

# TWIST & SCOUT: Grounding Multimodal LLM-Experts by Forget-Free Tuning

Aritra Bhowmik<sup>\*1</sup> Mohammad Mahdi Derakhshani<sup>\*1</sup> Dennis Koelma<sup>1</sup>  
 Yuki M. Asano<sup>2</sup> Martin R. Oswald<sup>1</sup> Cees G. M. Snoek<sup>1</sup>  
<sup>1</sup>University of Amsterdam <sup>2</sup>University of Technology Nuremberg

\*

## Abstract

*Spatial awareness is key to enable embodied multimodal AI systems. Yet, without vast amounts of spatial supervision, current Multimodal Large Language Models (MLLMs) struggle at this task. In this paper, we introduce TWIST & SCOUT, a framework that equips pre-trained MLLMs with visual grounding ability without forgetting their existing image and language understanding skills. To this end, we propose TWIST, a twin-expert stepwise tuning module that modifies the decoder of the language model using one frozen module pre-trained on image understanding tasks and another learnable one for visual grounding tasks. This allows the MLLM to retain previously learned knowledge and skills, while acquiring what is missing. To fine-tune the model effectively, we generate a high-quality synthetic dataset we call SCOUT, which mimics human reasoning in visual grounding. This dataset provides rich supervision signals, describing a step-by-step multimodal reasoning process, thereby simplifying the task of visual grounding. We evaluate our approach on several standard benchmark datasets, encompassing grounded image captioning, zero-shot localization, and visual grounding tasks. Our method consistently delivers strong performance across all tasks, while retaining the pre-trained image understanding capabilities.*

## 1. Introduction

Multimodal Large Language Models (MLLMs) have greatly advanced vision and language tasks, excelling in image captioning and visual question answering [2, 10, 22, 28]. Models like Flamingo, BLIP-2, InstructBLIP, and VisualGLM leverage large image-caption datasets to integrate vision and language, addressing complex multimodal challenges. However, due to their caption-based design, these models often lack visual grounding, limiting their suitability for tasks requiring precise spatial understanding [9, 14, 19, 32, 42].

While extensive pre-training can equip models with localization capabilities [6, 40], it requires massive datasets, human-annotated bounding boxes, and substantial computational resources, making it impractical for many setups. Instead, we focus on fine-tuning pre-trained MLLMs to instill spatial understanding in a forget-free manner, preserving existing language and vision comprehension skills.

Closest to our work is PIN by Dorckenwald *et al.* [13], which addresses single-object localization in pre-trained autoregressive MLLMs through two key innovations: modifying the vision encoder with learned spatial parameters for bounding box prediction and introducing a synthetic dataset of superimposed object renderings to remove reliance on human annotations. However, PIN’s architectural modifications cause catastrophic forgetting, erasing pre-trained image understanding. Additionally, its simplistic object-pasting approach introduces domain shift, limiting applicability to complex tasks requiring multi-object reasoning and richer spatial relationships [6, 40]. Another approach is parameter-efficient tuning via LoRA [17], which adds low-rank weight updates to a frozen backbone. While LoRA preserves pre-trained strengths for tasks close to its domain, its low-rank constraints and limited capacity fail to capture new spatial relationships and bounding-box nuances, leading to suboptimal grounding. Consequently, neither PIN nor LoRA retains vision-language skills while adding robust grounding—an issue our work addresses without full model finetuning.

To equip autoregressive MLLMs with robust grounding while ensuring forget-free performance, we introduce TWIST & SCOUT. TWIST stands for **TW**In-expert **Stepwise Tuning**, a framework with two parallel modules and a stepwise loss function inspired by Lightman *et al.* [23]. We treat the pre-trained backbone as one “expert” and add a Mixture of Experts (MoE) as the second expert for grounding, providing enough capacity to handle unfamiliar demands without overwriting pre-trained understanding. Akin to LoRA, we add new parameters; however, rather than relying on low-rank residuals, we fuse old and new knowledge via a learnable gating mechanism, enabling robust grounding without erasing existing skills. Stepwise

<sup>\*</sup>Joint first authors. Corresponding authors: {a.bhowmik, m.m.derakhshani}@uva.nl

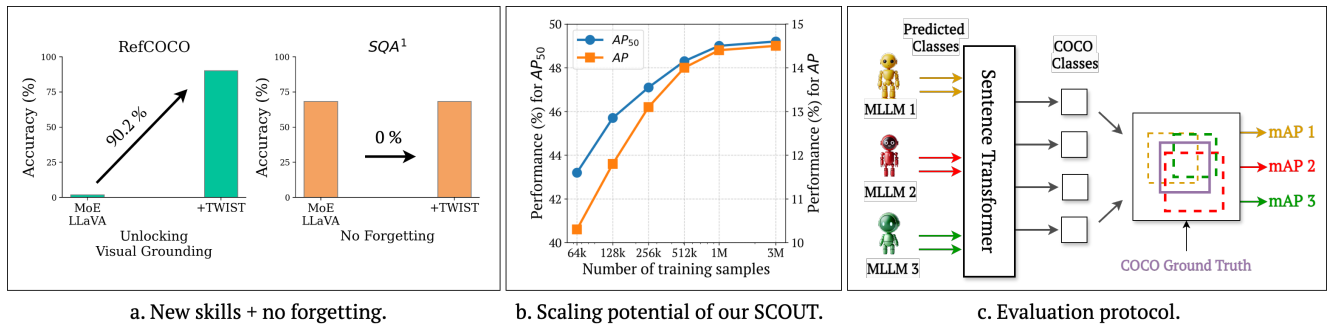


Figure 1. **TWIST & SCOUT contributions.** Our contributions include (a) TWIST, a framework that fine-tunes a pre-trained caption-based MLLM to acquire new grounding skills while retaining existing image understanding capabilities, (b) SCOUT, a scalable synthetic dataset that enhances model performance through step-by-step grounded chain-of-thought annotations, and (c) an evaluation protocol tailored for assessing MLLMs on grounded image captioning tasks.

