Skip-Vision: Efficient and Scalable Acceleration of Vision-Language Models via Adaptive Token Skipping

Supplementary Material

9. Proof and detailed analysis of Skip-Vision

In this section, we provide a theoretical analysis to justify the rationale behind Skip-Vision and quantify the performance loss introduced by skipping FFN computations for redundant visual tokens.

9.1. Bounded error of Skip FFN

At the core of this analysis lies the layer error incurred when bypassing the FFN layer. For a given layer l, the original output $h^{(l)}_{\text{original}}$ is :

$$h^{(l)}_{\text{original}} = h^{(l)}_{\text{attn}} + \text{FFN}^{(l)}(h^{(l)}_{\text{attn}}). \tag{14}$$

When skipping the FFN, the output becomes:

$$h^{(l)}_{\text{skip}} = h^{(l)}_{\text{attn}}. \tag{15}$$

The per-layer skipping error is:

$$\epsilon^{(l)} = \|h^{(l)}_{\text{original}} - h^{(l)}_{\text{skip}}\|_2 = \|\text{FFN}^{(l)}(h^{(l)}_{\text{attn}})\|_2. \tag{16}$$

For redundant tokens, such as those in homogeneous image regions, this error is negligible ($\epsilon^{(l)} \approx 0$) due to minimal feature transformations by the FFN.

However, errors propagate through subsequent layers, amplified by the recursive nature of transformer architectures. Leveraging Lipschitz continuity assumptions for self-attention and FFN operations ($L^{(l+1)}_{attn}$ and $L^{(l+1)}_{FFN}$), the cumulative error at layer $l+1$ is bounded by:

$$\epsilon^{(l+1)} \leq (L^{(l+1)}_{\text{attn}} + L^{(l+1)}_{\text{FFN}}) \cdot \epsilon^{(l)} + \epsilon^{(l+1)}_{\text{skip}}, \tag{17}$$

where ($\epsilon^{(l+1)}_{\text{skip}}$ represents new errors from skipping deeper layers ($l+1$).

Over L layers, the total error telescopes to:

$$\epsilon_{\text{total}} \leq \sum_{l=1}^{L} \epsilon^{(l)}_{\text{skip}} \cdot \prod_{i=1}^{L-l} (L^{(i+1)}_{\text{attn}} + L^{(i+1)}_{\text{FFN}}). \tag{18}$$

Theorem 9.1 ***Lipschitz Constants for Causal Attention and FFN in Transformers*** *Assume:*

1. *Inputs are normalized (e.g., via LayerNorm), bounding intermediate feature norms.*
2. *Weight matrices in attention (W_Q, W_K, W_V) and FFN (W_1, W_2) have bounded spectral norms (maximum singular values).*

Then the Lipschitz constants of these components satisfy:

1. *Causal Attention:*

$$L(Attn) \leq \frac{\|W_Q\|_2\|W_K\|_2\|W_V\|_2}{\sqrt{d_k}}, \tag{19}$$

where d_k is the key dimension. The causal mask further restricts attention dependencies, preserving this bound.

2. *Feed - Forward Network (FFN):*

$$L(FFN) \leq \|W_1\|_2\|W_2\|_2, \tag{20}$$

assuming the activation function (e.g., ReLU, GELU) is 1-Lipschitz.

Proof.

The Lipschitz constant of causal attention

1.Linear Transformation:

The input sequence $X \in \mathbb{R}^{n\times d}$ undergoes three linear transformations to obtain the query $Q = XW_Q$, the key $K = XW_K$, and the value $V = XW_V$. The Lipschitz constant of each linear transformation is the spectral norm (the largest singular value) of its weight matrix, denoted as $\sigma_Q = \|W_Q\|_2$, $\sigma_K = \|W_K\|_2$, and $\sigma_V = \|W_V\|_2$ respectively.

2.Attention Score Calculation:

The scaled dot - product $S = QK^\top/\sqrt{d_k}$. The Lipschitz constant of the bilinear mapping is related to σ_Q and σ_K. If the norm of the input X is bounded (for example, through LayerNorm), then:

$$\text{Lip}(S) \leq \frac{\sigma_Q\sigma_K}{\sqrt{d_k}}. \tag{21}$$

3. Softmax Activation:

After applying the causal mask, softmax is performed on each row. The Lipschitz constant of Softmax under the ℓ_2 norm is less than 1, that is:

$$\text{Lip}(\text{softmax}) \leq 1. \tag{22}$$

4. Weighted Sum of Values: The output $\text{Attn}(X) = AV$, where $A = \text{softmax}(S)$. The Lipschitz constant of this step is determined by the spectral norm σ_V of the linear transformation of V.

