ID-Align: RoPE-Conscious Position Remapping for Dynamic High-Resolution Adaptation in Vision-Language Models

Anonymous ACL submission

Abstract

The rapid advancement of Vision-Language Models (VLMs) has driven researchers to increase image token counts through dynamic high-resolution strategies to enhance the capabilities of VLMs, typically involving image upscaling, grid-based cropping, and joint encoding of multi-resolution patches. Although this approach enriches visual detail, it inadvertently introduces challenges due to the long-range decay characteristics of Rotary Position Embedding (RoPE). Specifically, excessive positional gaps between high- and low-resolution tokens disrupt their spatial correspondence, limiting the model's fine-grained perception capabilities. To address this issue, we introduce **ID-Align**, an innovative positional encoding strategy designed to preserve hierarchical relationships focusing on the alignment of image token position IDs across varying resolutions. In this method, high-resolution tokens inherit IDs from their corresponding low-resolution counterparts while constraining the overexpansion of positional indices. Our experiments conducted within the LLaVA-Next framework demonstrate that ID-Align achieves significant improvements, including a 6.07% enhancement on MMBench's cross-instance fine-grained perception tasks and notable gains across multiple benchmarks.

1 Introduction

The swift advancement in large language models (Achiam et al., 2023; Cai et al., 2024; Yang et al., 2024; Liu et al., 2024a), has not only revolutionized the field of natural language processing but also catalyzed the emergence of vision-language models (Liu et al., 2024d; Wu et al., 2024; Chen et al., 2024c; Li et al., 2023a; Wang et al., 2024). In the architecture of these advanced VLMs, visual encoders like CLIP (Radford et al., 2021) ViT (Dosovitskiy, 2020) or SigLip (Zhai et al., 2023) ViT are primarily utilized to encode images. Following this, mechanisms—such as MLP (Liu et al., 2024d) or Q-former (Li et al., 2023a) —are employed to fuse the encoded visual information with textual data. This multimodal information is then processed by LLMs, enabling comprehensive understanding and generation of contextually relevant responses across both visual and textual domains (Yin et al., 2023).

In pursuit of developing more effective VLMs, researchers undertake multifaceted efforts, including curating higher-quality training datasets (Bai et al., 2024) and refining model architectures (Cha et al., 2024). Beyond these strategies, researchers have discovered that one efficient approach to boost the performance of VLMs involves increasing the number of image tokens generated through image encoding (Dai et al., 2024; Deitke et al., 2024; Wu et al., 2024; Chen et al., 2024b; Liu et al., 2024b). Since the majority of ViTs are limited to processing images of specific, fixed resolutions, many current models, including those mentioned before, adopt a two-step strategy: initially resizing images to higher resolutions followed by cropping these images into manageable patches compatible with ViT requirements. Subsequently, the encoded image embeddings will be processed in the manner described previously.

Despite being straightforward and effective, this method exhibits several critical shortcomings. Following the initial step, the processed image embeddings are treated identically to text embeddings and are directly input into the LLM. However, the application of Rotary Position Embedding (RoPE) (Su et al., 2024), the most widely used method for position encoding, may pose specific challenges owing to its characteristic of long-range decay. This characteristic can result in the following problems:

- **Interaction Issues between Image Embedding and Text Embedding:** Implementing a high-resolution strategy leads to an overproduction of image embeddings. This surplus

can adversely impact the interaction between text embeddings and those image embeddings that occur earlier in the sequence.

- **The Loss of Correspondence between Low-Resolution and High-Resolution Images:** Given that there inherently exists a correspondence between high-resolution and low-resolution images, it is preferable to maintain this relationship when processing image embeddings. However, the long-range decay characteristic of RoPE can weaken this correspondence.

Neglecting this correspondence weakens the effectiveness of the high-resolution strategy, as it prevents high-resolution image embeddings from effectively interacting with their corresponding low-resolution image embeddings. This is particularly detrimental to the model's performance on multimodal tasks that require fine-grained perception, such as handling tasks involving multiple instances or rich text tasks like those including charts or other detailed elements.

To address this issue, we propose **ID-Align**: by rearranging the position IDs of embeddings, we preserve the correspondence between high-resolution image embeddings and their low-resolution counterparts by assigning identical positional encodings to both. This method not only maintains the relationship between high-resolution and low-resolution image embeddings but also mitigates the problem of excessive growth in position IDs caused by a large number of image embeddings. We conducted experiments on the LLaVA-Next (Liu et al., 2024c) architecture, and the results demonstrate that our approach significantly enhances the model's capabilities, particularly in aspects related to fine-grained perception of global information. Our contributions can be summarized into the following two points:

- We first analyzed the adverse effects of the long-range attenuation characteristics of RoPE when increasing the number of image embeddings using the aforementioned super-resolution methods.

- On this basis, we introduce ID-Align, a technique for reorganizing position IDs. This method is aimed at maintaining the correspondence between image embeddings across different resolutions and mitigating the excessive growth of position IDs caused by dynamic adjustments to higher resolutions. Our experiments on the architecture and datasets of LLaVA-Next confirm the effectiveness of ID-Align.

2 Background & Related Work

2.1 Vision Language Model

LLMs have exhibited outstanding cognitive and reasoning capabilities. This has naturally prompted the idea of utilizing these models as a foundational basis for processing visual information. A common practice is to employ a projector to connect a pre-trained LLM with a visual encoder, thereby enabling the LLM to interpret visual information (Zhang et al., 2024a). For image inputs I_{image}, it is usual to first encode them using vision encoders such as SigLIP (Zhai et al., 2023) or CLIP (Radford et al., 2021) ViT (Dosovitskiy, 2020):

$$F_{image} = VE(I_{image}) \tag{1}$$

Subsequently, the projector processes the encoded image features F_{image}:

$$P_{image} = Projector(F_{image}, I_{text}) \tag{2}$$

where I_{text} represents the text input. In certain architectures, such as BLIP-2 (Li et al., 2023a), I_{text} also interacts with F_{image} at this stage. Following this, the LLM backbone processes I_{text} alongside P_{image}, generating the corresponding output:

