ViCO: A Training Strategy towards Semantic Aware Dynamic High-Resolution

Long Cui[1,2] **Weiyun Wang**[3,2] **Jie Shao**[4,2] **Zichen Wen**[1,2]
Gen Luo[2] **Linfeng Zhang**[1] **Yanting Zhang**[5*] **Wenhai Wang**[6,2*]
[1]Shanghai Jiao Tong University
[2]Shanghai Artificial Intelligence Laboratory
[3]Fudan University [4]Nanjing University
[5]Donghua University
[6]The Chinese University of Hong Kong

Abstract

Existing Multimodal Large Language Models (MLLMs) suffer from increased inference costs due to the additional vision tokens introduced by image inputs. In this work, we propose Visual Consistency Learning (ViCO), a novel training algorithm that enables the model to represent images of varying semantic complexities using different numbers of vision tokens. The key idea behind our method is to employ multiple MLP connectors, each with a different image compression ratio, to downsample the vision tokens based on the semantic complexity of the image. During training, we minimize the KL divergence between the responses conditioned on different MLP connectors. At inference time, we introduce an image router, termed Visual Resolution Router (ViR), that automatically selects the appropriate compression rate for each image patch. Compared with existing dynamic high-resolution strategies, which adjust the number of visual tokens based on image resolutions, our method dynamically adapts the number of visual tokens according to semantic complexity. Experimental results demonstrate that our method can reduce the number of vision tokens by up to 50% while maintaining the model's perception, reasoning, and OCR capabilities. We hope this work will contribute to the development of more efficient MLLMs. The code and models will be released to facilitate future research.

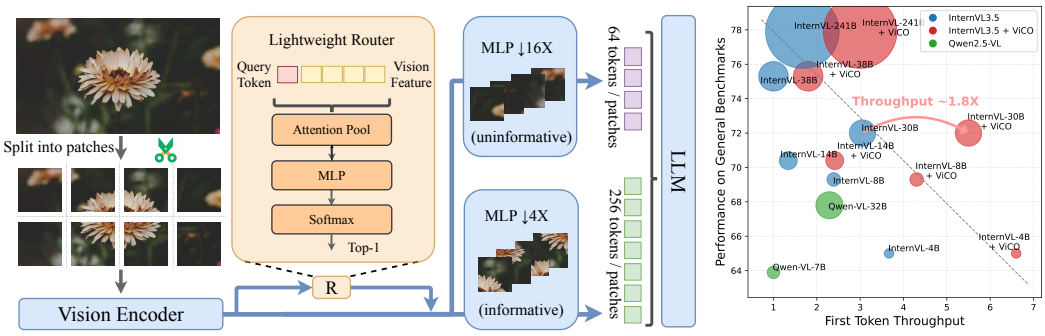
1 Introduction

Recent advancements in Multimodal Large Language Models (MLLMs) [1–7] have demonstrated remarkable performance across a wide range of tasks, showing tremendous potential for real-world applications. Despite these advancements, MLLMs still suffer from the increased inference costs due to the additional vision tokens introduced by image inputs. Taking InternVL3.5 [8] as an example, under its default configuration [9], each image is divided into up to 13 patches (including one thumbnail). Each patch is then represented by 256 visual tokens, resulting in a maximum of 3,328 tokens per image. In real-world scenarios such as document understanding or video comprehension, models are often required to process multiple images simultaneously. In such cases, the visual component will serve as the main body of the token sequence and constitutes the primary source of inference cost.

To address these challenges, we propose Visual Consistency Learning (ViCO), which introduces semantic-level adaptivity in the number of visual tokens. As shown in Fig. 1(a), given an image feature

*Corresponding authors.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).



(a) (b)

Figure 1: **Overall view of the ViCO pipeline and model performance.** (a) The ViCO inference pipeline, illustrating the image processing flow. (b) Bubble chart showing model performance on general benchmarks versus first token throughput of LLM. Bubble size is positively correlated with the number of model parameters. First token throughput is reported relative to InternVL3.5-38B, which is set as the baseline value of 1. Detailed experimental settings are provided in Section 4.8.

map, we introduce a Visual Resolution Router (ViR) that routes each patch to different compression rates: either a high-resolution representation with 256 tokens or a low-resolution representation with 64 tokens. These tokens are then concatenated with text tokens. The training procedure of ViCO consists of two stages: (1) *Consistency Training*: The model is trained to minimize the KL divergence between the responses conditioned on visual tokens with different compression rates. This encourages the model to generate accurate responses even when using highly compressed visual tokens, thereby improving its performance and robustness under compressed visual tokens. (2) *Router Training*: Although consistency training improves performance under high compression, inevitable information loss still causes a drop in performance. To mitigate this, we introduce ViR, which automatically selects the appropriate compression rate for each image patch. Those containing complex semantic information are represented using more tokens, while simpler patches are represented using fewer tokens. Compared with existing dynamic high-resolution strategies [10, 11], which adjust the number of visual tokens based on image resolutions, our method further determines the number of visual tokens allocated for each image patch at the semantic level. This fine-grained control enables substantial efficiency gains with minimal performance loss.

To evaluate the effectiveness of our method, we conducted extensive experiments on benchmarks spanning OCR, document understanding, video understanding, and multi-image reasoning. Experimental results show that our method can reduce the number of visual tokens by up to 50% while preserving the model's perception, reasoning, and OCR capabilities. As shown in Fig. 1(b), our method maintains the original performance while improving the first token throughput of InternVL3.5 series [8] across different model scales.

Our main contributions are as follows:

(1) We introduce a novel training strategy, termed Visual Consistency Learning, which minimizes the response distribution gap of the model between different visual token compression rates. This enables the model to effectively utilize highly compressed visual representations without significant performance loss.

(2) Building on ViCO, we develop a Visual Resolution Router that dynamically allocates visual tokens to image patches based on their semantic complexity. This provides fine-grained control over image patch representation and achieves a better trade-off between efficiency and accuracy.

(3) We perform large-scale experiments on benchmarks covering diverse multimodal recognition and reasoning tasks. Our results demonstrate that our method can halve the number of visual tokens while maintaining strong performance, which almost doubles inference throughput.

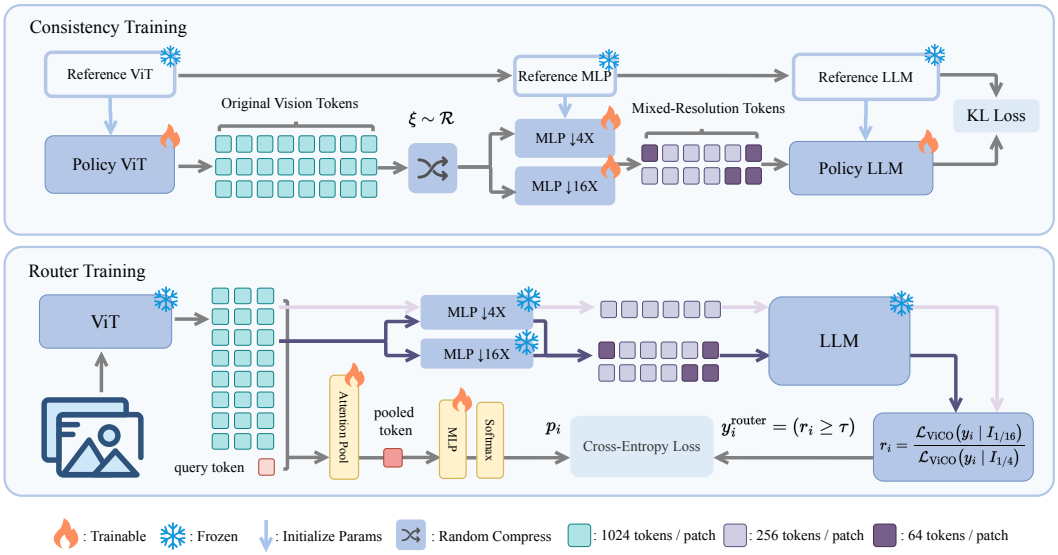


Figure 2: **Training procedure of Visual Consistency Learning (ViCO).** During the Consistency Training stage, the model aligns outputs under different compression rates. During the Router Training, the Visual Resolution Router (ViR) is trained to determine the appropriate compression for each patch based on its effect on model predictions.