Overall Lipschitz Constant

Combining the upper bounds of each step:

$$\text{Lip}(\text{CausalAttention}) \leq \frac{\sigma_Q\sigma_K\sigma_V}{\sqrt{d_k}}. \tag{23}$$

Impact of Causal Mask: The mask restricts the attention range, which may reduce the sensitivity to the input.

Therefore, the actual Lipschitz constant will not exceed the above-mentioned upper bound.

The Lipschitz Constant of FFN

The FFN is usually expressed as:

$$\text{FFN}(x) = W_2 \cdot \text{Activation}(W_1 x + b_1) + b_2, \tag{24}$$

where the Lipschitz constant of activation functions (such as ReLU, GELU) is 1.

Derivation of Lipschitz Constant

1. Linear Layer W_1: The spectral norm is $\sigma_1 = \|W_1\|_2$.
2. Activation Function: $\text{Lip}(\text{Activation}) = 1$.
3. Linear Layer W_2: The spectral norm is $\sigma_2 = \|W_2\|_2$.

Follow the equation 7 in [103], the overall Lipschitz constant is the product of the spectral norms of the two linear layers:

$$\text{Lip}(\text{FFN}) \leq \sigma_1 \sigma_2. \tag{25}$$

□

Corollary 9.1 ***Bounded Lipschitz*** *If W_Q, W_K, W_V, W_1, W_2 are orthogonal matrices (spectral norm = 1), then:*

- *$L(attn) \leq 1/\sqrt{d_k}$*
- *$L(FFN) \leq 1$*

If we assume Lipschitz constants $L_{\text{attn}} + L_{\text{FFN}} \leq \gamma$ and skipping errors $\epsilon_{\text{skip}}^{(l)} \leq \epsilon$, the total error is scaled to:

$$\epsilon_{\text{total}} \leq \epsilon \cdot \frac{\gamma^L - 1}{\gamma - 1}, \tag{26}$$

Theorem 5.1 establish that the skip error is bounded when $\gamma < 1$, provided the model is trained with modern regularization techniques. This ensures that the Multimodal Large Language Model (MLLM) remains less sensitive to the effects of skipping.

This error also impacts the KL divergence between the original and skipped outputs, bounded by:

$$\mathcal{D}_{\text{KL}}(p_{\text{skip}} \parallel p_{\text{original}}) \leq \frac{1}{2\sigma^2} \cdot \epsilon_{\text{total}}^2, \tag{27}$$

where σ^2 is the variance of the logits.

Proof. For two Gaussian distributions $p = \mathcal{N}(\mu_p, \Sigma_p)$ and $q = \mathcal{N}(\mu_q, \Sigma_q)$, their KL divergence is:

$$\begin{aligned} &\mathcal{D}_{\text{KL}}(p\|q) \\ &= \frac{1}{2}(\text{tr}(\Sigma_q^{-1}\Sigma_p) + (\mu_q - \mu_p)^T \Sigma_q^{-1}(\mu_q - \mu_p) \\ &- k + \ln \frac{|\Sigma_q|}{|\Sigma_p|}) \end{aligned} \tag{28}$$

where k is the dimension. If we assume that the covariances of the two distributions are the same, i.e., $\Sigma_p = \Sigma_q = \sigma^2 I$, and the total difference in means is ϵ_{total}, then:

$$\mathcal{D}_{\text{KL}}(p\|q) = \frac{1}{2\sigma^2}\|\mu_p - \mu_q\|^2. \tag{29}$$

Here, $\|\mu_p - \mu_q\|^2$ is $\epsilon_{\text{total}}^2$, so:

$$\mathcal{D}_{\text{KL}}(p\|q) \leq \frac{1}{2\sigma^2}\epsilon_{\text{total}}^2. \tag{30}$$

□

Further integrating feature similarity errors ($\epsilon_{\text{sim}} = O(\sqrt{1-\theta})$) from low-attention tokens, the final bound becomes:

$$\mathcal{D}_{\text{KL}} \leq \frac{1}{2\sigma^2} \cdot \left(\epsilon_{\text{total}} + \epsilon_{\text{sim}}\right)^2. \tag{31}$$

Practically, this analysis motivates a layer-wise skipping strategy, alongside token selection and token merge based on feature similarity (θ).

