
VITRIX-CLIPIN: Enhancing Fine-Grained Visual Understanding in CLIP via Instruction Editing Data and Long Captions

Ziteng Wang*

The Chinese University of Hong Kong, Shenzhen
tommmmywangzt@gmail.com

Siqi Yang*†

Meituan
siqi.yang@uq.net.au

Limeng Qiao

Meituan
qiaolm@pku.edu.cn

Lin Ma†

Meituan
forest.linma@gmail.com

Abstract

Despite the success of Vision-Language Models (VLMs) like CLIP in aligning vision and language, their proficiency in detailed, fine-grained visual comprehension remains a key challenge. We present CLIP-IN, a novel framework that bolsters CLIP’s fine-grained perception through two core innovations. Firstly, we leverage instruction-editing datasets, originally designed for image manipulation, as a unique source of hard negative image-text pairs. Coupled with a symmetric hard negative contrastive loss, this enables the model to effectively distinguish subtle visual-semantic differences. Secondly, CLIP-IN incorporates long descriptive captions, utilizing rotary positional encodings to capture rich semantic context often missed by standard CLIP. Our experiments demonstrate that CLIP-IN achieves substantial gains on the MMVP benchmark and various fine-grained visual recognition tasks, without compromising robust zero-shot performance on broader classification and retrieval tasks. Critically, integrating CLIP-IN’s visual representations into Multimodal Large Language Models significantly reduces visual hallucinations and enhances reasoning abilities. This work underscores the considerable potential of synergizing targeted, instruction-based contrastive learning with comprehensive descriptive information to elevate the fine-grained understanding of VLMs. Project is available here.

1 Introduction

The Contrastive Language-Image Pre-training (CLIP) model [30] has revolutionized vision-language representation learning by aligning visual and textual concepts within a unified embedding space. Trained on extensive web-scraped image-text pairs [34], CLIP exhibits remarkable zero-shot generalization across diverse tasks, serving as a foundational model for applications like image classification, cross-modal retrieval, object detection and segmentation, and multimodal large language models (MLLMs) [31, 11, 20, 3].

Despite its success in high-level semantic understanding, CLIP demonstrates limitations in capturing fine-grained visual details such as color, quantity, and spatial relationships [16, 59, 50, 15, 23, 22, 49]. For instance, distinguishing between a "black cat with a yellow bow tie" and one with a "red bow tie"

*Equal contribution. Work done when Ziteng Wang worked as an intern with Meituan.

†Corresponding author.

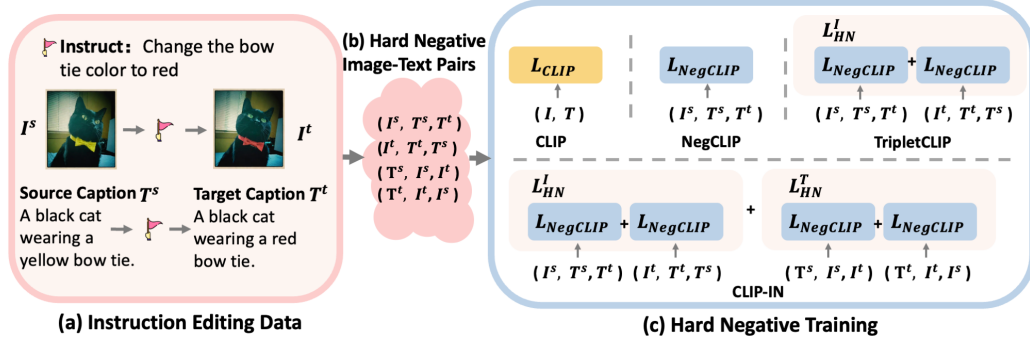


Figure 1: **Instruction Editing Data as Hard Negatives.** (a) We illustrate how instruction editing data provides challenging negative examples for CLIP. Given a source image and caption, an editing instruction leads to a target image and caption with subtle, fine-grained changes. The (source_image, target_caption) and (target_image, source_caption) pairs serve as hard negatives in (b), requiring the model to distinguish these nuanced visual-semantic differences. (c) We propose a symmetric hard negative contrastive loss to explicitly train the model to discern these subtle visual-semantic differences from both image-to-text and text-to-image perspectives.

can be challenging for CLIP when the overall scene context is similar (Figure 1). These limitations are inherited by MLLMs built upon CLIP’s visual encoders, impacting their perceptual accuracy and contributing to issues like object hallucination [39, 46]. Benchmarks like Multimodal Visual Patterns (MMVP) [39] specifically highlight these "CLIP-blind spots" by presenting perceptually distinct image pairs that CLIP often confuses.

These shortcomings primarily arise from two aspects of CLIP’s pre-training. First, the global image-text alignment objective can overlook subtle visual details and intricate inter-object relationships [16, 59, 23, 50]. Second, CLIP’s text encoder, typically employing absolute positional embeddings with a fixed 77-token limit, restricts the effective utilization of longer, more descriptive captions that could provide richer supervision for fine-grained details [1, 55, 24, 58].

Existing approaches to enhance fine-grained understanding in CLIP have explored region-level contrastive learning [16, 59, 50], which often requires complex region proposals or additional supervision. Self-distillation methods [47, 23, 4, 22] aim to improve local-to-global consistency without explicit regional annotations but may suffer from weakened semantic grounding or be limited by the teacher model’s capabilities. Hard negative mining strategies [53, 56, 27] focus on challenging the model with difficult negative examples, often generated by perturbing captions or using text-to-image synthesis, which may lack the necessary visual similarity or fine-grained control. Long-caption methods [1, 55, 24] extend CLIP’s text processing capacity, but simply increasing the token limit does not guarantee improved fine-grained alignment or prevent performance degradation on short text inputs. This motivates the central question: How can we effectively and scalably enrich vision-language models like CLIP with robust fine-grained visual understanding by harnessing data sources that offer explicit, targeted supervision for subtle visual-semantic distinctions, while simultaneously leveraging the contextual richness of descriptive language?

In this work, we introduce **CLIP-IN** (CLIP with INstruction edit data and INformative long data), a novel framework that addresses this challenge by synergistically integrating instruction editing data and long descriptive captions. Our core innovation lies in repurposing instruction editing datasets, originally designed for image manipulation, as a valuable source of hard negative image-text pairs for contrastive learning. Datasets like UltraEdit [57] provide tuples of (source_image, source_caption, editing_instruction, target_image, target_caption), inherently offering hard negative examples with fine-grained differences in objects, attributes, and spatial relationships. This contrasts with synthetic hard negatives generated by text-to-image models, as seen in TripletCLIP [27], which often lack precise control and visual similarity to the source image. We propose a symmetric hard negative contrastive loss to explicitly train the model to discern these subtle visual-semantic differences from both image-to-text and text-to-image perspectives, implicitly learning fine-grained visual distinctions and their linguistic descriptions, unlike explicit regional contrastive learning [50, 15]. Additionally, We propose that long caption data is complementary to the instruction editing data. We leverage long descriptive captions to provide broader contextual richness. To enable existing CLIP

Figure 2: CLIP-IN Framework Overview. Stage1. We adapt the CLIP text encoder to process long captions using Rotary Positional Embeddings (RoPE) via knowledge distillation. Stage2. Our proposed framework, CLIP-IN, leverages two complementary data sources: instruction editing data and long descriptive captions. Instruction editing data excels at teaching the "where" and "how" of subtle visual details, while long captions provide the broader "what" and "why" of the scene, capturing complex relationships and contextual information.

models to effectively process long text, we adapt the text encoder by incorporating Rotary Positional Embeddings (RoPE) through knowledge distillation from the pretrained clip text encoder model extended context.

By training on both instruction editing data and long captions, CLIP-IN aims to learn representations that are simultaneously semantically rich and visually grounded at a fine-grained level. Instruction editing data excels at teaching the "where" and "how" of subtle visual details, while long captions provide the broader "what" and "why" of the scene, capturing complex relationships and contextual information. This synergy enables CLIP-IN to outperform models trained on either data source alone.

Our primary contributions are: (1) A novel approach to utilize instruction-editing datasets as a rich source of hard negative image-text pairs for enhancing CLIP's fine-grained visual perception, extending the utility of these datasets beyond image generation tasks. (2) A synergistic training framework for CLIP that combines instruction-based image editing data and long descriptive captions to significantly improve its fine-grained visual-linguistic understanding by leveraging explicit and implicit supervision signals. (3) A methodology to adapt CLIP's text encoder for processing long captions by integrating Rotary Positional Embeddings (RoPE) via knowledge distillation, overcoming the inherent context length limitations. (4) Extensive experiments demonstrating consistent improvements on zero-shot visual benchmarks, fine-grained visual recognition tasks, and evaluations on MMVP and MLLM benchmarks.