2 Related Work

2.1 Multimodal Large Language Models.

Driven by the success of large language models [12–14], multimodal large language models (MLLMs) have advanced rapidly. Researchers have explored various alignment architectures, ranging from using connectors as soft prompts [15–19] and adding vision-language fusion layers [20, 21], to designs without vision encoders [22] that jointly process modalities through a unified Transformer. Although MLLMs vary in their architectural designs, most of them adopt a dynamic high-resolution strategy [2, 10], which segments images into patches based on their resolution to enhance the model's perceptual capabilities. However, this approach requires a large number of tokens to represent each image, thereby increasing the inference cost of MLLMs. In this paper, we propose a dynamic resolution strategy that is compatible with this paradigm. Our method introduces semantic-level adaptivity in determining the number of visual tokens needed to represent each image patch. This reduces the number of visual tokens and consequently the inference cost, while maintaining nearly the same performance.

2.2 Efficient Vision Language Models.

Improving vision-language models efficiency has led to extensive research on visual token compression. Early approaches, such as LLaMA-VID [23], DeCo [24], and MADTP [25], leverage context tokens, adaptive pooling, or cross-modal alignment to prune redundancy while retaining essential information. Other works like AVG-LLaVA[26] adaptively selects visual granularity based on textual instructions, while LLaVA-Mini[27] achieves extreme compression to a single token through modality pre-fusion. Additionally, training-free methods like FastV [28], SparseVLM [29], ToMe [30], and ToFu [31] reduce costs via attention-based pruning or token merging. However, these strategies often struggle with vision-sensitive tasks like OCR, where aggressive compression can sacrifice the fine-grained details essential for high-fidelity representation.

3 Visual Consistency Learning

To enhance recognition and perception capabilities, most existing MLLMs adopt a dynamic high-resolution strategy, which introduces a large number of visual tokens and greatly increases the

inference cost of these models. In this work, we propose Visual Consistency Learning (ViCO), a novel training algorithm that enables the model to represent images of varying semantic complexities using different numbers of vision tokens.

3.1 Consistency Training

As shown in Fig. 2, in the Consistency Training stage, the model is trained to generate consistent output conditioned on different patch compression rates. In practice, we introduce an extra reference model, which is frozen during training. The trained model is required to minimize the KL divergence between its response distribution and that of the reference model. The training objective is formulated as follows:

$$\mathcal{L}_{\text{ViCO}} = \mathbb{E}_{\xi \sim \mathcal{R}} \Bigg[\frac{1}{N} \sum_{i=1}^{N} \text{KL} \Big(\pi_{\theta_{ref}}(y_i \mid y_{<i}, I) \Big\| \pi_{\theta_{policy}}(y_i \mid y_{<i}, I_\xi) \Big) \Bigg], \tag{1}$$

where KL denotes the KL divergence, N is the sequence length, and ξ denotes the compression ratio of the patches in each image, which is uniformly sampled from $\mathcal{R} = [0, 1]$. The corresponding ratio of patches in image I_ξ is represented as 64 tokens, while the others are represented as 256 tokens. The reference model always performs inference with visual tokens without any compression.

3.2 Router Training

3.2.1 Implementation Architecture.

As illustrated in Fig. 2, the Router Training stage focuses on training the Visual Resolution Router (ViR), which is responsible for selecting an appropriate resolution for each input patch to balance efficiency and fidelity. ViR is implemented as a binary classifier and trained with a standard cross-entropy loss, while the main MLLM backbone remains frozen throughout this stage. To generate supervision signals for ViR, we first measure the effect of patch compression on the model's predictions. Concretely, for each patch, we calculate a loss ratio that quantifies how much the model's output degrades under compression. This ratio then serves as a guide for the router, indicating which patches can be safely compressed without significantly affecting overall performance. Specifically, for each patch, we compute a *loss ratio* defined as

$$r_i = \frac{\mathcal{L}_{\text{ViCO}}\big(y_i \mid I_{\frac{1}{16}}\big)}{\mathcal{L}_{\text{ViCO}}\big(y_i \mid I_{\frac{1}{4}}\big)}, \tag{2}$$

where $\mathcal{L}_{\text{ViCO}}$ denotes the consistency loss introduced in Section 3.1. This ratio captures the relative increase in loss caused by compressing the visual tokens, providing a principled measure of each patch's sensitivity to compression. Patches with low loss ratios can be safely compressed with minimal impact on the model's output, whereas patches with high loss ratios require higher resolution to preserve critical visual information. The binary ground-truth label for the router is then defined based on r_i:

$$y_i^{\text{router}} = \begin{cases} 0, & r_i < \tau \quad \text{(low compression impact)} \\ 1, & r_i \geq \tau \quad \text{(high compression impact)} \end{cases}, \tag{3}$$

where $y_i^{\text{router}} = 0$ indicates that the patch can be compressed with a more aggressive strategy, and $y_i^{\text{router}} = 1$ indicates that the patch can be compressed with a more conservative strategy. To maintain a balanced training signal, we store historical r_i values in a sliding window and dynamically set the threshold τ as the *k-th* percentile of these values. This approach typically results in roughly half of the patches being assigned to compression, which ensures a balanced distribution of target labels across patches. Notably, for each training sample, we randomly select a patch to compress and estimate its pseudo-label according to Eq. (3). The loss is only computed on this patch.

In practice, the predicted router value for each patch is obtained by first extracting visual token features from the ViT backbone, aggregating them using attention pooling, and passing the resulting

patch-level feature through a lightweight MLP. This process can be defined as:

$$p_i = \text{Softmax}\Big(\text{MLP}\big(\text{AttnPool}\big(\text{ViT}(I_i)\big)\big)\Big), \tag{4}$$

where p_i^0 and p_i^1 denote the predicted probabilities for the patch being assigned a high or low compression rate, respectively. The ViR is trained to match the ground-truth labels y_i^{router} using standard cross-entropy loss.

