Accelerating Multimodal Large Language Models via Dynamic Visual-Token Exit and Empirical Findings

Qiong Wu[12], **Wenhao Lin**[12], **Yiyi Zhou**[12]*, **Weihao Ye**[1], **Zhanpeng Zeng**[1], **Xiaoshuai Sun**[12], **Rongrong Ji**[12]

[1] Key Laboratory of Multimedia Trusted Perception and Efficient Computing, Ministry of Education of China, Xiamen University, 361005, P.R. China.

[2] Institute of Artificial Intelligence, Xiamen University, 361005, P.R. China.

{qiong, wenhaolin}@stu.xmu.edu.cn, zhouyiyi@xmu.edu.cn, weihaoye@stu.xmu.edu.cn, {zzeng, xssun, rrji}@xmu.edu.cn

Abstract

In this paper, we study the visual redundancy problem of *multimodal large language models* (MLLMs) from the perspective of attention behaviors. Via extensive empirical experiments, we observe and conclude three main inference stages of MLLMs: *(i) Early fusion* between tokens is first accomplished quickly. *(ii) Intra-modality modeling* then comes to play. *(iii) Multimodal reasoning* resumes and lasts until the end of inference. In particular, we reveal that visual tokens will stop contributing to reasoning when the text tokens receive enough image information. Based on this observation, we propose an effective method to improve the efficiency of MLLMs, termed *dynamic visual-token exit* (DyVTE), which is orthogonal but collaborative to previous token-wise visual compression methods. To validate the efficacy of DyVTE, we apply it to a set of MLLMs, including LLaVA, VILA, EAGLE and InternVL. The experimental results not only show the effectiveness of our DyVTE in improving MLLMs' efficiency, *e.g.*, DyVTE reduces the computation overhead of LLaVA-1.5 by up to 45.7% without performance drop, but also reveal a general pattern across multiple MLLMs, well facilitating the in-depth analysis of MLLMs. Our code is released at `https://github.com/DoubtedSteam/DyVTE`.

1 Introduction

Recently, the rapid development of vision-language learning has been witnessed with the great success of *large language models* (LLMs) [2, 6, 16, 46, 52, 53, 54]. Numerous efforts are devoted to equip LLMs with multimodal capability, *i.e.*, *multimodal large language models* (MLLMs) [11, 14, 29, 31, 36, 40, 56]. To overcome visual shortcoming [34, 49] in extreme high computation overhead, recent MLLMs often resort to higher-resolution images as input, which is often accompanied with a multitude of visual tokens [11, 15, 29, 43].

However, the multitude of visual tokens lead to prohibitively expensive computation. For instance, compared with LLaVA 1.5 [40], LLaVA-NeXT [29] adopts about 1728 more visual tokens, resulting in about 4 times more FLOPs. Despite the performance gains, recent works[62, 7, 37] also show that these large amount of tokens are obviously redundant, leading to a great waste of computation. For instance, Ye *et al.* [62] show that randomly dropping half visual tokens barely affects performance on common VL tasks, such as MMB [41] and SQA [42]. In this case, the efficient use of visual tokens has recently become a research hot-spot, and attracts an influx of attention [8, 13, 55, 57]. However,

*Corresponding Author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

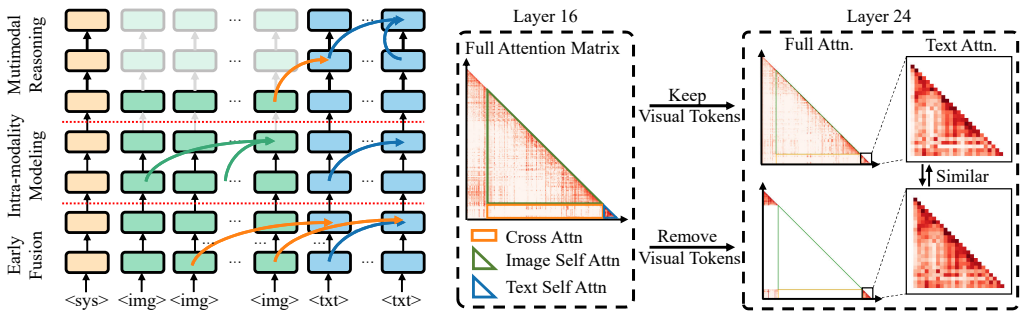

Figure 1: **Left**: Illustration of three main stages observed in MLLMs. During inference, an MLLM will go through three main stages, *i.e.*, *early fusion* between all tokens, *intra modality modeling* of the same-modal tokens, and *multimodal reasoning* between all tokens again. **Right**: The impact of visual tokens on the text self attention at the multimodal reasoning stage. Removing visual tokens at an appropriate time barely changes the text attention patterns, indicating that visual tokens contribute little to multi-modal reasoning.

existing research mainly focuses on evaluating token-wise redundancy, *e.g.*, token pruning [7, 62] and token merging [32, 47], lacking in-depth exploration of the intrinsic behaviors of MLLMs.

In this paper, we are dedicated to understand how MLLMs use visual tokens and how they behave during multimodal reasoning. To approach this target, we conduct extensive empirical studies about the attention behaviors of MLLMs, and observe and summarize three main stages of their inference, as illustrated in Fig.1-Left. At the first stage, an MLLM quickly accomplishes the exchange of multimodal information at its shallow layers, which we term *(i)* ***early fusion***. Afterwards, this information exchange decreases, and the tokens primarily engage in intra-modal interactions, termed *(ii)* ***intra-modality modeling***. At the last stage, visual tokens will resume the information propagation to the text ones, and this process will continue until a certain layer, and we term it *(iii)* ***multimodal reasoning***. Through extensive comparisons, we reveal that these observed behaviors of MLLMs are common across different models [12, 36, 40, 48] and tasks [22, 41, 63].

Behind these shared behaviors of MLLMs, we observe a key finding in terms of visual redundancy, that is *visual tokens will continue to propagate their semantics to the text ones at the multimodal reasoning stage, while their impact can be very limited.* As shown in the right sub-figure of Fig.1, removing visual tokens at a suitable layer of MLLMs does not produce significant changes to the text self-attention distributions. This case suggests that visual tokens actually contribute little to multimodal reasoning at the last inference stage. In other words, multimodal reasoning only happens within text tokens after they receive enough visual semantics. This finding is also confirmed in some very recent works [37], where the manual removal of visual tokens does not decline the performance of MLLMs on some tasks. But in this paper, we also found that the optimal time to remove visual tokens is distinct for different examples, tasks and even MLLMs. And the manual exploration is experimentally expensive and hard to meet the global optimum. Therefore, *when and how to automatically remove visual tokens* is still a thorny challenge.

Motivated by these observations, we propose a novel and effective method to improve MLLMs' efficiency, termed ***dy**namic **v**isual-**t**oken **e**xit* (DyVTE). In particular, we apply lightweight hyper-networks to perceive text token status and then dynamically decide the exit of all visual tokens, thereby speeding up inference. As discussed above, DyVTE mainly focuses on the overall contribution of all visual tokens during multimodal inference, which is orthogonal but collaborative to token-wise solutions like *token pruning* [7, 27, 62]. In our experiments, we show that these two paradigms can be easily combined to boost the efficiency of MLLMs. In addition, DyVTE also differs from previous *dynamic early exiting* methods [19, 21, 51], which often refer to an inference break *i.e.*, skipping rest layers for direct predictions. In contrast, text tokens continue to transform in DyVTE, which better meets our empirical findings. Moreover, we also introduce an efficient training regime for DyVTE independent to the global gradient computation of MLLMs, achieving effective and dynamic visual-token exit with very cheap training expenditure.

To validate DyVTE, we apply it to a set of advanced MLLMs with varying scales, including LLaVA-1.5 [40], VILA [36], EAGLE [48] and InternVL [12], and conduct experiments on a bunch of widely-used VL and MLLM benchmarks [20, 22, 23, 34, 49, 63]. The experimental results show

that our DyVTE can greatly reduce the computation overhead of MLLMs, while retaining their competitive performance on various benchmarks. For instance, our DyVTE reduces the computation overhead of LLaVA-1.5 by up to 45.7% without performance drop. When combined with token pruning methods like FastV [7], DyVTE can help LLaVA-1.5 reduce up to 51.5% computation with only 0.7% performance drop on average.

Overall, our contributions are three-fold:

- We study the problem of visual redundancy from the perspective of MLLMs' behaviors, and reveal the dependency between text and visual tokens.
- Based on the empirical findings, we propose a novel and effective approach to reduce visual redundancy of MLLMs, termed *dynamic visual-token exit* (DyVTE), which can dynamically evaluate and schedule the contributions of visual tokens to multimodal reasoning.
- The extensive experiments on a set of MLLMs well validate the motivation and effectiveness of DyVTE, also providing insights into the principle of MLLMs.

2 Related Work

Based on the successful *large language models* (LLMs) [1, 16, 2, 14, 25], numerous *multimodal large language models* (MLLMs) have been recently proposed [31, 40, 3, 29, 11] and achieved remarkable progresses on various vision-language tasks [22, 20, 23, 60, 33]. In terms of methodology, most MLLMs often project the extracted visual features onto the semantic space of LLM for both multimodal fusion and reasoning [40, 56, 48, 12]. For instance, LLaVA [40] uses a projection layer to transform visual features into input tokens for LLaMA [54]. BLIP-2 [31] introduces *QFormer* to aggregate query-related visual features as the input visual tokens. Similarly, QWen-VL also adopts learnable tokens as queries to obtain a limited number of visual tokens [3]. To improve the ability in fine-grained tasks [49, 44, 45], some recent MLLMs resort to increasing the resolution of input images [29, 56, 10] for better visual understanding. For instance, LLaVA-NeXT [39] concatenates the tokens of each crop of the image as the input of the LLM. InternVL2.5 [10] and Qwen2-VL [4] introduce dynamic resolution designs to match different images and preserve their visual details. Despite success, these high-resolution settings also inevitably increase the number of visual tokens, leading to excessive computation overhead [7, 62].