$$Output = LLM(I_{text}, P_{image}) \tag{3}$$

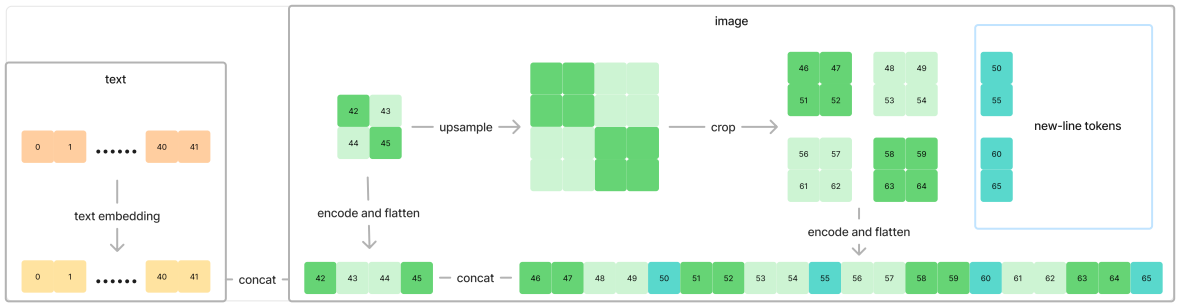
The architecture of the projector has many possible designs, and currently, a mainstream choice is to use a two-layer Multilayer Perceptron (MLP) to process F_{image} independently of I_{text}, as exemplified by the LLaVA architecture (Liu et al., 2024d):

$$P_{image} = MLP(F_{image}) \tag{4}$$

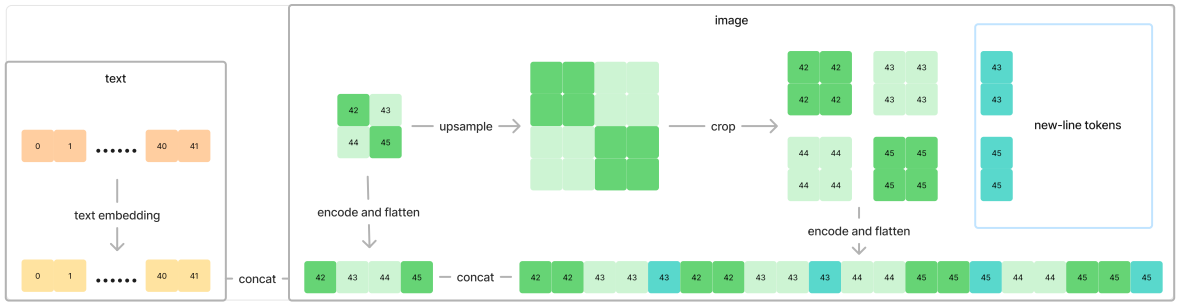
2.2 Dynamic High-resolution

The performance of VLMs can be influenced by a variety of factors, with the resolution of input images and the number of image tokens playing a crucial role (Li et al., 2024a).

Therefore, a reasonable approach is to use higher-resolution image inputs to obtain a greater number of image embeddings. However, ViTs are designed to handle images of a fixed resolution only. Currently, a mainstream approach is to



(a) the original method



(b) ID-Align

Figure 1: Intuitive presentation of the original high-resolution method and ID-Align.

upsample the original image to a higher resolution and then divide the high-resolution image into crops that are suitable for processing by ViTs. Subsequently, both the original image and the crops are processed separately. This approach has been widely embraced by a multitude of leading VLMs (Dai et al., 2024; Deitke et al., 2024; Wu et al., 2024; Chen et al., 2024b; Liu et al., 2024b).

For an input image of dimensions (H_0, W_0), the process involves resizing it into a lower-resolution image, denoted as I_{low} with dimensions (H_v, W_v), where (H_v, W_v) refer to the height and width dimensions that the ViT can process, and also upscaling it to a higher resolution based on its aspect ratio, resulting in I_{high} with dimensions $(m{*}H_v, n{*}W_v)$. For the I_{low}, a ViT directly encodes it. In contrast, for the I_{high}, it is cropped into $m \cdot n$ segments, each of size (H_v, W_v), which are encoded separately. Since m and n are typically selected from a set of preset values based on the aspect ratio of the image, this method is referred to as "dynamic".

The visual embeddings obtained from the high-resolution image are then rearranged, flattened, and combined with the embedding from the low-resolution image. This combined representation is processed by a projector before being fed into the LLM. The procedure is illustrated by Figure 1a.

This strategy allows for more detailed image information to be captured and processed. Notably, while some models such as Qwen2-VL (Wang et al., 2024) have integrated modified ViTs, such as NaViT (Dehghani et al., 2024), capable of handling varied input sizes directly, this remains an exception rather than the norm.

2.3 RoPE

The sequential nature of natural language is pivotal for understanding its semantics. However, the attention mechanism employed in the Transformer (Vaswani, 2017) architecture does not inherently capture this sequential information. Consequently, it is essential to incorporate positional encoding within the Transformer model to enable the processing of sequence-dependent information. For the query $\boldsymbol{q}$ with the position ID m and key $\boldsymbol{k}$ with the position ID n, positional encoding is applied to incorporate positional information into them:

$$\hat{\boldsymbol{q}} = PE(\boldsymbol{q}, m), \hat{\boldsymbol{k}} = PE(\boldsymbol{k}, n) \quad (5)$$

Positional encoding can be implemented in various ways (Gehring et al., 2017; Liu et al., 2020; Shaw et al., 2018; Dai, 2019; Raffel et al., 2020;

He et al., 2020; Wang et al., 2019). Nowadays, in the choice of positional encoding methods, Rotary Position Embedding (RoPE) (Su et al., 2024) has become a prevalent encoding method. The implementation of RoPE is as follows:

$$RoPE(\mathbf{q}, m) = \mathcal{R}_m \mathbf{q} \tag{6}$$

where:

$$\mathcal{R}_m = \begin{pmatrix} A_0 & 0 & \cdots & 0 \\ 0 & A_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_{d/2-1} \end{pmatrix} \tag{7}$$

$$A_i = \begin{pmatrix} \cos m\theta_i & -\sin m\theta_i \\ \sin m\theta_i & \cos m\theta_i \end{pmatrix} \tag{8}$$

$$\theta_i = \theta^{-\frac{2i}{d}} \tag{9}$$

Where d is the dimensionality of $\mathbf{q}$, θ is a hyperparameter, typically taking values ranging from 10^4 to 10^7.