3.2.2 Pseudo-Label Supervision.

Training based on pseudo-label loss enables the router to intelligently assess patches based on model-perceived semantic complexity, rather than human-perceived visual complexity. Recognizing that multimodal model training often diverges from human visual intuition, models can exhibit idiosyncratic sensitivity to regions that appear visually simple to humans. By leveraging pseudo-labels generated according to the model's inherent sensitivity, this data-driven supervision correctly handles such counterintuitive cases. In this way, the pseudo-label training effectively adapts to model artifacts and intrinsic feature distributions. Evaluations across a broad spectrum of model scales and comprehensive benchmarks in Section 4 further validate the stability and robustness of this pseudo-labeling strategy.

3.2.3 Text-Independent Routing.

As defined in Eq. (4), our routing decision depends solely on visual features and ignores textual instructions. Given the prevalence of multi-turn dialogues in practical scenarios, this text-independent routing is deliberately chosen to avoid recalculating the visual token compression strategy for each user turn, which would otherwise disrupt and disable KV cache reuse. Since ViCO natively achieves nearly lossless performance (Section 4), the marginal accuracy gain from text-aware routing is negligible and not worth the significant trade-off in inference efficiency.

4 Experiments

ViCO is evaluated on InternVL3.5 across general, OCR, multi-image, and video benchmarks (Section 4.2). We conduct comparative (Section 4.3) and ablation (Section 4.4) studies with a primary focus on OCR tasks due to their token sensitivity. We further verify generalization on native-resolution Qwen2.5-VL (Section 4.5), investigate routing granularity (Section 4.6), and demonstrate inference speedup (Section 4.8).

4.1 Experimental Settings

Benchmarks. Our evaluation spans diverse benchmarks: OCR-related (OCRBench [32], ChartQA [33], TextVQA [34], InfoVQA [35], DocVQA [36], AI2D [37]), general (MMStar [38], POPE [39], MME [40], MMBench [41], RealWorldQA [42]), reasoning (MMMU [43]), multi-image (BLINK [44], MMT-Bench [45], MMIU [46]), and video understanding (Video-MME [47]).

Training Details. We conduct training on H200 GPUs in 2 stages. For consistency learning, we use a 32K text length and a batch size of up to 256, with AdamW optimizer and cosine decay (4×10^{-6} to 1×10^{-7}). In the router learning stage, only the router is unfrozen with a batch size of 8 and threshold τ set to the 60th percentile of scores.

4.2 Main Results

4.2.1 Results on General Benchmarks.

Table 1 compares InternVL3.5 models (4B–241B) with their ViCO counterparts across nine benchmarks. ViCO consistently retains over 99.6% of original performance on average across all scales. For instance, the 8B model with ViCO achieves nearly identical scores to its baseline on POPE (88.7 vs. 88.4) with about 81% patch compression. This robustness extends to the 241B-MoE model ($\sim$99.6% retention), and remains stable even in MMMU "thinking mode" with outputs up to 64K tokens. These results demonstrate that ViCO is highly effective and scalable from 4B to 241B-MoE.

Table 1: **Performance of ViCO on InternVL3.5 models across general benchmarks, including strongly OCR-related benchmarks.** MME is normalized to 0–100 for overall scoring. RWQA: RealWorldQA; MMB: MMBench. *Ratio** denotes the ratio of compressed patches

Model	OCRBench	ChartQA	TextVQA	MMStar	POPE	MME	MMB V1.1	RWQA	MMMU	Avg.
InternVL3.5-4B	82.2	86.4	77.8	65.2	88.9	2272	80.3	66.3	66.6	100.0%
+ ViCO	83.0	85.2	77.6	65.5	88.6	2239	80.5	66.5	65.9	99.7%
*Ratio**	*73%*	*62%*	*54%*	*64%*	*80%*	*76%*	*73%*	*76%*	*69%*	*69.7%*
InternVL3.5-8B	84.0	86.5	77.6	68.5	88.7	2380	79.5	67.5	73.4	100.0%
+ ViCO	83.9	86.7	77.8	67.5	88.4	2381	79.8	66.8	71.9	99.6%
*Ratio**	*71%*	*47%*	*58%*	*67%*	*81%*	*77%*	*75%*	*75%*	*71%*	*69.1%*
InternVL3.5-14B	83.2	86.1	77.3	67.7	87.7	2398	81.5	70.5	73.3	100.0%
+ ViCO	84.3	86.2	77.7	67.7	87.7	2392	81.0	71.0	73.2	100.2%
*Ratio**	*71%*	*44%*	*64%*	*66%*	*82%*	*76%*	*75%*	*79%*	*70%*	*69.7%*
InternVL3.5-30B-MoE	88.1	87.5	80.2	71.6	89.6	2461	84.8	72.3	75.6	100.0%
+ ViCO	87.7	87.4	79.6	70.6	89.7	2462	83.8	71.8	75.9	99.6%
*Ratio**	*71%*	*62%*	*55%*	*69%*	*82%*	*76%*	*75%*	*64%*	*74%*	*69.8%*
InternVL3.5-38B	88.0	88.9	82.8	71.6	90.4	2492	87.3	75.9	76.9	100.0%
+ ViCO	87.8	88.8	82.4	71.6	91.4	2496	86.8	74.8	75.0	99.6%
*Ratio**	*67%*	*32%*	*53%*	*56%*	*70%*	*70%*	*73%*	*73%*	*64%*	*62.0%*
InternVL3.5-241B-MoE	91.1	88.6	84.6	74.3	90.7	2525	87.4	75.2	77.7	100.0%
+ ViCO	90.7	88.3	84.2	73.7	90.5	2527	87.2	74.6	76.9	99.6%
*Ratio**	*73%*	*55%*	*55%*	*60%*	*55%*	*62%*	*65%*	*77%*	*78%*	*64.4%*

Table 2: **Performance of ViCO on InternVL3.5 models across multi-image and video benchmarks.** Video-MME is evaluated using 64-frame inputs with and without subtitles.

Model	BLINK	MMT-Bench	MMIU	Video-MME		Avg.
				no sub	with sub	
InternVL3.5-4B	59.5	64.3	49.2	65.4	68.6	100.0%
+ ViCO	58.3	64.2	49.1	65.1	68.1	99.2%
Ratio of Compressed Patches	*74%*	*62%*	*66%*	*47%*	*47%*	*59.2%*
InternVL3.5-8B	59.5	66.7	49.4	66.0	68.6	100.0%
+ ViCO	58.9	67.1	50.7	66.1	68.7	100.4%
Ratio of Compressed Patches	*74%*	*61%*	*56%*	*40%*	*40%*	*54.2%*
InternVL3.5-14B	57.6	68.0	51.3	67.9	68.6	100.0%
+ ViCO	59.2	67.9	50.5	67.9	68.7	100.3%
Ratio of Compressed Patches	*77%*	*63%*	*67%*	*55%*	*55%*	*63.4%*
InternVL3.5-30B-MoE	60.0	66.6	55.1	68.7	71.8	100.0%
+ ViCO	60.2	66.9	54.5	69.0	71.4	99.9%
Ratio of Compressed Patches	*72%*	*60%*	*56%*	*42%*	*42%*	*54.4%*
InternVL3.5-38B	60.9	71.8	58.9	70.9	74.2	100.0%
+ ViCO	62.2	71.5	59.6	71.3	74.0	100.6%
Ratio of Compressed Patches	*64%*	*54%*	*51%*	*63%*	*63%*	*59%*
InternVL3.5-241B-MoE	61.4	72.7	61.3	72.9	76.5	100.0%
+ ViCO	64.8	72.0	61.4	73.7	76.6	101.1%
Ratio of Compressed Patches	*61%*	*58%*	*55%*	*70%*	*70%*	*62.8%*

4.2.2 Results on OCR-related Benchmarks.

As shown in Table 1, the 8B model maintains near-original performance across OCR benchmarks, specifically achieving 83.9 on OCRBench (vs. 84.0) with a 71% compression rate, and negligible variations on ChartQA (86.7 vs. 86.5) and TextVQA (77.8 vs. 77.6), demonstrating the robustness of our approach. This performance retention stems from ViCO's adaptive routing, which aggressively compresses simpler patches via lighter modeling while preserving complex modeling for semantically critical information. Consequently, the model focuses on important regions without losing essential details, maintaining high fidelity performance despite substantial token compression.