2 Related Works

Contrastive Language-Image Pre-training. Contrastive Language-Image Pre-training (CLIP) [36] has demonstrated remarkable success in learning aligned visual and textual representations by contrasting positive image-text pairs against negative ones on a massive scale. Subsequent works like EVA-CLIP [37], MetaCLIP [51], SigLIP [54], and SigLIP2 [40] have further improved the performance and robustness of CLIP models through architectural modifications and training strategies.

Fine-grained Understanding in CLIP. Despite its effectiveness, standard CLIP often struggles with fine-grained visual understanding, including discerning subtle attributes, complex inter-object relationships, and precise object counts. This limitation stems, in part, from the typically concise nature of image captions used in pre-training, which may lack the detailed descriptions necessary for capturing such granular distinctions. To address this, several approaches have been proposed. GLIP [16], RegionCLIP [59], and FG-CLIP [50] leverage grounding data to explicitly align image regions with corresponding textual phrases, thereby enhancing the model's ability to understand local-

ized visual details. Complementary to this, methods like CLIP37, [SILC [23], MaskEmb [4], and TIPS [22] focus on improving the consistency between global and local representations through self-supervised learning objectives, such as local-to-global correspondence learning via self-distillation and masked patch embeddings. Furthermore, DMM explores the use of generative feedback from text-to-image diffusion models to refine CLIP representations using only image data. However, these methods are primarily designed and evaluated on datasets with short captions, potentially limiting their effectiveness in scenarios requiring the processing of more detailed textual descriptions.

Hard Negative Mining and Generation for CLIP. Hard negatives, semantically distinct samples close in the embedding space, are crucial for learning discriminative features in contrastive learning. Prior works explored identifying hard negatives within datasets or generating synthetic negative captions [53, 56, 52, 35, 7, 6]. More recently, generating synthetic hard negative images has gained attention [28, 43, 33, 45]. For example, Peng et al. [28] synthesized images by rearranging segmented objects, potentially lacking real-world complexity and scalability. Saharia. [33] used inpainting for limited synthetic image generation, while Wang et al. [45] manipulated discrete visual tokens without generating corresponding negative texts. TripletCLIP, [47], the most related work, generates hard negative image-text pairs using LLMs and text-to-image models for triplet loss. However, its generated images often lack control and deviate significantly, primarily enabling image-to-text hard negative learning. Our approach uniquely leverages large-scale instruction editing datasets like UltraEdit [57], which provide real images and localized visual changes, mitigating domain biases of purely generated data and yielding challenging hard negatives. This explicit grounding allows for symmetric hard negative losses addressing both image and text mismatches.

Extending CLIP to Long Captions. Standard CLIP models are typically limited in their text input length (e.g., 77 tokens), which can hinder their ability to process detailed and descriptive long captions. DCI [41] highlighted this limitation and, along with DOCC [25], introduced datasets with dense, long captions for benchmarking. DreamL [58] proposed leveraging MLLM-generated long captions and dynamically sampling sub-captions to create multiple positive pairs, although it does not directly process the full long captions. Recent efforts [15, 50] have extended CLIP’s text capacity to 248 tokens by employing positional embedding interpolation and fine-tuning existing CLIP models. LoTLIP [48] incorporates long captions during the pre-training stage and introduces corner tokens to aggregate diverse textual information. TULIP [24] adopts a distillation approach, transferring the knowledge of a CLIP text encoder enhanced with relative positional embeddings (RoPE). While these methods demonstrate progress in handling long captions, they often rely on large-scale datasets of long caption-image pairs (e.g., millions in [58, 55, 15, 24] and 1.6 billion in FG-CLIP [50]). Direct fine-tuning on such data can sometimes lead to performance degradation on tasks involving short text inputs, such as image classification. In this work, we aim to exploit the synergistic benefits of instruction-editing hard image-text pairs and long caption data to achieve improved fine-grained understanding without compromising performance on standard CLIP benchmarks.

3 Methodology

3.1 Preliminaries

Contrastive Language-Image Pre-training (CLIP). Given a batch of N image-text pairs $(X; Y) = (x_i; y_i)_{i=1}^N$, CLIP aims to align semantically similar image-text pairs in a shared embedding space while separating dissimilar ones. The architecture consists of an image encoder and a text encoder $g(\cdot)$, both mapping their respective inputs to this common space. CLIP employs the InfoNCE loss [26] to achieve this alignment. For an image x_i as the anchor, the image-to-text contrastive loss, L_{I2T} , is defined as:

$$L_{I2T}(X; Y) = \frac{1}{N} \sum_{i=1}^N \log \frac{\exp(\text{sim}(f(x_i); g(y_i)))}{\sum_{k=1}^N \exp(\text{sim}(f(x_i); g(y_k)))}, \quad (1)$$

where sim denotes the cosine similarity and τ a temperature parameter. The overall CLIP loss, L_{CLIP} , is a symmetric combination of the image-to-text and text-to-image contrastive losses:

$$L_{CLIP} = L_{CL}(X; Y) + L_{CL}(Y; X). \quad (2)$$

Overview of CLIP-IN. As depicted in Figure 2, our primary objective is to significantly enhance CLIP’s capacity for discerning fine-grained visual details. To this end, we introduce CLIP-IN, a

two-stage training framework that leverages two complementary data modalities. **Instruction Editing Data** provides explicit supervision for learning localized visual transformations guided by precise textual instructions, thereby improving the model's sensitivity to subtle visual attributes and intricate relationships. **Long Captions** offer rich semantic context and detailed descriptions of complex visual scenes, enabling a more comprehensive understanding beyond localized features. In the first stage of our framework, we adapt the text encoder to effectively process long captions by replacing absolute positional embeddings with RoPE and performing knowledge distillation. Subsequently, the second stage involves a joint contrastive learning process utilizing both instruction editing data and long captions to achieve the desired enhancement in fine-grained visual perception.

3.2 Instruction Editing Data as Novel Source for Hard Negative Training

Instruction editing data offers a unique form of supervision by explicitly linking textual instructions to precise visual modifications in real images, as exemplified in Figure 1. This structure is particularly valuable for generating challenging negative samples that target fine-grained visual distinctions.

Formulating Hard Negatives from Instruction Editing Data. Each instance in an instruction editing dataset is a tuple $(I^s; I^t; T^e; T^s; T^t)$, where I^s is the source image, I^t is the target (edited) image, T^e is the editing instruction, and T^s and T^t are the corresponding source and target captions. From this data, we construct positive pairs $(I^s; T^s)$ and $(I^t; T^t)$, and critically, we define hard negative pairs $(I^s; T^t)$ and $(I^t; T^s)$. These hard negatives are challenging because the image pairs $(I^s; I^t)$ and text pairs $(T^s; T^t)$ are minimally but semantically distinct due to the fine-grained edits described by T^e . This structured approach to generating hard negatives from real image edits provides a strong learning signal for enhancing fine-grained visual understanding. We utilize the recent UltraEdit dataset [57], comprising approximately 4 million high-quality instruction-based editing samples across diverse editing types and instructions, as our primary source of instruction editing data.

Symmetric Hard Negative Contrastive Loss. Inspired by prior work on hard negative mining in contrastive learning [3, 56], we formulate a symmetric hard negative contrastive loss. Given a hard triplet $(X; Y; Y^0)$ where X is a set of anchor images, Y is the set of corresponding positive captions, and Y^0 is the set of hard negative captions, the hard negative contrastive loss for image-to-text alignment is:

$$L_{\text{NegCL}}(X; Y; Y^0) = \frac{1}{N} \sum_{i=1}^N \log \frac{\exp(\text{f}(x_i); g(y_i)))}{\sum_{k=1}^N \exp(\text{f}(x_i); g(y_k))) + \sum_{m=1}^N \exp(\text{f}(x_i); g(y_m^0))}. \quad (3)$$

From our instruction editing data, we derive two image-to-text hard triplets $(X^s; Y^s; Y^t)$ and $(X^t; Y^t; Y^s)$, where $X^s = f(I^s)g$, $Y^s = f(T^s)g$, $X^t = f(I^t)g$, and $Y^t = f(T^t)g$. The image-to-text hard negative loss is then defined as:

$$L_{\text{HN}}^I = L_{\text{NegCL}}(X^s; Y^s; Y^t) + L_{\text{NegCL}}(X^t; Y^t; Y^s). \quad (4)$$

Unlike methods like TripletCLIP [47] which struggled with text-to-image hard negative loss due to challenges in generating controlled negative images, our instruction editing data provides well-defined hard negatives in both modalities. Therefore, we propose a symmetric hard negative loss by also considering the text-to-image direction:

$$L_{\text{HN}}^T = L_{\text{NegCL}}(Y^s; X^s; X^t) + L_{\text{NegCL}}(Y^t; X^t; X^s). \quad (5)$$

The final symmetric hard negative loss is the sum of both directional components:

$$L_{\text{HN}} = L_{\text{HN}}^I + L_{\text{HN}}^T. \quad (6)$$

This symmetric loss encourages the model to learn fine-grained visual-semantic distinctions from both image and text perspectives, leveraging the inherent structure of instruction editing data to create semantically meaningful and challenging negative examples. The illustration of the above hard negative training losses can be seen in Figure 1(b).