tuning strengthens learning by breaking down complex tasks into simpler subtasks, enhancing vision-language performance. Complementing TWIST, we present SCOUT, short for **S**ynthetic **C**hain-of-**T**hought with **G**rounding, a high-quality synthetic dataset capturing meaningful spatial relationships, inter-object reasoning, and stepwise thought processes—providing a rich training signal for fine-tuning MLLMs. Recognizing the limitations of evaluation methods focused solely on object localization, we introduce a protocol for assessing MLLMs on free-form grounded image captioning, which requires both visual grounding and image understanding. Our contributions can be summarized as:

1. We propose TWIST, a TWIn-expert Stepwise Tuning framework that fine-tunes pre-trained MLLMs via two parallel modules without forgetting. TWIST employs step-by-step training, breaking complex grounding tasks into simpler subtasks (Figure 1 (a)).
2. We present SCOUT, a synthetic dataset with stepwise grounded chain-of-thought annotations. SCOUT facilitates fine-tuning for grounding and reasoning, providing a rich, spatially complex training signal (Figure 1 (b)).
3. We create an evaluation protocol for assessing MLLMs on free-form grounded image captioning (Figure 1 (c)).

Our experiments show strong performance in grounded image captioning and visual grounding while retaining initial image understanding.

## 2. Related Work

**Multimodal LLMs.** Large Language Models (LLMs), known for their instruction-following and generalization abilities, have been effectively integrated with vision encoders, achieving strong multimodal performance [1, 2, 4, 10, 11, 20, 22, 24, 33, 37, 41, 43, 47]. Pioneering models like Flamingo [2] and BLIP-2 [22] integrate vision and language by combining CLIP-based image encoders with LLMs—Flamingo using perceiver and gated cross-attention blocks, while BLIP-2 employs a lightweight Query-

ing Transformer. Recent efforts have optimized training strategies [4, 43], improved image resolution [4, 20, 40], and enhanced image encoders [8, 48]. Additional advancements refine input alignment [24] and projection layers [5, 10], while expanding instruction-tuning datasets has further improved performance and versatility [27, 49]. However, despite these improvements, instruction-tuned MLLMs mainly excel at image captioning and simple QA but struggle with spatial reasoning and precise object grounding [13]. Our work addresses these gaps by equipping MLLMs with spatial understanding for visual grounding and object localization.

### Grounded Multimodal Models and Object Detection.

Extending MLLMs beyond image and language understanding, several models have been developed to enable visual grounding and object localization [4, 6, 7, 31, 38–41, 44]. Pix2Seq [7] pioneered treating object detection as an autoregressive language modeling task, inspiring models like OFA [39], Unified-IO [31], UniTab [44], and Vision-LLM [41] to integrate language and coordinate vocabularies for grounding. Meanwhile, Shikra [6], CogVLM [40], and Qwen-VL [4] further advance positional representations in natural language, facilitating seamless interleaved grounded captions. Despite these advancements, most models rely on large annotated datasets and extensive pre-training. Grounding DINO [29] takes a different approach, using a transformer-based architecture trained with contrastive and bounding box regression losses for object detection. However, unlike autoregressive MLLMs, Grounding DINO is optimized specifically for detection and lacks the ability to generate grounded image captions in free-form text. PIN [13] attempts to bridge the gap by introducing a learnable positional insert module and a synthetic dataset for fine-tuning. Yet, its reliance on purely synthetic data leads to domain shift, causing it to forget previous vision-language abilities and remain limited to single-object localization. Our approach addresses these challenges through TWIST, a two-module framework that preserves pre-trained

vision-language skills while incrementally adding grounding capabilities. Paired with SCOUT, our synthetic dataset featuring chain-of-thought reasoning, TWIST enables MLLMs to handle complex, multi-object grounding tasks requiring both spatial reasoning and image understanding.

### 3. TWIST

In the following sections, we briefly review standard MLLMs and the concept of Mixture of Experts (MoE). We then introduce TWIST, a TWIn-expert Stepwise Tuning framework with two parallel modules and a step-by-step training objective. Finally, we explain how the step-by-step learning strategy adjusts the training loss.

#### 3.1. Preliminaries

**Multimodal Large Language Models (MLLMs).** MLLMs process both image and text data for multimodal generative tasks. These models consist of a vision encoder  $\psi(\cdot)$ , a language decoder,  $\phi(\cdot)$ , and a mapper function  $f(\cdot)$ . The language decoder takes a sequence of tokens as inputs  $[v_1, v_2, \dots, v_m, t_1, t_2, \dots, t_n]$  being composed of visual and textual tokens. Visual tokens are computed from an image  $\mathbf{x}$  as  $[v_1, v_2, \dots, v_m] = f(\psi(\mathbf{x}))$ , and textual tokens are computed from the text input  $\mathbf{t}$  as  $[t_1, t_2, \dots, t_n] = \text{Tokenizer}(\mathbf{t})$ . MLLMs are trained via the cross-entropy loss.

**Mixture of Experts (MoEs).** MoEs are a way to increase small model capacity to compete with large models performance without a proportional increase in computational cost [36]. Specifically, an MoE layer is composed of  $E$  ‘‘experts’’ and a gating network  $g(\cdot)$ . The gating network decides which expert is most suitable for a given token:

$$l_n = \text{MoE}(l_{n-1}) = \sum_{i=1}^E g_i(l_{n-1}) \cdot e_i(l_{n-1}), \quad (1)$$

where  $l_n$  represents the output of the  $n$ -th layer,  $l_{n-1}$  its input,  $E$  the total number of experts,  $g_i(\cdot)$  the gating function’s weight for the  $i$ -th expert, and  $e_i(\cdot)$  the  $i$ -th expert’s output. During inference, only the top- $k$  experts can be used, reducing inference costs considerably.

#### 3.2. TWIST Workflow

In Figure 2 (a), we present the general workflow of the TWIST model, which consists of a vision encoder, tokenizer and an LLM, taking image-text pairs as inputs and generating grounded free-form texts. Below, we detail each component of the TWIST workflow.