4.2.3 Results on Multi-Image and Video Understanding Benchmarks.

As shown in Table 2, ViCO maintains strong performance despite substantial token reduction. On Video-MME (with subtitles), the 241B-MoE model achieves $\sim$70% compression while scoring 76.6,

Table 3: **Comparative study of ViCO and training-free methods across general and strongly OCR-dependent benchmarks.** Experiments are conducted on InternVL3.5-8B. For DocVQA and InfoVQA, the validation sets are used for performance evaluation.

Model	DocVQA	ChartQA	InfoVQA	TextVQA	OCRBench	AI2D	MMStar	BLINK	Avg.
Vanilla	91.3	86.5	79.1	77.6	84.0	84.0	68.5	59.5	100.0%
FastV	87.5	84.0	71.6	76.9	81.0	83.0	64.2	56.8	95.9%
SparseVLM	75.9	84.9	55.2	76.5	77.7	84.1	66.4	54.5	91.2%
ToFu	74.1	71.3	69.6	74.5	50.6	83.5	63.3	57.4	86.3%
ViCO	90.8	86.7	78.6	77.8	83.9	83.7	67.5	58.9	99.6%
Ratio of Compressed Patches	*60%*	*47%*	*21%*	*58%*	*71%*	*42%*	*67%*	*74%*	*55.0%*

Table 4: **Comparative study of ViCO and training-aware methods on InternVL3.5-2B and InternVL3.5-8B across general and strongly OCR-dependent benchmarks.** Efficiency metrics are converted to the ratio of compressed patches (*Ratio**).

Model	↑ Ratio*	OCRBench	ChartQA	TextVQA	MMStar	POPE	MMBench	↑ Avg.
InternVL3.5-2B	-	83.7	80.8	76.7	57.5	87.2	76.6	100.0%
+ LLaVA-Mini	-	72.1	75.2	73.6	52.9	84.7	73.4	93.4%
+ AVG-LLaVA	39.1%	80.0	78.4	74.6	57.3	87.3	76.4	98.2%
+ ViCO	62.2%	83.3	79.8	76.2	57.7	87.2	76.7	99.6%
InternVL3.5-8B	-	84.0	86.5	77.6	68.5	88.7	79.5	100.0%
+ LLaVA-Mini	-	72.3	79.5	74.5	60.6	85.7	77.1	92.8%
+ AVG-LLaVA	42.4%	81.1	80.8	74.8	67.2	87.0	76.9	96.5%
+ ViCO	66.5%	83.9	86.7	77.8	67.5	88.4	79.8	99.9%

slightly exceeding the baseline. Similarly, the 38B model compresses 63% of patches with near 100% retention, while smaller models consistently preserve over 99% of original scores. These results demonstrate that ViCO effectively accelerates long sequence processing through adaptive modeling. Thus, the model efficiently handles long visual sequences, ensuring robust performance across multi-image and video understanding.

4.3 Comparison with Existing Methods

As shown in Table 3, we first evaluate ViCO against representative training-free baselines, including FastV, SparseVLM, and ToFu on the InternVL3.5-8B backbone. For a fair comparison, we tune their hyperparameters to match ViCO's average compression ratio. Unlike our approach, these baselines rely on manually pre-defined hyperparameters and cannot adapt compression dynamically across tasks. Consequently, although their overall scores decrease only slightly, they exhibit substantial performance drops on vision-sensitive benchmarks such as OCR. In contrast, our method automatically assesses the semantic importance of tokens, allowing it to apply stronger compression on vision-insensitive tasks like BLINK while retaining more tokens on vision-critical benchmarks such as InfoVQA, thereby avoiding significant degradation in performance.

To further verify the effectiveness of our training algorithm, we compare ViCO with training-aware baselines, namely LLaVA-Mini and Avg-LLaVA. These experiments are conducted using the InternVL3.5-2B and InternVL3.5-8B models. As summarized in Table 4, while these methods were effective for the LLaVA series, they encounter severe performance bottlenecks on advanced models like InternVL3.5. Specifically, on InternVL3.5-8B, although LLaVA-Mini provides extreme compression (1 token), it suffers from a sharp performance decline (92.8% retention), especially in OCR tasks that require fine grained details. Avg-LLaVA maintains a higher accuracy (96.5% retention) but offers very limited token reduction. In contrast, ViCO significantly outperforms these approaches by achieving a near lossless 99.9% performance retention while compressing 66.5% of image patches. On InternVL3.5-2B, ViCO similarly retains 99.6% of baseline performance while compressing 62.2% of image patches. These results further verify ViCO's unique advantage in balancing practical inference efficiency with high fidelity multimodal reasoning.

Table 5: **Ablation study of ViCO across general and strongly OCR-dependent benchmarks.** Experiments are conducted on InternVL3.5-8B. For DocVQA and InfoVQA, the validation sets are used for performance evaluation.

Model	DocVQA	ChartQA	InfoVQA	TextVQA	OCRBench	AI2D	MMStar	BLINK	Avg.
Vanilla	91.3	86.5	79.1	77.6	84.0	84.0	68.5	59.5	100.0%
Vanilla + ViR	57.9	78.0	69.4	71.1	75.1	83.5	64.7	54.7	87.9%
Vanilla w/o dynamic res.	56.2	75.0	38.6	64.0	68.7	83.0	64.1	56.0	80.2%
ViCO (All Compress)	85.8	84.5	68.9	74.8	80.6	83.2	66.3	55.4	95.1%
ViCO (Random Compress)	88.8	85.9	73.3	76.8	83.0	83.3	66.6	57.4	97.6%
ViCO (Image-level ViR)	89.2	86.5	79.1	76.7	82.9	84.2	67.4	56.8	98.8%
Ratio of Compressed Patches	*69%*	*36%*	*12%*	*65%*	*78%*	*32%*	*75%*	*88%*	*56.9%*
ViCO	90.8	86.7	78.6	77.8	83.9	83.7	67.5	58.9	99.6%
Ratio of Compressed Patches	*60%*	*47%*	*21%*	*58%*	*71%*	*42%*	*67%*	*74%*	*55.0%*

Table 6: Sensitivity to the threshold percentile k. Each entry reports the ratio of compressed patches (%) / performance retention (%). The default setting is $k = 60$.