10. More experimental results

10.1. Efficiency.

Following the LLaVA-1.5-7B training setup, we conducted additional comparisons between Skip-Vision and several recent works, as shown in the table 5. MMVet, MMStar and MMBench highlight Skip-Vision's strength in **capturing causal and global information**. These benchmarks emphasize high-level reasoning and abstraction, which benefit from Skip-Vision's ability to **preserve essential information flow** while reducing redundant computations. By skipping FFN and KV-cache for less informative tokens, the model **amplifies signal from key visual cues** and enhances **causal token interactions**. While this comes with a **slight trade-off** in fine-grained tasks (OCR, Textvqa), it reflects a deliberate balance between perception and reasoning, favoring tasks that rely on semantic integration over detail fidelity.

Method	GQA	MMB	VQA^{Text}	MMVet	Avg.
Vanilla (576 tokens)	61.9	64.7	58.2	31.1	100%
SparseVLM [130] (64 tokens)	52.7	56.2	51.8	23.3	85.2%
VisionZip [115] (64 tokens)	55.1	60.1	55.5	31.7	93.7%
PDrop [111] (64 tokens)	47.5	58.8	50.6	-	-
FasterVLM [127] (58 tokens)	54.9	60.6	55.3	30.1	93.1%
LLaVA-PruMerge [91] (32 tokens)	-	60.9	56.0	-	-
Skip-Vision ($N_r = 64, N_s = 156$)	**60.8**	**65.1**	**57.4**	**32.5**	**100.0%**

Table 5. Comparison with more methods

Under the cos setting, we conduct more experiments to evaluate the training and inference efficiency. We compare with methods: FastV [22], Victor [108] and mean average pool, fine-tuning on LLaVA-665k using 8 NVIDIA A100 GPUs. As shown in Figure 11, our architecture outperforms in both metrics. Compared to the CoS_{1296} baseline, it achieves comparable performance with 35% less training time and 74% reduced inference computation. FastV, unable to utilize flash attention, shows a significant disadvantage, even surpassing baseline training time.

10.2. Ablation study

To validate the effectiveness of each component within the Skip-Vision framework, under CoS setting, we conducted

	SF	FS	LS	Merge	LV	SK	MME	Textvqa	MMB	MMVet	MMMU	MathV	OCRB	MMStar	Overall
0	CoS_{1296}						1585	64.4	77.1	39.4	39.2	21.5	39.2	41.2	46
1				✓			1548	64.5	75.3	39.7	39.1	21.8	37.7	40.9	45.6
2	✓						1589	63.6	74.4	36.9	38	19.8	365	39.8	44.2
3	✓				✓		1591	63.7	74.9	39.4	38.7	20.7	36.3	39.9	44.8
4	✓		✓				1560	63.8	75.5	39.0	39.2	21	37.1	40.0	45.0
5	✓		✓		✓		1580	63.5	74.2	40.6	39.2	21.6	37.2	40.9	45.3
6	✓	✓	✓				1570	63.5	74.1	40	39.8	20.1	39	41.4	45.4
7	✓		✓	✓			1593	62.6	74.7	40.3	40.2	21.6	36.6	41.1	45.3
8	✓	✓	✓	✓			1571	63.6	75.9	40.3	40.3	21	36.1	41.5	45.6
9	✓		✓	✓	✓		1547	64.0	75.9	38	41.2	21.9	36.9	40.6	45.5
10	✓	✓	✓	✓	✓		1562	63.9	76.5	40.2	40.2	21.2	37.2	41.9	45.9
11	✓	✓	✓	✓	✓	✓	1563	63.7	76.5	41.7	40.3	21.2	37.0	41.9	46

Table 6. **Ablation study.** To establish a strong baseline, we performed an ablation study on each component of the Skip-Vision framework with the LLAVA 665k SFT dataset. SF (skip FFN), FS (former summary token), LS (latter summary token), Merge (reducing local visual tokens from 1024 to 256), LV (passing the last local visual token through the FFN), SK (using skip KV-cache during inference). This analysis highlights the distinct contributions of each element to efficiency and performance.