RoPE exhibits several key characteristics:

- RoPE can be described as a form of absolute positional encoding because it uses the absolute positions of tokens during the encoding process. However, it also exhibits properties of relative positional encoding due to its mathematical property:

$$\begin{aligned} (\mathcal{R}_m \mathbf{q})^T (\mathcal{R}_n \mathbf{k}) &= \mathbf{q}^T \mathcal{R}_m^T \mathcal{R}_n \mathbf{k} \\ &= \mathbf{q}^T \mathcal{R}_{n-m} \mathbf{k} \end{aligned} \tag{10}$$

- RoPE exhibits a characteristic of long-range decay: for a query $\mathbf{q}$ at position m and a key $\mathbf{k}$ at position n, after encoding with RoPE, the dot product $(\mathcal{R}_m \mathbf{q})^T (\mathcal{R}_n \mathbf{k})$ generally decreases as the absolute value of $|m - n|$ increases.

- The value of θ controls the positional encoding's sensitivity to positional differences. A smaller θ makes the model more sensitive to position changes, whereas a larger one facilitates the capture of long-range dependencies. Generally, the value of θ should increase as the training length increases (Men et al., 2024).

In the domain of VLMs, researchers are exploring modifications to RoPE to better accommodate multimodal features. Approaches such as CCA (Xing et al., 2025) and PyPE (Chen et al., 2025) aim to reconfigure position IDs from distinct angles, whereas V2PE (Ge et al., 2024) narrows the incremental scale of positional encodings specifically for image embeddings. Despite these advancements, none of these proposed methods sufficiently consider the prevalent application of super-resolution techniques—a critical aspect of the current technological landscape.

3 Motivation

Let's first review the mainstream methods. Assuming an image input with the shape of (H_0, W_0), with $H_0 \geq W_0$, it is first resized to dimensions (H_v, W_v) to be suitable for processing the ViT. Subsequently, this image is super-resolved to dimensions (mH_v, nW_v) to obtain a greater number of visual tokens. Employing a ViT with patch size p to encode the input image, we derive a sequence of $(H_v W_v)/p^2 + (mnH_v W_v)/p^2$ image embeddings. After adding new-line tokens, we get $(H_v W_v)/p^2 + (mnH_v W_v)/p^2 + nW_v/p$ image embeddings Following the conventional processing method, we treat image embeddings in the same manner as text embeddings. Assuming the position ID of the first image embedding in the entire image-text sequence is i , according to this approach, the position IDs of this image token sequence will range from i to $i + (H_v W_v)/p^2 + (mnH_v W_v)/p^2 + nW_v/p - 1$. Based on these position IDs, RoPE is applied to the image tokens.

Due to the long-distance decay characteristic of RoPE, this method encounters the following issues.

3.1 Long-Distance Decay Issue

Taking a CLIP ViT with a patch size of 14 and a resolution of 336×336 as an example, if the input image is first resized to 336×336 and then upsampled to 672×672, followed by cropping into four 336×336 sections. Encoding both the low-resolution image and the four crops results in $24 \times 24 \times 5 + 24 \times 2 = 2928$ image tokens. If we adopt a ViT that generates an even greater number of image tokens or upscale the image to an even higher resolution, this issue becomes more pronounced. Considering the long-distance decay characteristic of RoPE, this abundance of image tokens leads to the model's inability to effectively handle interactions between image tokens and text tokens.

3.2 Neglect of Correspondence

The ViT trained with CLIP or SigLIP demonstrates shortcomings in terms of fine-grained perception (Tong et al., 2024). The dynamic super-resolution method employs a ViT to encode particular segments of the source image, facilitating the extraction of finer details. This technique relies on high-resolution embeddings to capture detailed, fine-grained features, while low-resolution embeddings encapsulate broader, more generalized attributes. There exists a correspondence between these two types of embeddings; if this relationship is effectively leveraged, it can better integrate local and global information, thus enhancing fine-grained perception, and multi-instance perception. For example, in the mini-Gemini model (Li et al., 2024c), the mechanism is designed such that during cross-attention calculations, low-resolution embeddings engage solely with their corresponding high-resolution embeddings. However, the current method of assigning position IDs overlooks this relationship. This issue manifests in two key areas.

On one hand, when the embedding of high-resolution images interacts with their corresponding low-resolution counterparts, this correspondence is not effectively preserved. In the computation of the dot product of query and key values within the attention mechanism for high-resolution image crops and the low-resolution image, long-distance decay reduces the correlation between these tokens. This effect is particularly pronounced for tokens located in the bottom-right corner of the high-resolution image, which are the farthest from their corresponding regions in the low-resolution image. Concerning the tokens located at the lower right corner, it is observed that, compared to their corresponding low-resolution tokens, numerous high-resolution tokens exhibit closer proximity in terms of their position IDs. Conversely, the corresponding low-resolution tokens are situated at a noticeable distance. This spatial arrangement poses a challenge in differentiating the associated low-resolution tokens from other low-resolution tokens.

Another critical aspect lies in the interaction between image embeddings and text embeddings. In the calculation of the dot product of query and key values between text embeddings and image embeddings, significant differences in position IDs between high-resolution image embeddings and their corresponding low-resolution image embeddings can hinder the model's ability to capture the correspondence between low-resolution and high-resolution regions.

The dynamic characteristics of the aforementioned upsampling approach additionally serve to obscure these correspondences. The variability in the number of high-resolution embeddings results in discrepancies in the distances between high-resolution embeddings and their corresponding low-resolution counterparts across images with differing aspect ratios.

Consequently, it is necessary to explore and implement additional methodologies that can effectively preserve such correspondences.