**Twin-expert module.** We start with a caption-based mixture-of-expert MLLM [25] adept at visual question answering tasks, and extend it for the task of visual grounding as depicted in Figure 2 (b). A transformer block of the language decoder of a MLLM is composed of multi-head attention

(MHA), a feed-forward network (FFN) and a layer norm (LN), which processes the input tokens as follows:

$$\hat{l}_n = \text{MHA}(\text{LN}(l_{n-1})) + l_{n-1}, \quad (2)$$

$$l_n = \text{FFN}(\text{LN}(\hat{l}_n)) + \hat{l}_n, \quad (3)$$

where  $l_{n-1}$  is the input from layer  $n-1$ ,  $\hat{l}_n$  is the hidden representation at layer  $n$ , and  $l_n$  is the output of the  $n$ -th layer. The mixture of expert module only modifies Eq. (3) by replacing the FFN module with a MoE in the transformer block computation as follows:

$$l_n = \text{MoE}(\text{LN}(\hat{l}_n)) + \hat{l}_n. \quad (4)$$

We introduce a parallel MoE module for visual grounding and modify the above equations as follows:

$$l_n^{\text{IU}} = \text{MoE}^{\text{IU}}(\text{LN}(\hat{l}_n)) + \hat{l}_n, \quad (5)$$

$$l_n^{\text{VG}} = \text{MoE}^{\text{VG}}(\text{LN}(\hat{l}_n)) + \hat{l}_n,$$

$$l_n = \alpha \cdot l_n^{\text{IU}} + (1 - \alpha) \cdot l_n^{\text{VG}}, \quad (6)$$

where  $\text{MoE}^{\text{IU}}$  is a frozen MoE module pre-trained on image understanding tasks,  $\text{MoE}^{\text{VG}}$  is a learnable MoE module trained for visual grounding task, and  $\alpha$  is a learnable coefficient weight adjusting the contribution of each MoE module. This design choice prevents catastrophic forgetting of pre-trained image understanding skills of MLLMs. Moreover, the shared modules allow knowledge transfer from the pre-trained image understanding MoE into the grounding MoE, helping the latter to better interpret grounding tasks.

**Training step.** We train our model using a cross-entropy loss for the next token prediction task:

$$L = - \left[ \sum_{i=1}^N \log P_{\theta}(t_i | v_1, \dots, v_m, t_1, \dots, t_{i-1}) \right] + \lambda \cdot R(g), \quad (7)$$

where  $L$  is the next token prediction loss,  $N$  represents the length of the text sequence,  $v_i$  refers to the  $i$ -th visual token in the sequence,  $t_i$  denotes the  $i$ -th textual token in the sequence,  $\theta$  refers to the model parameters,  $\lambda$  is a regularization coefficient, and  $R(g)$  is a regularization term for sparsifying the gating mechanism. This loss function aims to minimize the discrepancy between the predicted and actual next token in the sequence.

**Step-by-step loss function.** To fully leverage the Twin-Expert module of TWIST, we implement a step-by-step loss inspired by Lightman et al. [23]. This approach decomposes complex tasks into sequential, easily digestible subtasks, each corresponding to a specific part of the overall reasoning process, as seen in Figure 2 (c). These steps are not separate tasks but subtasks of a unified task. To illustrate this concept mathematically, the loss function for training under step-by-step reasoning supervision can be expressed as:

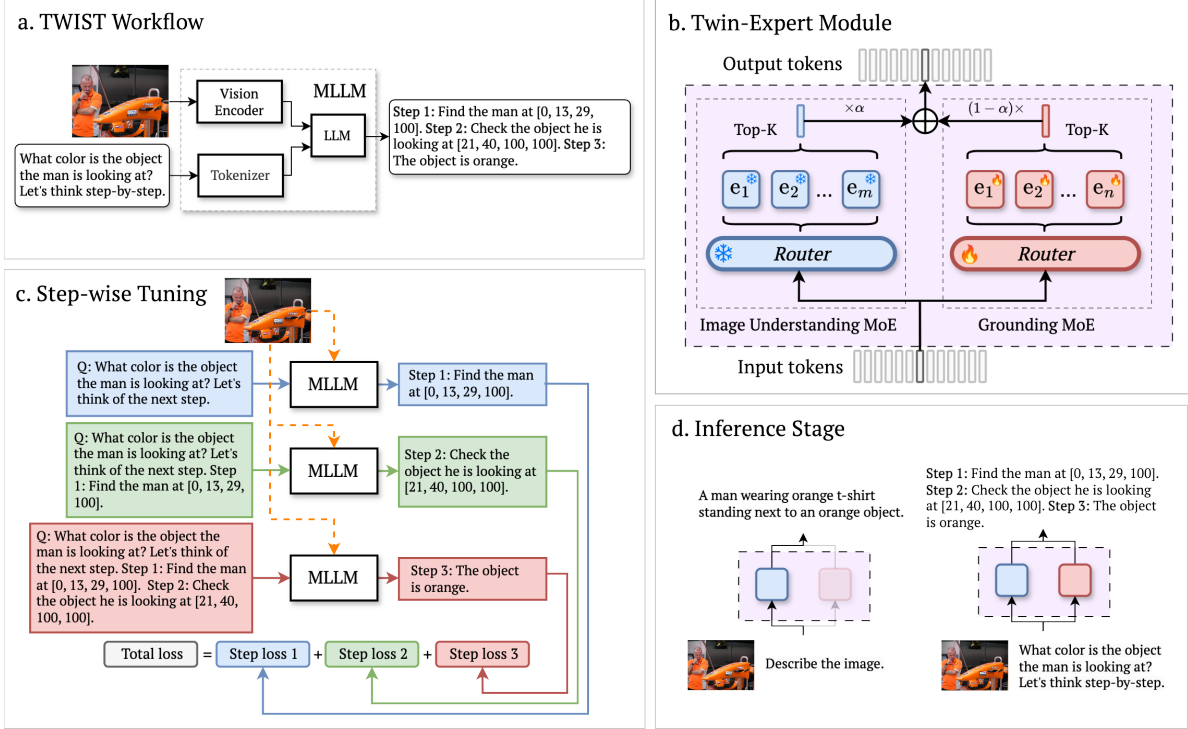


Figure 2. **TWIST system overview.** (a) The MLLM processes an image and text prompt via a vision encoder and language decoder to generate outputs, (b) Twin-Expert, featuring two parallel mixture of experts modules: a frozen one for image understanding and a trainable one for visual grounding, (c) The stepwise loss function breaks down complex reasoning into sequential subtasks, simplifying the training process, (d) During inference, information flows through the image understanding module (blue box) for those tasks and through both modules (blue and red box) for grounding tasks.