Model	$k = 40$	$k = 50$	$k = 60$	$k = 70$
InternVL3.5-8B	53.5 / 99.8	62.1 / 99.7	69.1 / 99.6	75.4 / 99.1
InternVL3.5-30B-MoE	54.2 / 99.8	62.8 / 99.7	69.8 / 99.6	76.1 / 99.2

4.4 Ablation Study

Settings. We conduct ablation experiments on the InternVL3.5-8B model. To evaluate the consistency training stage, we construct the Vanilla with ViR variant by applying the router decisions from the fully trained ViCO model to the original model. For the router training stage, we test three variants: (1) All Compression: all patches are compressed indiscriminately; (2) Random Routing: patches are routed randomly at the same compression ratios as the ViCO model; (3) Image-Level Routing: the router operates at the image level instead of the patch level. Additionally, we perform an experiment on the original InternVL3.5-8B model without dynamic resolution.

Effect of Consistency Training. As shown in Table 5, directly applying the ViCO router to the original model (Vanilla with ViR) causes a sharp performance drop (e.g., 84.0 to 75.1 on OCRBench), as InternVL3.5 cannot natively handle tokens interleaved at varying compression rates. Similarly, disabling dynamic resolution reduces tokens but severely hurts vision-sensitive tasks. These results emphasize the necessity of Consistency Training, which enables the model to process interleaved tokens at varying downsampling rates while maintaining strong performance.

Effect of Router Training. As shown in Table 5, indiscriminate compression and random routing both underperform the full ViCO. These results confirm that our router successfully learns to adjust compression based on semantic importance: patches with rich information are lightly compressed, while simpler ones are compressed more aggressively. This learned adaptive routing enables an optimal trade-off between inference efficiency and model performance.

Effect of Compression Granularity. As shown in Table 5, we compare patch-level routing with an image-level variant that decides compression based on global semantics. Image-level routing yields 98.8% retention, slightly below the 99.6% of patch-level routing. However, image-level decisions are overly coarse and sensitive to dominant content, causing drastic fluctuations in compression ratios across datasets (e.g., 12% on InfoVQA vs. 88% on BLINK). These results underscore the advantage of ViCO's patch-level routing in achieving more stable and fine grained semantic compression.

Sensitivity to the Threshold Percentile. We evaluate $k \in \{40, 50, 60, 70\}$ on InternVL3.5-8B and InternVL3.5-30B-MoE, with $k = 60$ as the default setting. As shown in Table 6, the model performance is highly robust and insensitive to the exact choice of k. While varying k influences the compression preference, every configuration consistently demonstrates the effectiveness of our method.

Table 7: **Generalization of ViCO to the native-resolution Qwen2.5-VL-3B.**

Model	OCRBench	ChartQA	TextVQA	MMStar	POPE	MMBench	↑ Avg.
Qwen2.5-VL-3B (ours)	78.1	82.5	77.2	61.0	87.3	79.8	100.0%
+ ViCO	78.2	82.0	76.9	60.3	86.9	79.7	99.6%
*Ratio**	*64%*	*43%*	*55%*	*64%*	*76%*	*67%*	*61.5%*

Table 8: **Granularity ablation on InternVL3.5-2B.** Binary and tri-level strategies are compared, with Ratio* standardized to the binary case.

Model	↑ Ratio*	OCRBench	ChartQA	TextVQA	MMStar	POPE	MMBench	↑ Avg.
$\{4\times, 16\times\}$	62.2%	83.3	79.8	76.2	57.7	87.2	76.7	99.6%
$\{4\times, 8\times, 16\times\}$	54.0%	83.6	80.1	76.1	57.7	87.1	76.5	99.7%

4.5 Generalization to Native-Resolution Architectures

To demonstrate the generality of ViCO, we extend our experiments to native-resolution models. While these architectures, such as Qwen2.5-VL [48], process images as variable length sequences, spatially adjacent tokens can still be effectively grouped into patches for compression. We validated this by applying ViCO to a Qwen2.5-VL-3B model trained on our data to ensure data consistency. As shown in Table 7, ViCO maintains over 99.5% of the baseline performance while achieving substantial token reduction. These results demonstrate ViCO's strong adaptability and robust generalization across different vision language modeling paradigms. Additional comparisons with training-free baselines on Qwen2.5-VL-7B are provided in Appendix G.

4.6 Analysis of Routing Granularity

Beyond binary routing, we evaluate a tri-level strategy using two thresholds τ to investigate finer grained compression. As shown in Table 8, this multi level approach yields only marginal gains relative to the increased system complexity. These diminishing returns confirm that our binary design optimally balances performance and simplicity. Additional results on InternVL3.5-8B show a similar trend and are provided in Appendix H.

4.7 Visualization of Image Routing

We visualize InternVL3.5-8B routing results using a 0.5 score threshold to distinguish between high and low compression. As shown in Fig. 3, the router effectively identifies semantic complexity: salient objects and informative text are assigned lower compression to maintain high resolution, while homogeneous backgrounds undergo higher compression. Importantly, higher compression indicates that the content can be adequately represented with fewer resources, rather than a lack of importance. This selective strategy focuses computation on semantically significant regions, enhancing efficiency without sacrificing critical details for the LLM.

4.8 Throughput Analysis

We deploy ViCO via LMDeploy [49] on 8B–241B models to evaluate practical inference. The LLM's *first-token throughput* is measured as relative speedup over 2000 concurrent requests, with each containing 10 visual patches and 64/512 text tokens. Tensor parallelism is 8 for the 241B model and 1 for others. As shown in Table 9, 50% compression consistently yields $> 1.75\times$ speedup, with shorter text inputs benefiting more due to higher visual token density. Furthermore, component level profiling confirms that our introduced modules incur minimal overhead, representing only $\sim$2% of total latency (see Appendix B for a detailed breakdown). These results demonstrate that our method not only reduces computational overhead theoretically but also yields significant practical acceleration during model inference, confirming the effectiveness of visual token compression for multimodal models.

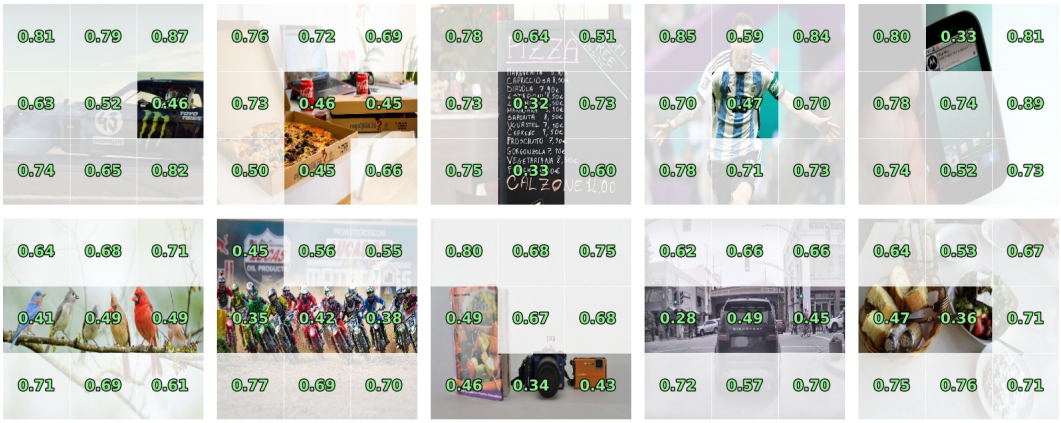


Figure 3: **Routing visualization for 8B model.** White/unshaded patches: 64/256 tokens. Annotated scores denote router confidence, where higher values indicate stronger compression preference.

