Exploring How Generative MLLMs Perceive More Than CLIP with the Same Vision Encoder

Siting Li, Pang Wei Koh, Simon Shaolei Du
University of Washington
{sitingli,pangwei,ssdu}@cs.washington.edu

Abstract

Recent research has shown that CLIP models struggle with visual reasoning tasks that require grounding compositionality, understanding spatial relationships, or capturing fine-grained details. One natural hypothesis is that the CLIP vision encoder does not embed essential information for these tasks. However, we find that this is not always the case: The encoder gathers query-relevant visual information, while CLIP fails to extract it. In particular, we show that another branch of Vision-Language Models (VLMs), Generative Multimodal Large Language Models (MLLMs), achieve significantly higher accuracy than CLIP in many of these tasks using the *same* vision encoder and weights, indicating that these Generative MLLMs *perceive more*—as they extract and utilize visual information more effectively. We conduct a series of controlled experiments and reveal that their success is attributed to multiple key design choices, including patch tokens, position embeddings, and prompt-based weighting. On the other hand, enhancing the training data alone or applying a stronger text encoder does not suffice to solve the task, and additional text tokens offer little benefit. Interestingly, we find that fine-grained visual reasoning is not exclusive to generative models trained by an autoregressive loss: When converted into CLIP-like encoders by contrastive finetuning, these MLLMs still outperform CLIP under the same cosine similarity-based evaluation protocol. Our study highlights the importance of VLM architectural choices and suggests directions for improving the performance of CLIP-like contrastive VLMs.

1 Introduction

Despite the success and widespread adoption of Contrastive Language-Image Pretraining (CLIP) (Radford et al., 2021), recent studies have pointed out that state-of-the-art CLIP models still fall short in various visual reasoning tasks, including Winoground (Thrush et al., 2022), SugarCREPE (Hsieh et al., 2024), and What'sUp (Kamath et al., 2023b). These benchmarks require vision-language models (VLMs) to pair images and captions, which are carefully designed to test model capabilities of visio-linguistic compositional reasoning, spatial reasoning, or fine-grained detail understanding—areas beyond standard zero-shot classification on ImageNet. While CLIP excels at the latter, its performance in these visual reasoning

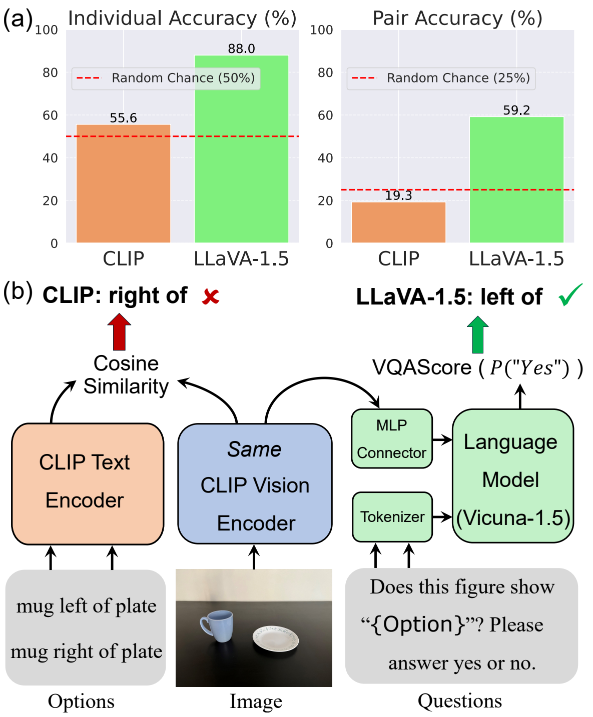

Figure 1: (a) Average two-way individual accuracy and pair accuracy of CLIP-ViT-L/14-336px and LLaVA-1.5-7B on various benchmarks (Kamath et al., 2023b; Hsieh et al., 2024; Thrush et al., 2022; Li et al., 2024; Yarom et al., 2024; Tong et al., 2024c). (b) CLIP and Generative MLLM architectures (using LLaVA-1.5 as an example) for fine-grained visual reasoning tasks. We observe that Generative MLLMs perform better in extracting and utilizing query-relevant information from the same vision encoder.

https://github.com/lst627/CLIP-Embeds

tasks remains poor.

One plausible explanation for these shortcomings is the potential information loss during the encoding process of the CLIP vision encoder (Tong et al., 2024c). For example, the encoder might behave like a bag-of-words model which only grasps the individual concepts in the image (“`mug`” and “`plate`” in Figure 1), but not the structural relationship (“`the mug is to the left of the plate`”) (Yuksekgonul et al., 2023).

In this work, we observe that the query-relevant visual information could still be preserved by CLIP vision encoder, but a better strategy is required to extract it: As shown in Figure 1, LLaVA-1.5-7B (Liu et al., 2024) with the *same* pretrained vision encoder, surpasses CLIP-ViT-L/14-336px by a large margin on many challenging visual reasoning benchmarks. Particularly, on spatial reasoning benchmark What'sUp, while CLIP's pair accuracy is lower than random chance (25%), LLaVA-1.5 achieves beyond 50% on all four subsets (Table 1). More evidence of other Generative MLLMs on various benchmarks showing this phenomenon is presented in Section 2. These results indicate that these Generative MLLMs extract and utilize query-relevant information more effectively from the same CLIP vision encoder. Notably, the vision encoder remains unchanged throughout training, ensuring a fair comparison.

What is the driving force behind Generative MLLMs' extracting more visual information and achieving strong visual reasoning performance? How can it benefit and improve CLIP-like contrastive VLMs? In Section 3, we investigate these questions by conducting controlled experiments on various factors as follows:

- **Training data.** In Section 3.1, we observe little performance gain after directly finetuning CLIP on LLaVA-1.5's training data and hard negatives, indicating that training data is not the only contributor.
- **Token usage and position embedding.** In Section 3.2, we observe that using patch tokens instead of the [CLS] token of CLIP (as proposed in PACL (Mukhoti et al., 2023)) brings improvement, and adding Rotary Position Embedding (RoPE) (Su et al., 2024) yields higher pair accuracy. However, using multiple text tokens from the CLIP text encoder as SPARC did (Bica et al., 2024) does not help.
- **Language models.** In Section 3.3, we replace the CLIP text encoder with a stronger, LLM-converted model (Huang et al., 2024), but it does not suffice to realize effective extraction and outperform random chance.
- **Architecture design for image-text alignment.** In Section 3.4, we find that text generation is not the only path to visual reasoning, as image-text matching through cosine similarity performed by contrastive VLMs can have strong performance on challenging benchmarks.
- **Training objective for image-text alignment.** In Section 3.4, we discover that finetuning with autoregressive loss is not necessary for deriving a VLM with fine-grained visual reasoning ability.
- **Question as prompt.** In Section 3.4, we also investigate the role of the question as a prompt for Generative MLLMs and find that, when fully fused with the image, it reweights the image tokens, significantly aiding in the extraction of relevant information and the enhancement of image embeddings.

In Section 4, we discuss the implications of our findings and their connection to prior work. Overall, we provide insights into VLM design and propose directions for improving contrastive VLMs.

2 Comparing CLIP and Generative MLLMs' visual reasoning performance

We begin by introducing the task setup for the comparison. Using score-based evaluation, we notice a significant performance gap between CLIP and Generative MLLMs with the same vision encoder across several challenging visual reasoning benchmarks, highlighting the latter's stronger ability to extract and utilize visual information for reasoning.

2.1 Task Setup

This paper focuses on the image-text matching task in which VLMs are asked to choose from captions for a given image or vice versa.

Benchmarks. We use several challenging benchmarks, **Winoground** (Thrush et al., 2022), **NaturalBench** (Li et al., 2024), **SeeTrue** (Yarom et al., 2024), **SugarCREPE** (Hsieh et al., 2024), for assessing VLMs' compositionality. In Winoground, each test case has two image+text pairs with the same words in different order. For NaturalBench, we use the retrieval version (denoted as NaturalBench-R) in the same format as Winoground provided by Lin et al. (2024). SeeTrue

	What'sUp Subset A				What'sUp Subset B			
	Left/Right		On/Under		Left/Right		Front/Behind	
	Indiv.	Pairs	Indiv.	Pairs	Indiv.	Pairs	Indiv.	Pairs
CLIP-ViT-L/14-336px	49.0	1.9	61.7	23.3	54.9	10.8	51.5	7.8
LLaVA-1.5-7B	96.6	93.2	76.2	52.4	98.5	97.1	**76.0**	**52.9**
Phi-3-V-3.8B	97.6	95.1	78.6	58.3	**100**	**100**	61.8	26.5
LLaMA-3-V-8B	**98.1**	**96.1**	**81.1**	**64.1**	**100**	**100**	73.0	47.1
Random chance	50.0	25.0	50.0	25.0	50.0	25.0	50.0	25.0

Table 1: The two-way individual accuracy and pair accuracy of CLIP-ViT-L/14-336px and Generative MLLMs in percentage points on four subsets of What'sUp. Generative MLLMs outperform CLIP by a large margin.

