated masks.

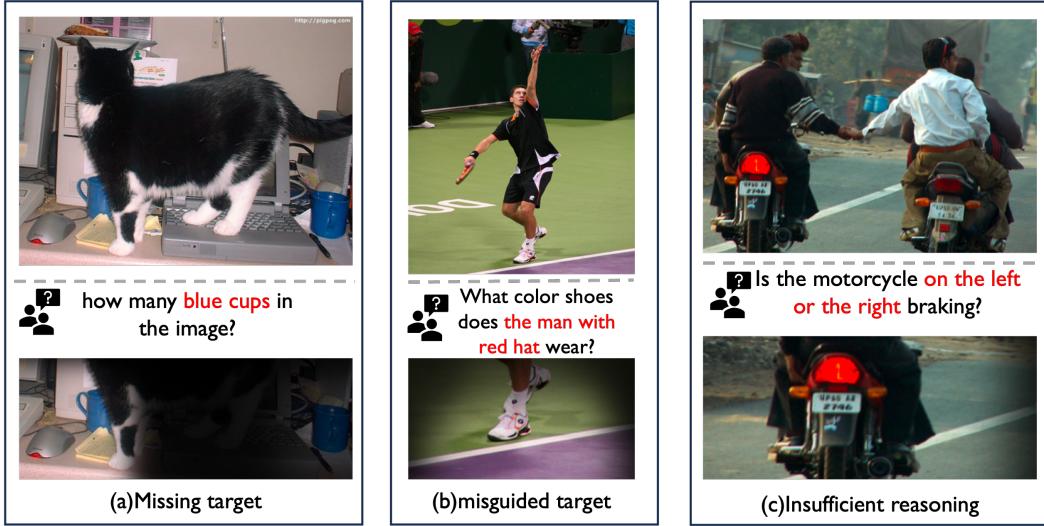


Figure 15: Some failure cases.

(b) **Misguided Target:** Accurately Localizing tiny target objects is a recognized challenge [58], especially when similar but more obvious objects are present. For instance, IVM incorrectly focuses on the more centrally located shoes of another man, instead of the shoes of the man wearing the **red hat** at the edge of the picture. However, this instruction is pretty challenging that at first glance, even a human might struggle to spot the man with the red hat in the left corner. We will leave challenge scenarios like this for future research.

(c) **Insufficient reasoning:** The objective of the IVM task is to assist LMMs in extracting visual features more effectively to better follow instructions. Thus, the demands on the model's reasoning capabilities extend far beyond mere object localization. Although IVM demonstrates strong performance, it sometimes overlooks additional image content necessary for accurately following instructions after correctly locating the target object. For instance, while IVM successfully identified the braking motorcycle, it failed to recognize that answering the question requires knowledge of the positions of both motorcycles simultaneously. We attribute this issue to biases in the training data. By incorporating more complex instructions and diversified labels, we anticipate that our model will achieve improved performance.

E.3 Robotics Result

Here, we provide more evaluation rollouts of the IVM-assisted LCBC agents under strong distractions. Figure 16 clearly demonstrates that even under strong distractions like the background are full of distracting objects with similar colors or shapes to the targeted objects, and strong human disturbances that adversarially attack the robots, the IVM-assisted LCBC agents can still complete the tasks pretty well, enjoying high-level of generalization and robustness thanks to the superior visual grounding ability injected by IVM. More videos can be found in the supplementary materials.