$$\mathcal{L}_{\text{step-by-step}} = \sum_{j=1}^J \left[ - \left( \sum_{i=1}^{N_j} \log P_{\theta}(t_i^{(j)} | v_1, \dots, v_m, t_1^{(j)}, \dots, t_{i-1}^{(j)}) \right) \right] + \lambda \cdot R(g), \quad (8)$$

where  $\mathcal{L}_{\text{step-by-step}}$  represents the step-by-step reasoning loss function,  $J$  is the number of reasoning steps,  $N_j$  is the number of tokens in step  $j$ ,  $t_i^{(j)}$  represents the  $i^{\text{th}}$  token in the  $j^{\text{th}}$  step output,  $v_1 \dots v_m$  are the image tokens,  $P_{\theta}$  is the probability predicted by the model and  $R(g)$  is the regularization term with weight  $\lambda$ .

**Inference step.** During inference, we determine the task type (image understanding or visual grounding) based on the input prompt and adjust  $\alpha$  accordingly. We employ a lightweight BERT-based classifier [12] which takes an input prompt and classifies it into one of the two task categories. Based on the classifier’s output,  $\alpha$  is adjusted dynamically:

$$\alpha = \begin{cases} 1 & \text{for Image Understanding,} \\ \text{unchanged} & \text{for Visual Grounding.} \end{cases} \quad (9)$$

Thus, at test time, the output of the twin-expert module, as

depicted in Figure 2 (d), is as follows:

$$l_{n+1} = \begin{cases} l_{n+1}^{\text{IU}} & \text{for Image Understanding,} \\ \alpha \cdot l_{n+1}^{\text{IU}} + (1-\alpha) \cdot l_{n+1}^{\text{VG}} & \text{for Visual Grounding.} \end{cases} \quad (10)$$

The BERT classifier adds minimal computational overhead, as it is an 8-bit quantized tiny model with approximately 1 million parameters, bringing the total active parameters from 1.67B to 1.671B. Our experiments show that the classifier achieves 99.98% accuracy, ensuring negligible impact on performance.

## 4. SCOUT

**Preliminaries.** Visual question answering datasets often involve spatial reasoning, such as “*What object is to the left of the girl?*” or “*Is there a bowl on top of the table?*”. Grounding tasks benefit from this reasoning, as describing relationships like “A cat at [x1, y1, x2, y2] sits to the left of a dog at [a1, b1, a2, b2]” provides clearer relative positioning, improving localization interpretation for MLLMs. Recent works like Shikra [6] have explored grounded chain-of-thought multimodal datasets, using LLMs to generate reasoning-based Q&A pairs from image captions—without direct visual access. However, relying solely on captions leads to hallucinated narratives that fail to reflect the actual

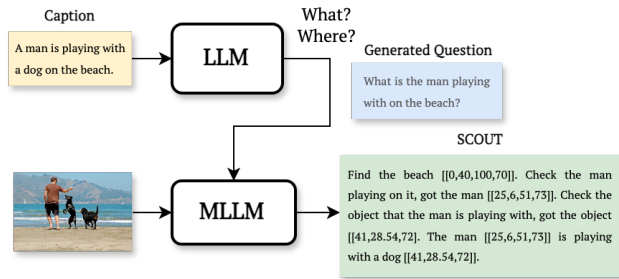


Figure 3. **SCOUT data generation.** We use an LLM to generate “what” and “where” questions from the input caption. With these questions and the image, we prompt an MLLM to produce the SCOUT grounding dataset, ensuring visually grounded and contextually relevant data.

image (see hallucination examples of the Shikra dataset in Figure A.3 of our Appendix).

**SCOUT data generation.** To generate our SCOUT dataset with high-quality, visually grounded data, we use a two-step process aimed at reducing text-only biases (see Figure 3). Starting with an image-caption pair from Flickr30k [34], we prompt an LLM like Mixtral [18] in-context to produce spatial reasoning questions of the “what” and “where” type, centered on captioned objects. This ensures relevance and grounding in the original description. To improve robustness, we also generate negative samples by asking about objects absent from the image, training the model to detect invalid or irrelevant queries. See Figures A.5 and A.6 in the Appendix. To reduce hallucinations—specifically, the kind caused by relying solely on text descriptions without verifying or aligning with the actual visual content—we use a state-of-the-art MLLM, CogVLM [40], for answer generation which is recognized for its strong performance in visual grounding and reasoning tasks. For the positive samples, we feed CogVLM the image and the question, prompting it to analyze the visual scene step-by-step, ensuring the answers accurately reflect objects and spatial relationships present in the image. For the negative samples, we skip feeding them to the MLLM and instead explicitly indicate that the referenced object is not in the image. This approach helps the model learn to distinguish valid questions from those that are irrelevant or incorrect. Since we rely on CogVLM for generating SCOUT, the quality of our data—and consequently, our model’s upper bound performance—is inherently tied to CogVLM’s reasoning capabilities.

**SCOUT data quality.** In a small-scale human analysis of 100 randomly selected samples, SCOUT achieved an accuracy of 94.7%, significantly outperforming Shikra’s 63.1%. A response was considered correct if the predicted object relationships and spatial positions matched the ground truth with at least 50% Intersection over Union (IoU) for bounding boxes and accurately described the relative spatial arrangement between objects based on the image content.

## 5. Experiments

We evaluate our approach on three grounding tasks: i) object localization, ii) grounded image captioning, and iii) visual grounding, as well as standard image understanding tasks. Below, we detail our architectural implementation and training datasets.