	Winoground	NaturalBench-R	MMVP	MMVP-VLM
CLIP-ViT-L/14-336px	27.8	47.8	14.0	20.7
LLaVA-1.5-7B	39.8	52.2	36.0	**49.6**
Phi-3-V-3.8B	35.8	50.5	30.7	31.9
LLaMA-3-V-8B	**46.3**	**64.7**	**50.0**	**49.6**
Random chance	25.0	25.0	25.0	25.0

Table 2: The pair accuracy of CLIP-ViT-L/14-336px and Generative MLLMs in percentage points on several paired benchmarks. Generative MLLMs achieve substantially better performance than CLIP.

consists of individual image-text pairs, while SugarCREPE has one image and two captions per test case. We use **MMVP(-VLM)** (Tong et al., 2024c) to test VLMs' ability to capture visual details like object existence, orientation, and counting. Since MMVP is not in paired image-text format, we manually convert it without altering content. We adopt **What'sUp A&B** with **COCO-spatial** and **GQA-spatial** (Kamath et al., 2023b) to evaluate VLMs' spatial reasoning. For What'sUp, each test case includes four captions (e.g., "`A dog left of/right of/on/under a table`") and corresponding images with minimal variation except for spatial relationships. We split each test case into two pairs—e.g., one pair contrasts "`left of`" versus "`right of`" with their ground truth images, and the other covers the remaining captions. This yields four benchmark subsets for A and B. COCO-spatial and GQA-spatial have one image and two captions per test case. More details are in Appendix A.1.

Models. Our main comparison is between **CLIP-ViT-L/14-336px** (Radford et al., 2021) and Generative MLLMs that use its pretrained vision encoder and keep the weights frozen during training: **LLaVA-1.5-7B** (Liu et al., 2024), along with **Phi-3-V-3.8B** and **LLaMA-3-V-8B** (Rasheed et al., 2024). In these MLLMs, the patch tokens from the CLIP vision encoder first pass through a two-layer MLP connector and are then used as input tokens for a generative language model which yields the token probability determining the model response. We also include results of **CLIP-ViT-L/14-224px**, **SigLIP-ViT-L/16-384px** (Zhai et al., 2023), and **EVA01-ViT-g-14** (Sun et al., 2023) for reference since they are of interest and widely used (Tong et al., 2024a).

Evaluation protocol. For CLIP-like contrastive VLMs, the matching score is the cosine similarity between its image embeddings and text embeddings. In prior works, Generative MLLMs are commonly evaluated by GPT-4 (Achiam et al., 2023) or human evaluators on generated responses. However, human evaluators are expensive for thousands of model responses, and GPT-4 as the judge can be incorrect and affected by user prompts. To ensure a fair comparison, we choose to use a score-based evaluation method and adopt the VQAScore (Lin et al., 2024), defined as

$$P(\text{“Yes”}|\texttt{image}, \text{“Does this figure show ‘}\texttt{text}\text{’? Please answer yes or no.”})$$

The question template remains the same across different benchmarks. We present the comparison between VQAScore and response-based evaluation in Appendix A.3.

Evaluation metrics. For SeeTrue, we report an average AUROC of three subsets. For other benchmarks, we use pair accuracy and individual accuracy when applicable. **Pair accuracy** (Tong et al., 2024c; Kamath et al., 2023b) requires correct

	SugarCREPE	SeeTrue	**What'sUp A**	**What'sUp B**	**COCO-spatial**		**GQA-spatial**	
					One-obj.	Two-obj.	One-obj.	Two-obj.
CLIP-ViT-L/14-224px	79.2	62.6	26.7	25.7	49.1	50.2	46.0	48.1
CLIP-ViT-L/14-336px	80.0	63.0	28.9	27.2	48.9	51.1	46.6	49.1
SigLIP-ViT-L/16-384px	85.2	66.8	26.7	28.7	50.3	48.6	47.8	48.7
EVA01-ViT-g-14	81.1	64.9	28.2	27.9	45.9	50.5	44.4	49.8
LLaVA-1.5-7B	88.5	76.0	**69.9**	**65.4**	89.9	**88.9**	94.6	**95.2**
Phi-3-V-3.8B	82.8	73.7	66.0	52.7	89.5	79.8	93.0	87.3
LLaMA-3-V-8B	**91.2**	**80.7**	66.7	58.6	**91.9**	78.9	**95.3**	91.4
Random chance	50.0	50.0	25.0	25.0	50.0	50.0	50.0	50.0

Table 3: Individual accuracy or AUROC of varied VLMs on visual reasoning benchmarks (spatial reasoning benchmarks in bold). The Generative MLLMs consistently outperform CLIP models (except on SugarCREPE, where SigLIP is better than Phi-3-V), with the largest performance gap observed in spatial reasoning.

matching for both images, and it only applies to benchmarks with two images in a test case. **Individual accuracy** refers to the accuracy of individual images. For MMVP(-VLM), we follow the original paper and use pair accuracy to represent the correct matching for both *captions* and individual accuracy for individual *captions* instead.

2.2 Results

We present the comparison in Table 1, 2, and 3. On these challenging benchmarks, Generative MLLMs outperform CLIP-ViT-L/14-336px with the same vision encoder, showing that (1) CLIP vision encoder has much query-relevant visual information not utilized by CLIP, and (2) Generative MLLMs can extract and align this information from the encoder more effectively. The performance gap is the most significant on **spatial reasoning**, where the CLIP models behave close to random chance for individual accuracy and lower than random chance for pair accuracy, but Generative MLLMs achieve high accuracies. We further find that the Generative MLLMs can even outperform XVLM (Zeng et al., 2021) specialized in spatial reasoning (See Appendix B.4).

3 Investigation of the Performance Gap

The gap observed in Section 2.2 could be the result of various factors, ranging from model training to architecture. In this section, we try to dissect and examine which factors contribute to Generative MLLMs' success and cause CLIP's failure by controlled experiments. We focus on the performance gap on **What'sUp**, of which the test cases are tightly controlled and balanced. A road map of the experiments is illustrated in Figure 2.

3.1 Training Data

First, we hypothesize that Generative MLLMs' visual information extraction ability benefits from training data. To check the effect of data, we use LLaVA-1.5's training data to finetune CLIP, SigLIP, and EVA-CLIP. We convert the datasets to the image-caption format (Details are deferred to Appendix B.2). By default, we freeze the vision encoder during finetuning for strict ablation. Considering that contrastive learning relies on negative samples beyond data quality (Robinson et al., 2020; Kalantidis et al., 2020), we also construct hard negative captions by switching the related phrases to their opposite (e.g., replacing "`on the left`" with "`on the right`"). In this setting, the training objective follows NegCLIP (Yuksekgonul et al., 2023).

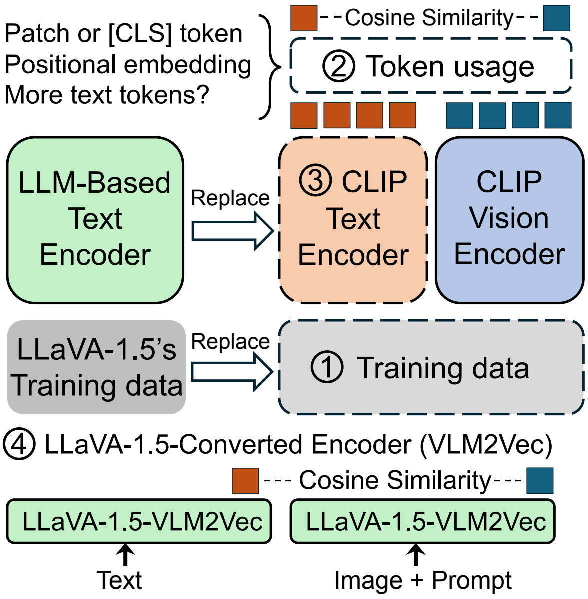

Figure 2: An illustration for CLIP-like contrastive VLMs and the controlled experiments in Section 3. We first investigate the effect of training data by replacing them with LLaVA-1.5's training data (①). Then, we try different token usage for CLIP vision encoder and text encoder (②) and discuss the influence of using stronger text encoders converted from LLMs (③). Finally, we convert LLaVA-1.5 to contrastive VLMs (④) to study the effect of the alignment architecture and training objective.