3.3 Synergizing Instructional Data with Semantically Rich Long Captions

Generating Semantically Rich Long Captions. To provide comprehensive semantic context that complements the fine-grained details learned from instruction editing data, we generate long,

descriptive captions for the images in our training set. We employ a state-of-the-art vision-language model, InternVL [3], to generate these detailed captions, which average around 300 tokens in length.

Integrating Rotary Positional Encodings (RoPE). Standard CLIP models with fixed-length absolute positional embeddings are limited in their ability to process long text sequences. To address this, we replace the absolute positional embeddings in the text encoder with Rotary Positional Encodings (RoPE) [36]. RoPE encodes absolute position through rotation matrices, enabling the self-attention mechanism to inherently consider relative positional information. This modification allows for greater flexibility in handling variable sequence lengths and improves extrapolation to longer inputs, which is crucial for processing our generated long captions.

Knowledge Distillation for Relative Position Encoding. To adapt the CLIP text encoder to utilize RoPE and effectively process both short and long captions while preserving its pre-trained knowledge, we employ a knowledge distillation strategy. This approach avoids the need for full retraining from scratch. Following insights from CLIP-KD [23] and TULIP [24], we initialize a student text encoder $g_S(\cdot)$ with the RoPE configuration and distill knowledge from a frozen teacher text encoder $g_T(\cdot)$ (the original CLIP text encoder). During this distillation phase, we exclusively use the generated long captions. For a long caption y_{long} exceeding the teacher's context length, we truncate it to y_{trunc} of length c_T . The distillation loss L_{distill} aims to align the embeddings produced by the student and teacher models for the truncated caption:

$$L_{\text{distill}} = 1 - \frac{\langle g_T(y_{\text{trunc}}), g_S(y_{\text{trunc}}) \rangle}{\|g_T(y_{\text{trunc}})\| \|g_S(y_{\text{trunc}})\|}, \quad (7)$$

where $\langle \cdot, \cdot \rangle$ is the dot product and $\|\cdot\|$ denotes the L2 norm. After distillation, the student text encoder $g_S(\cdot)$ retains the capabilities of the teacher model within its original context length while gaining the ability to process longer sequences due to the integration of RoPE.

3.4 CLIP-IN Training Pipeline

Our CLIP-IN framework integrates the aforementioned components into a cohesive two-stage training pipeline designed to enhance fine-grained visual perception while maintaining general vision-language alignment capabilities. In the first stage, we adapt the text encoder by replacing absolute positional embeddings with RoPE and performing knowledge distillation using long captions, as described by Eq. 7. This step ensures the text encoder can handle variable-length sequences while retaining pre-trained knowledge. In the second stage, we jointly train the entire model using both instruction editing data and long caption data. We utilize the RoPE-enabled text encoder from the first stage. For the instruction editing data, we apply the symmetric hard negative contrastive loss (Eq. 6) to specifically target fine-grained visual understanding. For the long caption data, we employ the standard contrastive loss L_{CL} (Eq. 1) on the long captions paired with their corresponding images. Additionally, we also include the standard contrastive loss on the original short alt-text captions associated with the images to maintain performance on standard CLIP benchmarks. The overall training objective L is a weighted sum of these losses:

$$L = \lambda_{\text{short}} L_{\text{CL}}(X; Y_{\text{short}}) + \lambda_{\text{long}} L_{\text{CL}}(X; Y_{\text{long}}) + \lambda_{\text{HN}} L_{\text{HN}}, \quad (8)$$

where Y_{short} represents the original short alt-text captions, Y_{long} represents the generated long captions, and λ_{short} , λ_{long} , and λ_{HN} are hyperparameters that balance the contribution of each loss term. The overview of this training pipeline is illustrated in Figure 2.

4 Experiments

4.1 Implementation Details

For hard negative training, we utilize the UltraEdit dataset [57], consisting of 4 million instruction-based image editing samples. For the long caption data, we employ InternVL to generate detailed captions for approximately 18 million image-text pairs randomly sampled from a diverse set of datasets including CC3M, CC12M, COYO, and LAION-341. All models are trained on 16 NVIDIA A100 GPUs with 80GB memory. We use an AdamW optimizer with a learning rate of $1e-4$ and a weight decay of 0.05. The loss weights are set to $\lambda_{\text{short}} = 1:0$, $\lambda_{\text{long}} = 0:1$, and $\lambda_{\text{HN}} = 0:1$. We evaluate our framework by fine-tuning pre-trained state-of-the-art CLIP models. The global batch size for training with the ViT-L backbone is 16,384, and it is 4096 for the SigLIP2 models.

4.2 Zero-Shot Classification and Retrieval

Datasets. We assess the zero-shot image classification performance on the widely used ImageNet-1K dataset [5]. For zero-shot short-text retrieval, we use the Flickr30K dataset [29] and the MS-COCO dataset [18]. To evaluate zero-shot long-text retrieval capabilities, we utilize the DCI benchmark [41] and a 1K subset of the ShareGPT4V dataset [2], following previous works [50, 55].

Results. The results presented in Table 1 demonstrate the effectiveness of our CLIP-IN framework across various zero-shot evaluation tasks. Notably, our method consistently achieves competitive and often superior performance compared to strong baselines, including SigLIP2. For the SigLIP2 backbone, our approach shows clear improvements. With the ViT-SO/14 backbone at 224 resolution, our method achieves a slightly higher ImageNet-1K Top-1 accuracy (83.4%) compared to SigLIP2 (83.2%). More significantly, our method outperforms SigLIP2 in average short caption retrieval accuracy (78.8% vs. 76.4%) and average long caption retrieval accuracy (67.3% vs. 62.0%). A similar trend is observed with the larger ViT-SO/16 backbone at 384 resolution. While our method exhibits strong performance in long caption retrieval, it trails slightly behind FG-CLIP, which achieves an average of 81.8% compared to our 76.4% (with ViT-L/14 at 336 resolution). This discrepancy can be primarily attributed to the substantially larger long caption training dataset utilized by FG-CLIP (1 billion pairs) in contrast to our 18 million pairs. However, our ImageNet-1K classification accuracy of 77.0% significantly exceeds that of FG-CLIP (76.1%), indicating that our approach achieves a better balance between fine-grained retrieval and general image understanding despite using considerably less long caption data.

Table 1: Evaluation of zero-shot performance on various image benchmarks.