Moreover, recent studies also show that the excessive use of visual tokens brings in obvious redundancy in both information and computation [35, 58, 59]. In this case, the efficiency study of MLLMs also becomes a research hotspot, of which focus ranges from structure design [35, 58, 59], token pruning [27, 18, 5] and token merging [47, 5]. The research of structure design mainly aim to build a new and lightweight MLLMs [35, 65], and our work is closer to the token efficiency researches [62, 7]. For instance, FastV [7] applies the averaged attention scores to evaluate the importance of each visual token and then drop the less important ones to reduce computation. FitPrune [62] resort to a statistic principle to quickly produces a token pruning regime for MLLMs based on the metrics of visual and cross attentions. PruMerge [47] speeds up the inference of MLLMs by merging visual tokens that have similar semantics. G-Prune [26] proposes an graph-based algorithm to select the most representative visual tokens to avoid redundant semantics and computations. Compared with these token-wise approaches, our DyVTE focuses more on the overall contributions of visual tokens in MLLMs, providing an alternative but orthogonal way for efficient MLLMs. In terms of dynamic inference, our works are also related to the dynamic early-exit studies for LLMs [17, 9]. However, these methods focus on the skipping of redundant layers for early prediction akin to previous dynamic methods [51, 21]. Recently, LLaVA-Mini [64] introduces a pre-fusion module through full pre-training and SFT, reducing visual tokens to just one. And our DyVTE is to conduct the early removal of visual tokens while keeping the original inference pattern. Moreover, this paper also involves quantitative analyses about the attention behaviors of MLLMs, providing insights into the mechanisms for their multimodal reasoning.

3 Inference Pattern Analysis of MLLMs

We first quantitatively investigate the attention behaviors of MLLMs, and then examine the impacts of visual tokens during multimodal inference.

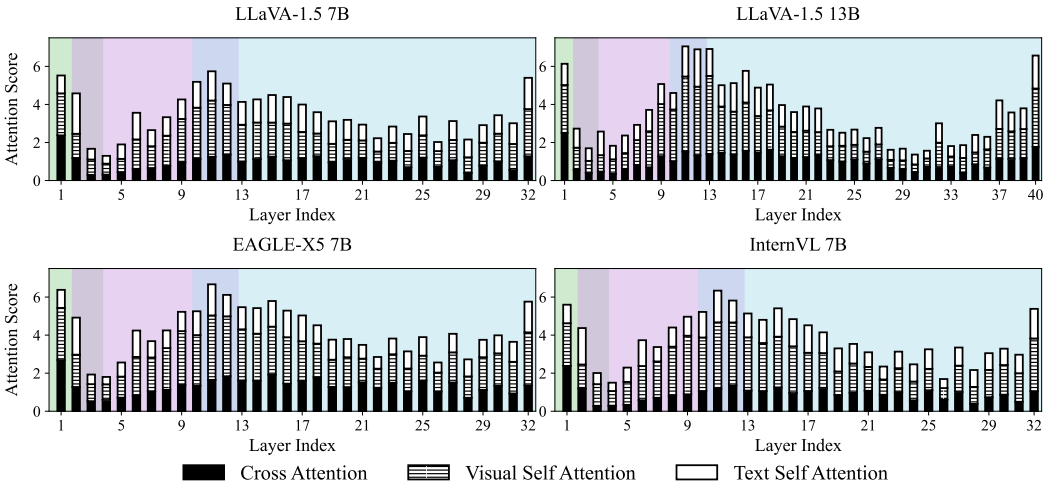

Figure 2: The averaged attention scores of four MLLMs in terms of cross, visual and text attentions. The background colors denote the three summarized stages of MLLMs, *i.e.*, *early fusion* (green), *intra-modality modeling* (purple) and multimodal reasoning (blue).

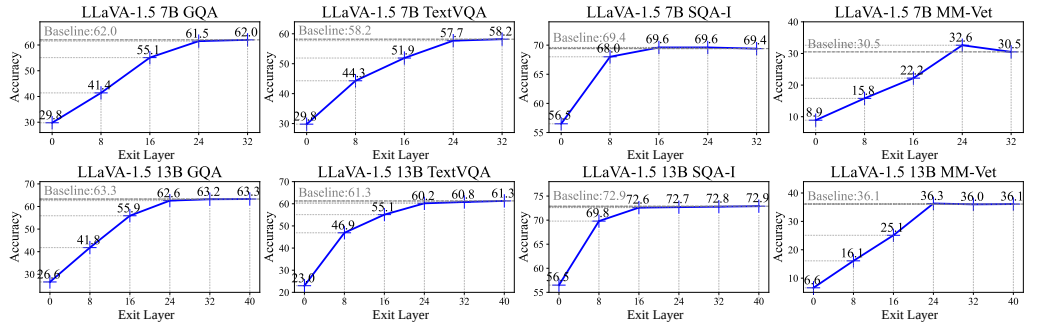

Figure 3: The relationship between manual visual-token exit and performance on LLaVA-1.5 7B and 13B. We show the results of removing all visual tokens at 0-th, 8-th, 16-th, 24-th and 32-th layers. "*Baseline*" represents the original performance of the MLLM. After a certain layer, the removal of all visual tokens barely impedes performance, but the layers for removal are different for different tasks.

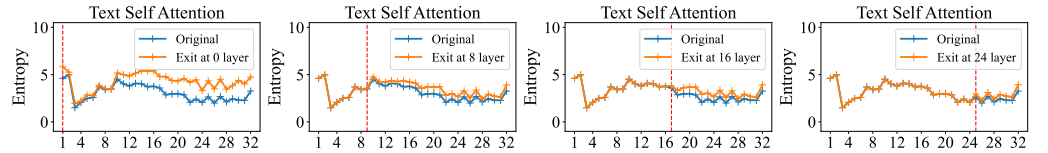

Figure 4: The entropy of text self-attention distribution with different layer to remove all visual tokens. We visualize the entropy of text self-attention on LLaVA-1.5 7B on GQA benchmark with different with different layer to remove all visual tokens, which shows that the removal of visual tokens after a certain layer barely affect the text self-attention.

Attention behaviors of MLLMs. We first visualize the attention patterns of four representative MLLMs [40, 48, 12] of two scales in Fig.2. These attention data includes *visual self-attention*, *visual-text cross-attention* and *text self-attention*[2]. Here, the values in these distributions are the column-wise summarized attention scores on the *LLaVA-split* [40]. From these plots, we can first observe that four MLLMs share similar patterns for three types of attentions. At the shallow layers, all MLLMs show high cross-attention values between visual and text tokens, suggesting the intensive interactions between two modalities. This corresponds to the *early fusion* stage we termed above, and it denotes the quick exchange of multimodal information. But *early fusion* declines quickly, and we can observe that the self-attention between the tokens of the same modality becomes the main activity at the second stage, *i.e.*, the modeling mainly involved in each modality. In this case, we can assume that MLLM starts the *intra-modality modeling* stage. At the last stage, the visual tokens will

[2]The visualization details are provided in our appendix A.1.

resume information propagation to the text tokens, and this process will continue to oscillate until the end of inference. Here, we term this process as the *multimodal reasoning* stage. Notably, we can find these patterns exist in MLLMs of different families and scales.

The impact of dropping all visual tokens. Next, we quantitatively measure the importance of visual tokens. In Fig.3, we present the results of removing visual tokens of LLaVA-1.5 7B and 13B on four benchmarks at different layers. From these plots, we can first observe that visual tokens are not always required throughout the entire process, especially the last stage. For instance, on GQA, removing all visual tokens at 24-th layer only has 0.5% and 0.6% performance drops for LLaVA-1.5 7B and 13B, receptively. This finding is also consistent with some recent efforts [37] that visual tokens receive little attention in deep layers. Another finding is that the removal of visual tokens have little impact on the modeling among text tokens. As shown in Fig.4, the removal of visual tokens that does not harm the performance and also has no impact on the modeling pattern among the text tokens, *e.g.*, after 24^{th} layer on GQA. The only difference lies in the slight change of attention intensity due to the shorter token sequence. More importantly, we can find that *the exit time of visual tokens is different for different examples and tasks.* For instance, the optimal exit layer of SQA is earlier than that of TextVQA, *i.e.*, 16 *v.s.* 25, suggesting the need of dynamic early-exist modeling. Overall, these results confirm our argument that visual tokens stop contributing to prediction after certain layer, and also shows the need of exploring dynamic and automatic strategies for effective visual token removal.

Discussion. Based on the above observations and analyses, we can categorize the inference of MLLMs into three main stages. (i) *Early Fusion*. In the initial stage, the MLLM quickly accomplish the exchange of multimodal information from visual to text tokens. (ii) *Intra-modality Modeling*. Next, the self-attention intra the modality becomes the main activity, enhancing vision and language understanding. (iii) *Multimodal Reasoning*. Finally, the visual tokens will resume the information propagation to the text tokens, though this transfer demonstrates limited impact on final response generation. In addition to these three stages, we can also conclude one common behavior of visual tokens in MLLMs, *i.e.*, the contribution of visual tokens to multimodal prediction diminish after some layers. Moreover, these results also show that the exist times for different types of VL examples are different, well motivating our exploration of dynamic visual-token exit.

4 Dynamic Visual-token Exiting

4.1 Method

In this paper, we propose a novel and effective method to reduce the visual redundancy of MLLMs, termed ***dy**namic **v**isual-**t**oken **e**xit* (DyVTE). DyVTE uses lightweight hyper-networks to perceive the text token status of MLLMs and then adaptively judge the right time to remove all visual tokens.

Concretely, given the text tokens $\mathbf{T}^{(k)}$ at the k-th layer of an MLLM, we use a simple MLP as the hyper-network to learn their status and decide whether to remove all visual tokens, defined by

$$\mathbf{p} = Softmax\big(\text{GELU}([avg(\mathbf{T}^{(k)}_{1:t-1}), \mathbf{T}^{(k)}_{t}]\mathbf{W}_1)\mathbf{W}_2\big), \tag{1}$$

where $\mathbf{p} \in \mathcal{R}^2$ is the binary prediction, $\mathbf{W}_1 \in \mathcal{R}^{2d\times h}$ and $\mathbf{W}_2 \in \mathcal{R}^{h\times 2}$ are weight matrices, and h is the dimension of the hidden states.