Results are shown in Table 4. Finetuning on

	What'sUp Subset A		What'sUp Subset B	
	Indiv.	Pairs	Indiv.	Pairs
CLIP	49.0	1.9	54.9	10.8
+ finetuning (ft)	50.5	1.9	53.9	5.9
+ ft + hard neg.	50.5	1.0	50.5	1.0
SigLIP	50.0	1.9	51.5	5.9
+ finetuning (ft)	49.0	1.0	51.0	3.9
+ ft + hard neg.	50.0	0.0	50.0	0.0
EVA-CLIP	49.0	1.0	50.1	4.9
+ finetuning (ft)	50.0	4.9	48.5	2.0
+ ft + hard neg.	50.0	1.9	48.0	2.0
Random chance	50.0	25.0	50.0	25.0

Table 4: The two-way individual accuracy and pair accuracy results of CLIP-ViT-L/14-336px, SigLIP-ViT-L/16-384px, and EVA01-ViT-g-14 focusing on the **Left/Right** subsets of What'sUp after finetuning on LLaVA-1.5's training data with or without hard negative captions. After direct finetuning, the accuracies are still quite low.

LLaVA-1.5's training data does not help these models, even with hard negatives. Still, their accuracy is around random chance. We also try to unlock the SigLIP vision encoder during finetuning, which does not increase the performance either (See results in Appendix B.3). We experiment with XVLM (Zeng et al., 2021) and observe similar results in Appendix B.4. This finding aligns with the previous failure on finetuning them on a much larger, preposition-focused subset of LAION (Kamath et al., 2023b), indicating that **data alone does not lead to stronger extraction ability.**

3.2 Token Usage

Patch tokens. The output of the CLIP vision encoder consists of two parts: The **`[CLS]` token**, functioning as the global feature of the image, and the **patch tokens**, containing local information of image patches. We notice that these Generative MLLMs employ all 576 patch tokens from the CLIP-ViT-L/14-336px vision encoder, in contrast to CLIP using only the projected `[CLS]` token.

We first perform an ablation study on LLaVA-1.5: We change the input of its language model to use only the `[CLS]` token, train this "`[CLS]`-LLaVA-1.5" model from scratch (pretraining + finetuning) using LoRA (Hu et al., 2021), and observe that its spatial reasoning performance is significantly worse than our reproduced LLaVA-1.5-LoRA in Table 5. This proves the importance of patch tokens to fine-grained visual reasoning: **Detailed information of images resides in these patch tokens.**

Inspired by this finding, we try incorporating patch tokens in standard CLIP models. We adopt the PACL method (Mukhoti et al., 2023) as it proposes to train a vision embedder e_v for patch tokens and a text embedder e_t on top of the frozen CLIP model (consisting of vision encoder f_v and text encoder f_t). For input image $\mathbf{x}$ and text $\mathbf{y}$, we calculate the image feature $\mathbf{v}(\mathbf{x})$ by

$$s(\mathbf{x}, \mathbf{y}) = e_v(f_v(\mathbf{x})) \cdot e_t(f_t(\mathbf{y}))$$

$$\mathbf{v}(\mathbf{x}) = e_v(f_v(\mathbf{x}))^\top \cdot \text{sigmoid}(10 \cdot s(\mathbf{x}, \mathbf{y}))$$

In other words, $s(\mathbf{x}, \mathbf{y})$ determines the weight for each projected patch token based on the text, and $\mathbf{v}(\mathbf{x})$ is a weighted sum of all projected patch tokens. Then we use $\mathbf{v}(\mathbf{x})$ and $e_t(f_t(\mathbf{y}))$ as the image and text features for CLIP training with the original contrastive objective. During the evaluation, we use the average of projected patch tokens $e_v(f_v(\mathbf{x}))$ as the image feature and $e_t(f_t(\mathbf{y}))$ as the text feature. The results of training on LLaVA-1.5's data are shown in the second row of Table 6. It brings higher pair accuracy to the Left/Right subset in Subset A.

Position embeddings. Considering that the average or weighted sum does not maintain the order/positional information of patch tokens, we add Rotary Position Embeddings (RoPE) (Su et al., 2024) to $f_v(\mathbf{x})$ before passing it to the vision embedder e_v, since RoPE is applied to visual tokens in the language model of Generative MLLMs we study. In the third row of Table 6, we find that this combination yields significantly higher pair accuracy on three subsets, showing that **part of the information comes from the order of patch tokens.** Nonetheless, the individual accuracy is not improved by much.

Multiple text tokens. Does using multiple text tokens of CLIP text encoder as well offer further performance gain? In Generative MLLMs, it is natural to use multiple visual tokens and text tokens, as they are concatenated as the input of an autoregressive language model. However, it is non-trivial to do so in contrastive VLMs. Therefore, we leverage the SPARC method (Bica et al., 2024) to implement the interaction between multiple visual tokens and text tokens for CLIP: We first obtain a weighted sum of patch tokens for each text token (named grouped visual tokens) and then perform local contrastive learning between grouped visual tokens and text tokens within each sample. The training objective is the sum of this local contrastive loss and the standard contrastive loss. For evaluation, we use the average of grouped visual tokens as the

	What'sUp Subset A				What'sUp Subset B			
	Left/Right		On/Under		Left/Right		Front/Behind	
	Indiv.	Pairs	Indiv.	Pairs	Indiv.	Pairs	Indiv.	Pairs
LLaVA-1.5-7B-LoRA	**84.5**	**68.9**	**76.2**	**52.4**	**89.2**	**78.4**	**86.3**	**72.5**
[CLS]-LLaVA-1.5-7B-LoRA	44.2	8.7	54.4	8.7	49.0	4.9	53.9	12.7
Random chance	50.0	25.0	50.0	25.0	50.0	25.0	50.0	25.0

Table 5: The results of [CLS]-LLaVA-1.5-7B-LoRA and reproduced LLaVA-1.5-7B-LoRA on all subsets of What'sUp, where [CLS]-LLaVA-1.5-7B-LoRA struggles with spatial reasoning.

	What'sUp Subset A				What'sUp Subset B			
	Left/Right		On/Under		Left/Right		Front/Behind	
	Indiv.	Pairs	Indiv.	Pairs	Indiv.	Pairs	Indiv.	Pairs
CLIP-ViT-L/14-336px	49.0	1.9	**61.7**	**23.3**	**54.9**	10.8	51.5	7.8
+ Patch Tokens (PT)	47.6	9.7	52.9	10.7	52.9	9.8	51.5	6.9
+ PT + RoPE	**54.9**	**22.3**	46.1	13.6	52.0	**20.6**	45.6	12.7
+ PT + RoPE + Multiple Text Tokens	48.1	0.0	50.0	2.9	50.0	6.9	48.0	7.8
+ PT + RoPE + Stronger Text Encoder	50.5	10.7	48.5	6.8	50.0	15.7	**53.9**	**21.6**
LLM2CLIP (Huang et al., 2024)	49.5	1.0	58.7	17.4	49.0	1.0	55.4	14.7
Random chance	50.0	25.0	50.0	25.0	50.0	25.0	50.0	25.0

Table 6: The results of different token usage and leveraging a stronger text encoder for CLIP-ViT-L/14-336px on the What'sUp benchmark after finetuning on LLaVA-1.5's training data. CLIP with Patch tokens + RoPE has the highest average pair accuracy.

image feature and the average of text tokens as the text feature. Details are deferred to Appendix B.5. Despite the complexity, this method does not help our task (See the fourth row of Table 6). This failure might result from the hardness of training and the insufficiency of token interaction.

3.3 Language Model

Previous research suggests that the CLIP text encoder fails to capture changed word orders, negation, and spatial or numerical details (Tong et al., 2024b; Kamath et al., 2023a; Yuksekgonul et al., 2023), while Generative MLLMs employ powerful pretrained LLMs, which is supposed to be stronger than the CLIP text encoder at reasoning.

