LION : Empowering Multimodal Large Language Model with Dual-Level Visual Knowledge

Gongwei Chen, Leyang Shen, Rui Shao[†], Xiang Deng, Liqiang Nie[†]
School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen
{chengongwei, shaorui, dengxiang, nieliqiang}@hit.edu.cn
https://github.com/rshaojimmy/JiuTian

Abstract

*Multimodal Large Language Models (MLLMs) have endowed LLMs with the ability to perceive and understand multi-modal signals. However, most of the existing MLLMs mainly adopt vision encoders pretrained on coarsely aligned image-text pairs, leading to insufficient extraction and reasoning of visual knowledge. To address this issue, we devise a dual-**L**evel v**I**sual kn**O**wledge e**N**hanced Multimodal Large Language Model (**LION**), which empowers the MLLM by injecting visual knowledge in two levels. **1) Progressive incorporation of fine-grained spatial-aware visual knowledge.** We design a vision aggregator cooperated with region-level vision-language (VL) tasks to incorporate fine-grained spatial-aware visual knowledge into the MLLM. To alleviate the conflict between image-level and region-level VL tasks during incorporation, we devise a dedicated stage-wise instruction-tuning strategy with mixture-of-adapters. This progressive incorporation scheme contributes to the mutual promotion between these two kinds of VL tasks. **2) Soft prompting of high-level semantic visual evidence.** We facilitate the MLLM with high-level semantic visual evidence by leveraging diverse image tags. To mitigate the potential influence caused by imperfect predicted tags, we propose a soft prompting method by embedding a learnable token into the tailored text instruction. Comprehensive experiments on several multi-modal benchmarks demonstrate the superiority of our model (e.g., improvement of 5% accuracy on VSR and 3% CIDEr on TextCaps over InstructBLIP, 5% accuracy on RefCOCOg over Kosmos-2).*

1. Introduction

Recently, Large Language Models (LLMs) have demonstrated remarkable zero-shot abilities on various linguistic tasks. Assisted by LLMs, several multimodal large language models (MLLMs), such as MiniGPT-4 [54], Otter [22], and InstructBLIP [7], achieve significant improvements in reasoning abilities to deal with various vision-language (VL) tasks.

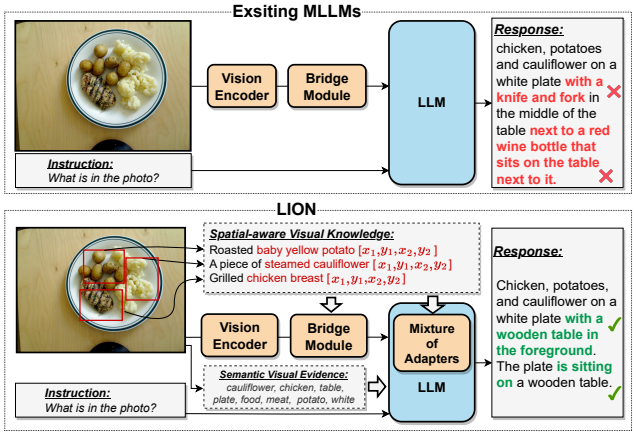

Figure 1. Comparison between existing MLLMs and LION. The existing MLLM generates a vague and inaccurate response, while LION provides a more precise and contextually accurate description by progressively incorporating spatial-aware knowledge and softly prompting semantic visual evidence.

In most of the existing MLLMs, the visual information is mainly extracted from a vision encoder pretrained with image-level supervision (*e.g.*, CLIP [36]), and then are adapted to a LLM by using a tiny bridge module. This makes these MLLMs inherently possess limited image understanding capabilities [24]. As shown in Fig. 1, the insufficient visual information misleads MLLMs to provide erroneous and hallucinated responses. An intuitive solution to this problem is to replace or tune the vision encoder [41]. However, it requires pretraining on massive data or suffers from the catastrophic forgetting issue [50], which diminishes the practical efficacy of this strategy. These predicaments highlight that the insufficient extraction of visual knowledge has become a central obstacle impeding the

[†]Corresponding authors

development of MLLMs.

To overcome this dilemma, as depicted in Fig. 1, we devise a dual-**L**evel v**I**sual kn**O**wledge e**N**hanced Multimodal Large Language Model (**LION**), which enriches the visual information in MLLMs in two levels. ***1) Progressive incorporation of fine-grained spatial-aware visual knowledge.*** LION enhances the MLLM with more fine-grained perceptual abilities by studying the region-level VL tasks involving the spatial coordinates. However, we find that simply training on region-level and the original image-level VL tasks* simultaneously hurts the general performances of the MLLM due to the conflicts between these two kinds of tasks. To address this issue, we propose a novel stage-wise instruction-tuning strategy to perform image-level and region-level VL tasks separately with two different visual branches and task adapters. In addition, we devise mixture-of-adapters with a router to dynamically fuse visual information across various granularities in a unified MLLM. This progressive incorporation of fine-grained visual knowledge contributes to the mutual promotion between these two kinds of VL tasks, and spawns LION to excel in capturing fine-grained visual information and performing spatial reasoning, as shown in Fig. 1. ***2) Soft prompting of high-level semantic visual evidence.*** Alongside the improvement of MLLMs in fine-grained perceptual capabilities, there is also an opportunity to enhance their high-level semantic understanding. LION uses an off-the-shelf vision model to extract high-level semantic knowledge, *i.e.*, image tags, as supplementary information for the MLLM. However, as off-the-shelf vision models are typically not flawless, errors in tag predictions are inevitable. Inspired by prompt tuning, we propose a soft prompting method to mitigate the potential negative influence resulting from the imperfect predicted tags. As shown in Fig. 1, injection of semantic visual evidence alleviates the hallucination issue substantially.

Our main contributions are summarized as follows:

- To address the internal conflict between region-level and image-level VL tasks, we propose a progressive incorporation of fine-grained spatial-aware visual knowledge with a novel stage-wise instruction-tuning strategy. It achieves mutual promotion between two kinds of VL tasks and equips LION with advanced holistic and fine-grained visual perceptual abilities.
- As a powerful complement, we propose to integrate image tags as high-level semantic visual evidence into MLLMs, and design a soft prompting method to alleviate the bad influence from incorrect tags. This mitigates the hallucination issue and showcases positive effects on various VL tasks.
- We evaluate LION on a wide range of VL tasks, including image captioning, visual question answering (VQA), and visual grounding, and demonstrate its superiority over the baselines as illustrated in Fig. 2. LION outperforms InstructBLIP by around 5% accuracy on VSR, and around 3% CIDEr on TextCaps, Kosmos-2 by around 5% accuracy on RefCOCOg. The evaluations on POPE and MM-Bench exhibit the remarkable abilities of LION in alleviating object hallucination and various perceptual dimensions.

*Here, image-level VL tasks denote image captioning and visual question answering, region-level VL tasks mean visual grounding tasks.