Table 9: **LLM throughput speedup across compression rates.** Speedup factors (×) are reported relative to the original non-compressed baseline (Ori.).

Model	Input Type	LLM Throughput @ token compression			
		Ori.	25%	50%	75%
8B	10 patches + 64 text tokens	1.00×	1.29×	1.97×	3.76×
	10 patches + 512 text tokens	1.00×	1.30×	1.76×	2.77×
30B	10 patches + 64 text tokens	1.00×	1.30×	1.99×	3.77×
	10 patches + 512 text tokens	1.00×	1.29×	1.80×	2.72×
38B	10 patches + 64 text tokens	1.00×	1.33×	1.99×	3.80×
	10 patches + 512 text tokens	1.00×	1.30×	1.77×	2.76×
241B	10 patches + 64 text tokens	1.00×	1.34×	2.00×	3.76×
	10 patches + 512 text tokens	1.00×	1.32×	1.81×	2.87×

5 Conclusion

In this work, we propose Visual Consistency Learning (ViCO), which enables the model to represent images of varying semantic complexities using different numbers of vision tokens. Unlike resolution-based strategies, ViCO dynamically adapts the token count according to semantic complexity. Experiments show our method reduces tokens by up to 50% while preserving perception, reasoning, and OCR performance. Additional ablation studies further validate the effectiveness of each proposed module. By making decisions based on visual semantics, ViCO enables the model to efficiently focus computation on the most informative regions of an image, providing insights for future research on adaptive visual representations. We hope ViCO facilitates more efficient MLLMs by reducing inference-time computational costs.

Acknowledgments

This work is supported by the National Key R&D Program of China (No. 2022ZD0161302) and the Fundamental Research Funds for the Central Universities (2232026D60). The authors declare no competing interests.

References

[1] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

[2] 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.

[3] LLM-Core-Team Xiaomi. Mimo-vl technical report, 2025. URL `https://arxiv.org/abs/2506.03569`.

[4] Team Kwai Keye, Biao Yang, Bin Wen, Changyi Liu, Chenglong Chu, Chengru Song, Chongling Rao, Chuan Yi, Da Li, Dunju Zang, et al. Kwai keye-vl technical report. *arXiv preprint arXiv:2507.01949*, 2025.

[5] Anthropic. Introducing claude 4: Claude sonnet 4 and claude opus 4, May 2025. URL `https://www.anthropic.com/news/claude-4`.

[6] DeepMind. Gemini 2.5 pro. `https://deepmind.google/technologies/gemini/pro/`, 2025.

[7] Weiyun Wang, Zhe Chen, Wenhai Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Jinguo Zhu, Xizhou Zhu, Lewei Lu, Yu Qiao, and Jifeng Dai. Enhancing the reasoning ability of multimodal large language models via mixed preference optimization. *arXiv preprint arXiv:2411.10442*, 2024.

[8] Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

[9] Jiabo Ye, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Guohai Xu, Chenliang Li, Junfeng Tian, Qi Qian, Ji Zhang, et al. Ureader: Universal ocr-free visually-situated language understanding with multimodal large language model. *arXiv preprint arXiv:2310.05126*, 2023.

[10] 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.

[11] 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.

[12] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[13] OpenAI. Introducing gpt-oss. https://openai.com/index/introducing-gpt-oss/, 2025.

[14] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *NeurIPS 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL `http://papers.nips.cc/paper_files/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html`.

[15] Weiyun Wang, Shuibo Zhang, Yiming Ren, Yuchen Duan, Tiantong Li, Shuo Liu, Mengkang Hu, Zhe Chen, Kaipeng Zhang, Lewei Lu, Xizhou Zhu, Ping Luo, Yu Qiao, Jifeng Dai, Wenqi Shao, and Wenhai Wang. Needle in a multimodal haystack. *arXiv preprint arXiv:2406.07230*, 2024.

[16] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in Neural Information Processing Systems*, 36, 2023.

[17] Zhaoyang Liu, Yinan He, Wenhai Wang, Weiyun Wang, Yi Wang, Shoufa Chen, Qinglong Zhang, Zeqiang Lai, Yang Yang, Qingyun Li, Jiashuo Yu, et al. Interngpt: Solving vision-centric tasks by interacting with chatgpt beyond language. *arXiv preprint arXiv:2305.05662*, 2023.

[18] Weiyun Wang, Zhangwei Gao, Lianjie Chen, Zhe Chen, Jinguo Zhu, Xiangyu Zhao, Yangzhou Liu, Yue Cao, Shenglong Ye, Xizhou Zhu, et al. Visualprm: An effective process reward model for multimodal reasoning. *arXiv preprint arXiv:2503.10291*, 2025.

[19] 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.

[20] 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.

[21] Gen Luo, Wenhan Dou, Wenhao Li, Zhaokai Wang, Xue Yang, Changyao Tian, Hao Li, Weiyun Wang, Wenhai Wang, Xizhou Zhu, Yu Qiao, and Jifeng Dai. Mono-internvl-1.5: Towards cheaper and faster monolithic multimodal large language models. *arXiv preprint arXiv:2507.12566*, 2025.

[22] Gen Luo, Xue Yang, Wenhan Dou, Zhaokai Wang, Jifeng Dai, Yu Qiao, and Xizhou Zhu. Mono-internvl: Pushing the boundaries of monolithic multimodal large language models with endogenous visual pre-training. *arXiv preprint arXiv:2410.08202*, 2024.

[23] Yanwei Li, Chengyao Wang, and Jiaya Jia. Llama-vid: An image is worth 2 tokens in large language models. In *European Conference on Computer Vision*, pages 323–340. Springer, 2024.

[24] Linli Yao, Lei Li, Shuhuai Ren, Lean Wang, Yuanxin Liu, Xu Sun, and Lu Hou. Deco: Decoupling token compression from semantic abstraction in multimodal large language models, 2024. URL `https://arxiv.org/abs/2405.20985`.

[25] Jianjian Cao, Peng Ye, Shengze Li, Chong Yu, Yansong Tang, Jiwen Lu, and Tao Chen. Madtp: Multimodal alignment-guided dynamic token pruning for accelerating vision-language transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15710–15719, 2024.

[26] Zhibin Lan, Liqiang Niu, Fandong Meng, Wenbo Li, Jie Zhou, and Jinsong Su. Avg-llava: An efficient large multimodal model with adaptive visual granularity. *arXiv preprint arXiv:2410.02745*, 2024.

[27] Shaolei Zhang, Qingkai Fang, Zhe Yang, and Yang Feng. Llava-mini: Efficient image and video large multimodal models with one vision token. *arXiv preprint arXiv:2501.03895*, 2025.

[28] 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, 2024. URL `https://arxiv.org/abs/2403.06764`.

[29] Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, et al. Sparsevlm: Visual token sparsification for efficient vision-language model inference. *arXiv preprint arXiv:2410.04417*, 2024.

[30] 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.