Are pretrained LLMs the missing piece to effectively extracting visual information? We perform further experiments on finetuning CLIP with patch tokens and RoPE on LLaVA-1.5 training data but replacing the original CLIP text encoder with a stronger one provided by LLM2CLIP (Huang et al., 2024). This text encoder is converted from Llama-3-8B-Instruct (Dubey et al., 2024) by contrastive finetuning and is shown to bring performance boost to state-of-the-art CLIP models on benchmarks such as MS COCO (Lin et al., 2014). We keep this text encoder and the CLIP vision encoder frozen during our finetuning. The results are shown in the fifth row of Table 6, where we also attach the results of the original LLM2CLIP checkpoint of CLIP-ViT-L/14-336px for reference[1]. We find that **a stronger text encoder does not suffice to effectively extract more information towards solving the task.**

3.4 Alignment Architecture, Training Objective, and Prompt

A major difference between CLIP-like contrastive VLMs and LLaVA-like Generative MLLMs is how they align images and texts. However, it is hard to examine every factor involved separately: The alignment architecture of CLIP—cosine similarity between image embeddings and text embeddings—is bound to its training objective (contrastive loss) and contrastive VLM structure (dual encoders). On the other hand, it is plausible to hypothesize that contrastive VLMs cannot perform fine-grained visual reasoning since cosine similarity might be overly coarse-grained both for training and evaluation, compared with text generation and autoregressive loss used by Generative MLLMs.

[1] The original LLM2CLIP is not a fair comparison as its implementation unfreezes the CLIP vision encoder during finetuning.

	What'sUp Subset A				What'sUp Subset B			
	Left/Right		On/Under		Left/Right		Front/Behind	
	Indiv.	Pairs	Indiv.	Pairs	Indiv.	Pairs	Indiv.	Pairs
CLIP-ViT-L/14-336px	49.0	1.9	61.7	23.3	54.9	10.8	51.5	7.8
LLaVA-1.5-7B-VLM2Vec-LoRA	**97.1**	**95.1**	**68.0**	**35.9**	**100**	**100**	**60.8**	**22.5**
w/o Question in Prompt	49.5	0.0	50.5	1.9	46.6	2.0	50.5	1.0
Random chance	50.0	25.0	50.0	25.0	50.0	25.0	50.0	25.0

Table 7: The two-way individual accuracy and pair accuracy of CLIP-ViT-L/14-336px and LLaVA-1.5-converted models in percentage points on four subsets of What'sUp. LLaVA-1.5-7B-VLM2Vec-LoRA outperforms CLIP on all subsets. When there is no question in the prompt, its performance degenerates to the standard CLIP.

We bypass this obstacle in comparison by converting a Generative MLLM to a CLIP-like contrastive VLM. On LLaVA-1.5-7B, we use the converting method proposed by VLM2Vec (Jiang et al., 2024): Specifically, we take the last layer vector representation of the last output token of LLaVA-1.5-7B as the output embedding. In this way, we get the encoder for image+question(prompt) and pure text simultaneously since LLaVA-1.5 allows using one or zero images in the input. Following the original paper, we use the prompt templates: "`Represent the given image with the following question: {Question}`" while encoding the image if there is a question in the sample; "`Find the text that can answer the given query: {Question}`" when there is no image; and no additional prompt for encoding image+question of LCS-558K and the captions. Then, we finetune this encoder using contrastive loss and LoRA (Hu et al., 2021) on LLaVA-1.5's training data with the CLIP vision encoder frozen. Surprisingly, they exhibit strong performance without using a large batch size (256) in Table 7 (LLaVA-1.5-7B-VLM2Vec-LoRA). The question used for evaluation is listed in Appendix A.4. **This proves that text generation+autoregressive loss is not the only solution to fine-grained visual reasoning.**

What could be the key factor of the success of this contrastive LLaVA-1.5 compared with CLIP models, including the standard ones and ours with patch tokens plus RoPE in Section 3.2? We verify that the additional question added in the prompt when obtaining the image embeddings plays an important role here. When we change the prompt template to "`Represent the given image.`" without any question, the model performance degenerates to the standard CLIP performance as shown in the third row of Table 7. Therefore, we conclude that **the question greatly helps the extraction and utilization of visual information from the vision encoder.** The question helps to reweight the patch tokens according to the context. Without the question, the image embeddings remain the same regarding different tasks (e.g., coarse-grained classification like "`dog/cat`", versus fine-grained visual reasoning like "`dog to the left/right of the table`"), which could be suboptimal and cause difficulty in alignment.

4 Discussion and Connection to Prior Work

In this section, we first discuss how our findings connect to the observations and conclusions in existing literature. Then, we list two directions for improving VLM's visual reasoning ability based on our results.

4.1 Connection to Prior Work

Recent results of VLMs on various benchmarks for testing fine-grained visual reasoning ability (e.g., compositionality, spatial reasoning, counting) reveal that they fail to solve simple tasks unexpectedly and often ignore visual patterns in the image (Thrush et al., 2022; Yuksekgonul et al., 2023). Researchers are actively exploring the root causes of such failures. Lin et al. (2024) notices the advantage of Generative MLLMs over CLIP in image-text matching tasks. We observe the significant discrepancy in performance when controlling the vision encoder and thus focus on how Generative MLLMs could outperform CLIP-like contrastive VLMs with the same vision encoder.

Vision encoder and token usage. Tong et al. (2024c) observes that the CLIP vision encoder could encode visually distinct images into highly similar embeddings, omitting essential information and thus resulting in low accuracy on tasks regarding the visual semantic difference. Hence, they suggest using features from multiple vision encoders, which is adopted by later works (Kar et al., 2024; Tong et al., 2024a; Xu et al., 2024). However, we observe that this part of information could be captured by the CLIP vision encoder but is not extracted or aligned properly. Similar to our observation, Koishigarina et al. (2025) argues that CLIP is not bag-of-words uni-modally, and the real issue of CLIP's compositionality lies in poor cross-modal alignment. Besides, while Tong et al. (2024c) only calculates the similarity between `[CLS]` tokens used by CLIP as evidence, we argue that detailed information is preserved in patch tokens and their positions.

Text encoder. Kamath et al. (2023a) and Tong et al. (2024b) point out that the CLIP text encoder might discard relevant information during encoding so that the model could not discriminate images that differ in key aspects. Following previous efforts in converting LLM to an encoder (BehnamGhader et al., 2024), recent works explore using LLM-converted encoders as the text encoder for CLIP: LLM2CLIP (Huang et al., 2024) finds that this practice boosts performance on several retrieval tasks on top of state-of-the-art CLIP models, but we observe its unsatisfying performance on What'sUp; Stone et al. (2024) achieves high accuracy on challenging benchmarks for compositionality after large-scale pretraining, although they reported struggles on Left/Right spatial relations. We discover that a stronger text encoder is not enough for solving the fine-grained visual reasoning task.

Training data and objective. Data-centric methods for improving CLIP-like models include selecting or synthesizing higher-quality image-text pairs (Gadre et al., 2024; Nguyen et al., 2024; Zheng et al., 2024), involving more negative samples by manual design (Yuksekgonul et al., 2023; Paiss et al., 2023) or larger batch size (Stone et al., 2024). But Kamath et al. (2023b) observes that CLIP cannot learn spatial relations even after training on a large amount of relevant data, suggesting that we might need inductive bias or denser supervision like XVLM (Zeng et al., 2021). Others try applying autoregressive loss, such as Cap/CapPa (Tschannen et al., 2023), or combining it with contrastive loss, like CoCa (Yu et al., 2022). Inspired by VLM2Vec (Jiang et al., 2024), we train LLaVA-1.5-VLM2Vec and verify that task-specific inductive bias, additional supervision, manually designed hard negatives, or finetuning with autoregressive loss is not necessary for contrastive VLMs to learn spatial relations.

Alignment architecture of contrastive VLMs. Cross-modal alignment can be implemented by cross-modal matching through cosine similarity (Radford et al., 2021), matching by directly outputting a score (Li et al., 2023), and generation (outputting a response) (Liu et al., 2024; Awadalla et al., 2023). The cross-modal contrasting is efficient, yet unable to perform complex reasoning like the generative models where Chain-of-Thought is applicable (Wei et al., 2022). Nevertheless, our LLaVA-1.5-VLM2Vec experiments show that advanced techniques can ignite the potential of contrastive VLMs in visual information extraction and improve their visual reasoning performance.

4.2 Discussion on Improving VLMs' Visual Reasoning Ability

Promptable image embeddings boost performance on fine-grained tasks. CLOC (Chen et al., 2024) formulates the idea of promptable embedding for regional understanding of images. They pass image embeddings and spatial hints to a prompter for obtaining region representations of images and perform localized contrastive training. In this way, when a grounding task only requires information from part of the image, the representation will not be distracted by other parts and thus lead to higher accuracy. VLM2Vec (Jiang et al., 2024) extends the spatial hints to general prompts and proposes a method for converting Generative MLLMs to encoders. Our ablation study of questions in prompts for LLaVA-1.5-VLM2Vec demonstrates the effectiveness of this technique.