In particular, the hyper-network of Eq.1 uses two text token representations, *i.e.*, $avg(\mathbf{T}_{1:t-1})$ and $\mathbf{T}_t$. The former is the averaged text tokens representing their overall state. The latter $\mathbf{T}_t$ is the last token, which often plays an important role for decoding under the *uni-directional self-attention* setting of MLLMs [38, 50]. With these two types of text representations, DyVTE can well perceive the state of MLLMs and automatically decide whether additional visual information is still needed for reasoning.

When the prediction $\mathbf{p}$ is *exit*, DyVTE will remove all visual tokens after this layer, while the text ones are kept in the rest layers of MLLMs. This process can be denoted by

$$P'_l = G_{l+1:L}(\mathbf{T}^{(l)}), \tag{2}$$

where $G_{l+1:L}(\cdot)$ denote the remaining layers in the MLLM. Compared with previous dynamic exit methods that skip layers for early prediction [17, 9], the principle of our DyVTE is to make the removal of visual tokens based on the text token status. Thus, its implementation requires no structure tweaks, *e.g.*, adding new prediction layers, and also no the update of MLLMs.

4.2 Optimization

In particular, given sufficient examples of visual-token exit, DyVTE can learn to judge the text token status at each layer of the MLLM. Thus, its accurate removals of visual tokens will greatly speed up the inference in the rest layers of the MLLM, while retaining similar predictions.

Thus the objective of DyVTE can be defined by

$$\underset{\theta_h}{\operatorname{argmin}}\, d(P, P'_l), \tag{3}$$

where θ_h denotes the weights of hyper-networks in DyVTE. $d(\cdot, \cdot)$ is KL-Divergence [28]. And P'_l and P are the predictions of an MLLM with and without early visual token exit, respectively.

To optimize Eq.3, a direct approach is to merge the predictions of hyper-networks with visual tokens of MLLMs, *e.g.,* implementing attention masks based on *Gumbel softmax* [24], thus using the *next token prediction* loss to indirectly update DyVTE. Although feasible, this approach still requires the computation of the model's all gradients, which is still inefficient and expensive.

In this case, we propose an efficient training regime independent to the gradient back-propagation of MLLMs. Concretely, we can compare the discrete outputs of the MLLM with and without DyVTE, *e.g.*, the answer strings A. If the answers are exactly the same, we can give a positive feedback to hyper-networks, and *vice verse*. By comparing it with the default output A, we can judge that whether the visual tokens should be exited at l-th layer:

$$\mathbf{y} = \begin{cases} 1, & A'_l = A, \\ 0, & A'_l \neq A. \end{cases} \tag{4}$$

Here, A'_l denotes the answer predicted with visual-token exit at the l-th layer. Besides, to make this supervision more robust, we also consider the prediction uncertainty as a regularization term, thereby , making the MLLM behaviors closer to the default inference:

$$\mathbf{y} = \begin{cases} 1, & A'_l = A \wedge \rho'_c < \tau, \\ 0, & otherwise. \end{cases} \tag{5}$$

Here, ρ'_c denotes the prediction uncertainty valued by *cross-entropy* and multiplied by a scaling factor, and τ denotes the threshold.

With this supervision, the hyper-networks in DyVTE can be optimized by the cross-entropy loss:

$$\mathcal{L}_D = -\big(\mathbf{y} \cdot \log(\mathbf{p}_1) + (1 - \mathbf{y}) \cdot \log(\mathbf{p}_0)\big). \tag{6}$$

Compared with the indirect optimization using *next token prediction* [61], Eq.5 can provide more effective supervisions to DyVTE. We randomly sample exit layers at each training step, and compute the labels $\mathbf{y}$ according to Eq.5. Finally, the hyper-networks are optimized based loss defined in Eq.6. The overall expenditure of DyVTE training will be much cheaper than tuning MLLMs or learning new prediction layers in previous dynamic exit approaches [9, 61].

5 Experiment

5.1 Datasets and Metrics

The benchmarks used in this paper consist of four conventional vision-language (VL) benchmarks and five newly introduced MLLM benchmarks. The traditional VL benchmarks include VQAv2 [22], GQA [23], ScienceQA [42], and TextVQA [49]. The MLLM benchmarks comprise POPE [34], MME [20], MMB [41], SEED [30] and MM-Vet [63]. Different from general VL evaluation, the MLLM benchmarks focus more on the evaluations of MLLMs, such as *fine-grained reasoning* [63] and *visual hallucination* [34].

5.2 Implementation Details

We apply DyVTE to five popular MLLMs, namely EAGLE-X5 7B [48], VILA 7B [36], InternVL 7B [12], LLaVA-1.5 7B [40] and LLaVA-1.5 13B [40]. The scale for ρ' in Eq.5 is set to 1.03. For all MLLMs, the hidden dimension of the hyper-network is set to 2,048. During training, all MLLMs are frozen, and the training rate of hyper-networks is set to 4×10^{-5}. And we randomly sample 1% examples of the *LLaVA-665k* instruction set [40] to train DyVTE for 1 epoch. More details can refer to our code project.

Table 1: Results of MLLMs with and without DyVTE on five MLLM benchmarks. The accuracy (higher is better) and TFLOPs (lower is better) are reported. The relative percentage change from the baseline model to DyVTE is also shown in parentheses.

Method	SEED		MME		MMB		POPE		MM-Vet	
	Accuracy ↑	TFLOPs ↓	Score ↑	TFLOPs ↓	Accuracy ↑	TFLOPs ↓	Accuracy ↑	TFLOPs ↓	Accuracy ↑	TFLOPs ↓
EAGLE-X5 7B	73.9	47.8	1528.0	27.8	68.4	29.6	88.8	27.7	37.4	27.6
EAGLE-DyVTE 7B	73.6 (-0.4%)	43.0 (-10.0%)	1581.7 (+3.5%)	20.3 (-27.0%)	68.8(+0.6%)	23.7 (-19.9%)	88.4 (-0.5%)	20.0 (-27.8%)	37.8 (+1.1%)	23.5 (-14.9%)
VILA 7B	61.7	9.2	1489.2	8.9	69.9	9.5	86.3	8.8	36.3	8.7
VILA-DyVTE 7B	61.8 (+0.2%)	5.9 (-35.9%)	1503.1 (+0.1%)	4.6 (-48.3%)	69.8 (-0.1%)	6.0 (-36.8%)	85.6 (-0.8%)	4.5 (-48.9%)	36.7 (+1.1%)	6.6 (-24.1%)
InternVL 7B	59.2	16.0	1525.1	15.5	64.6	16.2	86.4	15.4	31.2	15.4
InternVL-DyVTE 7B	59.1 (-0.2%)	11.9 (-25.6%)	1474.1 (-3.3%)	10.9 (-29.7%)	64.4 (-0.3%)	12.0 (-25.9%)	81.3 (-5.9%)	10.9 (-29.2%)	29.5 (-5.4%)	13.0 (-15.6%)
LLaVA-1.5 7B	58.6	9.2	1510.7	8.9	64.3	9.6	85.9	8.8	30.5	8.7
LLaVA-DyVTE 7B	58.6 (0.0%)	5.0 (-45.7%)	1491.4 (-1.3%)	4.3 (-51.7%)	64.7 (+0.6%)	5.4 (-43.8%)	81.6 (-5.0%)	4.1 (-53.4%)	31.9 (+4.6%)	6.3 (-27.6%)
LLaVA-1.5 13B	61.6	17.6	1531.3	16.9	67.7	18.3	85.9	16.8	36.1	16.7
LLaVA-DyVTE 13B	59.3 (-3.7%)	7.1 (-59.7%)	1546.4 (+1.0%)	7.2 (-57.4%)	66.0 (-2.5%)	7.8 (-57.4%)	84.8 (-1.3%)	7.6 (-54.8%)	34.8 (-3.6%)	10.6 (-36.5%)

Table 2: Results of MLLMs with and without DyVTE on four VL benchmarks. The accuracy (higher is better) and TFLOPs (lower is better) are reported. The relative percentage change from the baseline model to DyVTE is also shown in parentheses.

Method	GQA		VQA		TextVQA		SQA-I		Average	
	Accuracy ↑	TFLOPs ↓	Score ↑	TFLOPs ↓	Accuracy ↑	TFLOPs ↓	Accuracy ↑	TFLOPs ↓	Accuracy ↑	TFLOPs ↓
EAGLE-X5 7B	64.9	27.8	83.4	27.8	71.2	29.5	69.8	29.2	72.3	28.6
EAGLE-DyVTE 7B	62.4 (-3.9%)	21.7 (-21.9%)	82.6 (-1.0%)	21.6 (-22.3%)	70.2 (-1.4%)	24.5 (-16.8%)	71.7 (+2.7%)	23.5 (-24.3%)	71.7 (-0.8%)	22.8 (-20.3%)
VILA 7B	63.1	8.8	80.3	8.8	62.6	9.5	69.5	9.8	68.8	9.2
VILA-DyVTE 7B	61.9 (-1.9%)	5.5 (-37.5%)	79.2 (-1.4%)	5.4 (-38.6%)	61.2 (-2.2%)	7.2 (-24.2%)	69.5 (0.0%)	6.1 (-37.8%)	67.9 (-1.3%)	6.0 (-34.8%)
InternVL 7B	62.9	15.4	79.3	15.4	57.0	16.1	66.2	16.4	66.4	15.8
InternVL-DyVTE 7B	61.3 (-2.5%)	11.8 (-23.4%)	77.6 (-2.1%)	11.7 (-24.0%)	55.8 (-2.1%)	13.5 (-16.1%)	66.2 (0.0%)	12.1 (-26.2%)	65.2 (-1.8%)	12.3 (-22.2%)
LLaVA-1.5 7B	62.0	8.8	78.5	8.8	58.2	9.5	69.4	9.8	67.0	9.2
LLaVA-DyVTE 7B	60.0 (-3.2%)	5.3 (-39.8%)	76.6 (-2.4%)	5.1 (-42.0%)	56.6 (-2.7%)	6.7 (-29.5%)	69.6 (+0.3%)	5.5 (-43.9%)	65.7 (-1.9%)	5.6 (-39.1%)
LLaVA-1.5 13B	63.3	16.8	80.0	16.8	61.3	18.1	72.9	18.6	69.4	17.6
LLaVA-DyVTE 13B	62.3 (-1.6%)	9.0 (-46.4%)	78.8 (-1.5%)	8.9 (-47.0%)	58.9 (-3.9%)	10.8 (-40.3%)	72.3 (-0.8%)	8.2 (-55.9%)	68.1 (-1.9%)	9.2 (-47.7%)