4 Methods

Considering the operational mechanism of a casual language model and the implementation form of RoPE (Rotary Position Embedding), it is the relative positional relationship that dictates whether attention computation occurs. The difference in position IDs between entities represents the true distance, rather than the distance based on embeddings within the sequence. Therefore, without altering the order of embeddings, one viable method to modify the interaction patterns among embeddings involves adjusting their position IDs. By assigning closer position IDs to corresponding image embeddings, it becomes possible to enhance their attention score during the computation process. Adopting this viewpoint, we have reorganized the position IDs assigned to image embeddings, meticulously aligning the embeddings derived from high-resolution images with their corresponding counterparts in low-resolution images. This alignment is aimed at boosting their attention scores. We denote this strategy as **ID-Align**. Our approach is as follows:

- For the embeddings of low-resolution images, we adopt the same position IDs as those used in the previously established approach.
- For the embeddings of high-resolution images, we adjust their embeddings to match the position IDs of their corresponding low-resolution embeddings.

The algorithm is presented in Algorithm 1. We provide an illustrative example in Figure 1b.

Through the reorganization of position IDs, the "distance" between low-resolution embeddings and

their corresponding high-resolution embeddings is reduced. This adjustment not only brings related embeddings closer in terms of positional encoding but also effectively restricts the growth of position IDs. Consequently, this approach prevents the issue of position IDs increasing by thousands when processing a single image, which could otherwise lead to exceeding the maximum position ID values encountered during training.

Algorithm 1 ID-Align with RoPE

Require:

E_{text}: Sequence of text embeddings
E_{low}: Sequence of low-resolution image embeddings
E_{high}: Sequence of high-resolution image embeddings
$\mathcal{M} : E_{\text{high}} \rightarrow E_{\text{low}}$: Return the E_{low} corresponding to E_{high}

Ensure:

$max\_pid \leftarrow 0$
$E_{\text{merged}} \leftarrow \text{Concat}(E_{\text{text}}, E_{\text{low}}, E_{\text{high}})$
for each embedding $e_i \in E_{\text{merged}}$ **do**
  if $e_i \in E_{\text{text}} \cup E_{\text{low}}$ **then**
    $\text{pos\_id}(e_i) \leftarrow max\_pid$
    $max\_pid \leftarrow max\_pid + 1$
  else if $e_i \in E_{\text{high}}$ **then**
    $\text{pos\_id}(e_i) \leftarrow \text{pos\_id}(\mathcal{M}(e_i))$
    $max\_pid \leftarrow \max(max\_pid, \mathcal{M}(e_i) + 1)$
  end if
end for

function APPLYROTARYENCODING(E_{merged})
  for each $e_i \in E_{\text{merged}}$ **do**
    $e_i \leftarrow \text{RoPE}(e_i, \text{pos\_id}(e_i))$
  end for
  return E_{merged}
end function

5 Experiments and Results

5.1 Experiments Setup

We adopted the LLaVA-Next architecture (Liu et al., 2024c), utilizing Vicuna-1.5 7B (Zheng et al., 2023) as the base model and CLIP ViT-L/14 (336) (Radford et al., 2021) as the visual encoder. For training data, we used the dataset provided by LLaVA-Next (Liu et al., 2024c). As for the hyperparameter settings, we adopted the configurations from Open-LLaVA-Next (Chen and Xing, 2024). We will also list these hyperparameters in Appendix A.

The Vicuna model employs a RoPE θ value of 10^4, indicating it is relatively sensitive to positional changes. Given this characteristic, we opted for the Qwen-2.5-7B-Instruct (Yang et al., 2024) model with a θ value of 10^7 , alongside using SigLip 400M (Zhai et al., 2023) as the visual encoder. Compared to the Vicuna and CLIP models, these selections offer enhanced capabilities.

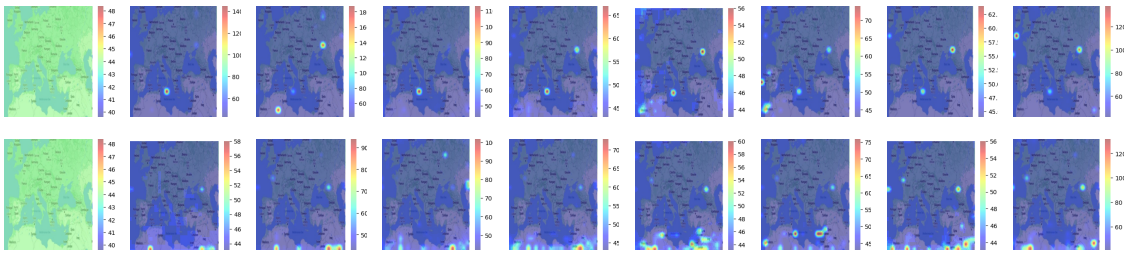
All experiments were conducted using eight A800 GPUs.

5.2 Benchmarks

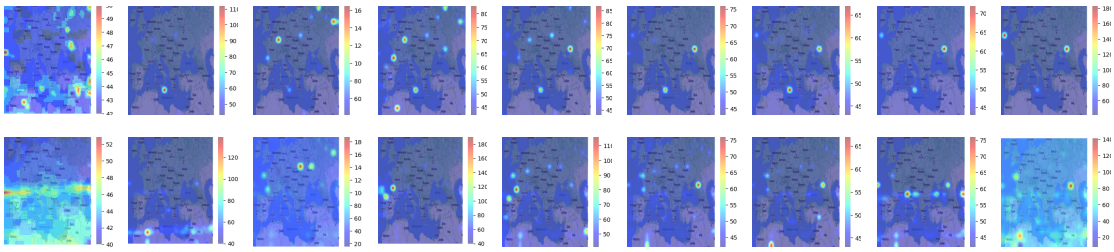
Focusing on the overall and various hierarchical capabilities of models, we primarily adopted three benchmarks—MMBench (Liu et al., 2024e), MME (Yin et al., 2023), and MMStar (Chen et al., 2024a). Additionally, SeedBench-2-Plus (Li et al., 2024b) and AI2D (Kembhavi et al., 2016) were utilized to assess the models' capability in processing rich text images such as charts, maps, and web pages. RealWorldQA was employed to evaluate the models' effectiveness in handling real-world images, whereas POPE (Li et al., 2023b) was used to examine the phenomenon of model hallucinations. To evaluate the model's performance on QA tasks, we will utilize the VQAv2 (Goyal et al., 2017) and ScienceQA (Lu et al., 2022) datasets. We utilized LMMS-Eval (Zhang et al., 2024b) for the evaluation of our model.