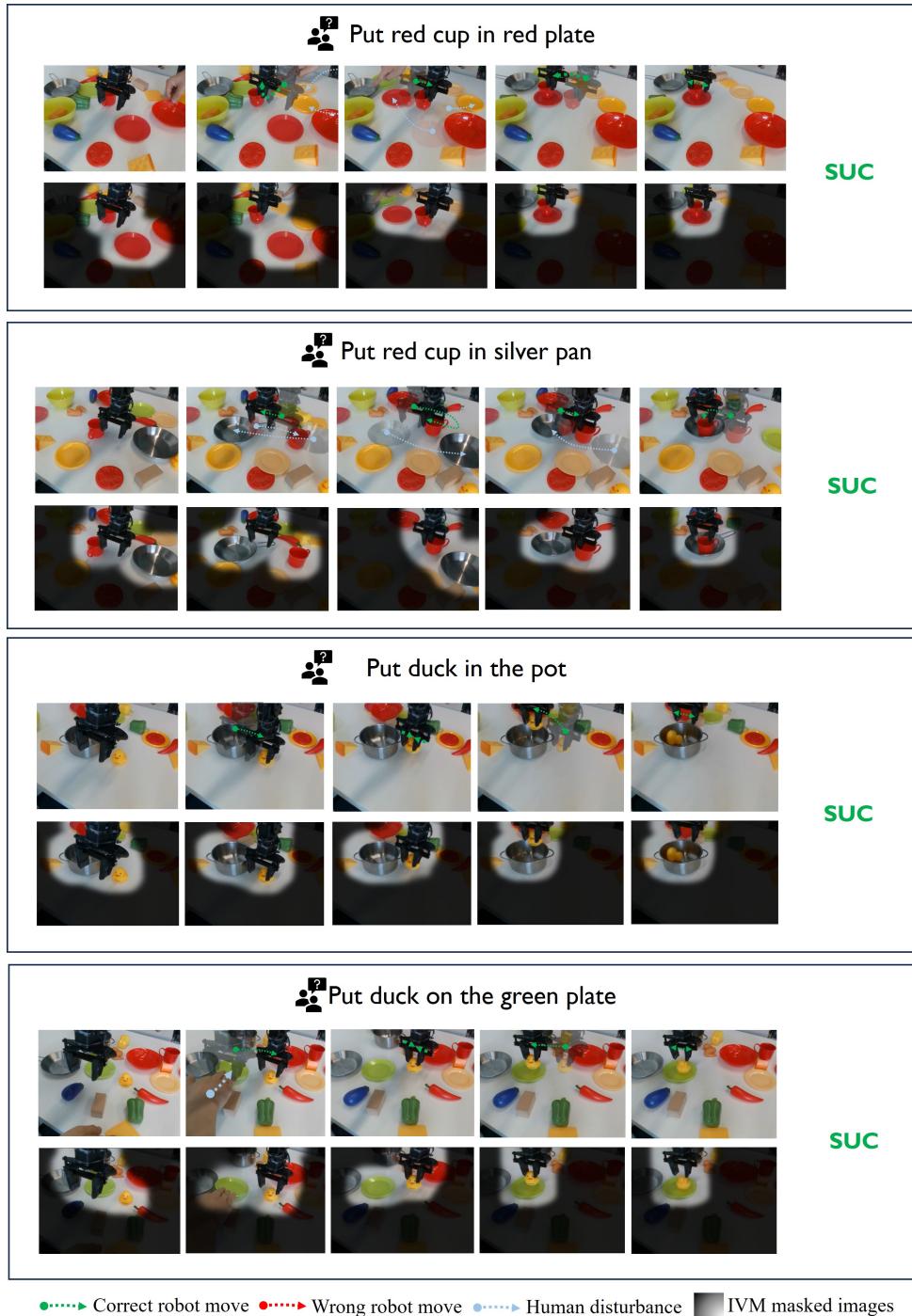


Figure 16: Real robot LCBC results with IVM assistance.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: Please see Abstract and Introduction for details.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: Please see Conclusion and Appendix A for details.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory Assumptions and Proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[NA\]](#)

Justification: This paper has no theory.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental Result Reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Please see Appendix C for details.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Code, model and data are available at <https://github.com/2to1inf/IVM>.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental Setting/Details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Please see Appendix C for details.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment Statistical Significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: Error bars are not reported because it would be too computationally expensive.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments Compute Resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [\[Yes\]](#)

Justification: Please see Appendix C for details.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code Of Ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [\[Yes\]](#)

Justification: N/A

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader Impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [\[Yes\]](#)

Justification: Please see Appendix B for details.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [\[Yes\]](#)

Justification: All data are collected from open-sourced and peer-reviewed dataset. The models used for annotations are also open-sourced and peer-reviewed.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [\[Yes\]](#)

Justification: N/A

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New Assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: We will release our IVM-Mix-1M dataset and detailed documentations after the acceptance.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [Yes]

Justification: All crowdsourcing labels in the IVM-Mix-1M dataset are annotated by the authors.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: This paper has no human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.