5.3 Quantitative Analysis

Results of DyVTE on different MLLMs. In Tab.1 and Tab.2, we report the results of applying DyVTE to a set of MLLMs of different families and sizes, including EAGLE [48], VILA [36], InternVL [12], and LLaVA-1.5 [40]. From these tables, we can first observe that DyVTE significantly reduces the computational overhead of existing MLLMs. For example, DyVTE reduces FLOPs of LLaVA 7B by 45.7% on SEED without dropping performance. Additionally, the actual reduction of FLOPs are distinct for different tasks. For instance, on SQA with multiple-choice questions, DyVTE reduces computational overhead by up to 43.9%, and on MM-Vet about granular answers, the reduction achieved is 27.6%. These results confirm our intuition that the optimal removal of visual tokens are different for different tasks. Another observation is that DyVTE's effects vary across these MLLMs. Specifically, DyVTE can reduce the computational overhead of VILA 7B by 48.3% on the MME benchmark with no performance drop, whereas InternVL-DyVTE 7B only achieve a reduction of 29.7%. When applying DyVTE to larger MLLMs, such as LLaVA-1.5 13B, we can observe a more significant efficiency improvement. For instance, DyVTE reduces the computational overhead of LLaVA-1.5 13B by 55.9% on SQA with only -0.8% on performance. Furthermore, DyVTE is more effective for MLLMs that uses stronger visual representations. For instance, EAGLE-X5 is a SOTA MLLM with strong and hybrid visual backbones, and it works very well with our DyVTE. DyVTE can reduce its computation by up to 6.2 TFLOPs with only 0.8% performance drop on the VL benchmarks. Overall, these results not only validate the effectiveness of our DyVTE but also confirm our motivations and findings about MLLMs.

Table 3: Comparison of real inference budget between token pruning methods and vanilla decoding. "*Acc.*" denotes the accuracy.

Method	SQA-I		MMB	
	Latency ↓	Acc. ↑	Latency ↓	Acc. ↑
LLaVA 13B	0.237s	72.9	0.236s	67.7
FastV	0.171s (-27.7%)	73.1 (+0.3%)	0.174s (-26.2%)	68.6 (+1.3%)
ToMe	0.175s (-26.2%)	73.2 (+0.4%)	0.179s (-24.2%)	67.4 (-0.4%)
DyVTE	0.161s (-32.1%)	72.3 (-0.8%)	0.163s (-30.9%)	66.0 (-2.5%)
DyVTE+FastV	0.154s (-34.9%)	73.0 (+0.2%)	0.155s (-34.4%)	68.5 (+1.2%)

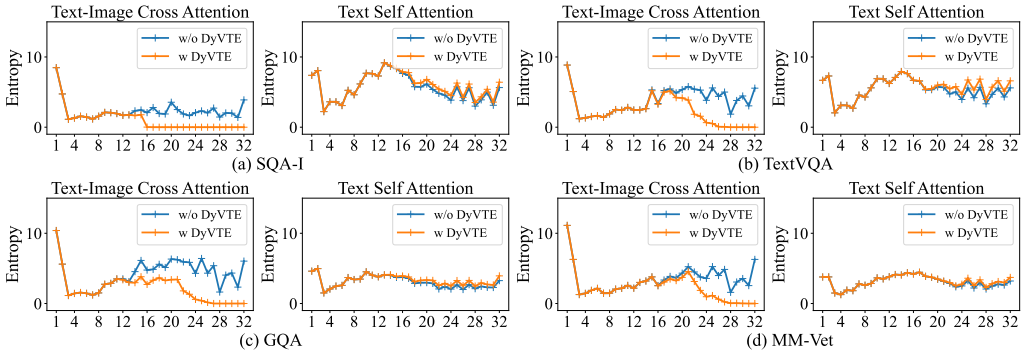

Figure 5: The entropy of two attention distributions. We visualize the entropy of cross-attention matrices and text self-attention on LLaVA-1.5 7B with and without our DyVTE, which shows that the removal of visual tokens barely affect the text self-attention.

Table 4: Ablation study of different token statuses for DyVTE. "*Mean Visual*" denotes the average of all visual tokens except the last one, similar with "*Mean Text*". "*Last Visual*" and "*Last Text*" denotes the last visual or text token. "*Exit Layer*" denotes the averaged layer numbers selected to remove by DyVTE. "*Acc.*" denotes the accuracy. The default setting of DyVTE is the last row. The best and second best results are marked in **bold** and underline, receptively.

State				GQA		TextVQA		MM-Vet		SQA-I		Average	
Mean Visual	Last Visual	Mean Text	Last Text	Acc. ↑	Exit Layer ↓	Acc. ↑	Exit Layer ↓	Acc. ↑	Exit Layer ↓	Acc. ↑	Exit Layer ↓	Acc. ↑	Exit Layer ↓
✓				**61.4**	21.6	57.3	21.0	**31.5**	21.4	69.7	21.7	**55.0**	21.4
	✓			61.2	21.3	57.1	21.1	30.0	**21.0**	69.7	21.4	54.5	21.2
		✓		60.1	17.4	**57.6**	22.1	31.1	22.1	69.7	20.4	54.6	20.5
			✓	59.8	**16.3**	56.3	**19.1**	29.9	**21.0**	**69.8**	**13.8**	54.0	**17.6**
✓	✓			**61.1**	21.2	57.3	21.3	31.6	21.3	**69.7**	21.0	**54.9**	21.2
✓		✓		58.7	16.5	**57.6**	22.2	**32.7**	22.5	**69.7**	21.3	54.7	20.6
✓			✓	59.4	**16.3**	56.5	19.9	30.9	21.8	**69.7**	14.4	54.1	18.1
	✓	✓		59.4	16.7	57.4	21.8	31.3	22.0	69.6	20.3	54.4	20.2
	✓		✓	59.3	16.4	56.5	19.9	31.2	21.6	69.6	14.4	54.1	18.1
		✓	✓	60.0	16.4	56.6	**19.7**	31.9	**21.1**	69.6	**13.4**	54.5	**17.6**

Inference Latency. In Tab. 3, we compare the actual inference latency of DyVTE with the token prune and the vanilla decoding methods. From the table, we can first observe that token prune methods can significantly reduce the inference latency. For instance, FastV can reduce the inference time by -27.7% and -26.2% on SQA and MMB benchmarks, receptively. Compared with these token prune methods, our DyVTE further enhances inference efficiency. Specifically, DyVTE improve the inference efficiency by 32.1% and 30.9% on SQA and MMB benchmarks with competitive performance. Moreover, the proposed DyVTE can work together with existing token prune methods and achieve better efficiency. For example, when combined with FastV, DyVTE achieves even greater reductions in inference latency, *i.e.*, 34.9% and 34.4% on SQA and MMB benchmarks. Overall, these results validate our DyVTE can accelerate the inference.

The attention entropy with visual tokens exit. In Fig.5, we visualize the entropy distributions of cross-attention matrices and text self-attention matrices of LLaVA-1.5 7B. And we compare the results with and without DyVTE across four benchmarks. We first observe that there are differences between the multimodal reasoning procedures across different benchmarks. Specifically, in SQA and TextVQA, the changes in entropy during text modeling are more obvious than those in GQA and MM-Vet. This phenomenon fully illustrates the necessity of adopting a dynamic approach to exit visual tokens. Another observation is that DyVTE can effectively remove all visual tokens at a specific layer according to the status of multimodal reasoning. Specifically, after removing all visual tokens by DyVTE, the entropy of text self-attention remains unaffected. In this way, the exiting determined by DyVTE only blocks visual information's cross-modality propagation while preserving text tokens' modeling. Overall, these findings validate that visual tokens are not always necessary in the reasoning process, and also confirm the effectiveness of the proposed DyVTE.

Ablation study. In Tab.4, we perform a set of experiments to analyze the effectiveness of different token status for DyVTE. In this table, various tokens are used as the token statuses in Eq.1. As shown in Tab.4, when using only image-related tokens, the model tends to remove all visual tokens at a later stage. Specifically, the average exit layer are 21.4 and 21.2. Using the "*Mean Text*" token leads to an earlier exit from the layers, but still retains the computation redundancy. When only the "*Last Text*" token is used, DyVTE removes all visual tokens too early, *i.e.*, at the 17-th layer, resulting

Table 5: Comparison between DyVTE and token pruning methods. The best and second best results are marked in **bold** and underline.

Method	SQA-I		MM-Vet		SEED		MMB		Average	
	Accuracy ↑	TFLOPs ↓	Score ↑	TFLOPs ↓	Accuracy ↑	TFLOPs ↓	Accuracy ↑	TFLOPs ↓	Accuracy ↑	TFLOPs ↓
LLaVA 7B	69.4	9.8	30.5	8.7	58.6	9.2	64.3	9.6	55.7	9.3
ToMe [5]	**69.6** (+0.3%)	5.9 (-39.8%)	30.6 (+0.3%)	4.9 (-43.7%)	57.8 (-1.4%)	5.5 (-40.2%)	63.7 (-0.9%)	5.7 (-40.6%)	55.4 (-0.5%)	5.5 (-40.9%)
FastV [7]	69.0 (-0.6%)	6.2 (-36.7%)	31.3 (+2.6%)	5.2 (-35.8%)	58.2 (-0.7%)	5.8 (-37.0%)	64.4 (+0.2%)	6.0 (-37.5%)	55.7 (0.0%)	5.8 (-37.6%)
DyVTE	**69.6** (+0.3%)	5.5 (-43.9%)	**31.9** (+4.6%)	6.3 (-27.6%)	**58.6** (0.0%)	5.0 (-45.7%)	**64.7** (+0.6%)	5.4 (-43.8%)	**56.2** (+0.9%)	5.5 (-40.9%)
DyVTE+FastV	68.9 (-0.7%)	**4.8** (-51.0%)	29.8 (-2.3%)	**4.0** (-54.0%)	58.2 (-0.7%)	**4.6** (-50.0%)	64.4 (+0.2%)	**4.6** (-52.1%)	55.3 (-0.7%)	**4.5** (-51.5%)

Table 6: Results of MLLMs with DyVTE that trained on themselves and others on five benchmarks. "*Trained*" denotes the MLLM that DyVTE trained on. "*Exit Layer*" denotes the averaged layer numbers selected to remove by DyVTE. "*Acc.*" denotes the accuracy.