Inference method	Skip window size	MME	Textvqa	MMB	MMVet	MMMU	MathV	OCRB	MMStar	Overall
Without skip KV-cache	-	1562	**63.9**	**76.5**	40.2	40.2	**21.2**	**37.2**	**41.9**	45.9
Skip KV-cache	middle+small	1562	61.8	76.5	32.6	**40.4**	**21.2**	22.6	**41.9**	42.4
Skip KV-cache	small	**1563**	63.7	**76.5**	**41.7**	40.3	**21.2**	37.0	**41.9**	**46**

Table 7. **Ablation study of skip KV-cache.**We report skip KV-cache performance across different visual token window sizes. Skip-Vision enables task-specific optimization by adjusting skip KV-cache levels for tailored acceleration.

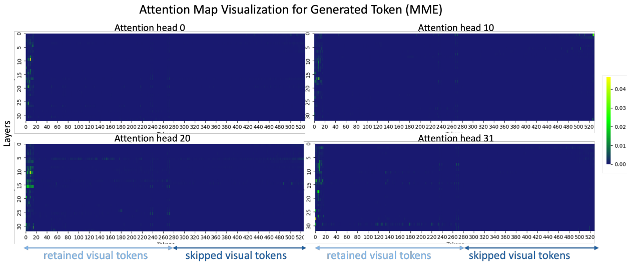

Figure 9. **Visualization of attention map in MME.**

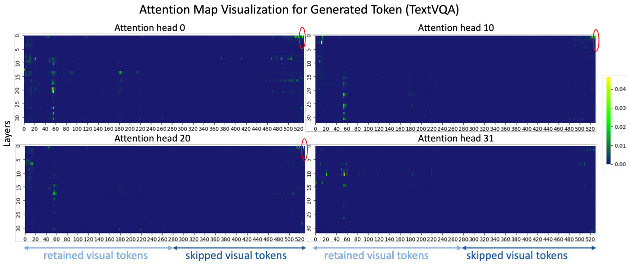

Figure 10. **Visualization of attention map in TextVQA.**

ablation and comparative experiments on the skip FFN, summary token, token merge, last visual token, and skip KV-cache. The detailed experimental results are presented in the table 6.

The last summary token or final visual token must pass through the FFN. As discussed in Section 4.2, the final visual token plays a crucial role in predicting the subsequent text token, thereby requiring access to the textual knowledge encoded within the FFN layers. This integration of information is essential. Compare (2, 3, 4, 5) in Table 6, employing a summary token as the final visual token has demonstrated enhanced effectiveness compared to a standard visual token. Furthermore, our experimental findings reveal that optimal performance is achieved when both the summary token and the last local visual token are processed through the FFN.

The former summary token enhances the model's comprehension of overall visual information. Compare (4,6), (7,8), (9,10) in Table 6, the former summary token enhances the emphasis on crucial information by adaptively merging large-scale visual features. This approach addresses the challenge posed by overly lengthy sequences of visual tokens bypassing the FFN, which may result in the omission of critical large-scale visual context.

The local token merge strategy seamlessly aligns with the skip-vision framework. Compare (0,1), (4,7), (6,8) in Table 6, when loacl token merge is directly applied to the CoS_{1296} baseline, the model's overall performance declines, reflecting its dependency on redundant visual data when all tokens pass through the FFN. In contrast, within the skip-vision framework, merging local tokens results in improved performance, indicating that our architecture efficiently leverages visual information without requiring excessive redundancy.