**Implementation details.** Our twin-expert module is built on MoE-LLaVA [25], which uses Phi-2 as its pre-trained language model. MoE-LLaVA has four experts for image understanding tasks, and we add a separate MoE with two experts for grounding, initialized from the image understanding MoE in the same decoder layer. The vision encoder and multi-head attention layers remain frozen, as do the interleaved decoder layers. We optimize the model with AdamW [30] using a  $2e-5$  learning rate, training on four A6000 GPUs for 1.5 days. The model contains 1.67B trainable parameters, with 0.8B active. We will release the code and our synthetic datasets.

**Datasets.** In addition to SCOUT, we train our model on two established datasets. We use the RefCOCO dataset [46], comprising three splits—RefCOCO, RefCOCO+, and RefCOCO<sub>g</sub>—with a total of 128,000 image-referential expression pairs from COCO2014 [26]. We also use 108k images from COCO2017, excluding the 6549 unique RefCOCO val/test images to prevent data leakage. Since COCO provides only object labels and bounding boxes, we use CogVLM to generate grounded image captions, forming the GIC dataset. TWIST & SCOUT refers to TWIST trained on REC, GIC, and SCOUT, using 512k samples from SCOUT unless otherwise specified.

### 5.1. Object Localization

**Setup.** We evaluate single-object localization—a core grounding capability—by comparing TWIST & SCOUT with PIN [13] and LoRA [17] in Table 1. PIN’s evaluation requires generating bounding boxes when prompted with object names. Their evaluation is conducted on *subsets* of COCO [26], Pascal VOC (PVOC) [15], and LVIS [16], with up to three objects per image, totaling 3,582, 2,062, and 6,016 test images, respectively. The mean Intersection over Union (mIoU) is reported for all bounding boxes, along with separate scores for medium ( $32 \times 32$  to  $96 \times 96$  pixels) and large (over  $96 \times 96$  pixels) objects, quantifying overlap between predicted and true boxes.

**Results.** Although TWIST and PIN use different backbones—complicating direct comparisons—Table 1 shows that TWIST & SCOUT outperforms PIN trained on the OpenFlamingo [3] backbone in single-object localization, improving mIoU by 22% on PVOC, 32% on COCO, and 39% on LVIS, particularly excelling with medium objects. Meanwhile, fine-tuning MoE-LLaVA via LoRA underperforms across all datasets, reinforcing the need for our approach. Notably, TWIST’s improvement over its LoRA counterpart

Method	Model	PVOC <sub>≤3 Objects</sub>			COCO <sub>≤3 Objects</sub>			LVIS <sub>≤3 Objects</sub>		
		mIoU	mIoU <sub>M</sub>	mIoU <sub>L</sub>	mIoU	mIoU <sub>M</sub>	mIoU <sub>L</sub>	mIoU	mIoU <sub>M</sub>	mIoU <sub>L</sub>
PIN	OpenFlamingo	0.45	0.27	0.62	0.35	0.26	0.59	0.26	0.24	0.61
LoRA	OpenFlamingo	0.44	0.26	0.62	0.33	0.23	0.58	0.23	0.19	0.55
LoRA	MoE-LLaVA	0.43	0.21	0.65	0.36	0.29	0.60	0.24	0.21	0.62
<b>TWIST &amp; SCOUT</b>	<b>MoE-LLaVA</b>	<b>0.68</b>	<b>0.58</b>	<b>0.81</b>	<b>0.66</b>	<b>0.57</b>	<b>0.78</b>	<b>0.65</b>	<b>0.55</b>	<b>0.76</b>

Table 1. **Object localization comparison** with PIN [13] and LoRA [17] on three benchmarks. TWIST consistently outperforms PIN across various datasets and metrics. Although PIN and TWIST use different backbones, making direct comparisons tricky, the LoRA variants perform on par, but TWIST shows a much larger improvement over its LoRA variant compared to PIN.

exceeds that of PIN over its own LoRA variant, demonstrating TWIST’s superior adaptability to new grounding tasks without erasing pre-trained vision-language expertise.

## 5.2. Visual Grounding

**Setup.** We compare our models to existing literature on the following two types of visual grounding tasks:

▷ **Grounded Image Captioning.** Grounded image captioning extends object detection by requiring models to recognize and localize objects within free-form text. Unlike standard detection tasks with predefined categories, this task generates structured outputs while aligning textual and visual elements. The lack of a standard evaluation protocol complicates model comparisons. To address this, we propose a protocol that maps object names from different MLLMs to COCO class labels using a sentence transformer [35] (Figure 1 (c)). We then evaluate models with COCO-style metrics, leveraging standardized annotations for consistency and fairness.

▷ **Referential Expression Comprehension.** Referential expression comprehension (REC) focuses on identifying a single object in an image based on a descriptive query. We evaluate this task using the RefCOCO [46] dataset, where models must accurately localize the target object given natural language descriptions which requires a deeper understanding of contextual relationships.

**Results.** Table 2 compares models on referential expression comprehension (REC) and grounded image captioning (GIC), highlighting their strengths and limitations. For REC, Grounding DINO [29] achieves the highest accuracy (green), as expected for a specialized object detector, while Ferret-7B [45] and Shikra-7B [6] perform competitively (orange) due to large-scale pre-training. TWIST & SCOUT remains on par, showing that fine-tuning preserves strong grounding capabilities, whereas PIN underperforms (red), revealing the limitations of its synthetic training. For GIC, Grounding DINO fails entirely (red) due to its lack of language capabilities. TWIST & SCOUT achieves the best performance (green), surpassing Ferret-7B by 2.2 AP<sub>50</sub>, reinforcing the advantage of fine-tuning VLMs for multi-object grounding. While Ferret-7B and Shikra-7B perform well (orange), they still fall short, showing that pre-training alone is insufficient

for mastering both spatial and semantic reasoning. Check Table A.3 and A.4 in Appendix for full comparison.

These results confirm our core hypothesis: models trained for one task struggle with another. Grounding DINO excels in REC but fails in GIC, while Ferret-7B and Shikra-7B perform moderately in both but do not surpass our fine-tuned approach. TWIST & SCOUT bridges this gap, adding grounding abilities to MLLMs while preserving vision-language understanding—without full retraining.