MLLM	Trained	GQA		VQAv2		SEED		MMB		MME	
		Acc. ↑	Exit Layer ↓	Acc. ↑	Exit Layer ↓	Acc. ↑	Exit Layer ↓	Acc. ↑	Exit Layer ↓	Acc. ↑	Exit Layer ↓
VILA 7B	-	63.1	-	80.3	-	61.7	-	69.9	-	1489.2	-
	VILA 7B	61.9	17.4	79.2	16.7	61.8	17.6	69.8	16.2	1503.1	13.1
	LLaVA 7B	61.3	17.1	79.1	17.1	61.8	15.8	69.8	16.5	1504.9	14.6
InternVL 7B	-	62.9	-	79.3	-	59.2	-	64.6	-	1525.1	-
	InternVL 7B	61.3	16.1	77.6	15.8	59.1	13.9	64.4	13.6	1474.1	12.1
	LLaVA 7B	61.3	17.0	77.7	16.8	59.2	18.1	64.9	17.8	1516.9	14.1

in an obvious performance drop, *i.e.*, performance drops by 1.0% on average. For combinations of two representations, we can observe that the "*Mean Text + Last Text*" is the most effective way. Specifically, it removes visual tokens at about 17-th layer, while causing only a 0.5% performance drop. On the other hand, using a combination of visual token statuses, *i.e.*, "*Mean Visual + Last Visual*", yields results similar to using a single visual representation. In this way, the visual tokens are retained until the 21-st layer. Overall, these results confirm that the key to judging whether visual tokens have a necessary impact on the prediction lies in the status of text tokens.

Generalization to different MLLMs. To evaluate the generalization capability of DyVTE across different MLLMs, we conduct experiments where DyVTE is trained on one MLLM and directly applied to another without additional fine-tuning. As shown in Tab.6, we compare the performance of DyVTE trained on the same MLLM versus trained on LLaVA 7B and applied to VILA 7B and InternVL 7B. The results demonstrate that DyVTE exhibits strong cross-model generalization. For instance, when applying DyVTE trained on LLaVA to VILA, the model achieves 61.3% accuracy on GQA with an average exit layer of 17.1, which is comparable to the specifically trained DyVTE (61.9% accuracy with exit layer 17.4). Another notable observation is that the optimal exit layers identified by cross-model DyVTE are close to those of specifically tuned DyVTE. On the MME benchmark, both VILA with LLaVA-trained DyVTE (exit layer 14.6) and VILA with self-trained DyVTE (exit layer 13.1) achieve similar efficiency gains. These results indicate that DyVTE has learned generalized patterns of visual token importance across different MLLM architectures. This strong generalization capability stems from the task definition of DyVTE as a simple binary prediction problem, which makes the optimization independent of specific MLLM gradient updates and enables direct supervision via binary labels based on MLLMs' predictions. Overall, the cross-model experiments validate that DyVTE can be trained once and effectively deployed across different MLLMs with minimal performance degradation.

Comparison with token pruning methods. In Tab.5, we compare the performance and efficiency of DyVTE with the visual token pruning methods on LLaVA-1.5 7B. From the table, we can observe that DyVTE outperforms token pruning methods in both performance and efficiency. For example, when compared with ToMe on SQA, DyVTE not only maintains the performance but also further reduces FLOPs by 4.1%. When compared with FastV, the advantage of DyVTE is more significant. Specifically, DyVTE improves performance by 0.4% while reducing computation overhead by 8.7% on the SEED. Another observation is that DyVTE can significantly improve performance in complex tasks. For example, in the MM-Vet benchmark, which evaluates multiple functions including mathematics and OCR, DyVTE outperforms ToMe by 4.3Additionally, DyVTE shows excellent compatibility with existing methods. When combined with FastV, DyVTE reduces the computational overhead by 51.5% while only decreasing the performance by 0.7% on average. Overall, these results validate the effectiveness and efficiency of DyVTE.

5.4 Qualitative Analysis

To gain insight into the proposed DyVTE, we visualize the effect of applying it on LLaVA-1.5 7B in Fig. 6. We can first observe that DyVTE maintains consistent predictions with the default LLaVA. As shown in Figure 6-(a), DyVTE can correctly identify the object in question even when the foreground is complex. Additionally, DyVTE provides much faster inference than the default LLaVA. Specifically, in the two given examples, DyVTE can improve inference speed by up to 10.2% compared to the default LLaVA. Another observation is that removing redundant visual information may enhance performance. As shown in Fig.6-(b), applying DyVTE can help LLaVA to better recognize the numbers on the sign. Overall, these results confirm that DyVTE enhances inference efficiency without harming the reasoning process of the default model, aligning well with our motivation.

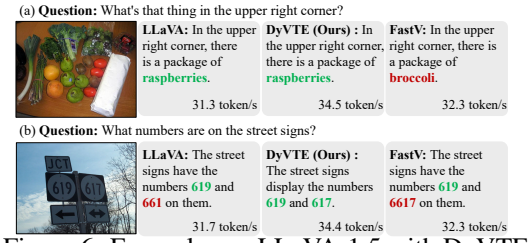

Figure 6: Examples on LLaVA-1.5 with DyVTE. Our DyVTE can help the MLLM answer the questions as accurately as default, while being faster.

6 Limitation

While DyVTE shows promising efficiency gains across multiple MLLMs and benchmarks, DyVTE currently performs full visual-token exit, which may not be optimal for cases where partial visual information is still beneficial in later layers.

7 Conclusion

In this paper, we investigate the visual redundancy problem of MLLMs via analyzing their inference behaviors, and summarize three key stages of MLLMs during multimodal inference, *i.e.*, *early fusion*, *inter-modality modeling*, and *multimodal reasoning*. Moreover, we also reveal that visual tokens stop contributing to multimodal reasoning after some layers. Motivated by this insight, we propose a simple yet effective method for MLLMs, termed Dynamic Visual-Token Exit (DyVTE) which optimally determines the layer at which visual tokens can be removed, thereby reducing computational overhead without compromising performance. Extensive experiments on nine benchmarks demonstrate that DyVTE significantly enhances the efficiency of MLLMs while maintaining their performance.

8 Acknowledgment

This work was supported by the National Science Fund for Distinguished Young Scholars (No.62025603), the National Natural Science Foundation of China (No. U21B2037, No. U22B2051, No. U23A20383, No. U21A20472, No. 62176222, No. 62176223, No. 62176226, No. 62072386, No. 62072387, No. 62072389, No. 62002305 and No. 62272401), and the Natural Science Foundation of Fujian Province of China (No. 2021J06003, No.2022J06001).

References

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

[2] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

[3] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023.

[4] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[5] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. *arXiv preprint arXiv:2210.09461*, 2022.

[6] Tom B Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[7] Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models. *arXiv preprint arXiv:2403.06764*, 2024.

[8] Mengzhao Chen, Wenqi Shao, Peng Xu, Mingbao Lin, Kaipeng Zhang, Fei Chao, Rongrong Ji, Yu Qiao, and Ping Luo. Diffrate: Differentiable compression rate for efficient vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17164–17174, 2023.

[9] Yanxi Chen, Xuchen Pan, Yaliang Li, Bolin Ding, and Jingren Zhou. Ee-llm: Large-scale training and inference of early-exit large language models with 3d parallelism. *arXiv preprint arXiv:2312.04916*, 2023.

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

[11] Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *arXiv preprint arXiv:2404.16821*, 2024.

[12] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *arXiv preprint arXiv:2312.14238*, 2023.

[13] Peiyan Dong, Mengshu Sun, Alec Lu, Yanyue Xie, Kenneth Liu, Zhenglun Kong, Xin Meng, Zhengang Li, Xue Lin, Zhenman Fang, et al. Heatvit: Hardware-efficient adaptive token pruning for vision transformers. In *2023 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, pages 442–455. IEEE, 2023.

[14] Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Bin Wang, Linke Ouyang, Xilin Wei, Songyang Zhang, Haodong Duan, Maosong Cao, et al. Internlm-xcomposer2: Mastering free-form text-image composition and comprehension in vision-language large model. *arXiv preprint arXiv:2401.16420*, 2024.

[15] Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Bin Wang, Linke Ouyang, Songyang Zhang, Haodong Duan, Wenwei Zhang, Yining Li, et al. Internlm-xcomposer2-4khd: A pioneering large vision-language model handling resolutions from 336 pixels to 4k hd. *Advances in Neural Information Processing Systems*, 37:42566–42592, 2025.

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

[17] Siqi Fan, Xin Jiang, Xiang Li, Xuying Meng, Peng Han, Shuo Shang, Aixin Sun, Yequan Wang, and Zhongyuan Wang. Not all layers of llms are necessary during inference. *arXiv preprint arXiv:2403.02181*, 2024.

[18] Mohsen Fayyaz, Soroush Abbasi Koohpayegani, Farnoush Rezaei Jafari, Sunando Sengupta, Hamid Reza Vaezi Joze, Eric Sommerlade, Hamed Pirsiavash, and Jürgen Gall. Adaptive token sampling for efficient vision transformers. In *European Conference on Computer Vision*, pages 396–414. Springer, 2022.

[19] Zhengcong Fei, Xu Yan, Shuhui Wang, and Qi Tian. Deecap: Dynamic early exiting for efficient image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12216–12226, 2022.

[20] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, and Rongrong Ji. Mme: A comprehensive evaluation benchmark for multimodal large language models. *Computing Research Repository (CoRR)*, 2023.

[21] Amir Ghodrati, Babak Ehteshami Bejnordi, and Amirhossein Habibian. Frameexit: Conditional early exiting for efficient video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15608–15618, 2021.

[22] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 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, 2017.

[23] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*, 2019.