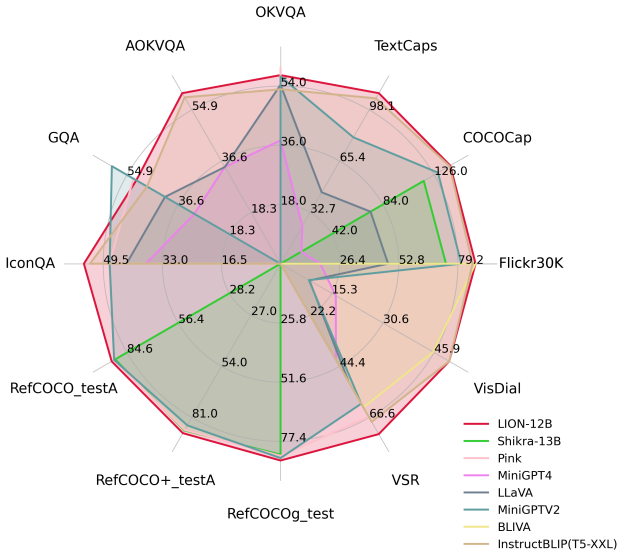

Figure 2. Compared to recently proposed MLLMs, LION achieves state-of-the-art performances across a wide range of VL tasks.

2. Related Work

2.1. Multimodal Large Language Models

Building on the success of LLMs, many researches have emerged to extend them to multimodal tasks, especially VL tasks. The common pipeline uses a vision model to transform the image into visual features, followed by a bridge module to align them with the feature space of LLMs. Some works [4, 13, 29, 35, 51] directly use a linear or MLP layer as bridge module, while others [1, 7, 22, 23, 46, 54] design more complicated bridge networks to compress or adaptively select visual information. Despite their impressive performance on VL tasks, there is still a lack of exploration on the effectiveness and limitation of the visual branch in a MLLM. Recently, Wang *et al.* [41] empirically investigate factors contributing to the formation of an effective vision encoder in a MLLM from the perspective of pretraining. Differently, our work explores the effect of region-level VL tasks on the visual understanding abilities of the MLLM, and incorporates fine-grained and high-level visual knowledge to enrich the visual branch in the MLLM.

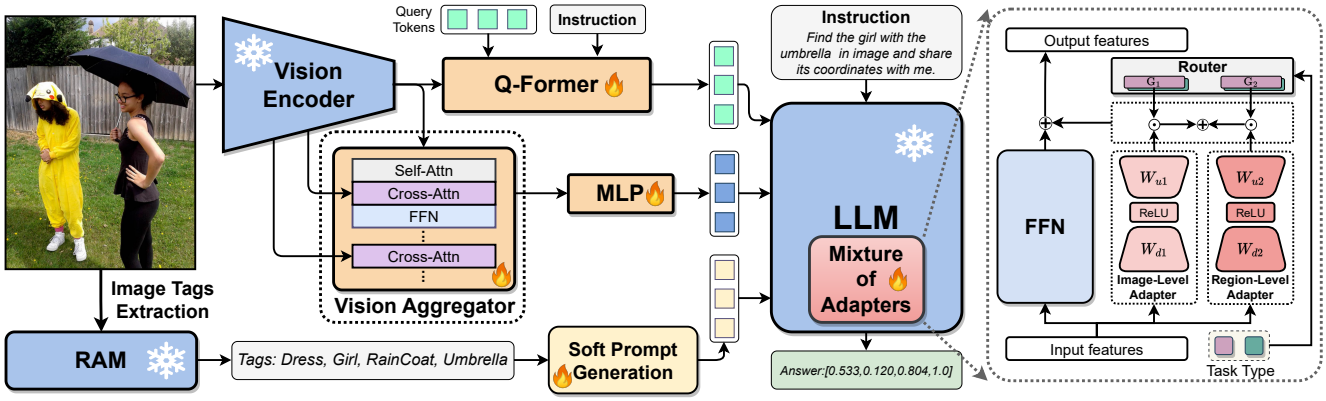

Figure 3. Overview of the proposed LION. The model extracts holistic visual features from Q-Former, and combines them with fine-grained spatial-aware visual features from the vision aggregator. The Mixture-of-Adapters with a router in the frozen LLM dynamically fuses visual knowledge learned from different visual branches and LLM adapters based on the task types (image-level and region-level).

2.2. Visual Grounding in the field of MLLMs

Visual grounding is a region-level VL task that aims to establish a connection between particular regions and their textual descriptors, which plays a vital role in human-machine interaction by enabling referential dialog. In the realm of MLLMs, there are some attempts to enhance MLLMs by leveraging visual grounding tasks. Works like Shikra [4], Kosmos-2 [35], Ferret [47] and Pink [45] demonstrate the promising direction of employing visual grounding datasets to endow MLLMs with region-level visual understanding abilities. They convert existing datasets equipped with spatial coordinates, like Visual Genome [20] and RefCOCO [19], into the textual instruction format and perform instruction tuning on MLLMs. Merely considering the visual grounding task as one of several instruction-tuning tasks, these works fall short in exploring the interactions among various tasks. In contrast, our work investigates the internal conflict between visual grounding tasks and image-level VL tasks (*e.g.*, image captioning and VQA), and proposes a stage-wise instruction-tuning strategy to address this issue, achieving a good balance between these two kinds of VL tasks.

3. LION

In this section, we present the dual-Level vIsual knOwlEdge eNhanced multimodal large language model (LION). The proposed LION aims to enrich the visual information that is fed to the LLM in two ways, *i.e., progressive incorporation of fine-grained spatial-aware visual knowledge* and *soft prompting of high-level semantic visual evidence*. The whole framework is depicted in Fig. 3.

3.1. Progressive Incorporation of Fine-grained Spatial-Aware Visual Knowledge

3.1.1 Reorganizing Visual Grounding Tasks

To incorporate fine-grained spatial-aware visual knowledge into MLLMs, we make use of region-level VL tasks, *i.e.*, visual grounding, and meticulously process the data with spatial coordinates in a unified format for instruction-tuning MLLM. Visual grounding requires the model to generate or comprehend natural language expressions referring to particular objects or regions within an image, *e.g.*, "a man with glasses". Referring to objects or regions in complex images needs an ability of precisely comprehending fine-grained visual information. Current MLLMs lack such referring comprehending, as they mainly target a coarse alignment of VL modalities when pretrained on massive image-text pairs [2, 41]. In this regard, we introduce visual grounding tasks as a kind of region-level VL tasks for the instruction-tuning of MLLMs. This aims to endow the model with fine-grained visual understanding ability such that better performance on image-level VL tasks (*e.g.*, image captioning and VQA) might be achieved.

We adopt two types of visual grounding tasks, including referring expression comprehension (REC) and referring expression generation (REG) [49]. We use the Visual Genome dataset [21], which associates a local area with one short description, to construct REC/REG tasks. The templates used to organize the Visual Genome dataset in a unified instruction-tuning format can be found in **Appendix**.