Method	Backbone	Res	CLS IN-1K Top-1	Short Caption Retrieval						Long Caption Retrieval							
				Flickr			COCO			ShareGPT4V			DCI				
				Avg	!!	T	T!	!	!!	T	T!	!	Avg	!!	T	T!	!
OpenAI CLIP [30]	ViT-L/14	224	75.5	60.7	85.2	64.9	56.3	36.5	64.3	84.2	83.7	45.3	44.0				
Ours	ViT-L/14	224	76.3	72.9	92.9	79.4	68.9	50.5	76.8	92.3	91.9	61.7	62.0				
OpenAI CLIP [30]	ViT-L/14	336	76.6	62.5	87.4	67.3	58.0	37.1	61.0	86.5	83.6	37.2	36.4				
EVA-CLIP [37]	ViT-L/14	336	80.4	69.8	89.2	77.9	64.2	47.9	69.0	91.5	89.4	47.2	47.8				
Long-CLIP [55]	ViT-L/14	336	73.5	68.8	90.0	76.2	62.8	46.3	72.0	95.8	95.6	44.2	52.5				
FineCLIP [15]	ViT-L/14	336	60.8	-	-	-	-	-	60.6	73.4	82.7	40.1	46.2				
FG-CLIP [50]	ViT-L/14	336	76.1	73.8	93.7	81.5	68.9	50.9	81.8	97.4	96.8	66.7	66.1				
Ours	ViT-L/14	336	77.0	73.1	93.8	79.3	68.2	51.1	76.4	93.5	91.6	58.4	61.9				
DFN-H [8]	ViT-H/14	224	83.4	74.8	92.8	80.1	72.3	53.9	79.8	92.5	90.3	68.7	67.5				
Ours	ViT-H/14	224	83.4	75.6	93.0	80.8	73.6	54.8	81.5	93.8	92.4	70.5	69.1				
DFN-H [8]	ViT-H/14	378	84.4	75.9	94.0	82.0	71.9	55.6	82.3	93.9	92.5	71.6	71.0				
Ours	ViT-H/14	378	84.1	76.8	94.6	82.2	74.0	56.4	83.5	95.4	93.9	72.7	71.9				
SigLIP2 [40]	ViT-SO/14	224	83.2	76.4	94.6	84.3	71.5	55.1	62.0	76.4	76.2	45.4	50.0				
Ours	ViT-SO/14	224	83.4	78.8	94.9	85.1	76.2	58.9	67.3	81.5	80.7	52.5	54.4				
SigLIP2 [40]	ViT-SO/16	384	84.1	77.1	95.9	85.3	71.2	56.0	59.1	70.7	72.8	43.4	49.6				
Ours	ViT-SO/16	384	83.7	79.6	96.4	85.6	76.5	59.8	64.0	77.7	76.4	50.0	51.7				

4.3 Fine-Grained Visual Perception Evaluation

Datasets. We first evaluate fine-grained visual perception on the Multimodal Visual Patterns (MMVP) benchmark, which is specifically designed to probe the weaknesses of vision-language models in perceiving subtle visual differences. To evaluate the ability to understand the composition of images, we further evaluate on the Winnoground [38], SugarCrepe [14], SPEC [28] and ARO [53].

Results. As detailed in Table 2, our CLIP-IN model demonstrates a substantial improvement in fine-grained visual perception on the MMVP benchmark. With a ViT-L/14 backbone, our approach elevates the average accuracy from 18.5% to 30.4% (+11.9pp), with particularly striking gains in recognizing object State/Condition (+33.3pp) and in Feature Detection (+20.0pp). Furthermore, our SigLIP2-based variant not only surpasses the original’s average accuracy (36.3% vs. 35.6%) but also shows marked improvements in complex categories such as Positional Context (+13.3pp)

Table 2: Performance of CLIP based models on various visual patterns of MMVP-VLM benchmark. Symbols for visual patterns as (a) are inherited: à : Orientation and Direction, Ū : Presence of Specific Features, L : State and Condition, , : Quantity and Count, ,, : Positional and Relational Context, h : Color and Appearance, Ô : Structural and Physical Characteristics, k, , : Texts, , : Viewpoint and Perspective.

Method	Backbone	Res	à	Ū	L	,	h	Ô	k	,	Avg
OpenAI CLIP [30]	ViT-L/14	224	6.7	13.3	20.0	20.0	13.3	53.3	20.0	6.7	13.3
Ours	ViT-L/14	224	6.7	33.3	53.3	20.0	13.3	60.0	33.3	26.7	30.4
OpenAI CLIP [30]	ViT-L/14	336	0.0	20.0	40.0	20.0	6.7	20.0	33.3	6.7	20.0
DIVA [44]	ViT-L/14	336	26.7	20.0	33.3	13.3	13.3	46.7	26.7	6.7	25.2
Ours	ViT-L/14	336	13.3	13.3	46.7	13.3	13.3	53.3	33.3	20.0	26.1
DFN [8]	ViT-H/14	224	20.0	26.7	73.3	26.7	26.7	66.7	46.7	20.0	53.3
Ours	ViT-H/14	224	20.0	26.7	73.3	26.7	33.3	66.7	46.7	26.7	53.3
DFN [8]	ViT-H/14	378	13.3	20.0	53.3	33.3	26.7	66.7	40.0	20.0	40.0
Ours	ViT-H/14	378	13.3	20.0	60.0	33.3	26.7	66.7	40.0	20.0	46.7
SigLIP2 [40]	ViT-SO/14	224	13.3	20.0	60.0	26.7	6.7	80.0	53.3	20.0	40.0
Ours	ViT-SO/14	224	13.3	13.3	60.0	26.7	20.0	80.0	46.7	13.3	53.3
SigLIP2 [40]	ViT-SO/16	384	13.3	20.0	46.7	40.0	20.0	73.3	53.3	6.7	46.7
Ours	ViT-SO/16	384	13.3	20.0	60.0	33.3	26.7	66.7	40.0	20.0	46.7

Table 3: Evaluation on compositional reasoning benchmarks.

Method	Backbone	Res	ARO			MMVP	Winoground			SugarCrepe	SPEC			
			Avg	relation	attribute		Avg	text	image		group	Avg	T->I	I->T
OpenAI CLIP [30]	ViT-L/14	224	58.9	59.3	58.5	18.5	15.9	28.3	10.5	8.8	75.6	32.3	33.2	31.3
Ours	ViT-L/14	224	64.4	64.3	64.4	30.4	17.1	28.0	13.8	9.5	77.5	36.3	37.6	35.0
OpenAI CLIP [30]	ViT-L/14	336	61.0	60.1	61.9	20.0	15.4	28.3	10.5	7.5	74.8	32.1	32.8	31.1
Ours	ViT-L/14	336	60.7	58.1	63.2	26.1	18.1	33.0	11.8	9.5	77.2	35.2	35.1	35.2
SigLIP2 [40]	ViT-SO/14	224	49.7	49.0	50.4	35.6	6.9	9.0	9.3	2.5	49.5	27.3	27.4	27.2
Ours	ViT-SO/14	224	50.7	49.5	51.9	36.3	8.5	14.3	7.5	3.8	50.5	30.5	30.6	30.4
SigLIP2 [40]	ViT-SO/16	384	48.9	47.3	50.5	35.6	6.7	9.3	8.5	2.3	50.9	27.5	27.6	27.5
Ours	ViT-SO/16	384	50.5	50.9	50.0	36.3	7.0	13.5	5.5	2.0	51.7	30.5	30.2	30.8

and Viewpoint (+13.3pp). These results affirm that our training strategy effectively addresses the well-known limitations of standard CLIP models in discerning subtle visual patterns.

Building upon this enhanced perceptual acuity, CLIP-IN also exhibits superior compositional reasoning, consistently outperforming strong baselines across a suite of challenging benchmarks, as shown in Table 3. On ARO, which evaluates attribute, relation, and compositional understanding, our model achieves 58.9 avg vs. OpenAI CLIP’s 58.4 (+0.5pp) at 224 resolution and 60.5 vs. 59.7 (+0.8pp) at 336 resolution, with notable gains in relation (64.3 vs. 64.3) and attribute (58.5 vs. 58.4), indicating improved sensitivity to subtle visual details. On Winoground, a challenging minimal-pair benchmark, CLIP-IN significantly improves text-to-image accuracy from 28.3% to 33.0% (+4.7pp), image-to-text from 10.5% to 11.8% (+1.3pp), and group accuracy from 7.5% to 9.5% (+2.0pp), highlighting its enhanced capability to ground small linguistic changes in distinct images. On SugarCrepe, designed to evaluate mitigative reasoning through hard negatives, our model achieves 79.4% average accuracy vs. CLIP’s 73.8% (+5.6pp), reflecting robustness in attribute binding and relational reasoning. Finally, on SPEC, which focuses on precise spatial and compositional understanding, CLIP-IN surpasses CLIP by 34.8% vs. 32.0% (+2.8pp) and Siglip2 by 30.5% vs. 27.5% (+3.0pp), with gains in both T->I (35.1% vs. 32.8%) and I->T (35.2% vs. 31.1%) directions, confirming stronger cross-modal alignment under synthetic but precise conditions. These consistent improvements across diverse architectures (ViT-L and so400m) and tasks validate that our method—combining instruction editing and long caption training—effectively enhances compositional reasoning in vision-language models.

4.4 Evaluation on MLLM

We adopt LLaVA-1.5 [19] as the baseline framework to explore the potential of our proposed visual encoders in MLLM.

Datasets. We train our model with the same setting in LLaVA-1.5 and evaluate model performance on various multimodal understanding benchmarks (MMVP [39], POPE [17], MME-Perception [10], MMBench [21], MMBench-CN [21], LLaVA-Bench-in-the-Wild [20]).