[24] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

[25] Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

[26] Yutao Jiang, Qiong Wu, Wenhao Lin, Wei Yu, and Yiyi Zhou. What kind of visual tokens do we need? training-free visual token pruning for multi-modal large language models from the perspective of graph. *arXiv preprint arXiv:2501.02268*, 2025.

[27] Zhenglun Kong, Peiyan Dong, Xiaolong Ma, Xin Meng, Wei Niu, Mengshu Sun, Xuan Shen, Geng Yuan, Bin Ren, Hao Tang, et al. Spvit: Enabling faster vision transformers via latency-aware soft token pruning. In *European conference on computer vision*, pages 620–640. Springer, 2022.

[28] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[29] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

[30] Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. Seed-bench: Benchmarking multimodal llms with generative comprehension. *Computing Research Repository (CoRR)*, 2023.

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

[32] Wentong Li, Yuqian Yuan, Jian Liu, Dongqi Tang, Song Wang, Jie Qin, Jianke Zhu, and Lei Zhang. Tokenpacker: Efficient visual projector for multimodal llm. *arXiv preprint arXiv:2407.02392*, 2024.

[33] Xudong Li, Mengdan Zhang, Peixian Chen, Xiawu Zheng, Yan Zhang, Jingyuan Zheng, Yunhang Shen, Ke Li, Chaoyou Fu, Xing Sun, et al. Zooming from context to cue: Hierarchical preference optimization for multi-image mllms. *arXiv preprint arXiv:2505.22396*, 2025.

[34] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *Computing Research Repository (CoRR)*, 2023.

[35] Bin Lin, Zhenyu Tang, Yang Ye, Jiaxi Cui, Bin Zhu, Peng Jin, Jinfa Huang, Junwu Zhang, Yatian Pang, Munan Ning, et al. Moe-llava: Mixture of experts for large vision-language models. *arXiv preprint arXiv:2401.15947*, 2024.

[36] Ji Lin, Hongxu Yin, Wei Ping, Yao Lu, Pavlo Molchanov, Andrew Tao, Huizi Mao, Jan Kautz, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models. *arXiv preprint arXiv:2312.07533*, 2023.

[37] Zhihang Lin, Mingbao Lin, Luxi Lin, and Rongrong Ji. Boosting multimodal large language models with visual tokens withdrawal for rapid inference. *arXiv preprint arXiv:2405.05803*, 2024.

[38] Daria Lioubashevski, Tomer Schlank, Gabriel Stanovsky, and Ariel Goldstein. Looking beyond the top-1: Transformers determine top tokens in order. *arXiv preprint arXiv:2410.20210*, 2024.

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

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

[41] 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? *Computing Research Repository (CoRR)*, 2023.

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

[43] Gen Luo, Yiyi Zhou, Yuxin Zhang, Xiawu Zheng, Xiaoshuai Sun, and Rongrong Ji. Feast your eyes: Mixture-of-resolution adaptation for multimodal large language models. *arXiv preprint arXiv:2403.03003*, 2024.

[44] Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and CV Jawahar. Infographicvqa. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1697–1706, 2022.

[45] Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2200–2209, 2021.

[46] Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

[47] Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. Llava-prumerge: Adaptive token reduction for efficient large multimodal models. *arXiv preprint arXiv:2403.15388*, 2024.

[48] Min Shi, Fuxiao Liu, Shihao Wang, Shijia Liao, Subhashree Radhakrishnan, De-An Huang, Hongxu Yin, Karan Sapra, Yaser Yacoob, Humphrey Shi, et al. Eagle: Exploring the design space for multimodal llms with mixture of encoders. *arXiv preprint arXiv:2408.15998*, 2024.

[49] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8317–8326, 2019.

[50] Alessandro Stolfo, Yonatan Belinkov, and Mrinmaya Sachan. A mechanistic interpretation of arithmetic reasoning in language models using causal mediation analysis. *arXiv preprint arXiv:2305.15054*, 2023.

[51] Shengkun Tang, Yaqing Wang, Zhenglun Kong, Tianchi Zhang, Yao Li, Caiwen Ding, Yanzhi Wang, Yi Liang, and Dongkuan Xu. You need multiple exiting: Dynamic early exiting for accelerating unified vision language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10781–10791, 2023.

[52] Qwen Team. Qwen2.5: A party of foundation models, September 2024.

[53] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[54] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[55] Hongjie Wang, Bhishma Dedhia, and Niraj K Jha. Zero-tprune: Zero-shot token pruning through leveraging of the attention graph in pre-trained transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16070–16079, 2024.

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

[57] Siyuan Wei, Tianzhu Ye, Shen Zhang, Yao Tang, and Jiajun Liang. Joint token pruning and squeezing towards more aggressive compression of vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2092–2101, 2023.

[58] Qiong Wu, Zhaoxi Ke, Yiyi Zhou, Gen Luo, Xiaoshuai Sun, and Rongrong Ji. Routing experts: Learning to route dynamic experts in multi-modal large language models. *arXiv preprint arXiv:2407.14093*, 2024.

[59] Qiong Wu, Wei Yu, Yiyi Zhou, Shubin Huang, Xiaoshuai Sun, and Rongrong Ji. Parameter and computation efficient transfer learning for vision-language pre-trained models. *Advances in Neural Information Processing Systems*, 36, 2024.

[60] Jiaer Xia, Yuhang Zang, Peng Gao, Yixuan Li, and Kaiyang Zhou. Visionary-r1: Mitigating shortcuts in visual reasoning with reinforcement learning. *arXiv preprint arXiv:2505.14677*, 2025.

[61] Long Xing, Qidong Huang, Xiaoyi Dong, Jiajie Lu, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, Jiaqi Wang, Feng Wu, et al. Pyramiddrop: Accelerating your large vision-language models via pyramid visual redundancy reduction. *arXiv preprint arXiv:2410.17247*, 2024.

[62] Weihao Ye, Qiong Wu, Wenhao Lin, and Yiyi Zhou. Fit and prune: Fast and training-free visual token pruning for multi-modal large language models. *arXiv preprint arXiv:2409.10197*, 2024.

[63] Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. *Computing Research Repository (CoRR)*, 2023.

[64] Shaolei Zhang, Qingkai Fang, Zhe Yang, and Yang Feng. Llava-mini: Efficient image and video large multimodal models with one vision token. In *The Thirteenth International Conference on Learning Representations*, 2025.

[65] Baichuan Zhou, Ying Hu, Xi Weng, Junlong Jia, Jie Luo, Xien Liu, Ji Wu, and Lei Huang. Tinyllava: A framework of small-scale large multimodal models. *arXiv preprint arXiv:2402.14289*, 2024.

A Appendix

In this paper, we first conduct extensive empirical studies on the attention behaviors of MLLMs, and summarize three main inference stages in MLLMs: *(i) Early fusion* between tokens is first accomplished quickly. *(ii) Intra-modality modeling* then comes to play. *(iii) Multimodal reasoning* resumes and lasts until the end of inference. To better confirm our findings, we conduct further experiments on more MLLMs in the supplementary materials.

A.1 Attention Visualization

In Fig.2, we visualize the attention weight of each part in attention matrix, and the attention weights are calculated as follow: We first calculate the attention matrix with system prompt $\mathbf{S} \in \mathcal{R}^{s\times d}$, visual tokens $\mathbf{V} \in \mathcal{R}^{v\times d}$ and text tokens $\mathbf{T} \in \mathcal{R}^{t\times d}$:

$$\mathbf{A} = Softmax(\frac{[\mathbf{S},\mathbf{V},\mathbf{T}]\mathbf{W}_q([\mathbf{S},\mathbf{V},\mathbf{T}]\mathbf{W}_k)^T}{\sqrt{d}} \cdot \mathbf{M}), \tag{7}$$

where $\mathbf{M}$ represents the attention mask for causal reasoning. Then, the weight of each part can be represent as

$$W_v = \sum_{i=s}^{s+v}\sum_{j=s}^{s+v}\mathbf{A}_{ij}/v, W_c = \sum_{i=s+v}^{s+v+t}\sum_{j=s}^{s+v}\mathbf{A}_{ij}/t, W_t = \sum_{i=s+v}^{s+v+t}\sum_{j=s+v}^{s+v+t}\mathbf{A}_{ij}/t \tag{8}$$

where W_v, W_c and W_t represent the attention weights of visual self attention, cross attention and text self attention, respectively. In addition, we combine the attention matrices of different heads in MHA through the *max pooling*.

A.2 Ablation Study

In Tab.7 and Tab.8, we perform a set of experiments to analyze the impact of different representation selections on DyVTE. As shown in Tab.7, when relying on more than two representations, we can find out that their performance is similar to "*mean text + last text*". For instance, the additional representation, *i.e.*, "*last visual*" and "*mean visual*", has no noticeable effect on the results. We can also observe that the missing of "*mean text*" or "*last text*" will lead to inaccurate exit. For example, the missing of "*mean text*" leads to later exit, *i.e.*, 2 layers latter, while the absence of "*last text*" does the opposite, *i.e.*, 2 layers earlier. As for taking attention scores be the representations, we find that hyper-network can not make the correct judgment. Specifically, the exit layers for all benchmarks can be much earlier. Overall, these results well confirm the use of text tokens status for dynamic visual token exiting in MLLMs.

Table 7: Ablation study of the token status selection for DyVTE. "*Mean Visual*" denotes the average of all visual tokens except the last one, similar with "*Mean Text*". "*Last Visual*" and "*Last Text*" denotes the last visual or text token. "*Exit Layer*" denotes the averaged layer numbers selected to remove by DyVTE. "*Acc.*" denotes the accuracy. The default setting of DyVTE is the last row. The best and second best results are marked in **bold** and underline, receptively.