The skip KV-cache mechanism enables adaptive selection of visual tokens to skip based on task-specific in-

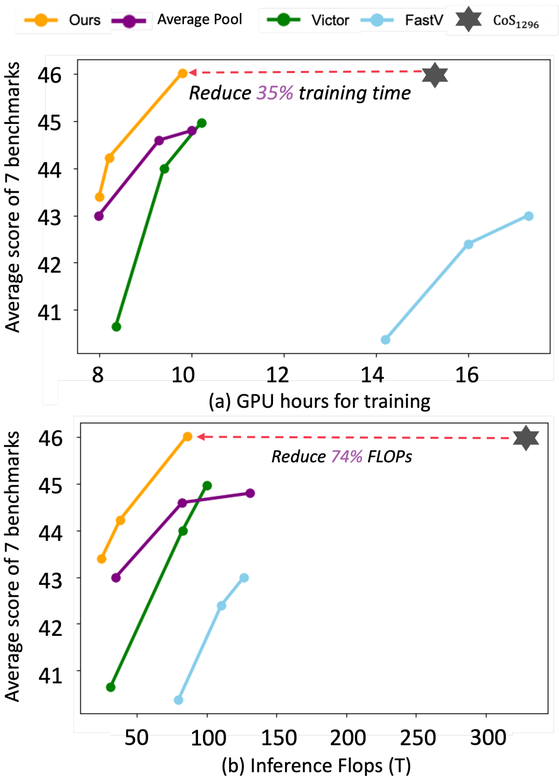

Figure 11. **Performance vs. Training-Time and Inference FLOPs.** Under the cos setting, we compare Skip-Vision with three MLLM acceleration methods, showing clear advantages in training speed and inference efficiency under equivalent computational constraints.

formation requirements. As demonstrated in Table 7, for tasks such as MMB, MMU, and MMStar that do not necessitate fine-grained information, both middle and small window-size visual tokens can be skipped, with only the initial and final summary tokens retained. For detail-sensitive tasks like TextVQA and OCRBench, we skip only the small window-size tokens that bypass the FFN, thereby preserving critical fine-grained details. Applying skip KV-cache with small window-size tokens during inference improves performance, particularly in tasks requiring extended responses, such as MMVet.