[31] Vittorio Pippi, Matthieu Guillaumin, Silvia Cascianelli, Rita Cucchiara, Maximilian Jaritz, and Loris Bazzani. Tofu: Visual tokens reduction via fusion for multi-modal, multi-patch, multi-image task. *arXiv preprint arXiv:2503.04444*, 2025.

[32] Yuliang Liu, Zhang Li, Hongliang Li, Wenwen Yu, Mingxin Huang, Dezhi Peng, Mingyu Liu, Mingrui Chen, Chunyuan Li, Lianwen Jin, et al. On the hidden mystery of ocr in large multimodal models. *arXiv preprint arXiv:2305.07895*, 2023.

[33] Ahmed Masry, Xuan Long Do, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 2263–2279, 2022.

[34] 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.

[35] 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.

[36] 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.

[37] Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *European Conference on Computer Vision*, pages 235–251, 2016.

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

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

[40] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui Yang, Xiawu Zheng, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023.

[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? *arXiv preprint arXiv:2307.06281*, 2023.

[42] X.AI Corp. Grok-1.5 vision preview: Connecting the digital and physical worlds with our first multimodal model. `https://x.ai/blog/grok-1.5v`, 2024.

[43] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. *arXiv preprint arXiv:2311.16502*, 2023.

[44] Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A Smith, Wei-Chiu Ma, and Ranjay Krishna. Blink: Multimodal large language models can see but not perceive. *arXiv preprint arXiv:2404.12390*, 2024.

[45] Kaining Ying, Fanqing Meng, Jin Wang, Zhiqian Li, Han Lin, Yue Yang, Hao Zhang, Wenbo Zhang, Yuqi Lin, Shuo Liu, Jiayi Lei, Quanfeng Lu, Runjian Chen, Peng Xu, Renrui Zhang, Haozhe Zhang, Peng Gao, Yali Wang, Yu Qiao, Ping Luo, Kaipeng Zhang, and Wenqi Shao. Mmt-bench: A comprehensive multimodal benchmark for evaluating large vision-language models towards multitask agi. *arXiv preprint arXiv:2404.16006*, 2024.

[46] Fanqing Meng, Jin Wang, Chuanhao Li, Quanfeng Lu, Hao Tian, Jiaqi Liao, Xizhou Zhu, Jifeng Dai, Yu Qiao, Ping Luo, et al. Mmiu: Multimodal multi-image understanding for evaluating large vision-language models. *arXiv preprint arXiv:2408.02718*, 2024.

[47] Chaoyou Fu, Yuhan Dai, Yondong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. *arXiv preprint arXiv:2405.21075*, 2024.

[48] 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.

[49] LMDeploy Contributors. Lmdeploy: A toolkit for compressing, deploying, and serving llm. `https://github.com/InternLM/lmdeploy`, 2023.

Supplementary Material

A Performance of ViCO on small-scale models

We also evaluate ViCO on smaller InternVL3.5 models 1B and 2B across a range of benchmarks, including general multimodal, OCR-related, multi-image, and video benchmarks. Tables 10 and 11 show that the models retain nearly all of their original performance, demonstrating that our approach remains effective even on smaller-scale models.

Table 10: **Performance of ViCO on InternVL3.5 small-scale models across general benchmarks.** MME is normalized from 0–2800 to 0–100. Abbreviations: OCR = OCRBench, Chart = ChartQA, Text = TextVQA, MMB = MMBench, RWQA = RealWorldQA, Ratio = Ratio of Compressed Patches.

Model	OCR	Chart	Text	MMStar	POPE	MME	MMB V1.1	RWQA	Avg.
InternVL3.5-1B	79.2	78.0	71.2	50.6	86.8	1910.2	69.9	57.6	100.0%
+ ViCO	78.8	77.4	71.1	50.8	86.2	1905.8	69.1	56.9	99.4%
Ratio	*70%*	*40%*	*56%*	*60%*	*80%*	*73%*	*72%*	*75%*	*65.8%*
InternVL3.5-2B	83.7	80.8	76.7	57.5	87.2	2123.3	76.6	62.0	100.0%
+ ViCO	83.3	79.8	76.2	57.7	87.2	2101.0	76.7	60.7	99.4%
Ratio	*68%*	*36%*	*60%*	*61%*	*77%*	*73%*	*71%*	*66%*	*64.0%*

Table 11: **Performance of ViCO on InternVL3.5 small-scale models across multi-image and video benchmarks.** For Video-MME, both the subtitle and non-subtitle settings are evaluated using 64-frame inputs. Abbreviations: MMT = MMT-Bench, Ratio = Ratio of Compressed Patches.

Model	**BLINK**	**MMT**	**MMIU**	**Video-MME**		**Avg.**
				no sub	**with sub**	
InternVL3.5-1B	44.0	54.5	45.2	52.4	55.0	100.0%
+ ViCO	43.9	54.3	43.9	52.9	54.9	99.5%
Ratio	*69%*	*57%*	*56%*	*63%*	*63%*	*61.6%*
InternVL3.5-2B	51.3	58.5	44.9	58.3	61.2	100.0%
+ ViCO	51.3	58.7	45.5	59.1	61.4	100.6%
Ratio	*69%*	*58%*	*48%*	*73%*	*73%*	*64.2%*

B Discussion on Additional Inference Latency

In this work, we introduce a new Router structure. Specifically, for models with 1B to 30B parameters, the Router has approximately 50M parameters, while for 38B and 241B models, it has around 490M parameters, which is very small relative to the overall model size. Additionally, the Attention Pooling module scales linearly with sequence length. Therefore, the theoretical computational overhead introduced by these additional structures is minimal.

To provide empirical evidence, we profile the component-level latency using an InternVL3.5-2B model on an NVIDIA H200 GPU. As summarized in Table 12, the combined execution time of the routing gate and MLP adapter represents only a negligible fraction ($\sim$2%) of total latency compared to the ViT and LLM backbone. These results confirm that our adaptive routing mechanism is computationally efficient and translates theoretical minimal overhead into practical inference efficiency.

C End-to-End Inference Speedup on Benchmark Workloads

To reflect the actual efficiency gains in real scenarios, we present the real end-to-end inference speedups of ViCO, measured in the same environment as described in the paper, across four model scales (8B, 30B-MoE, 38B, 241B-MoE) on various benchmarks:

Specifically, ViCO achieves consistent $1.72\times$ to $2.53\times$ end-to-end speedups across all model scales and benchmarks, with an average speedup exceeding $2.1\times$. These benchmark-specific real-world speedup metrics confirm that ViCO achieves consistent and substantial inference acceleration across model sizes in real deployment scenarios.

Table 12: **Latency breakdown of InternVL3.5-2B across components.**

Stage	ViT	Gate	MLP	LLM
Time (s)	0.0378	0.0037	0.0015	0.1826

Table 13: **End-to-end inference speedups of ViCO across model scales and benchmarks.** Speedups are measured relative to the corresponding uncompressed baselines. OCR: OCRBench; MMB: MMBench V1.1; RWQA: RealWorldQA. All values are speedup factors ($\times$).