One core point in reorganizing visual grounding tasks is the way of processing positions. Normally, the position of an object phrase is presented in the format of bounding box $[x_{min}, y_{min}, x_{max}, y_{max}]$. We use a natural language style to describe object positions along with the square brackets. A sample in the REC task is displayed as follows: "How

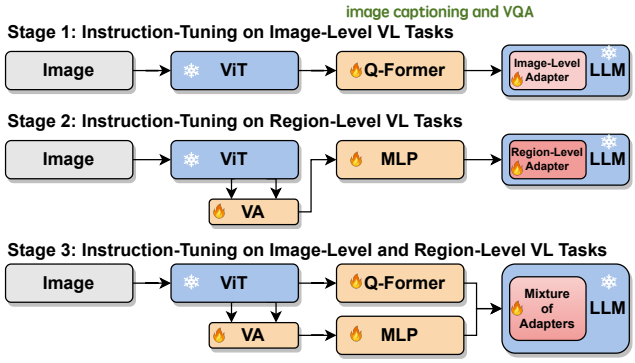

Figure 4. The stage-wise instruction-tuning strategy. **Stage 1**: We instruction-tune Q-Former and the image-level adapter on image-level VL tasks. **Stage 2**: We instruction-tune the vision aggregator (VA), MLP, and the region-level adapter on region-level VL tasks. **Stage 3**: The Mixture-of-Adapters is devised to form a unified model for instruction-tuning on both kinds of VL tasks.

can I locate a glass of beer in the image? Please provide the coordinates. Answer: $[0.525, 0.0, 0.675, 0.394]$".

3.1.2 The Stage-Wise Instruction-tuning Strategy

To facilitate MLLMs with fine-grained spatial-aware knowledge, the most intuitive way is to directly instruction-tune MLLMs with both image-level and region-level VL tasks in one stage. However, this single-stage instruction-tuning strategy is sub-optimal, and suffers from the internal conflict between these two kinds of VL tasks. We summarize two main issues leading to the internal conflict. **1)** One is the need of region-level modality-alignment pretraining. In concurrent works that integrate visual grounding ability, pretraining on the region-level multimodal datasets including visual grounding is a crucial step. Some works [35, 45, 47] elaborately create very large visual grounding datasets (*e.g.*, GRIT-20M in kosmos-2 [35]) to advance MLLM in fine-grained perception and understanding. The single-stage instruction-tuning makes it challenging to adapt visual representations learned for image-level alignment to region-level VL tasks under a limited training configuration. **2)** Another is the gap between the input-output modes of image-level VL tasks and region-level visual grounding tasks. The latter additionally requires MLLMs to understand specific phrases (in the format "$[x_{min}, y_{min}, x_{max}, y_{max}]$") about the positions of objects. They are semantically distinct from natural languages used in image-level tasks. This requirement necessitates the tuning of the LLM to adapt to region-level tasks, but may disrupt the internal state of the LLM suitable for image-level VL tasks. To address the above issues, we devise a stage-wise instruction-tuning strategy and mixture-of-adapters with a router.

The stage-wise instruction-tuning strategy is proposed to alleviate the internal conflict between image-level and region-level VL tasks during instruction-tuning. It is composed of three stages for instruction-tuning on image-level, region-level VL tasks and both, respectively, which is depicted in Fig. 4. In stage 1, we follow instructBLIP [7] and fine-tune Q-Former and the image-level adapter [5] in the LLM on image-level VL tasks, such as image captioning and VQA. In stage 2, we propose a vision aggregator for better capturing visual features in fine-grained understanding, which will be introduced later, and tune it with MLP and the region-level adapter on region-level VL tasks. The independent training in the first two stages greatly fulfills the requirements of sufficiently learning both image-level and region-level tasks, providing a solid foundation for subsequent joint training.

Mixture-of-Adapters with a Router. In stage 3 of our stage-wise instruction-tuning, we need a unified model but encounter a situation where adapters of LLM in stages 1 and 2 are different and suit distinct input-output modes. Inspired by Mixture-of-Experts, we treat each adapter as an expert, and propose a router module to avoid the potential interference between them, as depicted in Fig. 3.

An adapter [5] is inserted at each FFN layer in a parallel manner. Assuming $X \in \mathcal{R}^{L\times D}$ is the hidden representations generated by a self-attention / causal attention layer, the output representations after FFN (represented as $\mathbf{F}$) with the adapter (denoted by $\mathbf{H}$) layer are formulated as,

$$O = \mathbf{F}(X) + \mathbf{H}(X), \tag{1}$$

$$\mathbf{H}(X) = W_u(\sigma(W_d X)), \tag{2}$$

where σ is a non-linear function, ReLU. Our router module aims to dynamically aggregate the hidden features from the main branches and the multiple adapter branches according to task types. Given a set of adapters $\{\mathbf{H}_1, \ldots, \mathbf{H}_K\}$, each kind of task t defines a specific router function $\mathbf{R}^t$ to generate new hidden features, which can be formulated as,

$$O^t = \mathbf{F}(X) + \sum_{k=1}^{K} \mathrm{G}_k^t \odot \mathbf{H}_k(X). \tag{3}$$

where $\mathrm{G}_k^t \in \mathcal{R}^D$ is a trainable vector that modulates the hidden features from each adapter and makes them suitable for the target task. In practice, we define two types of tasks, one for image-level VL tasks (image captioning and VQA), the other for fine-grained VL tasks (visual grounding). Compared to directly incorporating multiple adapters, the router module provides a better way to maximize the complementarity of image-level and region-level tasks.

We use the standard language modeling loss in all instruction-tuning stages. In the experiments, we demonstrate that stage-wise training is superior to single-stage

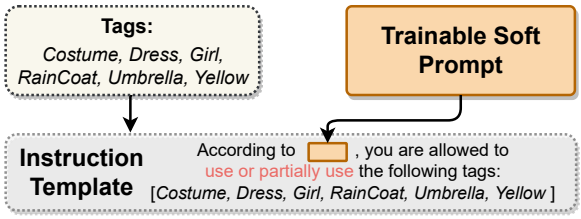

Figure 5. Instruction template with soft prompt. We use a well-designed instruction template with trainable soft prompts to inject the image tags generated by the RAM model into LION.

training, and ensures a good balance between high-level and fine-grained visual understanding capabilities, further achieves a significant mutual promotion between image-level and region-level VL tasks.

3.1.3 Vision Aggregator

To extract more sufficient visual details from input images, we devise a vision aggregator that integrates multi-level hidden features of the pretrained visual encoder. Although the vision encoder has a global reception field in all layers, it is verified that different transformer layers learn visual information at different scales [10], *e.g.*, lower layers learn visual details. Thus, our vision aggregator makes fine-grained spatial-aware visual knowledge more likely to be learned based on visual grounding tasks. Specifically, our vision aggregator can be regarded as a tiny transformer-style network, consisting of two transformer layers for aggregating the hidden features from the vision encoder. Given the hidden features $\{V_i, V_j, V_k\}$ from some middle layers in the vision encoder, the vision aggregation module uses two blocks to sequentially integrate the former two features with the last feature. Each block $\mathcal{B}$ is composed of self attention (Attn), cross attention (XAttn), and Feed-forward network (FFN) arranged in a sequential manner. Finally, the output features $\bar{V}$ is generated as follows,

$$\bar{V} = \mathcal{B}_2(\mathcal{B}_1(V_i; V_j); V_k), \tag{4}$$

$$\mathcal{B}(X; Y) = \mathrm{FFN}(\mathrm{XAttn}(\mathrm{Attn}(X), Y)). \tag{5}$$

In practice, we use the middle layers $\{i = L-1, j = 2L/3, k = L/3\}$ in the vision encoder to produce the hidden features as the input to VA, where L is the number of layers in the vision encoder.