## 5.3. Image Understanding

**Setup.** An appealing characteristic of TWIST & SCOUT is its ability to retain image understanding capabilities even after fine-tuning for grounding tasks.

**Results.** As shown in Table 3, our approach matches the performance of MoE-LLaVA (our base) and is better than much larger models like LLaVA-phi2 [28], despite being nearly ten times smaller. The reported numbers, except for MME, reflect accuracy scores, while MME represents a cumulative perception score with a maximum value of 2000.

## 5.4. Ablations

**Component ablation.** Table 4 breaks down the contribution of each component in TWIST & SCOUT. Without TWIST, the model completely lacks image understanding, as reflected in the 0 MM-Vet score. Introducing TWIST restores image understanding (34.3 MM-Vet) while slightly improving grounding performance (+0.8 in RefCOCO, +1.8 in COCO), indicating that retaining pre-trained knowledge benefits grounding to some extent. Adding SCOUT further enhances grounding, boosting RefCOCO by 1.3 and COCO by 2.3, confirming its role in improving spatial reasoning. Finally, applying step-wise loss leads to the best performance, particularly on COCO (+1.3), showing that structured learning helps integrate SCOUT’s knowledge more effectively.

Beyond these components, we analyze the impact of  $\alpha$ -gating, which facilitates knowledge transfer from image understanding to grounding. Instead of learning a standalone grounding module,  $\alpha$  controls how much pre-trained features are reused, ensuring the grounding module learns delta features rather than redundant representations. Replacing the

Method	Parameters	Type	RefCOCO			GIC		
			val	test-A	test-B	AP	AP <sub>50</sub>	AP <sub>L</sub>
Shikra-7B [6]	7.0B	pre-trained	87.0	90.6	80.2	13.2	46.8	16.7
Grounding DINO [29]	172M	pre-trained	90.6	93.2	88.2	0	0	0
Ferret-7B [45]	7.0B	pre-trained	87.5	91.3	82.4	13.9	47.1	17.4
PIN [13]	1.2M	fine-tuned	n.a.	26.4	n.a.	0	0	0
TWIST & SCOUT	1.6B	fine-tuned	87.2	90.2	80.3	15.0	49.3	19.1

Table 2. **Visual grounding.** Object detectors like Grounding DINO excel in REC but fail in GIC, while pre-trained models like Ferret-7B and Shikra-7B perform moderately in both. TWIST & SCOUT bridges this gap, achieving the best GIC performance while maintaining strong REC results, demonstrating the benefit of incremental fine-tuning over full retraining. Note that **red** indicates failure, **orange** represents moderate performance, and **green** highlights the best performance.

Method	Parameters	Image Question Answering			Benchmark Toolkit			
		GQA	SQA <sup>1</sup>	VQA <sup>T</sup>	POPE	MME	LLaVA <sup>W</sup>	MM-Vet
PIN [13]	1.2M	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.
LLaVA-phi2	13.0B	–	68.4	48.6	85.0	1335.1	–	28.9
MoE-LLaVA-phi2 (our base)	3.6B	61.4	68.5	51.4	86.3	1423.0	94.1	34.3
<b>TWIST &amp; SCOUT</b>	<b>1.6B</b>	<b>61.4</b>	<b>68.5</b>	<b>51.4</b>	<b>86.3</b>	<b>1423.0</b>	<b>94.1</b>	<b>34.3</b>

Table 3. **Image understanding comparison.** We retain the image understanding abilities of our base model (MoE-LLaVA) through the twin-expert step-wise tuning framework, while PIN fails in image understanding tasks. Note that “n.a.” denotes that the corresponding method is inherently incapable of performing the specified task and “–” means the numbers are not reported by the baselines.

TWIST	SCOUT	Step-wise	MM-Vet	RefCOCO	COCO
×	×	×	0	84.8	9.6
✓	×	×	34.3	85.6	11.4
✓	✓	×	34.3	86.9	13.7
✓	✓	✓	34.3	87.2	15.0

Table 4. **Component ablation.** TWIST preserves image understanding (MM-Vet), SCOUT enhances grounding abilities, and the step-wise loss simplifies SCOUT, making grounding easier to learn.

learned  $\alpha=0.31$  with  $\alpha=0$  disrupts this transfer, dropping model performance from 15 mAP to 0, confirming its necessity. These results validate our approach: TWIST ensures knowledge retention, SCOUT enhances grounding, stepwise tuning refines learning efficiency, and  $\alpha$ -gating enables effective feature reuse across tasks.

**Fine-tuning challenges.** We analyze the limitations of standard fine-tuning strategies in Table 5. Adapting a pre-trained MLLM for both image understanding and grounding is challenging—training on LLaVA-mix-665k (GQA, SQA, VQA) preserves image understanding but prevents grounding, while training on SCOUT erases image understanding, causing the model to generate bounding boxes instead of textual answers. Even training on both datasets together remains suboptimal, as the model struggles to balance both tasks. This issue worsens when adding a dataset with a domain shift. To test this, we train MoE-LLaVA in a multi-task setting using both VQA-RAD [21], a biomedical VQA dataset, and LLaVA-mix-665k simultaneously. As shown in

Table 6, MoE-LLaVA suffers a drop across all tasks, failing to generalize between biomedical reasoning and standard visual question answering. In contrast, TWIST & SCOUT fine-tunes each module separately while preserving pre-trained knowledge, maintaining strong performance across all tasks. These results show that standard fine-tuning struggles to integrate new abilities without degrading existing ones, especially with domain shifts. TWIST & SCOUT overcomes this by retaining image understanding while incorporating domain-specific reasoning, demonstrating the benefits of a modular, task-adaptive fine-tuning strategy.

**Effect of number of experts.** Table 7 examines the impact of varying experts in the grounding MoE module, evaluated on RefCOCO and RefCOCO+ test-A and test-B splits. A single expert (equivalent to a simple MLP) underperforms compared to multiple experts, reinforcing our choice of MoEs for flexible parameter allocation to meet grounding tasks’ computational demands. Increasing experts from 1 to 2 yields substantial gains, validating the need for multiple experts. However, increasing from 2 to 4 provides only marginal improvements while doubling trainable parameters, making it inefficient. Thus, we adopt the 2-expert configuration for the best balance of performance and efficiency.