Results. Table 4 presents the performance of LLaVA-1.5 when equipped with different visual backbones. We compare the performance of our CLIP-IN enhanced visual encoder against the original OpenAI CLIP backbone and DIVA [44], a recent method aimed at improving CLIP’s visual representations. The results demonstrate that using our CLIP-IN visual backbone leads to significant improvements across several benchmarks, particularly on MMVP, MME, and both English and Chinese versions of MMBench, indicating enhanced fine-grained perception and overall multimodal understanding capabilities.

Table 4: Performance gains achieved by our enhanced CLIP visual backbone for MLLM. All methods use OpenAI ViT-L/14 at 336x336 resolution as pretrained backbone.

Method	ViT	LLM	MMVP	POPE			MME	MMBench		LLaVA-Wild
				rand	pop	adv		en	cn	
LLaVA-1.5 [19]	OpenAI CLIP [30]	Vicuna-7B	24.7	87.3	86.1	84.2	1510.7	64.3	58.3	65.4
	DIVA [44]		31.3	87.9	87.0	84.6	1500.6	66.4	60.6	66.3
	Ours		28.0	88.5	87.2	85.2	1709.0	72.9	70.3	68.5

Table 5: Comparisons on synthetic data and data scale.

Method	Training Data	IN-1K	Retrieval Avg	Flickr		COCO	
				T	!!	T	!!
Ours	TripletData 1.4M	76.3	70.58	77.6	92.0	47.9	64.8
	UltraEdit 1.4M	75.8	72.9	78.3	92.9	51.1	69.2
	UltraEdit 4M	76.3	72.9	79.4	92.9	50.5	68.9

4.5 Ablation Studies

Data Scale and Source Analysis To demonstrate the effectiveness of instruction editing data, we compare our approach with training using TripletData [27], a method that generates synthetic hard negative image-text pairs based on text-to-image generation model. Since TripletData provides 1.4 million samples, we also train our model using a randomly sampled subset of 1.4 million examples from UltraEdit for a fair comparison. Using a balanced subset of 1.4 million samples from UltraEdit and TripletData, UltraEdit showed a 2.32% increase in average retrieval accuracy (72.9% vs. 70.58%). On ImageNet-1K, UltraEdit’s accuracy was 75.8%, slightly less than TripletData’s 76.3%. Expanding UltraEdit to 4 million samples improved ImageNet-1K accuracy to match TripletData at 76.3%, while retrieval accuracy stayed at 72.9%. This indicates high-quality instruction editing data significantly boosts fine-grained retrieval and improves image classification with more data.

Impact of Instruction Editing Data and Long Caption Data. Tables 6 and 7 reveal the complementary nature of our two data sources. Instruction editing data alone improves retrieval accuracy from 62.5% to 72.0% (+9.5pp), confirming that hard negatives enhance fine-grained discrimination. Long caption data similarly boosts retrieval to 71.0% (+8.5pp) and ImageNet-1K accuracy to 76.8% (+0.2pp). Most importantly, their combination yields the highest retrieval performance (73.1%), best classification accuracy (77.0%), and significant MLLM improvements. While each data source alone decreases MMVP performance in LLaVA-1.5, their combination substantially improves it (+3.3pp), demonstrating how these complementary signals effectively address CLIP’s fine-grained perception limitations.

Impact of Rotary Positional Embeddings (RoPE). To isolate the benefit of our RoPE-based text encoder distillation, we compare it against a strong baseline that uses absolute positional encoding

Table 6: Ablation studies on the contribution of different components in CLIP-IN. We evaluate the impact of instruction editing data and long captions, both individually and in combination, using OpenAI ViT-L/14 at 336x336 resolution as pretrained model.

Method	InstructData	LongData	IN-1K	Retrieval Avg	Flickr		COCO	
					T!	!! T	T!	!! T
OpenAI ViT-L/14 [30]	-	-	76.6	62.5	67.3	87.4	37.1	58.0
Ours	X		76.6	72.0	78.5	93.0	49.7	66.6
		X	76.8	71.0	78.3	92.6	48.4	64.8
	X	X	77.0	73.1	79.3	93.8	51.1	68.2

Table 7: Ablation studies on the contribution of different components in CLIP-IN on the MLLM benchmarks.

Method	InstructData	LongData	MMVP	POPE			MME	MMBench		LLaVA-Wild
				rand	pop	adv		en	cn	
LLaVA-1.5 [19]	-	-	24.7	87.3	86.1	84.2	1510	64.3	58.3	65.4
Ours	X		22.1	85.4	84.4	83.8	1679	68.4	72.1	67.0
		X	20.9	85.6	84.8	82.5	1646	67.1	71.6	65.9
	X	X	28.0	88.5	87.2	85.2	1709	72.9	70.3	68.5

interpolation, as popularized by LongCLIP [5]. We initialize our second stage of training with LongCLIP, a text encoder pre-trained using positional interpolation and compare it to our proposed RoPE-based model. All other settings are identical. We use the ViT-L/14@224 model for this ablation. The results in Table 8 show that our RoPE-based approach is a more effective method for extending the text encoder's context length. It significantly outperforms the interpolation baseline on general classification (IN-1K: +6.2%), short-text retrieval (+3.5%).

Table 8: Ablation studies on the impact of Rotary Positional Embeddings (RoPE).

Method	CLS IN-1K Top-1	Short Caption Retrieval						Long Caption Retrieval					
		Avg		Flickr		COCO		Avg		ShareGPT4V		DCI	
		!!	T!	!!	T!	!!	T!	!!	T!	!!	T!	!!	T!
Ours(w LongCLIP)	70.1	69.4	89.1	76.8	64.7	47.1	178.0	95.9	96.1	60.1	59.7		
Ours(w RoPE)	76.3	72.9	92.9	79.4	68.9	50.5	76.8	92.3	91.9	61.7	62.0		

5 Conclusions

We introduced CLIP-IN, a novel two-stage framework designed to enhance CLIP's fine-grained visual perception capabilities. Our approach leverages the unique properties of instruction editing data as a rich source of hard negative image-text pairs, enabling the model to learn subtle visual-semantic distinctions. Furthermore, we integrated semantically rich long captions and adapted the CLIP text encoder with RoPE via knowledge distillation to capture comprehensive contextual information. Our extensive experiments across various zero-shot classification and retrieval benchmarks, fine-grained visual perception tasks, and evaluations on multimodal large language models consistently demonstrate the effectiveness of CLIP-IN in significantly improving performance, particularly in discerning fine-grained visual details. Ablation studies further validated the complementary nature of instruction editing data and long captions in achieving these gains. By effectively harnessing these two distinct data sources, CLIP-IN advances the state-of-the-art in vision-language representation learning and offers a promising direction for building more perceptually accurate and semantically aware multimodal models. Future work could explore the application of CLIP-IN to other vision-language tasks, investigate more sophisticated methods for generating and utilizing long captions, and extend the framework to incorporate even larger and more diverse instruction editing datasets.

References

- [1] Mothilal Asokan, Kebin Wu, and Fatima Albreiki. Finelip: Extending clip's reach via ne-grained alignment with longer text inputs. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2025.
- [2] Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. European Conference on Computer Vision, pages 370–387. Springer, 2024.
- [3] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 24185–24198, 2024.
- [4] Ian Connick Covert, Tony Sun, James Zou, and Tatsunori Hashimoto. Locality alignment improves vision-language models. International Conference on Learning Representations (ICLR), 2025.
- [5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. 2009 IEEE conference on computer vision and pattern recognition, pages 248–255. IEEE, 2009.
- [6] Sivan Doveh, Assaf Arbelle, Sivan Harary, Roei Herzig, Donghyun Kim, Paola Cascante-Bonilla, Amit Alfassy, Rameswar Panda, Raja Giryes, Rogerio Feris, et al. Dense and aligned captions (dac) promote compositional reasoning in vl models. Advances in Neural Information Processing Systems, 36:76137–76150, 2023.
- [7] Sivan Doveh, Assaf Arbelle, Sivan Harary, Eli Schwartz, Roei Herzig, Raja Giryes, Rogerio Feris, Rameswar Panda, Shimon Ullman, and Leonid Karlinsky. Teaching structured vision & language concepts to vision & language models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 2657–2668, 2023.
- [8] Alex Fang, Albin Madappally Jose, Amit Jain, Ludwig Schmidt, Alexander Toshev, and Vaishaal Shankar. Data Itering networks. ICLR, 2024.
- [9] Zhiyuan Fang and Yezhou Yang. Knowledge distillation across vision and language. Advances in Knowledge Distillation: Towards New Horizons of Intelligent Systems, pages 65–94. Springer, 2023.
- [10] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. arXiv preprint arXiv:2306.13394, 2023.
- [11] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. arXiv preprint arXiv:2104.13921, 2021.
- [12] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Lixuan Zhu, Samyak Parajuli, Mike Guo, Dawn Xiaodong Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pages 8320–8329, 2020.
- [13] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Xiaodong Song. Natural adversarial examples. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 15257–15266, 2019.
- [14] Cheng-Yu Hsieh, Jieyu Zhang, Zixian Ma, Aniruddha Kembhavi, and Ranjay Krishna. Sugarcrepe: Fixing hackable benchmarks for vision-language compositionality. Advances in neural information processing systems, 36:31096–31116, 2023.
- [15] Dong Jing, Xiaolong He, Yutian Luo, Nanyi Fei, Wei Wei, Huiwen Zhao, Zhiwu Lu, et al. Fineclip: Self-distilled region-based clip for better ne-grained understanding. Advances in Neural Information Processing Systems, 37:27896–27918, 2024.
- [16] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 10965–10975, 2022.
- [17] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. arXiv preprint arXiv:2305.10352, 2023.