3.2. Soft Prompting of High-Level Semantic Visual Evidence

The vision encoder in a MLLM may be insufficient in comprehensively extracting visual information required by complex multi-modal tasks, although it has been trained on large-scale image-text pairs. It has been demonstrated that increasing the amount and quality of pretraining multi-modal datasets can significantly improve the visual understanding capability of MLLM [41], but inevitably induces prohibitive computational overhead. An appealing alternative is to harness the convenient and powerful off-the-shelf vision models to capture various aspects of visual content within an image as a supplement.

We choose the recognize anything model (RAM) [52] as an off-the-shelf vision model to provide diverse tags, encompassing objects, scenes, actions, and attributes, as visual evidence to support comprehensive visual perception. Instead of directly adding tags in the instruction, we design a soft prompting method to guide the model to adaptively use the inserted tags in order to avoid the potential negative influence caused by the imperfect predictions from RAM.

In Fig. 5, we present the instruction template of tags along with the soft prompt that is a trainable vector. Our soft prompting approach can be regarded as a kind of prompt tuning methods, which guides the model toward the right direction. In standard prompt tuning works, the right direction is directly formulated as the optimization for task goals. In our work, the right direction is specified by a tailored sentence, *"According to <hint>, you are allowed to use or partially use the following tags:"*, and "<hint>" will be replaced by the soft prompt. Our soft prompting method for inserting tags has some distinct properties. It is designed to adaptively select valuable information from tags, rather than serving a specific task, as seen in standard prompt tuning methods. Our method directly uses the output labels from a small off-the-shelf vision model to incorporate high-level semantic visual evidence into a MLLM, so as to eliminate extra computational overhead of the feature alignment.

4. Experiments

In this section, we conduct extensive experiments to demonstrate the effectiveness of our model along with the quantitative and qualitative analyses. Please refer to **Appendix** for implementation details and training details.

4.1. Evaluations on Image-Level VL Tasks

Here, we evaluate multi-modal understanding abilities of LION on two kinds of image-level VL tasks, image captioning and VQA. Image captioning requires the model to generate a text description of the input image. We use COCO caption [6], TextCaps [39] and Flickr30K [48] as benchmarks, and report CIDEr as the evaluation metric. We utilize greedy search for caption generation. VQA provides an image along with a specific question for the model, asking for the output as an answer. We evaluate LION on six VQA datasets, including OKVQA [34], AOKVQA [38], GQA [17], IconQA [32], Visual Spatial Reasoning [27], and Visual Dialog (VisDial) [8]. For OKVQA, A-OKVQA, and GQA, we employ an open-ended generation with a greedy

Table 1. Comparison on image captioning and VQA. "†" denotes including in-house data that are publicly inaccessible. "*" means our evaluated results by using publicly released checkpoints, which are only for reference as official evaluation settings are incomplete. We report CIDEr score for Flickr30K, COCOCap, and TextCaps, Mean Reciprocal Rank (MRR) for Visual Dialog (VisDial), and top-1 accuracy for others. The best and second performances for each benchmark are indicated in bold and underline, respectively.

Model	Flickr30K	COCOCap	TextCaps	OKVQA	AOKVQA	GQA	IconQA	VSR	VisDial
Flamingo-3B [1]	60.60	73.00	-	-	-	-	-	-	46.10
Flamingo-9B [1]	61.50	79.40	-	44.70	-	-	-	-	48.00
Kosmos-1 [16]	67.10	84.70	-	-		-	-	-	-
Kosmos-2 [35]	80.50	-	-	-	-	-	-	-	
AdapterV2 [13]	-	122.20	-	-	-	-	-	-	-
Shikra [4]	73.90	117.50	-	47.16	-	-	-	-	-
Pink [45]	-	-	-	**59.50**	-	52.60	47.80	66.30	-
MiniGPT4 [54]	17.75*	17.04*	24.06*	37.50	34.51*	30.80	37.60	41.60	16.52*
LLaVA [29]	48.03*	73.85*	45.54*	54.40	34.51*	41.30	43.00	51.20	8.65*
MiniGPTV2 [3]	80.75*	129.16*	80.60*	56.90	-	**60.30**	47.70	60.60	8.47*
BLIVA [15]	87.10	-	-	-	-	-	44.88	62.20	45.63
InstructBLIP† (T5XL) [7]	84.50	138.21	82.55	49.28	57.86	48.40	50.00	64.80	46.60
InstructBLIP† (T5XXL) [7]	83.50	138.28	82.53	48.59	56.16	47.90	51.20	65.60	48.50
InstructBLIP (T5XL) [7]	83.71	135.47	104.17	47.38	56.12	46.34	52.47	69.93	48.75
InstructBLIP (T5XXL) [7]	85.79	138.63	105.44	53.02	59.38	47.74	53.18	68.46	50.41
LION-4B	85.57	138.20	104.87	51.08	59.98	49.50	**54.91**	72.96	50.02
LION-12B	**87.12**	**139.25**	**108.76**	57.33	**60.87**	51.56	54.89	**73.77**	**50.42**

Table 2. Comparison on REC. "Avg." means the average of top-1 accuracy over all the 8 evaluation sets.

Model	RefCOCO			RefCOCO+			RefCOCOg		Avg.
	val	test-A	test-B	val	test-A	test-B	val	test	
Zero-shot Setting									
Kosmos-2 [35]	52.32	57.42	47.26	45.48	**50.73**	42.24	60.57	61.65	52.21
GRILL [18]	-	-	-	-	-	-	-	47.50	-
Pink [45]	54.10	**61.20**	44.20	43.90	50.70	35.00	59.10	60.10	51.00
LION-4B	57.89	56.07	58.40	**46.38**	45.29	47.50	64.74	63.56	54.98
LION-12B	**58.54**	56.41	**59.36**	45.93	45.73	**47.89**	**66.12**	**64.69**	**55.58**
Fune-tuning Setting									
OFA-L [42]	79.96	83.67	76.39	68.29	76.00	61.75	67.57	67.58	72.65
VisionLLM-H [43]	-	86.70	-	-	-	-	-	-	-
Shikra-7B [4]	87.01	90.61	80.24	81.60	87.36	72.12	82.27	82.19	82.93
Shikra-13B [4]	87.83	91.11	81.81	82.89	87.79	74.41	82.64	83.16	83.96
Pink [45]	88.30	91.70	84.00	81.40	87.50	73.70	83.70	83.70	84.25
Ferret-7B [47]	87.49	91.35	82.45	80.78	87.38	73.14	83.93	84.76	83.91
Ferret-13B [47]	89.48	92.41	84.36	82.81	88.14	75.17	**85.83**	**86.34**	85.57
MiniGPTv2 [3]	88.69	91.65	85.33	79.97	85.12	74.45	84.44	84.66	84.29
LION-4B	89.73	92.29	84.82	83.60	88.72	77.34	85.69	85.63	85.98
LION-12B	**89.80**	**93.02**	**85.57**	**83.95**	**89.22**	**78.06**	85.52	85.74	**86.36**

Table 3. The comparison of various strategies in the instruction-tuning period. "REC Avg." represents the average score of all REC tasks. "Held-in" denotes the average score of COCOCap, TextCaps, OKVQA, and AOKVQA, while "Held-out" means the average score of Flickr30K, GQA, IconQA, VSR, and VisDial.