**Backbone ablation.** We assess TWIST’s flexibility by testing different backbones, as shown in Table 8. While TWIST is built on MoE-LLaVA, replacing it with LLaVA still enables effective grounding, achieving 85.7 test-A / 77.2 test-B on RefCOCO and 81.9 test-A / 63.8 test-B on

Method	Datasets		Question Answering			Visual Grounding		
	LLaVA-mix-665k	SCOUT	GQA	SQA	VQA <sup>T</sup>	AP	AP <sub>50</sub>	AP <sub>L</sub>
MoE-LLaVA	✓	×	61.4	68.5	51.4	0	0	0
	✓	✓	53.1	56.9	46.3	8.1	32.6	10.3
	×	✓	0	0	0	10.7	35.2	12.9
<b>TWIST</b>	×	✓	61.4	68.5	51.4	15.0	49.3	19.1

Table 5. **Fine-tuning challenges for image understanding and grounding tasks.** Fine-tuning on one task leads to catastrophic forgetting of the other, while joint fine-tuning remains suboptimal. TWIST & SCOUT preserves both abilities effectively.

Methods	VQA <sup>T</sup>	VQA-RAD
MoE-LLaVA	31.7	28.5
<b>TWIST</b>	51.4	63.1

Table 6. **Fine-tuning with domain shift.** Adding the biomedical VQA-RAD degrades MoE-LLaVA’s performance, while TWIST & SCOUT maintains strong results across all tasks.

Experts	Parameters	RefCOCO		RefCOCO+	
		test-A	test-B	test-A	test-B
1	0.8B	79.8	71.6	78.4	60.2
2	1.6B	90.2	80.3	87.7	71.9
4	3.3B	90.3	80.5	88.0	72.1

Table 7. **Effect of number of experts.** A single expert (MLP) underperforms, validating the need for MoEs. Two experts match four in performance while using half the parameters.

RefCOCO+. Though MoE-LLaVA performs better due to its expert-based design, these results confirm that TWIST is a general framework adaptable to different base models without being architecture-specific.

Methods	RefCOCO		RefCOCO+	
	test-A	test-B	test-A	test-B
LLaVA	85.7	77.2	81.9	63.8
<b>MoE-LLaVA</b>	90.2	80.3	87.7	71.9

Table 8. **Backbone ablation.** TWIST generalizes across backbones, enabling grounding when replacing MoE-LLaVA with LLaVA.

**Impact of fine-tuning datasets.** Table 9 shows the effect of different fine-tuning datasets on TWIST’s performance. Using only Visual Genome (VG) degrades performance (AP: 9.2, AP<sub>50</sub>: 38.5) due to its noisy annotations. Adding SCOUT improves results (AP: 12.7, AP<sub>50</sub>: 44.1), while training exclusively on SCOUT yields the best performance (AP: 15.0, AP<sub>50</sub>: 49.3). This highlights the importance of high-quality, visually grounded data, with SCOUT providing

a cleaner, more informative signal than VG.

Dataset Type		COCO	
VG	SCOUT	AP	AP <sub>50</sub>
✓	×	9.2	38.5
✓	✓	12.7	44.1
×	✓	15.0	49.3

Table 9. **Impact of fine-tuning datasets.** Adding the visual genome (VG) dataset degrades performance due to noisy labels, while incorporating SCOUT enhances grounding effectiveness.

**Scaling properties of SCOUT.** We assess SCOUT’s impact on localization by varying dataset size from 64k to 3M samples, as shown in Table 10. TWIST’s performance improves steadily, particularly up to 1M samples, after which gains plateau. This saturation occurs because SCOUT inherits CogVLM’s 3-object-per-image limitation, meaning that beyond 512k samples, additional data increases quantity but not diversity in grounding information. Thus, further scaling becomes ineffective, emphasizing dataset quality over sheer volume for improving grounding performance.

Datasets	64k	128k	256k	512k	1M	3M
PVOC	0.21	0.34	0.59	0.68	0.69	0.69
LVIS	0.20	0.38	0.61	0.65	0.67	0.67
COCO	0.10	0.12	0.13	0.14	0.15	0.15

Table 10. **Scaling properties of SCOUT** on localization tasks, showing improvements until saturation at 1M samples.

## 6. Conclusion

We propose TWIST, a fine-tuning framework that equips pre-trained MLLMs with visual grounding while preserving their image understanding capabilities. By leveraging SCOUT, a high-quality synthetic dataset, our approach enables effective grounding without full model retraining. Through rigorous evaluation, we demonstrate TWIST & SCOUT’s ability to enhance multimodal reasoning and localization, providing a scalable solution for integrating new skills into MLLMs.

**Acknowledgement.** This work has been financially supported by TomTom, the University of Amsterdam and the allowance of Top consortia for Knowledge and Innovation (TKIs) from the Netherlands Ministry of Economic Affairs and Climate Policy. Cees G. M. Snoek is also (partially) funded by the Horizon Europe project ELLIOT (GA No. 101214398). We also extend our gratitude to the anonymous reviewers for their valuable feedback and insightful suggestions during the rebuttal stage, which considerably improved this work.