5.3 Results and Analysis

The primary experimental results are shown in Table 1. As can be observed from the table, the adoption of ID-Align has led to improvements in the model's performance metrics across various benchmarks. When using Vicuna and CLIP as pre-training models, there was a notable improvement across all benchmarks, with the exception of the perception subcategory in MME. However, the overall score for MME still showed an increase. These benchmarks cover a broad spectrum of capabilities, indicating the effectiveness of our approach. When employing Qwen2.5, which has a RoPE θ value of 10^7, and SigLIP as the base models, the performance gains were observed to decrease, and there was a decline in performance on several benchmarks. This observation aligns with our analysis, which suggests that these models are relatively insensitive to changes in positional encoding. However, after adopting ID-Align, the



(a) Query from the bottom right corner.



(b) Query from the central of the image.

Figure 2: The layer-wise attention visualization from the first to the thirty-second layer, with a stride of four, under conditions with and without ID-Align. Each subplot's first row represents the scenario without ID-Align, whereas the second row shows the results with ID-Align applied. Specifically, 2a refers to a query from an embedding located at the bottom right corner of the high-resolution image, while 2b corresponds to a query from the central embedding of the high-resolution image. The original image is sourced from the mmbench-test and depicts a map of Europe.

overall performance of the model showed an increasing trend.

In the appendix B, we also plot the learning curve. From these curves, it can be observed that after applying ID-Align, the training loss is slightly lower during the latter half of the training phase compared to when not using ID-Align. Additionally, the gradient norm is notably lower, indicating that the model is closer to achieving convergence. This effect is especially pronounced on Vinuca.

To further investigate which specific capabilities contributed most to the observed growth in benchmark performance, we have detailed the changes in various sub-metrics of MMbench, as shown in Table 2. We have also listed the subtasks of MMBench in Appendix C. As can be observed, when using vinca as the LLM base, although all sub-indicators showed improvement, the most significant growth was seen in the FP-S, FP-C, and RR metrics. Meanwhile, when employing qwen as the LLM backbone, it was the FP-C, RR, and LR metrics that maintained their growth. These sub-indicators are all related to fine-grained perception, with FP-C and RR also involving scenarios with multiple instances.

We also evaluated the performance of ID-Align in a training-free scenario on MMBench, where ID-Align was not employed during the training phase but was applied during inference. The results are shown in Table 2. It can be observed from the table that, in the training-free setting, the model's capability for fine-grained cross-instance perception and reasoning still improves, albeit with a smaller margin. However, its performance regarding single-instance tasks declines.

To better understand the reasons behind the changes, we also generated attention visualization images, as shown in Figure 2. The figure demonstrates that, relative to the baseline, the utilization of ID-Align increases the attention weights assigned to high-resolution image embeddings when mapped onto their corresponding low-resolution areas. This effect is especially pronounced for embeddings derived from the bottom right corner of the image. These findings are consistent with our previous analytical predictions.

Model	MMBench$_{test}$	MMStar	RealWorldQA	SEEDB2-Plus	POPE@ACC
Vicuna					
w/o ID-Align	64.46	36.65	58.69	52.04	87.49
w/ ID-Align	**67.79** (+3.33)	**38.12** (+1.47)	**59.18** (+0.49)	**53.27** (+1.23)	**87.61** (+0.12)
Qwen					
w/o ID-Align	78.14	**50.53**	**64.18**	61.00	**89.17**
w/ ID-Align	**78.48** +0.34	50.14 (-0.39)	63.79 (-0.39)	**62.06** (+1.06)	89.16 (-0.01)

	MME		AI2D	VQAV2$_{val}$	SQA$_{img}$
	cognition	perception			
Vicuna					
w/o ID-Align	248.93	**1502.72**	64.77	76.86	69.11
w/ ID-Align	**298.57** (+49.64)	1482.52 (-20.2)	**65.84** (+1.07)	**79.88** (+3.02)	**70.10** (+0.99)
Qwen					
w/o ID-Align	348.93	1530.18	74.84	79.88	80.61
w ID-Align	**349.64** (+0.71)	**1560.51** (+30.33)	**75.13** (+0.29)	**80.25** (+0.37)	**81.06** (+0.45)

Table 1: Performance on Different Benchmarks with and without ID-Align

Model	MMBench Test					
	CP	FP-S	FP-C	AR	RR	LR
Vicuna						
w/o ID-Align	76.02	66.33	57.09	76.39	52.13	34.68
w/ ID-Align	**77.73** (+1.71)	**71.11** (+4.78)	**63.16** (+6.07)	**78.13** (+1.74)	**57.35** (+5.22)	37.57 (+2.89)
Training-Free	77.09 (+1.07)	65.33 (-1.00	57.89 (+0.80)	78.12 (+1.73)	54.98 (+2.85)	**38.15** (+3.47)
Qwen						
w/o ID-Align	83.73	81.91	71.26	84.38	75.83	56.65
w/ ID-Align	82.87 (-0.86)	81.91 (+0.00)	**72.87** (+1.61)	83.33 (-1.05)	**77.72** (+1.89)	**59.54** (+2.89)
Training-Free	**83.94** (+0.21)	81.66 (-0.25)	72.47 (+1.21)	**85.07** (+0.69)	75.83 +0.00	58.38 (+1.73)

Table 2: The table presents the results on sub-metrics from the MMBench-Test. Specifically, **CP** stands for Coarse Perception, **FP-C** represents Fine-grained Perception (cross-instance), **FP-S** denotes Fine-grained Perception (single-instance), **AR** refers to Attribute Reasoning, **LR** indicates Logical Reasoning, **RR** represents Relation Reasoning.

6 Conclusion

In this paper, we analyze the potential issues of the dynamic high-resolution strategies adopted by current VLMs. Based on our analysis, we propose **ID-Align**: a method that aligns the position IDs of high-resolution embeddings with their corresponding low-resolution embeddings, preserving their relationship and constraining excessive growth in position IDs. We conducted experiments on the LLaVA-Next architecture, demonstrating the effectiveness of our approach—even when employing models with very large RoPE θ values, which are not sensitive to changes in position IDs.