Effectively utilizing vision encoders offers benefits without pretraining new vision models. Our results suggest that there is still room to enhance VLMs with a fixed, pretrained vision encoder by advanced extraction methods. We explore whether this also holds for Generative MLLMs in the Appendix B.7: We try an alternative decoding algorithm on LLaVA-1.5-7B for attending more to the visual information, named Multi-Modal Mutual-Information Decoding (Favero et al., 2024), which leads to performance gain (+6%), on par with using interleaved visual tokens from multiple vision encoders (I-MoF (Tong et al., 2024c)). This result indicates that LLaVA-1.5 still misses some key information for query answering and has room for further improvement apart from using a better vision encoder.

5 Conclusion

Our study first reveals that Generative MLLMs perceive fine-grained visual information more effectively using the same vision encoder than CLIP for visual reasoning tasks. Through controlled experiments, we find that patch tokens, position embeddings, and prompt-based image embeddings are key differences causing the gap; however, training data, multiple text tokens, and better text encoders are insufficient to bridge the gap. Additionally, text generation and finetuning with autoregressive loss are not mandatory for strong visual reasoning. These findings not only offer insights into VLM design but also provide practical guidelines for enhancing contrastive VLMs on visual reasoning.

6 Limitations

First, for controlled experiments on data in Section 3.1, we do not train models from scratch or use larger batch sizes due to the limited computing resources, so the conclusion regarding data might be restricted.

Second, the number of visual reasoning benchmarks we study is restricted. Therefore, we hope that more comprehensive, unbiased, and visual-centric reasoning benchmarks for VLMs can be available in the future.

Third, we only study the comparison between CLIP-ViT-L/14-336px and the Generative MLLMs that use its vision encoder and explore the reasons behind their discrepancy. Our conclusion is thus restricted to them. We do not claim that all Generative MLLMs are better than contrastive VLMs in all cases. Nevertheless, it is interesting to compare other pairs of contrastive VLMs and Generative MLLMs, and we leave this for future work.

Acknowledgments

We thank Amita Kamath for the discussion about the What'sUp benchmark. We appreciate the comments and advice from Rui Xin and Scott Geng on our drafts. PWK is supported by the Singapore National Research Foundation and the National AI Group in the Singapore Ministry of Digital Development and Innovation under the AI Visiting Professorship Programme (award number AIVP-2024-001) and by the AI2050 program at Schmidt Sciences. SSD acknowledges the support of NSF DMS 2134106, NSF CCF 2212261, NSF IIS 2143493, NSF IIS 2229881, Sloan Fellowship, and the AI2050 program at Schmidt Sciences.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, Jenia Jitsev, Simon Kornblith, Pang Wei Koh, Gabriel Ilharco, Mitchell Wortsman, and Ludwig Schmidt. 2023. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*.

Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. 2024. Llm2vec: Large language models are secretly powerful text encoders. *arXiv preprint arXiv:2404.05961*.

Ioana Bica, Anastasija Ilić, Matthias Bauer, Goker Erdogan, Matko Bošnjak, Christos Kaplanis, Alexey A Gritsenko, Matthias Minderer, Charles Blundell, Razvan Pascanu, et al. 2024. Improving fine-grained understanding in image-text pre-training. *arXiv preprint arXiv:2401.09865*.

Hong-You Chen, Zhengfeng Lai, Haotian Zhang, Xinze Wang, Marcin Eichner, Keen You, Meng Cao, Bowen Zhang, Yinfei Yang, and Zhe Gan. 2024. Contrastive localized language-image pre-training. *arXiv preprint arXiv:2410.02746*.

Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, et al. 2022. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Alessandro Favero, Luca Zancato, Matthew Trager, Siddharth Choudhary, Pramuditha Perera, Alessandro Achille, Ashwin Swaminathan, and Stefano Soatto. 2024. Multi-modal hallucination control by visual information grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14303–14312.

Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. 2024. Datacomp: In search of the next generation of multimodal datasets. *Advances in Neural Information Processing Systems*, 36.

Gregor Geigle, Radu Timofte, and Goran Glavaš. 2024. African or european swallow? benchmarking large vision-language models for fine-grained object classification. *arXiv preprint arXiv:2406.14496*.

Cheng-Yu Hsieh, Jieyu Zhang, Zixian Ma, Aniruddha Kembhavi, and Ranjay Krishna. 2024. Sugarcrepe: Fixing hackable benchmarks for vision-language compositionality. *Advances in neural information processing systems*, 36.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Weiquan Huang, Aoqi Wu, Yifan Yang, Xufang Luo, Yuqing Yang, Liang Hu, Qi Dai, Xiyang Dai, Dongdong Chen, Chong Luo, et al. 2024. Llm2clip: Powerful language model unlock richer visual representation. *arXiv preprint arXiv:2411.04997*.

Drew A Hudson and Christopher D Manning. 2019. 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.

Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. 2021. Openclip.

Ziyan Jiang, Rui Meng, Xinyi Yang, Semih Yavuz, Yingbo Zhou, and Wenhu Chen. 2024. Vlm2vec: Training vision-language models for massive multimodal embedding tasks. *arXiv preprint arXiv:2410.05160*.

Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. 2020. Hard negative mixing for contrastive learning. *Advances in neural information processing systems*, 33:21798–21809.

Amita Kamath, Jack Hessel, and Kai-Wei Chang. 2023a. Text encoders bottleneck compositionality in contrastive vision-language models. *arXiv preprint arXiv:2305.14897*.

Amita Kamath, Jack Hessel, and Kai-Wei Chang. 2023b. What's" up" with vision-language models? investigating their struggle with spatial reasoning. *arXiv preprint arXiv:2310.19785*.

Oğuzhan Fatih Kar, Alessio Tonioni, Petra Poklukar, Achin Kulshrestha, Amir Zamir, and Federico Tombari. 2024. Brave: Broadening the visual encoding of vision-language models. *arXiv preprint arXiv:2404.07204*.

Darina Koishigarina, Arnas Uselis, and Seong Joon Oh. 2025. Clip behaves like a bag-of-words model cross-modally but not uni-modally. *arXiv preprint arXiv:2502.03566*.

Tiep Le, Vasudev Lal, and Phillip Howard. 2024. Coco-counterfactuals: Automatically constructed counterfactual examples for image-text pairs. *Advances in Neural Information Processing Systems*, 36.

Baiqi Li, Zhiqiu Lin, Wenxuan Peng, Jean de Dieu Nyandwi, Daniel Jiang, Zixian Ma, Simran Khanuja, Ranjay Krishna, Graham Neubig, and Deva Ramanan. 2024. Naturalbench: Evaluating vision-language models on natural adversarial samples. *arXiv preprint arXiv:2410.14669*.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer.

Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. 2024. Evaluating text-to-visual generation with image-to-text generation. In *European Conference on Computer Vision*, pages 366–384. Springer.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306.

Jishnu Mukhoti, Tsung-Yu Lin, Omid Poursaeed, Rui Wang, Ashish Shah, Philip HS Torr, and Ser-Nam Lim. 2023. Open vocabulary semantic segmentation with patch aligned contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19413–19423.

Thao Nguyen, Samir Yitzhak Gadre, Gabriel Ilharco, Sewoong Oh, and Ludwig Schmidt. 2024. Improving multimodal datasets with image captioning. *Advances in Neural Information Processing Systems*, 36.

Roni Paiss, Ariel Ephrat, Omer Tov, Shiran Zada, Inbar Mosseri, Michal Irani, and Tali Dekel. 2023. Teaching clip to count to ten. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3170–3180.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Kanchana Ranasinghe, Satya Narayan Shukla, Omid Poursaeed, Michael S Ryoo, and Tsung-Yu Lin. 2024.

Learning to localize objects improves spatial reasoning in visual-llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12977–12987.

Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad S. Khan. 2024. Llava++: Extending visual capabilities with llama-3 and phi-3.

Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. 2020. Contrastive learning with hard negative samples. *arXiv preprint arXiv:2010.04592*.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. 2021. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*.

Gabriela Ben Melech Stan, Raanan Yehezkel Rohekar, Yaniv Gurwicz, Matthew Lyle Olson, Anahita Bhiwandiwalla, Estelle Aflalo, Chenfei Wu, Nan Duan, Shao-Yen Tseng, and Vasudev Lal. 2024. Lvlm-intrepret: An interpretability tool for large vision-language models. *arXiv preprint arXiv:2404.03118*.