Strategy	Image-Level		Region-Level
	Held-in	Held-out	REC Avg.
Single Stage	84.79	60.20	3.78
Stage-wise	**88.07**	61.91	54.46
w/ Router	87.67	**62.16**	**54.98**

decoding strategy, For IconQA, Visual Spatial Reasoning, and Visual Dialog, we match the output with various candidates, and select the candidate with the highest value as the prediction. We report Mean Reciprocal Rank (MRR) for Visual Dialog, and top-1 accuracy for other VQA tasks. The detailed descriptions of these datasets and inference instructions are presented in **Appendix**.

As shown in Table 1, LION achieves the best performance across 7 out of 9 benchmarks, and the second on the other 2 benchmarks. LION shares the same training datasets with InstructBLIP, except Visual Genome dataset adopted in our work and the in-house dataset, WebCapFilt, used in InstructBLIP. The amount of Visual Genome dataset (3.6M) is far smaller than WebCapFilt (14M). Compared to the original InstructBLIP trained with WebCapFilt, LION

Table 4. Ablation studies of dual-level visual knowledge. "VG" means visual grounding tasks. "Held-in" and "Held-out" denote the training images of tasks are seen and unseen, respectively. "REC Avg." means the average score of all REC tasks.

Components		Held-in				Held-out					REC Avg.
VG	Tags	COCOCap	TextCaps	OKVQA	AOKVQA	Flickr30K	GQA	VSR	IconQA	VisDial	
×	×	135.47	104.17	47.38	56.12	83.71	46.34	69.93	52.47	48.75	-
✓	×	137.87	104.84	51.07	56.90	83.99	49.22	**73.20**	54.67	49.70	54.98
✓	✓	**138.20**	**104.87**	**51.08**	**59.98**	**85.57**	**49.50**	72.96	**54.91**	**50.02**	54.92

Table 5. Evaluation of object hallucination on POPE benchmark. F1 score is the major metric for halluciantion evaluation.

Datasets	Metrics	LION	Shikra [4]	InstructBLIP [7]	MiniGPT-4 [54]	LLaVA [29]	mPLUG-Owl [46]
Random	**F1-Score** ↑	88.33	86.19	**89.27**	80.17	66.64	68.39
	Accuracy ↑	**88.97**	86.90	88.57	79.67	50.37	53.97
	Precision ↑	**97.12**	94.40	84.09	78.24	50.19	52.07
	Recall ↑	81.00	79.27	95.13	82.20	99.13	99.60
Popular	**F1-Score** ↑	**85.94**	83.16	84.66	73.02	66.44	66.94
	Accuracy ↑	**86.77**	83.97	82.77	69.73	49.87	50.90
	Precision ↑	**91.69**	87.55	76.27	65.86	49.93	50.46
	Recall ↑	80.87	79.20	95.13	81.93	99.27	99.40
Adversarial	**F1-score** ↑	**84.71**	82.49	77.32	70.42	66.32	66.82
	Accuracy ↑	**85.37**	83.10	72.10	65.17	49.70	50.67
	Precision ↑	**88.69**	85.60	65.13	61.19	49.85	50.34
	Recall ↑	81.07	79.60	95.13	82.93	99.07	99.33

exhibits superior performances on all zero-shot evaluation benchmarks, showcasing a better generalization ability. We also re-implemented InstructBLIP on the same instruction-tuning datasets. The comparison shows the consistent and significant improvements of LION over the re-implemented InstructBLIP, demonstrating the effectiveness of incorporating dual-level visual knowledge. Shikra, MiniGPTV2 and Pink are also integrated with visual grounding abilities. Their inferior results on most image-level VL tasks exhibit that our proposed stage-wise instruction-tuning strategy and soft prompting of high-level semantic knowledge are very helpful in enhancing holistic visual understanding abilities of MLLMs.

4.2. Evaluations on Region-Level VL Tasks

To assess the fine-grained perceptual and reasoning abilities of LION, we evaluate it on three REC datasets, RefCOCO [19], RefCOCO+ [19], RefCOCOg [33]. REC requires the model to locate the target object given a referring expression. We follow the standard setting, and use accuracy as an evaluation metric, which means it is correct when the IOU between prediction and ground-truth is no less than 0.5.

In Table 2, we show the comparison between LION and other MLLMs with respect to the grounding abilities, under the settings of zero-shot and fine-tuning evaluations, respectively. In the zero-shot evaluation setting, we directly employ LION to generate coordinates of referring expressions on three datasets. Our model shows significant improvements on most evaluation sets over Kosmos-2 and Pink, except test-A sets of RefCOCO and RefCOCO+. The languages used in RefCOCOg are more flowery than those used in RefCOCO and RefCOCO+. The significant improvements on RefCOCOg clearly demonstrate that LION can handle complex referring expressions and has superior zero-shot spatial-aware visual understanding abilities.

In the fine-tuning setting, we fine-tune LION with training samples from three REC datasets. As shown in Table 2, our model achieves the best performance on average and on most of the evaluation sets, indicating the advanced fine-grained perception ability of our model. Ferret proposes a spatial-aware visual sampler to handle free-form referred regions, and meticulously constructs an extensive grounding dataset with lots of efforts on data generation and filtering. However, LION can achieve superior performances compared to Ferret by using a simple vision aggregator and existing datasets, implying the effectiveness of fine-grained visual knowledge.

4.3. Ablation Study

The effect of vision aggregator. We conduct an ablation study of the vision aggregator with only visual grounding tasks on LION-4B during stage 2 of the stage-wise instruction-tuning. As illustrated in Fig. 6, the removal of the vision aggregator degrades REC performances, validating that aggregating multi-level vision features promotes the extraction of fine-grained spatial-aware visual knowledge.

Stage-wise instruction-tuning mitigates the conflict between image-level and region-level tasks. We investigate the performance of two types of VL tasks under three instruction-tuning strategies, i.e., single stage, stage-wise, and stage-wise with a router. As shown in Table 3, the stage-wise instruction-tuning strategy shows a significant im-

也是测 hallucination

Table 6. Evaluation on MMBench test set, all the reported results of compared models are from the leadboard of MMBench.