7 Limitation

Our study is subject to two primary limitations. Firstly, we have not explored the performance of our approach at larger resolutions involving a higher number of image tokens. Given that certain contemporary models are designed to process ultra-high-resolution images resulting in tens of thousands of image tokens, these extended sequence lengths present significant challenges. Secondly, our experimental design focused exclusively on single-image datasets, neglecting both multi-image scenarios and the opportunity to integrate with prevalent token reduction strategies.

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*.

Tianyi Bai, Hao Liang, Binwang Wan, Yanran Xu, Xi Li, Shiyu Li, Ling Yang, Bozhou Li, Yifan Wang, Bin Cui, et al. 2024. A survey of multimodal large language model from a data-centric perspective. *arXiv preprint arXiv:2405.16640*.

Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, et al. 2024. Internlm2 technical report. *arXiv preprint arXiv:2403.17297*.

Junbum Cha, Wooyoung Kang, Jonghwan Mun, and Byungseok Roh. 2024. Honeybee: Locality-enhanced projector for multimodal llm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13817–13827.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. 2024a. Are we on the right way for evaluating large vision-language models? *arXiv preprint arXiv:2403.20330*.

Lin Chen and Long Xing. 2024. Open-llava-next: An open-source implementation of llava-next series for facilitating the large multi-modal model community. https://github.com/xiaoachen98/Open-LLaVA-NeXT.

Zhanpeng Chen, Mingxiao Li, Ziyang Chen, Nan Du, Xiaolong Li, and Yuexian Zou. 2025. Advancing general multimodal capability of vision-language models with pyramid-descent visual position encoding. *arXiv preprint arXiv:2501.10967*.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. 2024b. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*.

Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. 2024c. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *Science China Information Sciences*, 67(12):220101.

Wenliang Dai, Nayeon Lee, Boxin Wang, Zhuolin Yang, Zihan Liu, Jon Barker, Tuomas Rintamaki, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2024. Nvlm: Open frontier-class multimodal llms. *arXiv preprint arXiv:2409.11402*.

Zihang Dai. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*.

Mostafa Dehghani, Basil Mustafa, Josip Djolonga, Jonathan Heek, Matthias Minderer, Mathilde Caron, Andreas Steiner, Joan Puigcerver, Robert Geirhos, Ibrahim M Alabdulmohsin, et al. 2024. Patch n'pack: Navit, a vision transformer for any aspect ratio and resolution. *Advances in Neural Information Processing Systems*, 36.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. 2024. Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models. *arXiv preprint arXiv:2409.17146*.

Alexey Dosovitskiy. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Junqi Ge, Ziyi Chen, Jintao Lin, Jinguo Zhu, Xihui Liu, Jifeng Dai, and Xizhou Zhu. 2024. V2pe: Improving multimodal long-context capability of vision-language models with variable visual position encoding. *ArXiv*, abs/2412.09616.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. In *International conference on machine learning*, pages 1243–1252. PMLR.

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

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. 2016. A diagram is worth a dozen images. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 235–251. Springer.

Bo Li, Hao Zhang, Kaichen Zhang, Dong Guo, Yuanhan Zhang, Renrui Zhang, Feng Li, Ziwei Liu, and Chunyuan Li. 2024a. Llava-next: What else influences visual instruction tuning beyond data?

Bohao Li, Yuying Ge, Yi Chen, Yixiao Ge, Ruimao Zhang, and Ying Shan. 2024b. Seed-bench-2-plus: Benchmarking multimodal large language models with text-rich visual comprehension. *arXiv preprint arXiv:2404.16790*.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023a. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR.

Yanwei Li, Yuechen Zhang, Chengyao Wang, Zhisheng Zhong, Yixin Chen, Ruihang Chu, Shaoteng Liu, and Jiaya Jia. 2024c. Mini-gemini: Mining the potential of multi-modality vision language models. *arXiv preprint arXiv:2403.18814*.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. 2023b. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 292–305.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. 2024a. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

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

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

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

Xuanqing Liu, Hsiang-Fu Yu, Inderjit Dhillon, and Cho-Jui Hsieh. 2020. Learning to encode position for transformer with continuous dynamical model. In *International conference on machine learning*, pages 6327–6335. PMLR.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. 2024e. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pages 216–233. Springer.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521.

Xin Men, Mingyu Xu, Bingning Wang, Qingyu Zhang, Hongyu Lin, Xianpei Han, and Weipeng Chen. 2024. Base of rope bounds context length. *arXiv preprint arXiv:2405.14591*.

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.

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

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*.

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

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

A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.

Benyou Wang, Donghao Zhao, Christina Lioma, Qiuchi Li, Peng Zhang, and Jakob Grue Simonsen. 2019. Encoding word order in complex embeddings. *arXiv preprint arXiv:1912.12333*.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang Ma, Chengyue Wu, Bingxuan Wang, et al. 2024. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding. *arXiv preprint arXiv:2412.10302*.

Yun Xing, Yiheng Li, Ivan Laptev, and Shijian Lu. 2025. Mitigating object hallucination via concentric causal attention. *Advances in Neural Information Processing Systems*, 37:92012–92035.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. 2023. A survey on multimodal large language models. *arXiv preprint arXiv:2306.13549*.

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.

Duzhen Zhang, Yahan Yu, Jiahua Dong, Chenxing Li, Dan Su, Chenhui Chu, and Dong Yu. 2024a. Mm-llms: Recent advances in multimodal large language models. *arXiv preprint arXiv:2401.13601*.

Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, and Ziwei Liu. 2024b. Lmms-eval: Reality check on the evaluation of large multimodal models. *Preprint*, arXiv:2407.12772.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.

A Training Scripts

The decision to utilize ID-Align can be controlled by setting the value of use-id-align.

In our experiment, we selected the most suitable high resolution based on the aspect ratio of the images. However, during upsampling, we did not maintain the aspect ratios nor did we perform any padding.