Austin Stone, Hagen Soltau, Robert Geirhos, Xi Yi, Ye Xia, Bingyi Cao, Kaifeng Chen, Abhijit Ogale, and Jonathon Shlens. 2024. Learning visual composition through improved semantic guidance. *arXiv preprint arXiv:2412.15396*.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. 2023. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*.

Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. 2022. Winoground: Probing vision and language models for visio-linguistic compositionality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5238–5248.

Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. 2024a. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *arXiv preprint arXiv:2406.16860*.

Shengbang Tong, Erik Jones, and Jacob Steinhardt. 2024b. Mass-producing failures of multimodal systems with language models. *Advances in Neural Information Processing Systems*, 36.

Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. 2024c. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9568–9578.

Michael Tschannen, Manoj Kumar, Andreas Steiner, Xiaohua Zhai, Neil Houlsby, and Lucas Beyer. 2023. Image captioners are scalable vision learners too. *Advances in Neural Information Processing Systems*, 36:46830–46855.

Tan Wang, Kevin Lin, Linjie Li, Chung-Ching Lin, Zhengyuan Yang, Hanwang Zhang, Zicheng Liu, and Lijuan Wang. 2023. Equivariant similarity for vision-language foundation models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11998–12008.

Ziqi Wang, Hanlin Zhang, Xiner Li, Kuan-Hao Huang, Chi Han, Shuiwang Ji, Sham M Kakade, Hao Peng, and Heng Ji. 2024. Eliminating position bias of language models: A mechanistic approach. *arXiv preprint arXiv:2407.01100*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Yifan Xu, Xiaoshan Yang, Yaguang Song, and Changsheng Xu. 2024. Libra: Building decoupled vision system on large language models. *arXiv preprint arXiv:2405.10140*.

Michal Yarom, Yonatan Bitton, Soravit Changpinyo, Roee Aharoni, Jonathan Herzig, Oran Lang, Eran Ofek, and Idan Szpektor. 2024. What you see is what you read? improving text-image alignment evaluation. *Advances in Neural Information Processing Systems*, 36.

Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. 2022. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*.

Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. 2023. When and why vision-language models behave like bags-of-words, and what to do about it? In *The Eleventh International Conference on Learning Representations*.

Yan Zeng, Xinsong Zhang, and Hang Li. 2021. Multi-grained vision language pre-training: Aligning texts with visual concepts. *arXiv preprint arXiv:2111.08276*.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986.

Yuhui Zhang, Alyssa Unell, Xiaohan Wang, Dhruba Ghosh, Yuchang Su, Ludwig Schmidt, and Serena Yeung-Levy. 2024. Why are visually-grounded language models bad at image classification? *arXiv preprint arXiv:2405.18415*.

Kecheng Zheng, Yifei Zhang, Wei Wu, Fan Lu, Shuailei Ma, Xin Jin, Wei Chen, and Yujun Shen. 2024. Dreamlip: Language-image pre-training with long captions. *arXiv preprint arXiv:2403.17007*.

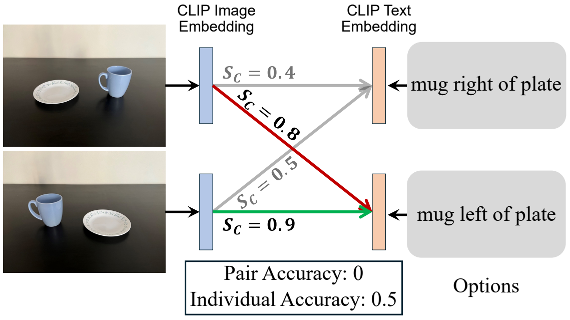

Figure 3: Example test case and evaluation method for CLIP-like models on What'sUp benchmark. In our two-way evaluation on benchmarks with paired images, a test case consists of two similar images and two captions. The model chooses one caption for each image, and it gets one point in pair accuracy only if choosing correctly for both images. The choices of CLIP-like models are determined by S_C, cosine similarity between image and text embeddings.

A Benchmarks and Additional Evaluations

A.1 Benchmark Information

What'sUp. The What'sUp benchmark (Kamath et al., 2023b) contains 820 images of pairs of household objects captured by the authors, 408 in Subset A and 412 in Subset B. For every object pair, all prepositions are present in the benchmark, and thus the images and captions are balanced, avoiding the bias in real-world images (e.g., a cup is usually on the table, not under the table). We corrected the mislabeled images in the GitHub Issues and reevaluated the pretrained VLMs. For CLIP and XVLM's evaluation, we refer to the official code provided by the What'sUp benchmark's authors in `https://github.com/amitakamath/whatsup_vlms`. The evaluation of SigLIP and EVA-CLIP directly follows the evaluation of CLIP in the official code. We offer an example in Figure 3 to demonstrate how pair accuracy and individual accuracy are computed on benchmarks with paired images like What'sUp.

COCO-spatial and GQA-spatial. Kamath et al. (2023b) also selects validation sample from COCO (Lin et al., 2014) and GQA (Hudson and Manning, 2019) targeting spatial relations (to the left of vs to the right of, above vs below). Each test case contains one image, one positive caption, and one negative caption. COCO-spatial has 2687 test cases, and GQA has 1451 test cases in total.

Winoground, NaturalBench, and SeeTrue. Winoground (Thrush et al., 2022) is a challenging benchmark consisting of 400 pairs of image-text pairs. It focuses on VLM's compositionality, with two images and two similar captions in one test case. A example of the captions is "`some plants surrounding a lightbulb`" vs "`a lightbulb surrounding some plants`." High pair accuracy requires VLM to match these images with their captions correctly at the same time. NaturalBench (Li et al., 2024) is a benchmark for testing Generative MLLMs on compositionality with unbiased Yes/No answers. In one test case, there are two images with two questions, and each question has "Yes" as the answer for one image and "No" for the other image. We use the retrieval version of NaturalBench provided by (Lin et al., 2024). SeeTrue (Yarom et al., 2024) is an alignment bench that has 6930 human labels for whether a given image is paired with the text or not. We report the AUROC (Area Under the Receiver Operating Characteristic curve) instead of accuracy on SeeTrue. We use VQAScore's official code for evaluation on these benchmarks in `https://github.com/linzhiqiu/t2v_metrics`.

SugarCREPE. SugarCREPE (Hsieh et al., 2024) is designed for evaluating VLM's compositionality with grammatical, sensical, and fluent hard negatives. Each test case contains one image, one positive caption, and one negative caption. There are 7512 test cases in total.

MMVP(-VLM). The MMVP benchmark contains 150 pairs of similar images, and the MMVP-VLM benchmark has 135 pairs of similar images, divided into nine categories. There is an overlap between the image pairs in these two benchmarks. We corrected the mislabeled images in the GitHub Issues and reevaluated the pretrained VLMs. Since MMVP is incompatible with CLIP, we convert its questions manually. We attach the converted version to the supplementary material for reference.

A.2 Model Weight Information

We use public pretrained weights of LLaVA-1.5-7B (`https://huggingface.co/llava-hf/llava-1.5-7b-hf`) under the Meta LLaMA License Agreement and the weights of Phi-3-V-3.8B in `https://huggingface.co/MBZUAI/LLaVA-Phi-3-mini-4k-instruct` and LLaMA-3-V-8B in `https://huggingface.co/MBZUAI/LLaVA-Meta-Llama-3-8B-Instruct` provided by (Ranasinghe et al., 2024) under MIT License since they are trained with vision encoder frozen. For contrastive VLMs, we use OpenAI's pretrained CLIP-ViT-L/14-224px and CLIP-ViT-L/14-336px model under MIT License, SigLIP-ViT-L/16-384px pretrained on the WebLI dataset (Chen et al., 2022) and EVA01-ViT-g-14 pretrained on the LAION400M-s11b-b41k dataset (Schuhmann et al., 2021) under Apache 2.0 License provided in the OpenCLIP repository. In Table 5, LLaVA-1.5-7B-LoRA is reproduced. In Table 6, the checkpoint for LLM2CLIP is from `https://huggingface.co/microsoft/LLM2CLIP-Llama-3-8B-Instruct-CC-Finetuned` under Apache 2.0 License. We also use the text encoder and the adapter of this checkpoint in our experiments of using a stronger text encoder. In Table 7, the LLaVA-1.5-7B-VLM2Vec-LoRA is trained by ourselves with the vision encoder frozen using the VLM2Vec method (Jiang et al., 2024).