## References

- [1] Pravesh Agrawal, Szymon Antoniak, Emma Bou Hanna, Devendra Chaplot, Jessica Chudnovsky, Saurabh Garg, Theophile Gervet, Soham Ghosh, Amélie Héliou, Paul Jacob, et al. Pixtral 12b. *arXiv preprint arXiv:2410.07073*, 2024. 2
- [2] 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. 2022. 1, 2
- [3] Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, et al. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*, 2023. 5
- [4] 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. 2
- [5] Junbum Cha, Wooyoung Kang, Jonghwan Mun, and Byungseok Roh. Honeybee: Locality-enhanced projector for multimodal llm. *arXiv preprint arXiv:2312.06742*, 2023. 2
- [6] 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. 1, 2, 4, 6, 7
- [7] Ting Chen, Saurabh Saxena, Lala Li, David J Fleet, and Geoffrey Hinton. Pix2seq: A language modeling framework for object detection. *arXiv preprint arXiv:2109.10852*, 2021. 2
- [8] Zhe Chen, Jiannan Wu, Wenhui Wang, Weijie Su, Guo Chen, Sen Xing, Zhong Muyan, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *arXiv preprint arXiv:2312.14238*, 2023. 2
- [9] Guangran Cheng, Chuheng Zhang, Wenzhe Cai, Li Zhao, Changyin Sun, and Jiang Bian. Empowering large language models on robotic manipulation with affordance prompting. *arXiv preprint arXiv:2404.11027*, 2024. 1
- [10] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *NeurIPS*, 2024. 1, 2
- [11] Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models. *arXiv preprint arXiv:2409.17146*, 2024. 2
- [12] Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 4
- [13] Michael Dorkenwald, Nimrod Barazani, Cees GM Snoek, and Yuki M Asano. Pin: Positional insert unlocks object localisation abilities in vlms. *CVPR*, 2024. 1, 2, 5, 6, 7
- [14] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023. 1
- [15] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. 5
- [16] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019. 5
- [17] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 1, 5, 6
- [18] Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024. 5
- [19] Chuhaio Jin, Wenhui Tan, Jiange Yang, Bei Liu, Ruihua Song, Limin Wang, and Jianlong Fu. Alphablock: Embodied fine-tuning for vision-language reasoning in robot manipulation. *arXiv preprint arXiv:2305.18898*, 2023. 1
- [20] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. 2
- [21] Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *NeurIPS*, 2023. 7
- [22] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. 2023. 1, 2
- [23] Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let’s verify step by step. *arXiv preprint arXiv:2305.20050*, 2023. 1, 3
- [24] Bin Lin, Bin Zhu, Yang Ye, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*, 2023. 2

- [25] Bin Lin, Zhenyu Tang, Yang Ye, Jiayi Cui, Bin Zhu, Peng Jin, Junwu Zhang, Munan Ning, and Li Yuan. Moe-llava: Mixture of experts for large vision-language models. *arXiv preprint arXiv:2401.15947*, 2024. 3, 5
- [26] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 5
- [27] Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. Mitigating hallucination in large multi-modal models via robust instruction tuning. In *ICLR*, 2023. 2
- [28] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *NeurIPS*, 2024. 1, 6
- [29] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 2, 6, 7
- [30] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 5
- [31] Jiasen Lu, Christopher Clark, Rowan Zellers, Roozbeh Motlaghi, and Aniruddha Kembhavi. Unified-io: A unified model for vision, language, and multi-modal tasks. In *ICLR*, 2022. 2
- [32] Sheng Luo, Wei Chen, Wanxin Tian, Rui Liu, Luanxuan Hou, Xiubao Zhang, Haifeng Shen, Ruiqi Wu, Shuyi Geng, Yi Zhou, et al. Delving into multi-modal multi-task foundation models for road scene understanding: From learning paradigm perspectives. *arXiv preprint arXiv:2402.02968*, 2024. 1
- [33] Brandon McKinzie, Zhe Gan, Jean-Philippe Fauconnier, Sam Dodge, Bowen Zhang, Philipp Dufter, Dhruvi Shah, Xianzhi Du, Futang Peng, Floris Weers, et al. Mm1: Methods, analysis & insights from multimodal llm pre-training. *arXiv preprint arXiv:2403.09611*, 2024. 2
- [34] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*, 2015. 5
- [35] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019. 6
- [36] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*, 2017. 3
- [37] Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *arXiv preprint arXiv:2406.16860*, 2024. 2
- [38] Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. Git: A generative image-to-text transformer for vision and language. *arXiv preprint arXiv:2205.14100*, 2022. 2
- [39] 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 *ICML*, 2022. 2
- [40] Weihang 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. 1, 2, 5
- [41] Wenhui 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. *NeurIPS*, 2024. 2
- [42] Licheng Wen, Xueming Yang, Daocheng Fu, Xiaofeng Wang, Pinlong Cai, Xin Li, Tao Ma, Yingxuan Li, Linran Xu, Dengke Shang, et al. On the road with gpt-4v (ision): Early explorations of visual-language model on autonomous driving. *arXiv preprint arXiv:2311.05332*, 2023. 1
- [43] Le Xue, Manli Shu, Anas Awadalla, Jun Wang, An Yan, Senthil Purushwalkam, Honglu Zhou, Viraj Prabhu, Yutong Dai, Michael S Ryoo, et al. xgen-mm (blip-3): A family of open large multimodal models. *arXiv preprint arXiv:2408.08872*, 2024. 2
- [44] Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Faisal Ahmed, Zicheng Liu, Yumao Lu, and Lijuan Wang. Unitab: Unifying text and box outputs for grounded vision-language modeling. In *ECCV*, 2022. 2
- [45] 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. *ICLR*, 2024. 6, 7
- [46] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *ECCV*, 2016. 5, 6
- [47] Haotian Zhang, Mingfei Gao, Zhe Gan, Philipp Dufter, Nina Wenzel, Forrest Huang, Dhruvi Shah, Xianzhi Du, Bowen Zhang, Yanghao Li, et al. Mm1. 5: Methods, analysis & insights from multimodal llm fine-tuning. *arXiv preprint arXiv:2409.20566*, 2024. 2
- [48] Pan Zhang, Xiaoyi Dong Bin Wang, Yuhang Cao, Chao Xu, Linke Ouyang, Zhiyuan Zhao, Shuangrui Ding, Songyang Zhang, Haodong Duan, Hang Yan, et al. Internlm-xcomposer: A vision-language large model for advanced text-image comprehension and composition. *arXiv preprint arXiv:2309.15112*, 2023. 2
- [49] Yanzhe Zhang, Ruiyi Zhang, Jiuxiang Gu, Yufan Zhou, Nedim Lipka, Diyi Yang, and Tong Sun. Lllavar: Enhanced visual instruction tuning for text-rich image understanding. *arXiv preprint arXiv:2306.17107*, 2023. 2