Listing 1: The script for the LLaVA-Next pretrain phase, using Vicuna and CLIP as the LLM backbone and visual encoder, respectively.

```
nnodes=1
num_gpus=8
deepspeed --num_nodes ${nnodes} --
    num_gpus ${num_gpus} --master_port
    =10270 llava/train/train_mem.py \
    --deepspeed ./scripts/zero2.json \
    --model_name_or_path ${MODEL_PATH} \
    --version plain \
    --data_path ${DATA_PATH} \
    --image_folder ${IMAGE_FOLDER} \
    --vision_tower ${VISION_TOWER} \
    --mm_projector_type mlp2x_gelu \
    --tune_mm_mlp_adapter True \
    --unfreeze_mm_vision_tower False \
    --mm_vision_select_layer -2 \
    --mm_use_im_start_end False \
    --mm_use_im_patch_token False \
    --mm_patch_merge_type spatial_unpad
        \
    --image_aspect_ratio anyres \
    --group_by_modality_length False \
    --bf16 True \
    --output_dir ./checkpoints/${
        RUN_NAME} \
    --num_train_epochs 1 \
    --per_device_train_batch_size 8 \
    --per_device_eval_batch_size 4 \
    --gradient_accumulation_steps 4 \
    --evaluation_strategy "no" \
    --image_grid_pinpoints "[(336, 672),
        (672, 336), (672, 672), (1008,
        336), (336, 1008)]" \
    --use_id_align True \
    --save_strategy "steps" \
    --save_steps 24000 \
    --save_total_limit 1 \
    --learning_rate 1e-3 \
    --weight_decay 0. \
    --warmup_ratio 0.03 \
    --lr_scheduler_type "cosine" \
    --logging_steps 1 \
    --tf32 True \
    --model_max_length 4096 \
    --gradient_checkpointing True \
    --dataloader_num_workers 4 \
    --lazy_preprocess True \
    --report_to None \
    --run_name ${RUN_NAME}
```

Listing 2: The script for the LLaVA-Next finetune phase, using Vicuna and CLIP as the LLM backbone and visual encoder, respectively.

```
nnodes=1
num_gpus=8

deepspeed --num_nodes ${nnodes} --
    num_gpus ${num_gpus} --master_port
    =10271 llava/train/train_mem.py \
    --deepspeed ./scripts/zero3.json \
    --model_name_or_path ${MODEL_PATH} \
    --version v1 \
    --data_path ${DATA_PATH} \
    --image_folder ${IMAGE_FOLDER} \
    --pretrain_mm_mlp_adapter ./
        checkpoints/${BASE_RUN_NAME}/
        mm_projector.bin \
    --unfreeze_mm_vision_tower True \
    --mm_vision_tower_lr 2e-6 \
    --vision_tower ${VISION_TOWER} \
    --mm_projector_type mlp2x_gelu \
    --mm_vision_select_layer -2 \
    --mm_use_im_start_end False \
    --use_id_align True \
    --mm_use_im_patch_token False \
    --group_by_modality_length True \
    --image_aspect_ratio anyres \
    --mm_patch_merge_type spatial_unpad
        \
    --bf16 True \
    --image_grid_pinpoints "[(336, 672),
        (672, 336), (672, 672), (1008,
        336), (336, 1008)]" \
    --output_dir ./checkpoints/${
        RUN_NAME} \
    --num_train_epochs 1 \
    --per_device_train_batch_size 8 \
    --per_device_eval_batch_size 4 \
    --gradient_accumulation_steps 2 \
    --evaluation_strategy "no" \
    --save_strategy "steps" \
    --save_steps 1000 \
    --save_total_limit 1 \
    --learning_rate 2e-5 \
    --weight_decay 0. \
    --warmup_ratio 0.03 \
    --lr_scheduler_type "cosine" \
    --logging_steps 1 \
    --tf32 True \
    --model_max_length 4096 \
    --gradient_checkpointing True \
    --dataloader_num_workers 4 \
```

```
    --lazy_preprocess True \
    --report_to none \
    --run_name ${RUN_NAME}
```

Listing 3: The script for the LLaVA-Next pre-train phase, using Qwen and SigLIP as the LLM backbone and visual encoder, respectively.

```
nnodes=1
num_gpus=8
deepspeed --num_nodes ${nnodes} --
    num_gpus ${num_gpus} --master_port
    =10270 llava/train/train_mem.py \
    --deepspeed ./scripts/zero2.json \
    --model_name_or_path ${MODEL_PATH} \
    --version plain \
    --data_path ${DATA_PATH} \
    --image_folder ${IMAGE_FOLDER} \
    --vision_tower ${VISION_TOWER} \
    --mm_projector_type mlp2x_gelu \
    --tune_mm_mlp_adapter True \
    --unfreeze_mm_vision_tower False \
    --mm_vision_select_layer -2 \
    --mm_use_im_start_end False \
    --mm_use_im_patch_token False \
    --mm_patch_merge_type spatial_unpad
        \
    --image_aspect_ratio anyres \
    --group_by_modality_length False \
    --bf16 True \
    --output_dir ./checkpoints/${
        RUN_NAME} \
    --num_train_epochs 1 \
    --per_device_train_batch_size 8 \
    --per_device_eval_batch_size 4 \
    --gradient_accumulation_steps 4 \
    --evaluation_strategy "no" \
    --image_grid_pinpoints "[(384, 768),
         (768, 384), (768, 768), (1152,
        384), (384, 1152)]" \
    --use_id_align True \
    --save_strategy "steps" \
    --save_steps 24000 \
    --save_total_limit 1 \
    --learning_rate 1e-3 \
    --weight_decay 0. \
    --warmup_ratio 0.03 \
    --lr_scheduler_type "cosine" \
    --logging_steps 1 \
    --tf32 True \
    --model_max_length 32768 \
    --gradient_checkpointing True \
    --dataloader_num_workers 4 \
    --lazy_preprocess True \
    --report_to none \
    --run_name ${RUN_NAME}
```