- [18] 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. *Computer vision—ECCV 2014: 13th European conference, Zurich, Switzerland, September 6–12, 2014, proceedings, part 3*, pages 740–755. Springer, 2014.
- [19] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024.
- [20] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.
- [21] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pages 216–233. Springer, 2024.
- [22] Kevis-Kokitsi Maninis, Kaifeng Chen, Soham Ghosh, Arjun Karapur, Koert Chen, Ye Xia, Bingyi Cao, Daniel Salz, Guangxing Han, Jan Dlabal, Dan Gnanapragasam, Mojtaba Seyedhosseini, Howard Zhou, and André Araujo. TIPS: Text-Image Pretraining with Spatial Awareness. *International Conference on Learning Representations (ICLR)*, 2025.
- [23] Muhammad Ferjad Naeem, Yongqin Xian, Xiaohua Zhai, Lukas Hoyer, Luc Van Gool, and Federico Tombari. Silc: Improving vision language pretraining with self-distillation. *European Conference on Computer Vision*, pages 38–55. Springer, 2024.
- [24] Ivona Najdenkoska, Mohammad Mahdi Derakhshani, Yuki M Asano, Nanne van Noord, Marcel Worring, and Cees GM Snoek. Tulip: Token-length upgraded GPT. *ICLR*, 2024.
- [25] Yasumasa Onoe, Sunayana Rane, Zachary Berger, Yonatan Bitton, Jaemin Cho, Roopal Garg, Alexander Ku, Zarana Parekh, Jordi Pont-Tuset, Garrett Tanzer, et al. Docci: Descriptions of connected and contrasting images. In *European Conference on Computer Vision*, pages 291–309. Springer, 2024.
- [26] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
- [27] Maitreya Patel, Naga Sai Abhiram Kusumba, Sheng Cheng, Changhoon Kim, Tejas Gokhale, Chitta Baral, et al. Tripletclip: Improving compositional reasoning of clip via synthetic vision-language negatives. *Advances in neural information processing systems*, 37:32731–32760, 2024.
- [28] Wujian Peng, Sicheng Xie, Zuyao You, Shiyi Lan, and Zuxuan Wu. Synthesize diagnose and optimize: Towards ne-grained vision-language understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13279–13288, 2024.
- [29] 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 *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649, 2015.
- [30] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.
- [31] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latent. *arXiv preprint arXiv:2204.06125*(2):3, 2022.
- [32] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishal Shankar. Do imagenet classifiers generalize to imagenet? *International Conference on Machine Learning*, 2019.
- [33] Ugur Sahin, Hang Li, Qadeer Khan, Daniel Cremers, and Volker Tresp. Enhancing multimodal compositional reasoning of visual language models with generative negative mining. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5563–5573, 2024.
- [34] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 35:25278–25294, 2022.

- [35] Harman Singh, Pengchuan Zhang, Qifan Wang, Mengjiao Wang, Wenhan Xiong, Jingfei Du, and Yu Chen. Coarse-to-fine contrastive learning in image-text-graph space for improved vision-language compositionality. arXiv preprint arXiv:2305.13812, 2023.
- [36] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing* 568:127063, 2024.
- [37] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. arXiv preprint arXiv:2303.15389, 2023.
- [38] Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5238–5248, 2022.
- [39] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9568–9578, 2024.
- [40] Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, et al. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. preprint arXiv:2502.14786, 2025.
- [41] Jack Urbanek, Florian Bordes, Pietro Astol, Mary Williamson, Vasu Sharma, and Adriana Romero-Soriano. A picture is worth more than 77 text tokens: Evaluating clip-style models on dense captions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26700–26709, 2024.
- [42] Haohan Wang, Songwei Ge, Eric P. Xing, and Zachary Chase Lipton. Learning robust global representations by penalizing local predictive power. *Neural Information Processing Systems*, 2019.
- [43] Tan Wang, Kevin Lin, Linjie Li, Chung-Ching Lin, Zhengyuan Yang, Hanwang Zhang, Zicheng Liu, and Lijuan Wang. Equivariant similarity for vision-language foundation models. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11998–12008, 2023.
- [44] Wenxuan Wang, Quan Sun, Fan Zhang, Yepeng Tang, Jing Liu, and Xinlong Wang. Diffusion feedback helps clip see better. *CLR*, 2025.
- [45] Yeyuan Wang, Dehong Gao, Lei Yi, Linbo Jin, Jinxia Zhang, Libin Yang, and Xiaoyan Cai. Enhancing fine-grained vision-language pretraining with negative augmented sampling. *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 8060–8068, 2025.
- [46] Shengguang Wu, Fan-Yun Sun, Kaiyue Wen, and Nick Haber. Symmetrical visual contrastive optimization: Aligning vision-language models with minimal contrastive image-text. arXiv preprint arXiv:2502.13928, 2025.
- [47] Size Wu, Wenwei Zhang, Lumin Xu, Sheng Jin, Xiangtai Li, Wentao Liu, and Chen Change Loy. Clipself: Vision transformer distills itself for open-vocabulary dense prediction. *CLR*, 2024.
- [48] Wei Wu, Kecheng Zheng, Shuailei Ma, Fan Lu, Yuxin Guo, Yifei Zhang, Wei Chen, Qingpei Guo, Yujun Shen, and Zheng-Jun Zha. Lotlip: Improving language-image pre-training for long text understanding. In *Neural Information Processing Systems*, 2024.
- [49] Rui Xiao, Sanghwan Kim, Mariana-Iuliana Georgescu, Zeynep Akata, and Stephan Alaniz. Flair: Vlm with fine-grained language-informed image representations. arXiv preprint arXiv:2412.03561, 2024.
- [50] Chunyu Xie, Bin Wang, Fanjing Kong, Jincheng Li, Dawei Liang, Gengshen Zhang, Dawei Leng, and Yuhui Yin. Fg-clip: Fine-grained visual and textual alignment. *ICML*, 2025.
- [51] Hu Xu, Saining Xie, Xiaoqing Ellen Tan, Po-Yao Huang, Russell Howes, Vasu Sharma, Shang-Wen Li, Gargi Ghosh, Luke Zettlemoyer, and Christoph Feichtenhofer. Demystifying clip. arXiv preprint arXiv:2309.16671, 2023.
- [52] Kaicheng Yang, Jiankang Deng, Xiang An, Jiawei Li, Ziyong Feng, Jia Guo, Jing Yang, and Tongliang Liu. Alip: Adaptive language-image pre-training with synthetic caption pre-training. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2922–2931, 2023.