State				GQA		TextVQA		MM-Vet		SQA-I		Average	
Mean Visual	Last Visual	Mean Text	Last Text	Acc. ↑	Exit Layer ↓	Acc. ↑	Exit Layer ↓	Acc. ↑	Exit Layer ↓	Acc. ↑	Exit Layer ↓	Acc. ↑	Exit Layer ↓
✓				**61.4**	21.6	57.3	21.0	**31.5**	21.4	69.7	21.7	**55.0**	21.4
	✓			61.2	21.3	57.1	21.1	30.0	**21.0**	69.7	21.4	54.5	21.2
		✓		60.1	17.4	**57.6**	22.1	31.1	22.1	69.7	20.4	54.6	20.5
			✓	59.8	**16.3**	56.3	**19.1**	29.9	**21.0**	**69.8**	**13.8**	54.0	**17.6**
✓	✓			**61.1**	21.2	57.3	21.3	31.6	21.3	**69.7**	21.0	**54.9**	21.2
✓		✓		58.7	16.5	**57.6**	22.2	**32.7**	22.5	**69.7**	21.3	54.7	20.6
✓			✓	59.4	**16.3**	56.5	19.9	30.9	21.8	**69.7**	14.4	54.1	18.1
	✓	✓		59.4	16.7	57.4	21.8	31.3	22.0	69.6	20.3	54.4	20.2
	✓		✓	59.3	16.4	56.5	19.9	31.2	21.6	69.6	14.4	54.1	18.1
		✓	✓	60.0	16.4	56.6	**19.7**	31.9	**21.1**	69.6	**13.4**	54.5	**17.6**
	✓	✓	✓	60.1	16.4	57.0	20.2	30.8	**21.3**	69.6	14.1	54.4	18.0
✓		✓	✓	**60.2**	16.7	57.1	20.3	31.0	21.6	**69.7**	14.3	**54.5**	18.2
✓	✓		✓	57.6	15.9	**57.3**	22.2	**31.7**	22.1	**69.7**	20.1	54.1	20.2
✓	✓	✓		57.8	**15.2**	56.2	**19.3**	30.0	21.6	69.3	**11.7**	53.3	**16.9**
✓	✓	✓	✓	60.5	16.8	57.1	20.5	31.1	21.8	69.8	14.1	54.6	18.3
LLaVA-1.5 7B				62.0	-	58.2	-	30.5	-	69.4	-	55.0	-

Table 8: Ablation study of the attention status selection for DyVTE. "*Visual Self*" denotes the attention score from visual modality of each visual tokens, similar with "*Text Self*". "*Cross*" denotes the attention score from visual to the text of each visual tokens. For attention representations whose dimensions are not 576, we use interpolation to align the dimensions. "*Exit Layer*" denotes the averaged layer numbers selected to remove by DyVTE. "*Acc.*" denotes the accuracy. The default setting of DyVTE is the last row. The best and second best results are marked in **bold** and underline.

State			GQA		TextVQA		MM-Vet		SQA-I		Average	
Visual Self	Cross	Text Self	Acc. ↑	Exit Layer ↓	Acc. ↑	Exit Layer ↓	Acc. ↑	Exit Layer ↓	Acc. ↑	Exit Layer ↓	Acc. ↑	Exit Layer ↓
✓			38.3	**0.5**	42.4	**0.7**	**12.2**	**0.9**	64.2	**0.6**	**39.9**	**0.7**
	✓		38.8	1.3	**43.1**	1.3	11.5	1.2	**65.2**	1.2	39.6	1.2
		✓	**39.6**	1.7	42.7	1.1	11.9	3.6	64.7	**0.6**	39.7	1.7
✓	✓		39.4	**2.7**	**43.1**	3.0	13.1	**2.6**	**65.4**	3.8	40.3	3.0
✓		✓	**40.7**	3.0	43.0	1.7	13.3	5.9	65.1	**1.0**	40.5	**2.9**
	✓	✓	**40.7**	3.0	**43.1**	**1.6**	**13.4**	6.0	65.2	**1.0**	**40.6**	**2.9**
✓	✓	✓	49.5	10.9	45.6	5.6	15.6	7.1	65.6	2.5	44.1	6.5
LLaVA-1.5 7B			62.0	-	58.2	-	30.5	-	69.4	-	55.0	-

A.3 Distribution of exit layers

In Fig. 7, we present the distributions of the visual token exit layers for LLaVA-DyVTE 7B and 13B. From the figure, we observe that DyVTE dynamically selects the exit layer based on the specific task requirements. For example, on the TextVQA dataset, which demands fine-grained information, the exit occurs later compared to other datasets. In contrast, on the GQA dataset, which involves open-ended word questions and simple true/false judgments, the exit layers are more evenly distributed across early and late layers. Another notable observation is that the timing of the exit layer shows similar distributions across MLLMs of different scales and architectures. For instance, both LLaVA-DyVTE 7B and 13B remove all visual tokens at the same layer early in the process on the SQA dataset. Similarly, for InternVL and LLaVA, peaks in the exit layer distributions occur at comparable layers. Overall, these results support our motivation and the proposed DyVTE.

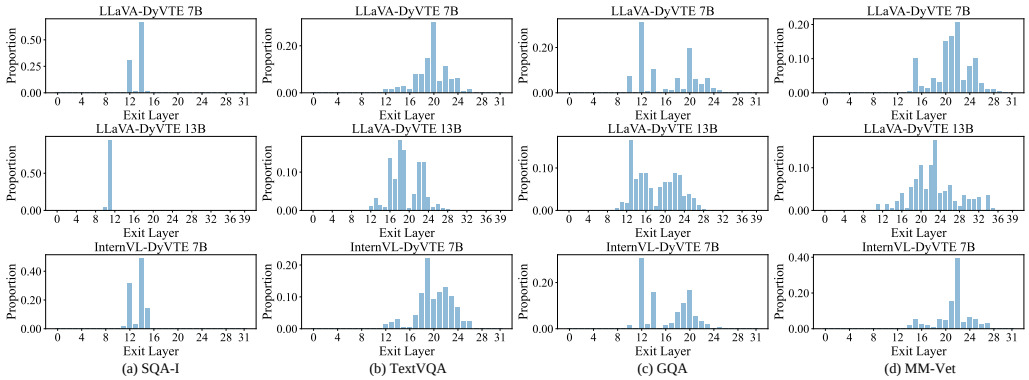

Figure 7: Statistics of the exiting layers decided by DyVTE on LLaVA-1.5 7B, LLaVA-1.5 13B and InternVL 7B. The horizontal axis represents the exit layer, and vertical axis represents the proportion. The exit time by is different for different tasks, but similar for MLLMs with the same sizes.

A.4 Distribution of Attention

In Fig.8-12, we further visualize the attention distributions on different MLLMs for different datasets. We can first observe that datasets produce similar attention patterns on the same MLLM. Another observation is that attention patterns across different MLLMs have similar trends and the distributions can generally be categorized into three distinct stages. These findings emphasize the universality of attention behavior in MLLMs, further validating our approach. Overall, the experiment results confirm that different MLLMs have similar attention patterns on the different data.

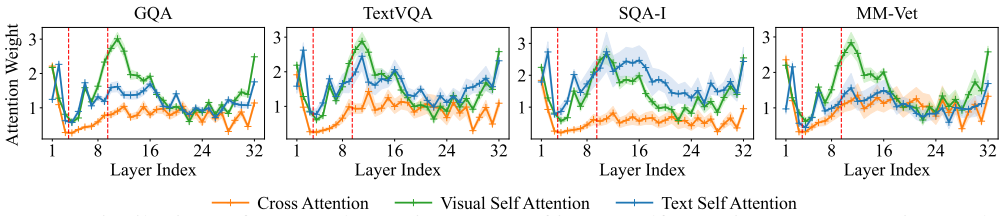

Figure 8: Distributions of averaged attention scores of image self-attention, cross-attention, and text self-attention of LLaVA-1.5 7B. We visualize the mean and variance of the attention weight of each part on GQA, TextVQA, SQA-I, MM-Vet datasets. From these distributions, we can summarize three main stages of MLLMs as introduced in the main paper, which are then marked by red lines.

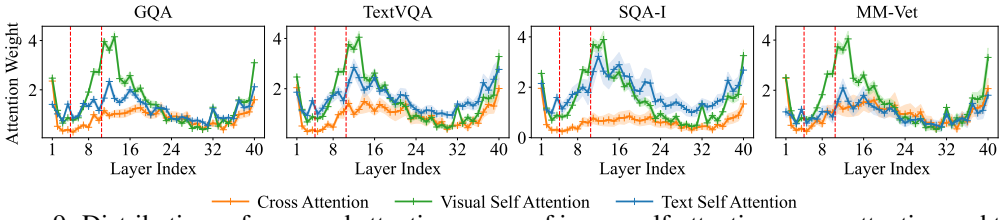

Figure 9: Distributions of averaged attention scores of image self-attention, cross-attention and text self-attention of LLaVA-1.5 13B. We visualize the mean and variance of the attention weight of each part on GQA, TextVQA, SQA-I, MM-Vet datasets. From these distributions, we can summarize three main stages of MLLMs as introduced in the main paper, which are then marked by red lines.

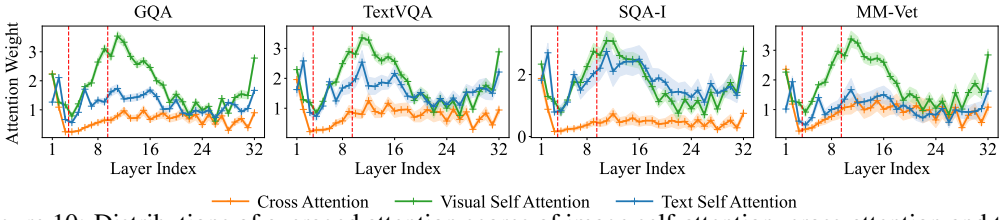

Figure 10: Distributions of averaged attention scores of image self-attention, cross-attention and text self-attention of InternVL 7B. We visualize the mean and variance of the attention weight of each part on GQA, TextVQA, SQA-I, MM-Vet datasets. From these distributions, we can summarize three main stages of MLLMs as introduced in the main paper, which are then marked by red lines.