Models	Text Encoder	Vision Encoder	Overall	LR	AR	RR	FP-S	FP-C	CP
MiniGPT-4 [54]	Vincuna-7B	EVA-G	23.0	13.6	32.9	8.9	28.8	11.2	28.3
PandaGPT [40]	Vincuna-13B	ImageBind ViT-H/14	42.5	23.1	61.5	34.1	32.7	28.7	57.6
VisualGLM [11]	ChatGLM-6B	EVA-CLIP	33.5	11.4	48.8	27.7	35.8	17.6	41.5
InstructBLIP [7]	Vicuna-7B	EVA-G	33.9	21.6	47.4	22.5	33.0	24.4	41.1
LLaVA-v1.5 [28]	Vicuna-v1.5-7B	CLIP ViT-L/14	59.5	32.4	72.6	49.3	62.3	52.2	67.7
Otter-I [22]	LLaMA-7B	CLIP ViT-L/14	48.3	22.2	63.3	39.4	46.8	36.4	60.6
Shikra [4]	Vincuna-7B	CLIP ViT-L/14	60.2	33.5	69.6	53.1	61.8	50.4	71.7
LMEye [26]	FlanT5-XL	CLIP ViT-L/14	62.6	41.0	74.3	55.9	61.6	58.7	69.2
MMICL [53]	FlanT5-XXL	EVA-G	65.2	44.3	77.9	64.8	66.5	53.6	70.6
mPLUG-Owl2 [44]	LLaMA2-7B	CLIP ViT-L/14	66.0	43.4	76.0	62.1	68.6	55.9	73.0
LLaVA-v1.5-13B	Vicuna-v1.5-13B	CLIP ViT-L/14	67.8	43.4	71.9	60.7	73.4	59.1	77.3
LION	FlanT5-XXL	EVA-G	**73.4**	**51.7**	**84.1**	**78.4**	**74.0**	**60.8**	**78.9**

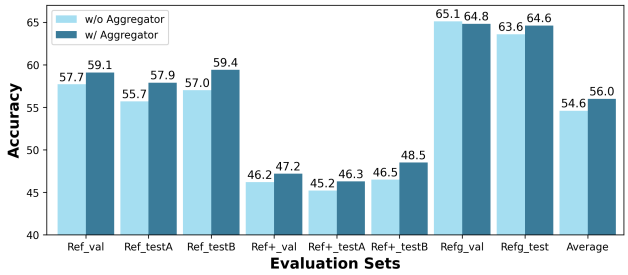

Figure 6. The effect of the vision aggregator. The results on RefCOCO, RefCOCO+, and RefCOCOg clearly show that the proposed module can overall improve REC performances across 8 evaluation sets.

provement on the average REC performance, which is completely damaged in the single stage instruction-tuning. The worse REC performance of the single stage strategy can be attributed to the lack of pretraining on large-scale grounding datasets, like in Kosmos-2 and Shikra, and the gap of their input-output modes. To address these challenges, stage-wise training progressively integrates fine-grained spatial-aware knowledge from visual grounding datasets by splitting the whole instruction-tuning process into three stages. The model can sufficiently learn diverse levels of visual knowledge in separate training stages (stages 1 and 2), and incorporate them in the final training stage (stage 3 in Fig. 4). This contributes to the performance improvements of all VL tasks. Furthermore, stage-wise instruction-tuning with the router improves the held-out and REC performance, with a slight degradation in the held-in performance. All these results demonstrate LION's ability to handle the potential conflict of various VL tasks and maximize the learning benefit. **Dual-level visual knowledge enhances multimodal understanding abilities of MLLMs.** We evaluate the performance of our model integrated with different levels of visual knowledge on various benchmarks in Table 4. It can be seen that dual-level visual knowledge can upgrade the performance of all VL tasks to varying degrees. When progressively incorporating fine-grained spatial-aware knowledge, the performances of four tasks, OKVQA, GQA, IconQA, and Visual Spatial Reasoning, are significantly improved, as they highly require region-level understanding and spatial reasoning. When inserting tags as high-level visual evidence, we can see substantial performance increases on Flickr30K and AOKVQA, which demand more comprehensive semantic knowledge than other tasks, like COCO caption and OKVQA.

4.4. Evaluations on Object Hallucination and MM-Bench

Li *et al.* [25] present an open-sourced evaluation benchmark, called POPE, to evaluate the object hallucination [37]. We follow the POPE evaluation pipeline to inspect LION. The results in Table 5 show that LION has superior results, especially under popular and adversarial settings, which means that incorporating fine-grained and high-level semantic visual knowledge into MLLM can mitigate the object hallucination to some degree. To comprehensively validate the effectiveness of our method, we further evaluate LION on MMBench [30]. The results are summarized in Table 6. Our strong performances across various skills demonstrate that the progressive incorporation of fine-grained knowledge significantly alleviates the hallucination phenomenon in MLLMs.

4.5. Qualitative Analysis

As shown in Fig. 7, we depict various examples to validate the advanced perceptual and reasoning abilities of LION. The left example exhibits our superior fine-grained understanding capability to correctly generate the right attributes "white", "yellow" and the character "Kwon". The middle example validates the advantage of our model in visual spatial reasoning. The right example shows that LION accurately localizes the referring object, while Shikra provides an incorrect response caused by the misunderstanding of fine-grained details "a soda bottle".

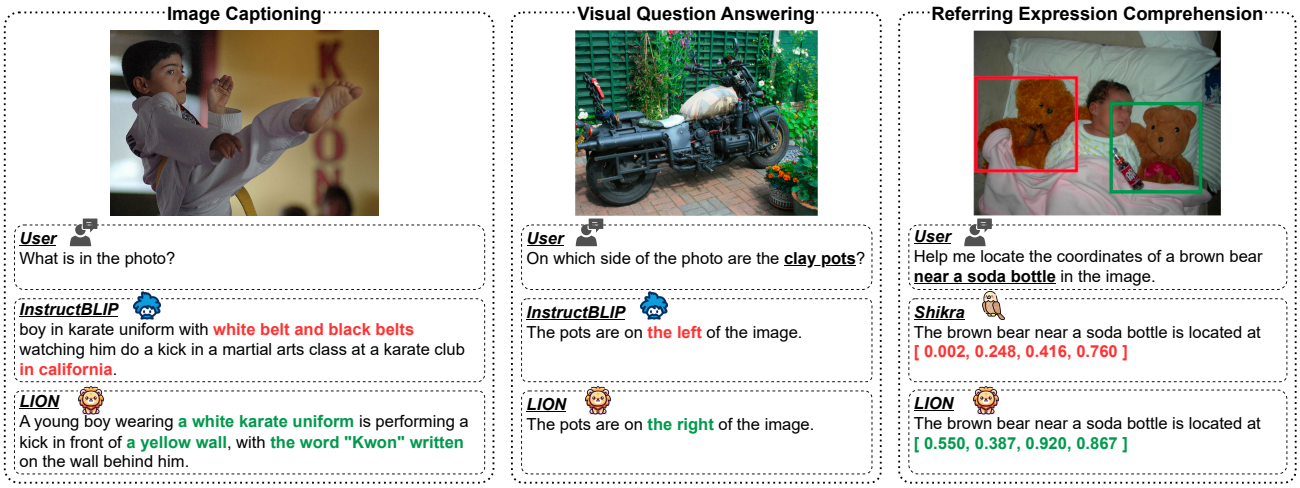

Figure 7. Qualitative comparison of InstructBLIP, Shikra, and LION. We mark the hallucination or incorrect part in red, and highlight the correct part in green for comparison. These samples exhibit that LION is able to achieve superior fine-grained understanding and visual spatial reasoning capabilities with fewer hallucinated responses.