10.3. More visualizations

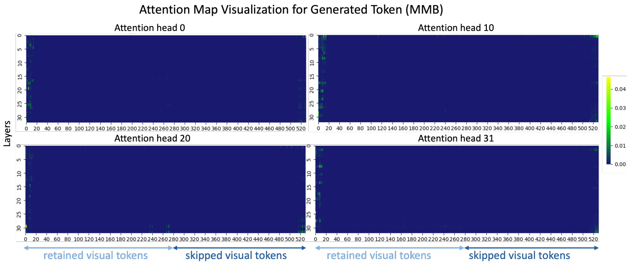

Figure 12. **Visualization of attention map in MMBench.**

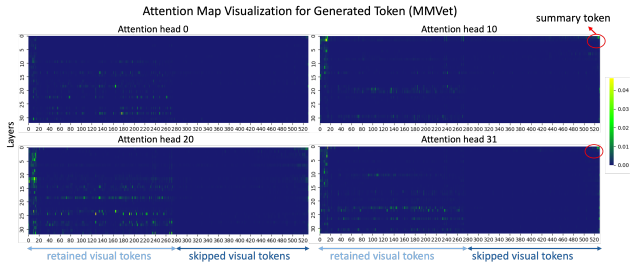

Figure 13. **Visualization of attention map in MMVet.**

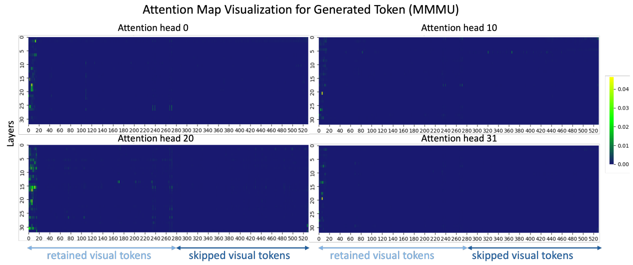

Figure 14. **Visualization of attention map in MMMU.**

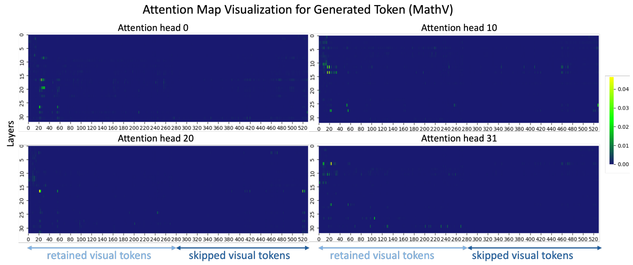

Figure 15. **Visualization of attention map in MathV.**

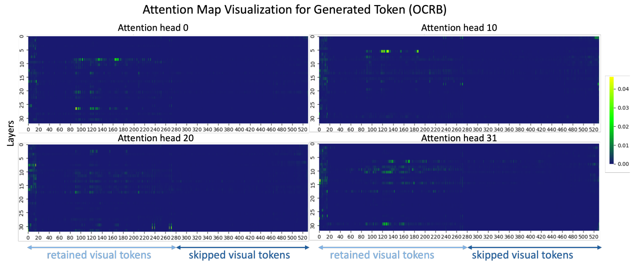

Figure 16. **Visualization of attention map in OCRBench.**

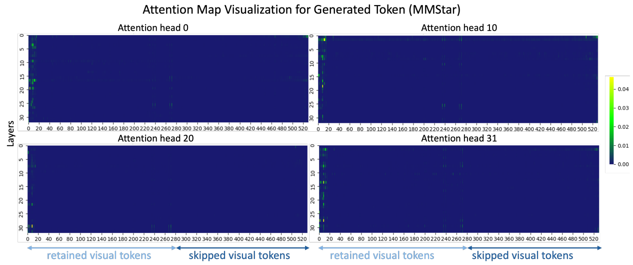

Figure 17. **Visualization of attention map in MMStar.**

10.4. Dataset

In Table 8 and Table 9, we introduce the two scaling datasets used by Skip-Vision during the SFT stage: SK-1M and SK-9M.

Task	Dataset
Visual Instruction Tuning	LLaVA-665k [69], SVIT [132]
VQA	CREPE [122],Imagenet multi task [89], VQA-rad [52]
Visual Reasoning	Wikitable [51], Super-CLEVR [62], VSR [65]
Knowledge	ViQuAE [55],Kvqa [90], Websrc [23]
Chart / Diagram / graph	ChartQA [79],Iconqa [74], Infographicvqa [82],
Document	DeepForm [98], TAT-QA [135], Visualmrc [99], Docvqa [81], Sujet-Finance-QA-Vision-100k [97]
Math	Mathverse [128]
OCR / Screen / Scene text	TQA [50], HW-SQuAD [80], TextVQA [95], ST-VQA [8],TextOCR-GPT4V [13], OCRbench-kv [71], Uber-text [129]
Science	AI2D [49]

Table 8. Datasets used by SK-1M at the SFT stage.

Task	Dataset
Visual Instruction Tuning	SVIT [132], ALLaVA [18], ShareGPT4V [20], cog-vlm-sft [105]
Caption	TextCaps [94], ShareGPT-4o [106]
VQA	CREPE [122],Imagenet multi task [89], VQA-rad [52], VQAv2 [4], Vizwiz [34]
Visual Reasoning	Wikitable [51], Super-CLEVR [62], VSR [65], FigureQA [45], TallyQA [1], Visual cot [92], CLEVR [43], Raven [126]
Knowledge	ViQuAE [55],Kvqa [90], Websrc [23], OK-VQA [78], Volcano [54], RLAIF-V [120]
Chart / Diagram / graph	ChartQA [79],Iconqa [74], Infographicvqa [82], MapQA [17], TabFact [110], Chart2Text [46], DVQA [44], Chartbench [113], MMC [66]
Document	DeepForm [98], TAT-QA [135], Visualmrc [99], Docvqa [81], Sujet-Finance-QA-Vision-100k [97], Docmatix [53], DocReason25K [37], DocStruct4M [38]
Math	Mathverse [128], MathOCR [16], MathV360K [93], GeoGPT4V [11], Geo170K QA [31]
OCR / Screen / Scene text	TQA [50], HW-SQuAD [80], TextVQA [95], ST-VQA [8],TextOCR-GPT4V [13], OCRbench-kv [71], Uber-text [129], OCR-VQA [85], ScreenQA [5], SynthText [33], ChromeWriting [109], K12 Printing [57], SQuAD [87], ICDAR19-LSVT [48], ICPR18-MTWI [36], ICDAR19-ArT [26], COCO-Text [102], Docscan [96], HierText [73]
Science	AI2D [49], Plotqa [84], ArXivQA [59], ScienceQA [75]

Table 9. Datasets used by SK-9M at the SFT stage.