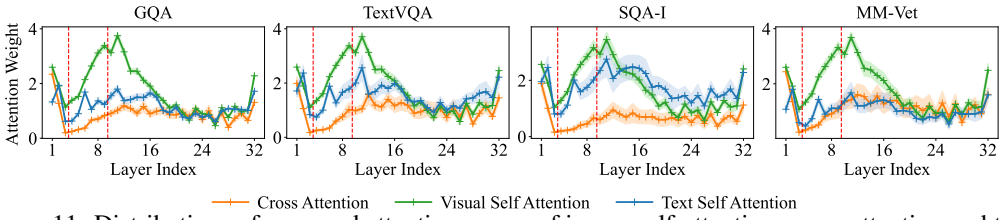

Figure 11: Distributions of averaged attention scores of image self-attention, cross-attention and text self-attention of VILA 7B. We visualize the mean and variance of the attention weight of each part on GQA, TextVQA, SQA-I, MM-Vet datasets. From these distributions, we can summarize three main stages of MLLMs as introduced in the main paper, which are then marked by red lines.

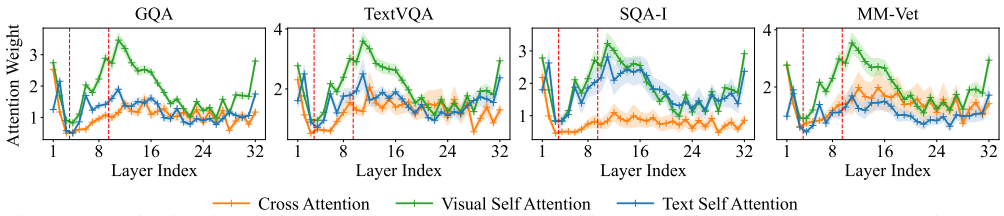

Figure 12: Distributions of averaged attention scores of image self-attention, cross-attention and text self-attention of EAGLE-X5 7B. We visualize the mean and variance of the attention weight of each part on GQA, TextVQA, SQA-I, MM-Vet datasets. From these distributions, we can summarize three main stages of MLLMs as introduced in the main paper, which are then marked by red lines.

A.5 Entropy of Attention

In Fig.13-17, we visualize the entropy distributions of cross-attention matrices and text self-attention matrices. For all these MLLMs and datasets, we can observe that removing all visual tokens at an appropriate time will not have a significant impact on the answer modeling process. Specifically, for all MLLMs and datasets, the removing of visual tokens only makes the entropy of cross-attention disappear, while the entropy of text self-attention maintains the same distribution. Overall, these results well confirm that cross-modal interaction in fact barely contributes to multimodal reasoning after some layers.

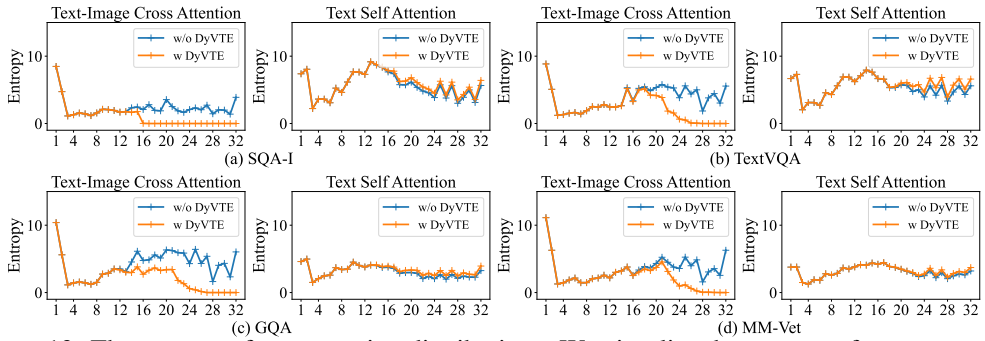

Figure 13: The entropy of two attention distributions. We visualize the entropy of cross-attention matrices and text self-attention on LLaVA-1.5 7B with and without our DyVTE, which shows that the removal of visual tokens barely affect the text self-attention.

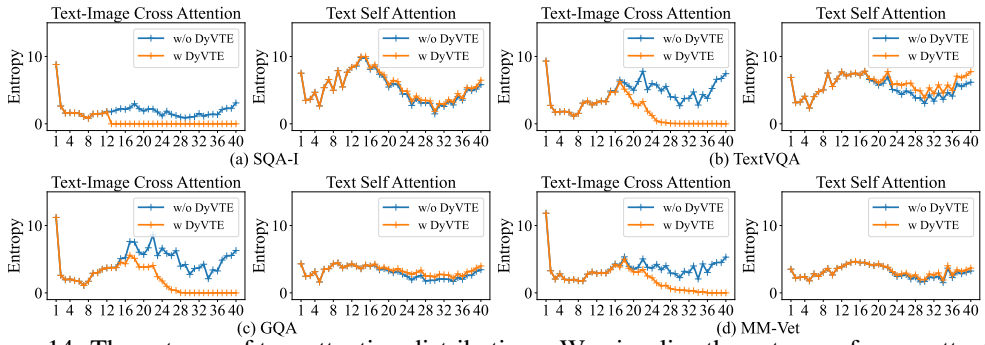

Figure 14: The entropy of two attention distributions. We visualize the entropy of cross-attention matrices and text self-attention on LLaVA-1.5 13B with and without our DyVTE, which shows that the removal of visual tokens barely affect the text self-attention.

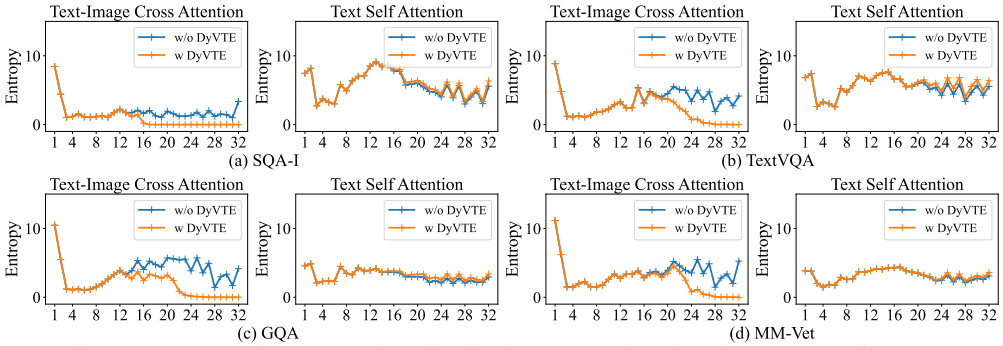

Figure 15: The entropy of two attention distributions. We visualize the entropy of cross-attention matrices and text self-attention on InternVL 7B with and without our DyVTE, which shows that the removal of visual tokens barely affect the text self-attention.

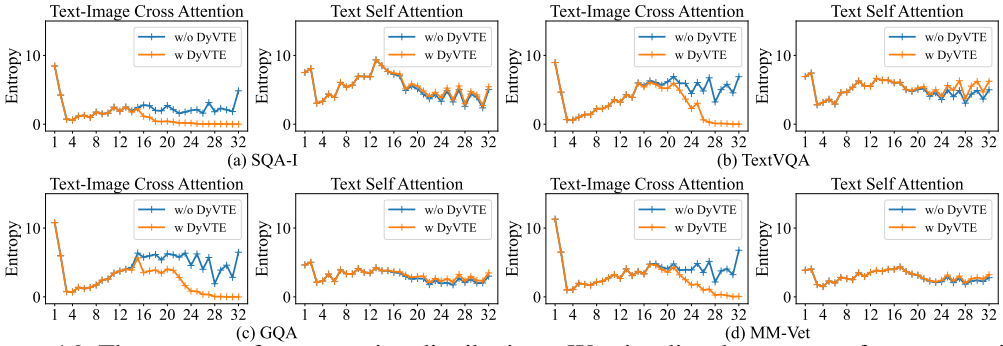

Figure 16: The entropy of two attention distributions. We visualize the entropy of cross-attention matrices and text self-attention on VILA 7B with and without our DyVTE, which shows that the removal of visual tokens barely affect the text self-attention.

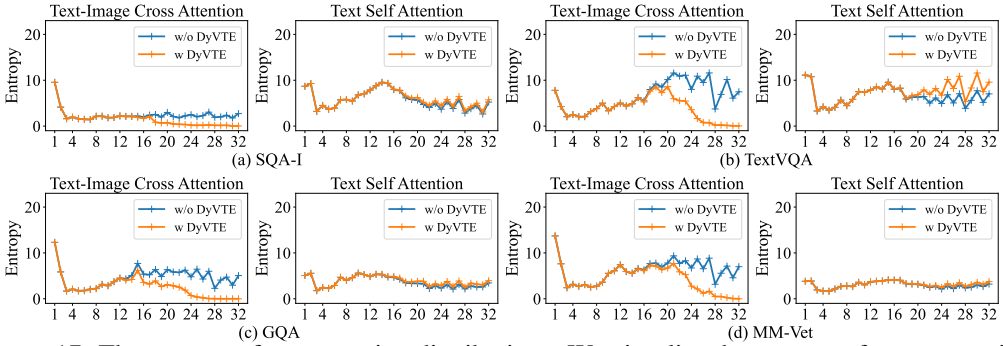

Figure 17: The entropy of two attention distributions. We visualize the entropy of cross-attention matrices and text self-attention on EAGLE-X5 7B with and without our DyVTE, which shows that the removal of visual tokens barely affect the text self-attention.

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 abstract and introduction explicitly outline the main contributions, including an empirical findings in the inference MLLM and a novel and effective DyVTE method to exit the visual tokens in an early stage.

   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 limitation have been provided in Sec. 6.

   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: [NA]

   Justification: The paper does not include any new theoretical results.

   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: We provide all the details in our paper, and our code is anonymously released.

   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: Our code is anonymously released and all datasets we used is public.

   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, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide all the details in our paper, and our code is anonymously released.

   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: [No]

   Justification: The paper does not include experiments involving random processes that would require the reporting of error bars or statistical significance.

   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 have provide these information in our experiment 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: We have strictly adhered to the guidelines.

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: [NA]

    Justification: The paper does not explicitly discuss the societal impacts of the work performed, as it primarily focuses on the technical development of multimodal large language models.

    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: Our work is based on public data and models.

    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: We have cited their works.

    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: Our code is produced in our anonymous project.

    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: [No]

    Justification: Our 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: Our 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 rigorousness, or originality of the research, declaration is not required.

    Answer: [No]

    Justification: LLM, as a part of MLLM, is the object of our study.

    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.