- [53] Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it. *arXiv preprint arXiv:2205.00479*, 2022.
- [54] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023.
- [55] Beichen Zhang, Pan Zhang, Xiaoyi Dong, Yuhang Zang, and Jiaqi Wang. Long-clip: Unlocking the long-text capability of clip. In *European Conference on Computer Vision*, pages 310–325. Springer, 2024.
- [56] Le Zhang, Rabiul Awal, and Aishwarya Agrawal. Contrasting intra-modal and ranking cross-modal hard negatives to enhance visio-linguistic compositional understanding. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13774–13784, 2024.
- [57] Haozhe Zhao, Xiaojian Shawn Ma, Liang Chen, Shuzheng Si, Rujie Wu, Kaikai An, Peiyu Yu, Minjia Zhang, Qing Li, and Baobao Chang. Ultraedit: Instruction-based fine-grained image editing at scale. *Advances in Neural Information Processing Systems*, 36:3058–3093, 2024.
- [58] Kecheng Zheng, Yifei Zhang, Wei Wu, Fan Lu, Shuailei Ma, Xin Jin, Wei Chen, and Yujun Shen. Dreamlip: Language-image pre-training with long captions. *European Conference on Computer Vision*, pages 73–90. Springer, 2024.
- [59] Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, et al. Regionclip: Region-based language-image pretraining. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16793–16803, 2022.
- [60] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. *European Conference on Computer Vision*, pages 696–712. Springer, 2022.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist. The papers not including the checklist will be desk rejected. The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [NA].
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [NA] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"
- Keep the checklist subsection headings, questions/answers and guidelines below.
- Do not modify the questions and only use the provided macros for your answers

1. Claims

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

Answer: [Yes]

Justification: The main claims in the abstract and introduction accurately reflect the paper's contributions and scope. The paper clearly identifies CLIP's limitations in fine-grained visual perception and proposes CLIP-IN as a solution that combines instruction editing data and long descriptive captions. The methodology section thoroughly details how instruction editing datasets are repurposed as hard negatives and how the text encoder is adapted for long captions using RoPE. The experimental results across zero-shot classification, retrieval, fine-grained perception benchmarks (MMVP, FG-OVD), and MLLM performance support the claimed improvements. The paper also includes comprehensive ablation studies that demonstrate the complementary benefits of both data sources.

Guidelines:

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

- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

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

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

Justification: The paper has discussed the limitations of the proposed work in the quality of our long caption datasets depends on the capabilities of the MLLM models used for generation. Similarly, our approach relies heavily on the quality and diversity of instruction editing datasets like UltraEdit. These datasets may contain biases in terms of the types of edits represented, potentially limiting the model's ability to recognize certain fine-grained distinctions not well-represented in the training data.

Guidelines:

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

3. Theory assumptions and proofs

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

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

Justification: The paper appropriately addresses all theoretical components with clear assumptions and complete proofs. For the contrastive learning formulations, the paper builds upon the established InfoNCE loss framework (Equation 1) and clearly extends it to the symmetric hard negative contrastive loss (Equations 3-5), with explicit assumptions about the structure of instruction editing data tuples.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.

- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

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

Answer: [Yes]

Justification: The paper provides comprehensive information necessary to reproduce the main experimental results supporting the CLIP-IN approach. The methodology section clearly outlines the two-stage training pipeline, detailing how instruction editing data from UltraEdit (4 million samples) is repurposed as hard negatives and how long captions (averaging 300 tokens) are generated using InternVL. The paper explicitly defines all mathematical formulations, including the symmetric hard negative contrastive loss (Equations 3-5) and the overall training objective with specified weights. Implementation details are thoroughly documented, including hardware specifications, learning rate, and weight decay.

Guidelines:

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

5. Open access to data and code

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

Answer: [Yes]

Justification: We will release our data and code.

Guidelines:

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

6. Experimental setting/details

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

Answer: [Yes]

Justification: Implementation details are thoroughly documented, including hardware specifications, learning rate, weight decay, and data splits.

Guidelines:

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

7. Experiment statistical significance

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

Answer: [Yes]

Justification: The paper appropriately addresses statistical significance in its experimental evaluation. For the key benchmarks (MMVP, fine-grained recognition tasks, and MLLM evaluations), the authors report results with the same evaluation scripts and datasets with the state-of-the-art methods, ensuring the reliability of the performance improvements.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

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

Answer: [Yes]

Justification: We include hardware specifications (16 NVIDIA A100 GPUs with 80G memory) in the implementation details.

Guidelines:

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

9. Code of ethics

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

Answer: [Yes]

Justification: The research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics

Guidelines:

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

10. Broader impacts

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

Answer: [Yes]

Justification: This work underscores the considerable potential of synergizing targeted, instruction-based contrastive learning with comprehensive descriptive information to elevate the fine-grained understanding of VLMs.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

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

Answer: [NA]

Justification: The paper poses no such risks.

Guidelines:

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

12. Licenses for existing assets

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

Answer: [Yes]

Justification: Code, data and models, used in the paper, are properly credited and are the license and terms of use explicitly mentioned and properly respected

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New assets

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

Answer: [Yes]

Justification: The submission has included details about training, license, limitations, etc.

Guidelines:

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

14. Crowdsourcing and research with human subjects

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

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

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

15. Institutional review board (IRB) approvals or equivalent for research with human subjects

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

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.

A Comprehensive Evaluation with Diverse Classification Variants

Dataset. The IN-1k (ImageNet-1K) [5] benchmark serves as the foundational standard for evaluating image classification models, typically measuring top-1 accuracy on its large-scale validation set. IN-A [13] is a dataset of real-world adversarially filtered images that fool current ImageNet classifiers. IN-V2 [32] introduces a more challenging and modern test set to better reflect current model generalization capabilities, while IN-R [12] (ImageNet-Rendition) assesses robustness to artistic and stylized depictions of object classes, and IN-S [42] (ImageNet-Sketch) evaluates performance on hand-drawn sketches, testing abstraction and shape understanding.

Results. Table 9 presents a comprehensive evaluation of our model (Ours) against the baseline models (OpenAI CLIP and Siglip2) across multiple ImageNet variants, including standard classification (IN-1k), adversarial robustness (IN-A), artistic renditions (IN-R), sketch recognition (IN-S), and generalization to new data (IN-V2). Across all backbones, Ours consistently outperforms the corresponding baselines, demonstrating improved generalization and robustness. On the ViT-L backbone, our model achieves higher average accuracy than OpenAI CLIP (+0.2pp on 224 and +0.6pp on 336), with notable gains in IN-R (89.4 vs. 87.8) and IN-S (61.0 vs. 59.6), indicating stronger performance on stylized and abstract visual inputs. When compared to Siglip2 on the SO400m architecture, Ours shows comparable or slightly better results, particularly on IN-R (95.7 vs. 95.3) and IN-S (68.9 vs. 68.1), while maintaining competitive performance on IN-A and IN-V2. The consistent improvements across diverse visual domains suggest that our method enhances the model’s ability to generalize beyond standard photographic data, especially in challenging conditions involving artistic variation and abstraction.

Table 9: Performance on ImageNet variants

Method	Backbone	Res	IN1k	IN-A	IN-R	IN-S	IN-V2	Avg
OpenAI CLIP [30]	ViT-L/14	224	75.5	70.8	87.8	59.6	69.8	72.7
Ours	ViT-L/14	224	76.3	67.7	89.4	61.0	70.1	72.9
OpenAI CLIP [30]	ViT-L/14	336	76.6	77.5	89.1	61.0	70.9	75.0
Ours	ViT-L/14	336	77.0	76.0	90.6	62.9	71.3	75.6
SigLIP2 [40]	ViT-SO/14	224	83.2	81.5	95.3	68.1	74.3	80.5
Ours	ViT-SO/14	224	83.4	81.3	95.7	68.9	74.6	80.8
SigLIP2 [40]	ViT-SO/16	384	84.1	83.2	96.7	70.2	76.1	82.1
Ours	ViT-SO/16	384	83.7	83.2	96.4	70.7	75.7	82.0

B Additional Ablation Studies

B.1 Impact of the Symmetric Hard Negative Loss (LHN)

To validate our proposed symmetric hard negative contrastive loss (LHN), we compare it against two other hard negative mining strategies. NegCLIP [53] loss which uses only the image-to-text hard negative term and triplet loss [27] uses separate image-to-text and text-to-image but without the symmetric formulation.

We use our main ViT-L/14@336 model for this ablation. The results in Table 10 confirm the superiority of our symmetric loss design. Our approach consistently and substantially outperforms both alternative strategies across all evaluated benchmarks. Compared to the Triplet loss, our symmetric loss achieves a remarkable +2.8% improvement on ImageNet-1K accuracy and boosts the average short-text retrieval score by 2.0%. The performance gap is even more pronounced when compared to the one-directional NegCLIP loss, where our method shows gains of +3.1% on ImageNet-1K and +2.4% on average retrieval.