5. Conclusion

To address the insufficient extraction and reasoning of visual information in MLLMs, we propose LION to exploit dual-level visual knowledge, *i.e.,* fine-grained spatial-aware visual knowledge and high-level semantic visual evidence, based on region-level and image-level VL tasks. To mitigate the internal conflict between these two kinds of tasks, LION proposes a stage-wise instruction-tuning strategy to progressively incorporate fine-grained spatial-aware visual knowledge into MLLMs, achieving the mutual promotion between these two kinds of VL tasks. We use image tags as high-level semantic visual evidence, and present a soft prompting method to alleviate the potential influence resulting from incorrect tags. Extensive experiments validate the superiority of LION in image captioning, VQA, and visual grounding tasks.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022. 2, 6

[2] Chi Chen, Ruoyu Qin, Fuwen Luo, Xiaoyue Mi, Peng Li, Maosong Sun, and Yang Liu. Position-enhanced visual instruction tuning for multimodal large language models. *arXiv preprint arXiv:2308.13437*, 2023. 3

[3] Jun Chen, Deyao Zhu, Xiaoqian Shen, Xiang Li, Zechun Liu, Pengchuan Zhang, Raghuraman Krishnamoorthi, Vikas Chandra, Yunyang Xiong, and Mohamed Elhoseiny. Minigpt-v2: large language model as a unified interface for vision-language multi-task learning. *arXiv preprint arXiv:2310.09478*, 2023. 6

[4] Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal llm's referential dialogue magic. *arXiv preprint arXiv:2306.15195*, 2023. 2, 3, 6, 7, 8

[5] Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. Adaptformer: Adapting vision transformers for scalable visual recognition. *Advances in Neural Information Processing Systems*, 35:16664–16678, 2022. 4

[6] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. 5

[7] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. In *NeurIPS*, 2023. 1, 2, 4, 6, 7, 8

[8] Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. Visual dialog. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 326–335, 2017. 5

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 11

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 5

[11] Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335, 2022. 8

[12] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. In *CVPR*, 2023. 11

[13] Peng Gao, Jiaming Han, Renrui Zhang, Ziyi Lin, Shijie Geng, Aojun Zhou, Wei Zhang, Pan Lu, Conghui He, Xiangyu Yue, et al. Llama-adapter v2: Parameter-efficient visual instruction model. *arXiv preprint arXiv:2304.15010*, 2023. 2, 6

[14] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 11

[15] Wenbo Hu, Yifan Xu, Y Li, W Li, Z Chen, and Z Tu. Bliva: A simple multimodal llm for better handling of text-rich visual questions. *arXiv preprint arXiv:2308.09936*, 2023. 6

[16] Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Qiang Liu, et al. Language is not all you need: Aligning perception with language models. *arXiv preprint arXiv:2302.14045*, 2023. 6

[17] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019. 5

[18] Woojeong Jin, Subhabrata Mukherjee, Yu Cheng, Yelong Shen, Weizhu Chen, Ahmed Hassan Awadallah, Damien Jose, and Xiang Ren. Grill: Grounded vision-language pre-training via aligning text and image regions. *arXiv preprint arXiv:2305.14676*, 2023. 6

[19] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 787–798, 2014. 3, 7

[20] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123:32–73, 2017. 3

[21] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123:32–73, 2017. 3

[22] 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. 1, 2, 8

[23] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 2023. 2

[24] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *CVPR*, 2022. 1

[25] 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. 8

[26] Yunxin Li, Baotian Hu, Xinyu Chen, Lin Ma, and Min Zhang. Lmeye: An interactive perception network for large language models. *arXiv preprint arXiv:2305.03701*, 2023. 8

[27] Fangyu Liu, Guy Emerson, and Nigel Collier. Visual spatial reasoning. *Transactions of the Association for Computational Linguistics*, 11:635–651, 2023. 5

[28] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023. 8

[29] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 2, 6, 7

[30] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*, 2023. 8

[31] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 12

[32] Pan Lu, Liang Qiu, Jiaqi Chen, Tony Xia, Yizhou Zhao, Wei Zhang, Zhou Yu, Xiaodan Liang, and Song-Chun Zhu. Iconqa: A new benchmark for abstract diagram understanding and visual language reasoning. *arXiv preprint arXiv:2110.13214*, 2021. 5

[33] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11–20, 2016. 7

[34] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pages 3195–3204, 2019. 5

[35] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023. 2, 3, 4, 6

[36] 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. 1

[37] Vipula Rawte, Amit Sheth, and Amitava Das. A survey of hallucination in large foundation models. *arXiv preprint arXiv:2309.05922*, 2023. 8

[38] Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge. In *European Conference on Computer Vision*, pages 146–162. Springer, 2022. 5

[39] Oleksii Sidorov, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. Textcaps: a dataset for image captioning with reading comprehension. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 742–758. Springer, 2020. 5

[40] Yixuan Su, Tian Lan, Huayang Li, Jialu Xu, Yan Wang, and Deng Cai. Pandagpt: One model to instruction-follow them all. *arXiv preprint arXiv:2305.16355*, 2023. 8

[41] Guangzhi Wang, Yixiao Ge, Xiaohan Ding, Mohan Kankanhalli, and Ying Shan. What makes for good visual tokenizers for large language models? *arXiv preprint arXiv:2305.12223*, 2023. 1, 2, 3, 5

[42] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International Conference on Machine Learning*, pages 23318–23340. PMLR, 2022. 6

[43] Wenhai Wang, Zhe Chen, Xiaokang Chen, Jiannan Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu Qiao, et al. Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. *arXiv preprint arXiv:2305.11175*, 2023. 6

[44] Haiyang Xu, Qinghao Ye, Ming Yan, Yaya Shi, Jiabo Ye, Yuanhong Xu, Chenliang Li, Bin Bi, Qi Qian, Wei Wang, et al. mplug-2: A modularized multi-modal foundation model across text, image and video. *arXiv preprint arXiv:2302.00402*, 2023. 8

[45] Shiyu Xuan, Qingpei Guo, Ming Yang, and Shiliang Zhang. Pink: Unveiling the power of referential comprehension for multi-modal llms. *arXiv preprint arXiv:2310.00582*, 2023. 3, 4, 6

[46] Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*, 2023. 2, 7

[47] Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao, Shih-Fu Chang, and Yinfei Yang. Ferret: Refer and ground anything anywhere at any granularity. *arXiv preprint arXiv:2310.07704*, 2023. 3, 4, 6

[48] 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. 5

[49] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *ECCV*, pages 69–85. Springer, 2016. 3

[50] Yuexiang Zhai, Shengbang Tong, Xiao Li, Mu Cai, Qing Qu, Yong Jae Lee, and Yi Ma. Investigating the catastrophic forgetting in multimodal large language models. *arXiv preprint arXiv:2309.10313*, 2023. 1

[51] Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*, 2023. 2

[52] Youcai Zhang, Xinyu Huang, Jinyu Ma, Zhaoyang Li, Zhaochuan Luo, Yanchun Xie, Yuzhuo Qin, Tong Luo, Yaqian Li, Shilong Liu, et al. Recognize anything: A strong image tagging model. *arXiv preprint arXiv:2306.03514*, 2023. 5, 11

[53] Haozhe Zhao, Zefan Cai, Shuzheng Si, Xiaojian Ma, Kaikai An, Liang Chen, Zixuan Liu, Sheng Wang, Wenjuan Han, and Baobao Chang. Mmicl: Empowering vision-language model with multi-modal in-context learning. *arXiv preprint arXiv:2309.07915*, 2023. 8

[54] 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. 1, 2, 6, 7, 8

A. Experimental Details

Architecture. We use the off-the-shelf ViT-G/14 from EVA-CLIP [12] without the last layer as our frozen vision backbone. The vision aggregator consists of two Bert Layers [9] with cross attention in each layer. The output from the Vision Aggregator undergoes a transformation via a two-layer MLP with GeLU [14] activation, and is projected into the latent feature space of the LLM. This output is then concatenated with the output from Q-Former and the textual inputs, forming the comprehensive inputs for the LLM. In the LLM, the hidden dimension of each adapter is set to 64. We implement LION on LLMs with two different size, including FlanT5-XL(3B) and FlanT5-XXL(11B), resulting in LION-4B and LION-12B, respectively.