A.3 Comparison between VQAScore and Response-Based Evaluation

We compare the score-based evaluation, VQAScore (Lin et al., 2024), and the standard response-based evaluation for Generative MLLMs on What'sUp. Response-based evaluation requires a question accompanied by a given image as the input, and the questions used for LLaVA-1.5's evaluation are listed in Table 8. Then, the question is concatenated with the fixed prompt template ("`USER: <image>\n{question} ASSISTANT:`"). Considering the position bias in LLMs (Wang et al., 2024), we exchange the position of two prepositions in the question with 50% probability on COCO-spatial and GQA-spatial benchmarks for fair results. On the What'sUp benchmark, the orders are always the same for two images. Then, we use greedy decoding to ensure reproducibility and evaluate the outputs by keyword matching since we observe that the outputs of Generative MLLMs are quite structured, showing their strong instruction-following ability.

The reason why we use different commands after the main question (e.g., "`Answer left or right`", "`Choose from the two options`", and "`Give a short answer`") is that we find the LLaVA-1.5 model sensitive to such command. For instance, we try "`Answer on or under`" and "`Answer with under or on`" for the On/Under subset in What'sUp Subset A, and the model accuracy is quite low. For Phi-3-V-3.8B and LLaMA-3-V-8B, we try these prompts and pick the one with the highest accuracy. This is one of their limitations that deserves future research. However, we aim to show that they can extract such information, so we use the best prompt to showcase its ability.

The results are shown in Table 10. We observe that the accuracy of LLaVA-1.5-7B is increased on On/Under and Front/Behind subsets. However, the performance of LLaMA-3-V-8B is worsened. Overall, they still surpass CLIP.

A.4 Evaluating VLM2Vec

For evaluation, we use the same question template as for training ("`Represent the given image with the following question: {Question}`"). We list the questions used for VLM2Vec's evaluation in Table 9. Similar to response-based evaluation for Generative MLLMs, we notice variance when using different questions. Here, we adopt the questions that lead to the best performance on the benchmarks.

In addition, we show that the benefit of using a question in the prompt generalizes beyond What'sUp. Here, we perform the same comparison as in Table 7 on MMVP and MMVP-VLM. We use the original question of the benchmark in the prompt. For MMVP-VLM which does not have questions, we manually add an MMVP-like question to each test case without altering content or tuning the prompt. We attached these questions to the updated supplementary material. We use the same prompt format as What'sUp ("`Represent the given image with the following question: {Question}`" or "`Represent the given image.`" without any question). We observe similar results in Table 11.

B Supplementary Experimental Details and Results

B.1 Hyperparameters

Our code for training standard CLIP, SigLIP, and EVA-CLIP is based on `https://github.com/mlfoundations/open_clip` (Ilharco et al., 2021). We finetune these models for five epochs with a learning rate of 5e-6 on the combination of converted LCS-558K plus converted DataMix-665K. We use 50 steps of warmup and AdamW optimizer with a cosine-annealing learning rate schedule. The batch size is 512, and we train the models on 4 GPUs. The training time is less than one day.

For the `[CLS]`-LLaVA-1.5-7B-LoRA and reproduced LLaVA-1.5-7B-LoRA in Table 5, we use the official LLaVA code in `https://github.com/haotian-liu/LLaVA` released under the Apache

Subset	Question
What'sUp Subset A&B, Left/Right	`Is the (object 1) to the left of or to the right of the (object 2)? Answer left or right.`
What'sUp Subset A, On/Under	`Is the (object 1) on or under the (object 2)? Choose from the two options.`
What'sUp Subset B, Front/Behind	`Is the (object 1) in front of or behind the (object 2)? Answer front or behind.`
What'sUp Subset A (four-way)	`Is the (object1) to the left of, to the right of, on, or under the (object2)? Choose from the four options.`
What'sUp Subset B (four-way)	`Is the (object1) to the left of, to the right of, in front of, or behind the (object2)? Answer front, behind, left, or right.`
COCO/GQA-spatial, One obj.	`Is the (object 1) on the (left/right/top/bottom) or on the (right/left/bottom/top)? Give a short answer.`
COCO-spatial, Two obj.	`Is the (object 1) (to the left of/to the right of/above/below) a (object 2) or (to the right of/to the left of/below/above) a (object 2)? Give a short answer.`
GQA-spatial, Two obj.	`Is the (object 1) to the (left/right/front/behind) of a (object 2) or to the (right/left/behind/front) of a (object 2)? Give a short answer.`

Table 8: Question formats for different subsets for LLaVA-1.5-7B.

Subset	Question
What'sUp Subset A&B, Left/Right	`Is the (object 1) to the left of or to the right of the (object 2)?`
What'sUp Subset A, On/Under	`Is the (object 1) at the bottom of the (object2) or at the top of the (object2)?`
What'sUp Subset B, Front/Behind	`Is the (object 1) in the back of the (object2) or in the front of the (object2)?`

Table 9: Question formats for evaluating LLaVA-1.5-7B-VLM2Vec-LoRA.

2.0 license. The batch size, learning rate, and other training settings are the same as described in LLaVA-1.5 paper (Liu et al., 2024).

For the experiments in Table 6, we start from an implementation of PACL (Mukhoti et al., 2023) in `https://github.com/NMS05/Patch-Aligned-Contrastive-Learning`. Since we only need to train the vision embedder and text embedder, we apply a larger batch size (4096) and train for 10 epochs on 8 GPUs on the combination of converted LCS-558K plus converted DataMix-665K. We use 0.1 as the fixed temperature, 1e-4 as the learning rate, and Adam as the optimizer. The training time is less than one day for all experiments. We adopt the data augmentation implemented in the codebase, except the `RandomHorizontalFlip`, which discourages models from learning about Left/Right spatial relations. When processing the captions, we follow the technique used in the codebase, which uses the original caption randomly with a probability of 0.5, and template+(a noun in the caption) otherwise. The templates are: "`a picture of {}.`", "`itap of {}.`", "`a photograph of {}.`", "`this picture contains {}.`", "`a good photo of {}.`". For experiments with a stronger text encoder, we do not apply this technique on DataMix-665K.

For LLaVA-1.5-7B-VLM2Vec-LoRA in Table 7, we refer to the VLM2Vec code in `https://github.com/TIGER-AI-Lab/VLM2Vec`. We use

	What'sUp Subset A				What'sUp Subset B			
	Left/Right		On/Under		Left/Right		Front/Behind	
	Indiv.	Pairs	Indiv.	Pairs	Indiv.	Pairs	Indiv.	Pairs
CLIP-ViT-L/14-336px	49.0	1.9	61.7	23.3	54.9	10.8	51.5	7.8
LLaVA-1.5-7B	99.0	98.1	80.1	60.2	**100**	**100**	**98.5**	**97.1**
Phi-3-V-3.8B	**100**	**100**	**85.4**	**70.9**	**100**	**100**	56.9	13.7
LLaMA-3-V-8B	90.3	80.6	57.8	20.4	71.1	46.1	69.1	41.2

Table 10: Results of CLIP-ViT-L/14-336px and Generative MLLMs on four subsets in What'sUp using standard response-based evaluation. The individual accuracy and pair accuracy are in percentage points.

	MMVP	MMVP-VLM
CLIP-ViT-L/14-336px	14.0	20.7
LLaVA-1.5-7B-VLM2Vec-LoRA	**30.0**	**37.8**
w/o Question in Prompt	9.3	11.9
Random chance	25.0	25.0

Table 11: The pair accuracy of CLIP-ViT-L/14-336px and LLaVA-1.5-converted models in percentage points on MMVP and MMVP-VLM. LLaVA-1.5-7B-VLM2Vec-LoRA continues to outperform CLIP. When there is no question in the prompt, its performance degenerates to the standard CLIP.

rank=8 for LoRA, 256 for the batch size, 1024 for maximum input token length, and 0.02 for the temperature. We train the model for only 900 steps on 4 GPUs for 40 hours on the combination of LCS-558K and DataMix-665K, with a linear learning rate schedule, 100 warmup steps, and 2e-5 as the learning rate. Although we do not train the model on full data, the model performance is remarkable on What'sUp.

B.2 Converting LLaVA-1.5's Training Data

We use LLaVA-1.5's training data for all finetuning experiments we include in the paper. The DataMix-665K is under CC BY 4.0 License, while the LCS-558K is under LAION/CC/SBU License for images and BSD 3-Clause "New" or "Revised" License for BLIP-generated captions. They are datasets of English conversations.