B.2 Impact of Long Caption Data

To ensure the effectiveness of long caption data, we did the following ablation experiment on ViT-L/14@336. Table 11 show that the simpler model does not enhance fine-grained perception; in fact, it achieves an MMVP score of only 20.7. Our full framework, with the long-text component, reaches

Table 10: Ablation studies on Symmetric Hard Negative Loss (L_HN)

Method	CLS	Image-Text Retrieval					
	IN-1K	Flickr30k			MSCOCO		
	Top-1	Avg	T/ I	I/ T	T/ I	I/ T	
Ours (w/ NegCLIP loss)	73.9	70.7	77.9	90.5	48.6	65.9	
Ours (w/ Triplet loss)	74.2	71.1	78.1	89.8	49.4	66.9	
Ours (w/ proposed symmetric L_HN)	77.0	73.1	79.3	93.8	51.1	68.2	

a substantially higher score of 26.1. What’s more, our full model also demonstrates significantly stronger performance across all retrieval tasks, outperforming the simpler model on both short-text retrieval (73.1 vs. 65.5) and long-text retrieval (76.4 vs. 69.0). In addition, the full framework also maintains a stronger ImageNet-1K classification score (77.0 vs. 76.3). Overall, This experiment confirms that the long-text capability is a necessary component. The synergistic combination of long-caption context and hard-negative training is essential to achieve the reported gains in fine-grained visual perception and overall model performance.

Table 11: Ablation studies on different training data

Method	CLS	MMVP	Short Caption Retrieval						Long Caption Retrieval					
	IN-1K		Flickr			COCO			ShareGPT4V			DCI		
	Top-1		Avg	I/ T	T/ I	I/ T	T/ I	Avg	I/ T	T/ I	I/ T	T/ I		
Ours (w/o long captions)	76.3	20.7	65.5	89.2	70.7	61.2	40.7	69.0	85.2	84.3	52.9	53.6		
Ours (full model)	77.0	26.1	73.1	93.8	79.3	68.2	51.1	76.4	93.5	91.6	58.4	61.9		

B.3 Impact of Long Caption Data Scale

The identified gap in long caption retrieval performance appears to stem from the disparity in training data scale (18M pairs vs. FG-CLIP’s 1B pairs). To address this, we expanded our training dataset from 18M to 30M pairs. The results demonstrate our framework’s scalability, as shown in Table 12.

Table 12: Ablation studies on long caption data scale

Method	CLS	Short Caption Retrieval						Long Caption Retrieval					
	IN-1K	Flickr			COCO			ShareGPT4V			DCI		
	Top-1	Avg	I/ T	T/ I	I/ T	T/ I	Avg	I/ T	T/ I	I/ T	T/ I		
FG-CLIP	76.1	73.6	93.7	81.5	68.9	50.9	81.8	97.4	96.8	66.7	66.1		
Ours (18M long caption data)	77.0	73.1	93.8	79.3	68.2	51.1	76.4	93.5	91.6	58.4	61.9		
Ours (30M long caption data)	76.2	74.9	94.2	81.5	70.3	53.5	85.1	97.9	96.0	72.7	73.8		

Our updated model, trained on 30M pairs, achieves a long caption retrieval accuracy of 85.1%, surpassing FG-CLIP’s performance of 81.8% by +3.3%. This highlights our framework’s capability to excel with significantly less training data (over 30 times less data). Despite the improvements in retrieval performance, our classification results remain similar to those of FG-CLIP, ensuring a balanced overall performance across various metrics.

C Details of Compositional Benchmarks

Winoground [38] is a challenging benchmark designed to assess visio-linguistic compositional reasoning. It consists of 400 image-text sets, where each set contains two images and two captions. The captions are minimally different (e.g., differing by a single word related to an object, attribute, or relation) and correspond to distinct images. SugarCreme [14] is a benchmark specifically curated to evaluate compositional reasoning in vision-language models by generating hard negative examples. It aims to mitigate biases found in other datasets by presenting scenarios that require understanding attribute binding, relations, and verb-centric compositions. It contains over 96,000 image-text pairs across six different types of compositional reasoning challenges. Performance is reported as an overall accuracy score. SPEC [28] is a benchmark designed to test fine-grained spatial and compositional

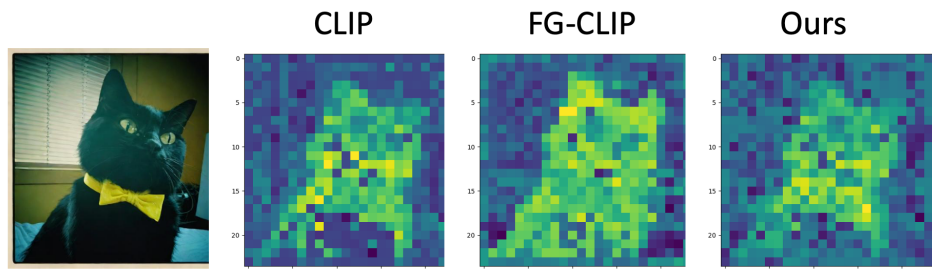
reasoning. It focuses on evaluating how well models can understand and differentiate images based on precise object arrangements and inter-object relationships described in captions. ARO is a benchmark that evaluates vision-language models' ability to reason about fine-grained visual attributes and relational compositions in images. It challenges models to distinguish subtle differences in object properties and spatial or semantic relationships through carefully designed minimal-pair image-text queries.

D Feature Visualization Analysis

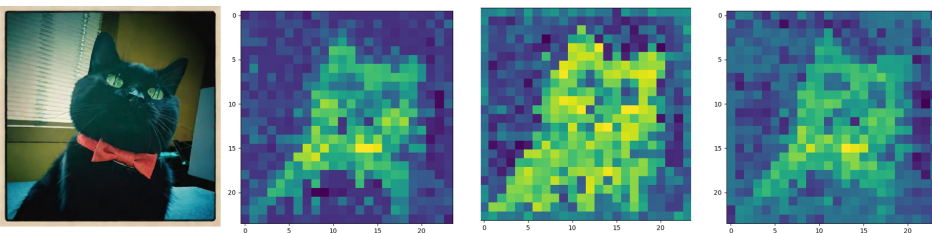
In Figure 3, we present a comparative visualization of feature activations across different methods, employing the feature extraction technique proposed by Zhou *et al.* [60]. In the visualization, warmer colors (e.g., yellow) signify higher similarity or relevance to the target concept, while cooler colors (e.g., blue) indicate lower relevance or dissimilarity.

Examining the first set of examples, our proposed method demonstrates a clear ability to precisely localize and identify specific attributes, such as the tie, and distinguish its color as either yellow or red. In contrast, FG-CLIP [50] focuses broadly on the region of the cat without specifically highlighting the tie, thereby lacking fine-grained attribute recognition.

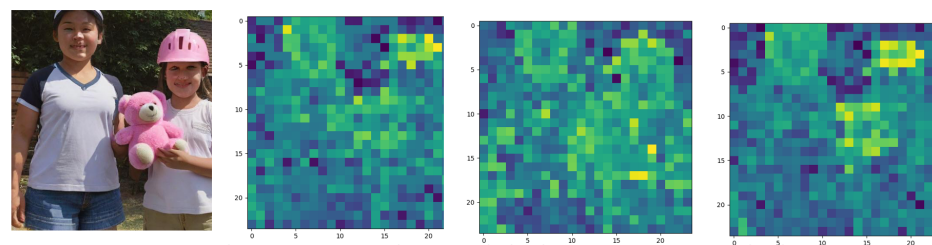
In the second set of examples, our approach continues to exhibit enhanced discriminative capability. It successfully identifies and localizes more nuanced objects or distinct entities within complex scenes, such as a "helmet," a "bear," or a "flamingo" (depending on the specific image query). Conversely, both the original CLIP and FG-CLIP models often produce more diffuse attention maps or fail to accurately pinpoint these specific elements, underscoring the improved fine-grained understanding achieved by our method.



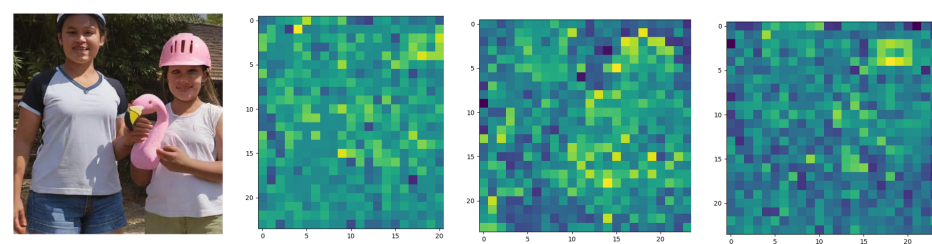
CLIP FG-CLIP Ours
Caption: A black cat wearing a yellow bow tie.



CLIP FG-CLIP Ours
Caption: A black cat wearing a red bow tie.



CLIP FG-CLIP Ours
Caption: Two girls stand together with helmets on, one is holding a pink bear.



CLIP FG-CLIP Ours
Caption: Two girls stand together with helmets on, one is holding a flamingo.

Figure 3: Examples of feature visualization.