Model	OCR	POPE	MME	MMB	RWQA	MMMU	BLINK
InternVL3.5-8B	2.17	2.51	2.38	2.31	2.30	2.15	2.26
InternVL3.5-30B-MoE	2.12	2.53	2.29	2.28	1.95	2.25	2.17
InternVL3.5-38B	2.07	2.12	2.17	2.28	2.22	1.96	1.91
InternVL3.5-241B-MoE	2.19	1.72	1.88	1.90	2.33	2.44	1.90

D Broader Impact

ViCO improves MLLM inference efficiency by reducing visual tokens, which contributes to the goals of Green AI through lower energy consumption. Additionally, it facilitates the deployment of advanced multimodal models on resource-constrained edge devices, enhancing accessibility without foreseeable negative societal consequences.

E Limitations

Unlike training-free methods, ViCO requires a one-time training overhead. This cost is a deliberate trade-off: since many practitioners are unwilling to sacrifice even 1% of performance or deployment compatibility for speed, we prioritize a solution that achieves virtually lossless acceleration. After this fixed training investment, ViCO provides long-term, high-fidelity efficiency gains during the entire inference lifecycle.

F Training Details and Computational Cost

Stage 1 (Consistency Training). Uses a 32k context length with sequence packing ($\sim$20 samples per sequence) on 32 GPUs. Trained for 6k steps (totaling $\sim$4M samples, comparable to $\sim$3M in LLaVA-Mini) using the AdamW optimizer with a learning rate of 4×10^{-6} and cosine decay down to 1×10^{-7} (taking $\sim$5 hours for the 8B model).

Stage 2 (Router Training). Uses 128k samples on 8 GPUs. Trained for 800 steps (batch size of 8, $\sim$20 packed samples per batch) using the AdamW optimizer with a learning rate of 1×10^{-4} and cosine decay down to 1×10^{-6} (taking < 1 hour for the 8B model).

Ultra-Large Models Scaling. For 38B and 241B-MoE, compute resources were scaled by $8\times$ with 256 GPUs allocated for Consistency Training. This resource scaling is not caused by the complexity of our algorithm, but is dictated by the distributed infrastructure required for ultra-large models. Crucially, our experimental results demonstrate that ViCO effectively scales to extreme parameter levels.

Sliding Window Size and Update Frequency. The sliding window size is set to 400 patch loss values and is updated at every step. We conducted experiments by changing the window size to 200, 300 and 800, and found that it has virtually no impact on the final training results. The sliding window is implemented using a simple, efficient FIFO data structure, which introduces zero overhead on training speed.

Controlled Baseline Comparison Statement. In Table 4, all baseline methods were trained on the exact same 4M dataset rather than their original reported data volumes (e.g., 3M in LLaVA-Mini) to maintain a strict controlled comparison under identical data scale.

Table 14: **Generalization of ViCO to the native-resolution Qwen2.5-VL-7B and comparison with training-free methods.**

Model	OCRBench	ChartQA	TextVQA	MMStar	POPE	MMBench	↑ Avg.
Qwen2.5-VL-7B	86.3	86.1	80.2	62.5	86.4	80.2	100.0%
+ FastV	75.4	83.1	79.6	61.2	86.3	80.2	96.70%
+ SparseVLM	74.0	60.7	72.2	57.5	85.3	79.7	89.15%
+ ToFu	74.8	79.9	75.1	59.2	85.1	79.3	94.13%
+ ViCO	86.1	86.0	80.5	62.5	85.9	79.8	99.81%
*Ratio**	*67%*	*44%*	*52%*	*69%*	*81%*	*67%*	*63.3%*

Table 15: **Granularity ablation on InternVL3.5-8B.** Binary and tri-level strategies are compared, with Ratio* standardized to the binary case.

Model	↑ Ratio*	OCRBench	ChartQA	TextVQA	MMStar	POPE	MMBench	↑ Avg.
$\{4\times, 16\times\}$	66.5%	83.9	86.7	77.8	67.5	88.4	79.8	99.86%
$\{4\times, 8\times, 16\times\}$	61.5%	83.9	86.2	77.6	68.3	88.6	80.0	99.96%

G Additional Results on Qwen2.5-VL-7B

Following the methodology in Section 4.5, we extend our evaluation to Qwen2.5-VL-7B and compare ViCO with representative training-free baselines, including FastV, SparseVLM, and ToFu. For a fair comparison, we tune their hyperparameters to match ViCO's overall average compression ratio. As shown in Table 14, ViCO retains 99.81% of baseline performance, outperforming all three baselines. These results further demonstrate ViCO's generalization to larger native-resolution models.

H Compression Granularity on InternVL3.5-8B

Following the same methodology as the main-text granularity ablation, we extend the comparison between binary and tri-level compression to InternVL3.5-8B.

As shown in Table 15, introducing an intermediate compression tier increases performance retention from 99.86% to 99.96%, while reducing Ratio* from 66.5% to 61.5%. These results indicate that the additional tier provides only a small improvement in performance retention, accompanied by lower compression efficiency. The diminishing returns observed on the 2B model therefore also hold at the 8B scale, supporting our choice of binary routing.

I Visualization of Router

As shown in Figures 4 and 5, we visualize the Visual Resolution Router from the InternVL3.5-8B ViCO model, demonstrating additional routing results across different patch layouts.



Figure 4: **Routing results of the InternVL3.5-8B ViCO model on images split into 3×3 patches.** Grey-shaded patches are compressed to 64 tokens, while unshaded patches retain 256 tokens. Each patch is annotated with the router's confidence score, where higher values indicate a stronger preference for compression.



Figure 5: **Routing results of the InternVL3.5-8B ViCO model on images split into 3×4 patches.** Grey-shaded patches are compressed to 64 tokens, while unshaded patches retain 256 tokens. Each patch is annotated with the router's confidence score, where higher values indicate a stronger preference for compression.

NeurIPS Paper Checklist

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 accurately reflect the paper's contributions and scope, which are empirically validated in Section 4, including the verification of generalization.

   Guidelines:

   - The answer [N/A] 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 [N/A] 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: We discuss the limitations in Section E.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [Yes]

Justification: The methodology in Section 3 is empirically validated by experiments in Section 4.

Guidelines:

- The answer [N/A] 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: Detailed implementation and architecture are provided in Sections 3 and 4.

   Guidelines:

   - The answer [N/A] 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: [No]

Justification: The data cannot be released due to proprietary constraints. The code will be released publicly.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.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://neurips.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) necessary to understand the results?

   Answer: [Yes]

   Justification: Training details are provided alongside their respective results in Sections 4, F, and other relevant sections.

   Guidelines:

   - The answer [N/A] 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: We report single-run results due to high computational costs, following standard practice in MLLM research.

   Guidelines:

   - The answer [N/A] 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: Details are provided in Sections 4, F.

   Guidelines:

   - The answer [N/A] 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: This research conforms to the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: We discuss societal impacts in Sections D.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] 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., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] 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: All external assets are properly credited and used under their respective licenses.

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: This work focuses on a training method. But the trained model will be released for academic research purposes.

    Guidelines:

    - The answer [N/A] 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: [N/A]

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

    Guidelines:

    - The answer [N/A] 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: [N/A]

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

    Guidelines:

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

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

16. **Declaration of LLM usage**

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

    Answer: [N/A]

    Justification: LLM is used only for writing.

    Guidelines:

    - The answer [N/A] 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 in the NeurIPS handbook for what should or should not be described.