We check the frequency of appearance of the following keywords in DataMix-665K and LCS-558K: `on the left`, `on the right`, `to the left`, `to the right`, `at the left`, `at the right`. In DataMix-665K, there are 12957 instances with at least one of the key phrases, among which 12658 have a paired image. For captions (ground truth answers), this number is 13473 since an instance is paired with a multi-turn conversation. In LCS-558K, there are 560 such instances and captions since each instance has only one question and one answer.

In our experiments in Section 3.1 and Section 3.2, LCS-558K was converted from image-text pair format to conversation format, so we revert this process by using ground truth answer as the caption. Since DataMix-665K is in a multi-turn conversation format, we randomly pick one answer as the caption in each epoch. In Section 3.3, the new text encoder can encode long paragraphs, so we use the concatenation of all answers in the multi-turn conversation as the ground truth caption. In practice, we calculate the text embeddings of all possible captions using the text encoder of LLM2CLIP before training to save memory and time. In Section 3.4, we randomly choose one turn from the multi-turn conversation.

B.3 Results of Unlocking Image Encoder

We try unlocking the image encoder during finetuning on the SigLIP-ViT-L/16-384px model. The results are in Table 12. Still, the individual accuracy remains low.

B.4 Evaluating and Finetuning XVLM

Observing the similar failure of the data-informed attempt, previous work concluded that even with relevant, high-quality data and hard negatives, denser supervision is likely required to let the model learn the basic spatial relations (Kamath

	What'sUp Subset A		What'sUp Subset B		COCO-spatial		GQA-spatial	
	Indiv.	Pairs	Indiv.	Pairs	One-obj.	Two-obj.	One-obj.	Two-obj.
SigLIP-ViT-L/16-384px	50.0	1.9	51.5	5.9	48.7	50.2	51.2	47.0
+ finetuning (ft)	50.5	2.9	51.5	5.9	48.7	57.7	50.5	48.1
+ ft + hard neg.	50.0	3.9	47.1	2.0	52.3	47.0	51.8	52.7
Random chance	50.0	25.0	50.0	25.0	50.0	50.0	50.0	50.0

Table 12: Results of SigLIP-ViT-L/16-384px focusing on the Left/Right subsets of What'sUp, COCO-spatial, and GQA-spatial benchmark with unlocked image encoder, after finetuning on LLaVA-1.5's training data with or without hard negative captions. The accuracy remains low on all benchmarks.

	What'sUp A	What'sUp B	COCO-spatial		GQA-spatial	
			One-obj.	Two-obj.	One-obj.	Two-obj.
XVLM-16M	**50.0**	32.8	65.4	64.6	**63.2**	**53.3**
+ finetuning	46.4	**34.6**	**66.8**	**65.2**	61.3	51.2
Random chance	25.0	25.0	50.0	50.0	50.0	50.0

Table 13: Individual accuracy of XVLM-16M on the Left/Right subsets of What'sUp, COCO-spatial, and GQA-spatial benchmark on LLaVA-1.5's training data. LLaVA-1.5's training data does not help improve XVLM-16M notably.

et al., 2023b), as in XVLM (Zeng et al., 2021), a VLM with supervision at the bounding-box level. We attach XVLM-16M's performance in Table 13. We find that Generative MLLMs still beat XVLM-16M, while they do not incorporate downstream task-related inductive bias or denser supervision.

We explore finetuning XVLM on LLaVA-1.5's training data based on their official code (`https://github.com/zengyan-97/X-VLM`), but no improvement is observed in the results (the last row in Table 13). The image encoder is locked during finetuning. We use both contrastive learning loss and image-text matching loss to finetune the XVLM-16M model for five epochs with a learning rate of 1e-5 and a weight decay rate of 0.01. We use 10% steps of warmup and AdamW optimizer with a lambda learning rate schedule. The batch size is 128, and we train the model on 4 GPUs. The evaluation is performed through the image-text matching score.

B.5 Implementation Details of PACL and SPARC

For PACL, the vision embedder applied on CLIP-ViT-L/14-336px is the sum of a one-layer linear projection and a two-layer nonlinear projection with GELU as the activation function. The input of the vision embedder is 576 1024-dimensional patch tokens after `LayerNorm` and `Dropout` (with probability = 0.1), and the output dimension is 768 for 576 tokens. The text embedder accepts one 768-dimensional text token and applies one-layer linear projection on it. All output embeddings are L2-normalized. The embedders used in SPARC experiments share the same model structures as in PACL. For experiments with the text encoder from LLM2CLIP, the input dimension of the text embedder is 1280 instead of 768, with other settings unchanged.

For RoPE, we refer to the implementation in the codebase for LLaMA in `https://github.com/huggingface/transformers`. It is applied before `LayerNorm`. Compared with learned or sinusoidal position embeddings, it maintains the relative positions of tokens. We choose to use it since it is applied in language models of Generative MLLMs listed in Section 2.

For SPARC, we follow the pseudocode in Appendix C of Bica et al. (2024). Specifically, the output of the vision embedder is 576 768-dimensional projected patch tokens, and the output of the text embedder is 77 768-dimensional projected text tokens (with padding). After multiplying them, we get a similarity matrix of size 77×576. Following the SPARC paper, we first apply min-max normalization to the matrix and then sparsify it by zeroing out all matrix entries below the threshold $1/576$. We normalize the rows of the similarity matrix, multiply it with the patch tokens, and obtain 77 grouped visual tokens. The global representation of the image is the mean of these grouped visual tokens after L2-normalization, and we get the global representation of the text similarly. During inference, we calculate the cosine similarity between the two global representations. When training, we use the global representations for the standard contrastive loss and apply a local contrastive loss to contrast the 77 grouped visual tokens and 77 text tokens within each sample. In this way, we align the patch tokens to individual concepts represented by text tokens. Unlike the original implementation, we do not use a learnable temperature for contrastive losses.

B.6 When Generative MLLMs are worse than CLIP

We also observe that in some cases, MLLMs have worse performance than CLIP (See Table 14). On EqBen-mini (Wang et al., 2023), their performance is close. On COCOCounterfactuals (Le et al., 2024), we notice that CLIP embeddings are involved in the construction of the benchmark as a metric, which could affect the comparison.

This phenomenon is also discussed in previous literature (Zhang et al., 2024; Geigle et al., 2024), where they find that training on web-crawled data teaches CLIP many rare concepts, while Generative MLLMs are not sufficiently exposed to such data for image-text alignment.

B.7 Alternative Decoding for Generative MLLMs

We apply Multi-Modal Mutual-Information Decoding (M3ID) (Favero et al., 2024) on LLaVA-1.5 for response-based evaluation. For token in each decoding step t, M3ID computes the output probability with the image and without any input image, denoted as $\mathbf{l_c}$ and $\mathbf{l_u}$ respectively. The latter corresponds to the language priors of the answer to the given question. Then a correction term $(\mathbf{l_c} - \mathbf{l_u})$ is added to $\mathbf{l_c}$ with weight $\frac{1-\exp(-\lambda t)}{\exp(-\lambda t)}$ if the model is not highly confident with the token in step t $(\max_k (l_c)_k < \log \alpha$ where α, λ are pre-defined hyperparameters). This correction prevents the VLM from omitting the visual input and relying on the language priors.

In Table 15, this method achieves gain (+6%) relative to the baseline LLaVA-1.5-7B on MMVP. We note that this is on par with I-MoF with interleaved CLIP and DINO features) (Tong et al., 2024c). This result suggests that LLaVA-1.5 did not attend to the visual input enough and thus might miss the key information for answering the query. A similar finding was described through the interpretability perspective on attention weights in Stan et al. (2024).

	EqBen-mini	COCOCounterfactuals
CLIP-ViT-L/14-336px	**40.0**	**87.7**
LLaVA-1.5-7B	32.9	57.9

Table 14: The pair accuracy of CLIP-ViT-L/14-336px and LLaVA-1.5-7B in percentage points.

	Indiv.	Pairs
LLaVA-1.5-7B	61.7	25.3
+ M3ID (Favero et al., 2024)	**64.3**	**31.3**
LLaVA-1.5-13B + I-MoF (Tong et al., 2024c)	–	**31.3**
Random chance	50.0	25.0

Table 15: Results of LLaVA-1.5-7B with M3ID ($\alpha = 0.6, \lambda = 0.15$) using response-based evaluation on MMVP benchmark with the original results. M3ID encourages LLaVA-1.5-7B to attend more to the visual input, achieving performance on par with using interleaved CLIP and DINO features.