Listing 4: The script for the LLaVA-Next finetune phase, using Qwen and SigLIP as the LLM backbone and visual encoder, respectively

```
nnodes=1
num_gpus=8
deepspeed --num_nodes ${nnodes} --
    num_gpus ${num_gpus} --master_port
    =10271 llava/train/train_mem.py \
    --deepspeed ./scripts/zero3.json \
    --model_name_or_path ${MODEL_PATH} \
    --version ${PROMPT_VERSION} \
    --data_path ${DATA_PATH} \
    --image_folder ${IMAGE_FOLDER} \
    --pretrain_mm_mlp_adapter ./
        checkpoints/${BASE_RUN_NAME}/
        mm_projector.bin \
    --unfreeze_mm_vision_tower True \
    --mm_vision_tower_lr 2e-6 \
    --vision_tower ${VISION_TOWER} \
    --mm_projector_type mlp2x_gelu \
    --mm_vision_select_layer -2 \
    --mm_use_im_start_end False \
    --use_id_align True \
    --mm_use_im_patch_token False \
    --group_by_modality_length True \
    --image_aspect_ratio anyres \
    --mm_patch_merge_type spatial_unpad
        \
    --bf16 True \
    --image_grid_pinpoints "[(384, 768),
         (768, 384), (768, 768), (1152,
        384), (384, 1152)]" \
    --output_dir ./checkpoints/${
        RUN_NAME} \
    --num_train_epochs 1 \
    --per_device_train_batch_size 8 \
    --per_device_eval_batch_size 4 \
    --gradient_accumulation_steps 2 \
    --evaluation_strategy "no" \
    --save_strategy "steps" \
    --save_steps 1000 \
    --save_total_limit 1 \
    --learning_rate 2e-5 \
    --weight_decay 0. \
    --warmup_ratio 0.03 \
    --lr_scheduler_type "cosine" \
    --logging_steps 1 \
    --tf32 True \
    --model_max_length 32768 \
    --gradient_checkpointing True \
    --dataloader_num_workers 4 \
    --lazy_preprocess True \
    --report_to none \
    --run_name ${RUN_NAME}
```

B Learning Curve

These plots were generated using a sliding average window with a window length of 100.

B.1 Vicuna

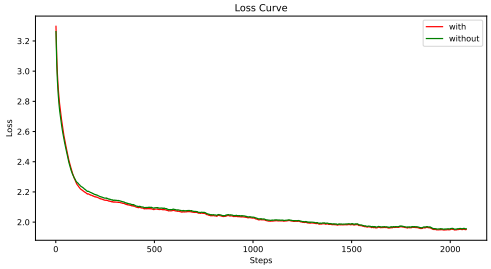


Figure 3: Pretrain Loss

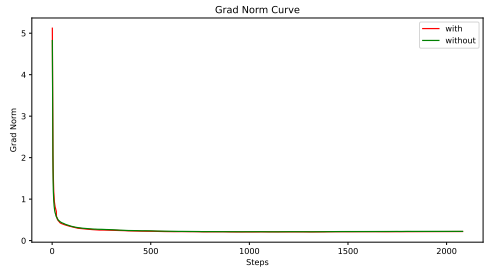


Figure 4: Pretrain Grad Norm

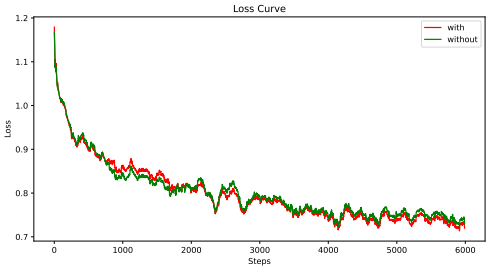


Figure 5: Finetune Loss

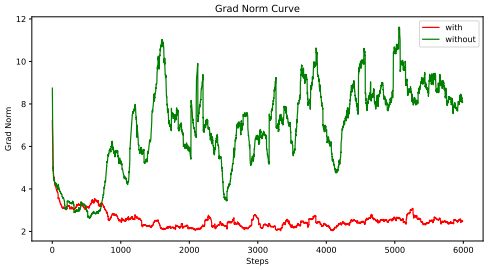


Figure 6: Finetune Grad Norm

B.2 Qwen

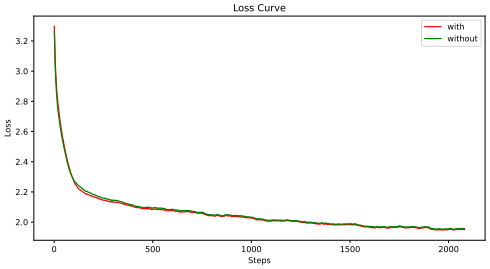


Figure 7: Pretrain Loss

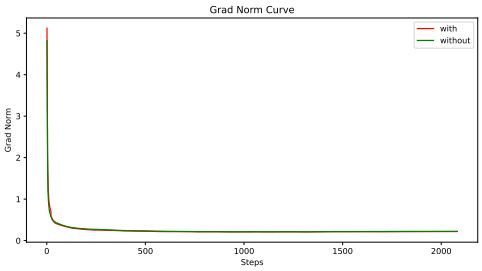


Figure 8: Pretrain Grad Norm

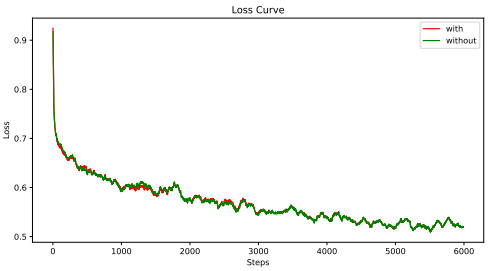


Figure 9: Finetune Loss

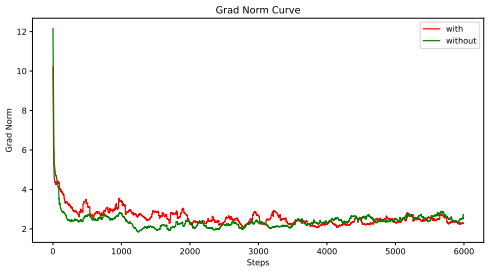


Figure 10: Finetune Grad Norm

C MMBench Leaf Tasks

Coarse Perception:

- Image Style
- Image Topic
- Image Scene
- Image Mood
- Image Quality

Fine-grained Perception (Single-instance):

- Attribute Recognition
- Celebrity Recognition
- Object Localization

- OCR

Fine-grained Perception (Cross-instance):

- Spatial Relationship
- Attribute Comparison
- Action Recognition

Attribute Reasoning:

- Physical Property Reasoning
- Function Reasoning
- Identity Reasoning

Relation Reasoning:

- Social Relation
- Nature Relation
- Physical Relation

Logic Reasoning:

- Future Prediction
- Structuralized Image-text Understanding