When incorporating the image tags as high-level semantic visual evidence, we use the recognize anything model (RAM-14M) [52] based on the backbone Swin-Large. All the image tags are generated by using a 384×384 image size and a 0.8 threshold across 4585 categories in the ram tag list. All other hyperparameters are set the same as in the RAM codebase [†].

Training Details. Our training process comprises three stages. In Stage 1, we use a batch size of 64 for 10 epochs over 30k steps, with a learning rate starting at 1e-5 and reducing to a minimum of 0. Stage 2 increases the batch size to 256 for another 10 epochs across 60k steps, beginning with a learning rate of 5e-4, which is reduced to a floor of 1e-6; notably, the learning rate for the Vision Aggregator is set to a constant 1e-5. Stage 3 reverts to a batch size of 64

[†]https://github.com/xinyu1205/recognize-anything

for 10 epochs and 60k steps, with an initial learning rate of 1e-5, descending to a minimum of 0. Throughout all stages, the AdamW [31] optimizer is employed with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and a weight decay of 0.05. The learning rate is warmed up linearly from 1e-8 across 1000 steps at the beginning of each stage.

Training Data. We describe all training datasets in Table 7. In stage 1, a part of LION is trained on image-level VL tasks, including COCO Caption, TextCaps, OKVQA, AOKVQA, VQAv2, OCR-VQA. Sepcifically, we follow InstructBLIP to define a visual question generation (VQG) task, which requires the model to generate a question given an answer. This VQG task is formed by using OKVQA, AOKVQA, and VQAv2 training datasets. We also use a dialogue dataset, LLaVA-Instruct-150K in this stage. In stage2, we use Visual Genome training dataset to construct referring expression comprehension (REC) and referring expression generation (REG) tasks. In final stage, all the mentioned datasets are used to train a unified model, resulting in the LION. We insert image tags in stage 3, firstly generate image tags for all training images, then use them with the soft prompting method. We provide evaluation metrics in Table 8.

B. Instruction Templates

B.1. Task Templates for Instruction-Tuning

We provide instruction templates for transform image-level and region-level VL tasks into a instruction-tuning format. For image-level VL tasks, we follow the setting in InstructBLIP. For region-level tasks, we use the templates in Shikra, which are generated by GPT-4 with carefully designed instructions. For each task listed in Table 9, we only show a few templates.

B.2. Instructions for Evaluation

We provide instructions for evaluation on various benchmarks. For instructions involving options, we arrange the options in the alphabetical order. For REC tasks, we randomly choose a template in training instruction lists for evaluation, which is the same as Shikra.

OKVQA, AOKVQA, GQA <Image> Question: {Question} Short answer:

COCOCap, Flickr30K, TextCaps <Image> A short image description:

IconQA <Image> {Question}

VSR <Image> Based on the image, is this statement true or false? "{Question}" Answer:

Visual Dialog <Image> Dialog history: {History}\n Question: {Question} Short answer:

Table 7. The training datasets used for instruction-tuning.

Task	Dataset	Stage 1	Stage 2	Stage 3	Data Number
Dialogue	LLaVA-Instruct-150K	✓		✓	361K
VQA	OKVQA, A-OKVQA, VQAv2, OCR-VQA	✓		✓	1.3M
VQG	OKVQA, A-OKVQA, VQAv2	✓		✓	470K
Image Captioning	COCO, TextCaps	✓		✓	524K
REC	Visual Genome		✓	✓	3.6M
REG	Visual Genome		✓	✓	3.6M

Table 8. Summary of the evaluation datasets.

Task	Dataset	Split	Metric
Image Captioning	Flickr30K	karpathy-test	CIDEr(↑)
	COCO	karpathy-test	CIDEr(↑)
	TextCaps	val	CIDEr(↑)
Visual Question Answering	OKVQA	val	Accuracy(↑)
	AOKVQA	val	Accuracy(↑)
	Visual Spatial Reasoning	val	Accuracy(↑)
	Visual Dialog	val	MRR(↑)
	IconQA	test	Accuracy(↑)
	GQA	test-dev	Accuracy(↑)
Referring Expression Comprehension	RefCOCO	val & testA & testB	Accuracy(↑)
	RefCOCO+	val & testA & testB	Accuracy(↑)
	RefCOCOg	val & test	Accuracy(↑)

Table 9. Examples of instruction templates for various tasks. "{expr}" represents the expression in the REC task. "{BBox}" refers to the bounding box of a user-specified location.

VQA	<Image>Given the image, answer the following question with no more than three words. {Question} <Image>Based on the image, respond to this question with a short answer: {Question}. Answer: <Image>Use the provided image to answer the question: {Question} Provide your answer as short as possible: <Image>What is the answer to the following question? "{Question}" <Image>The question "{Question}" can be answered using the image. A short answer is
VQG	<Image>Based on the image, provide a question with the answer: {Answer}. Question: <Image>Given the visual representation, create a question for which the answer is "{Answer}". <Image>From the image provided, craft a question that leads to the reply: {Answer}. Question: <Image>Considering the picture, come up with a question where the answer is: {Answer}. <Image>Taking the image into account, generate an question that has the answer: {Answer}. Question:
Image Captioning	<Image>Can you briefly explain what you see in the image? <Image>Could you use a few words to describe what you perceive in the photo? <Image>Please provide a short depiction of the picture. <Image>Using language, provide a short account of the image. <Image>Use a few words to illustrate what is happening in the picture.
REC	<image>Identify the position of {expr} in image and share its coordinates. <image>I'd like to request the coordinates of {expr} within the photo. <image>How can I locate {expr} in the image? Please provide the coordinates. <image>I am interested in knowing the coordinates of {expr} in the picture. <image>Assist me in locating the position of {expr} in the photograph and its bounding box coordinates. <image>In the image, I need to find {expr} and know its coordinates. Can you please help?
REG	<image>What are the unique characteristics of the rectangular section {BBox} in image? <image>Describe the novel qualities of the selected bounding box {BBox} in image. <image>What sets the chosen region {BBox} in image apart from its surroundings? <image>Provide a one-of-a-kind depiction for the area enclosed by {BBox} in image. <image>How would you portray the unique features of the designated box {BBox} in image? <image>Explain the distinguishing characteristics of the marked bounding box {BBox} in image.