Variation-aware Vision Token Dropping for Faster Large Vision-Language Models

Supplementary Material

In the appendix, we provide detailed experimental settings in Section 6, additional experimental results in Section 7, algorithmic descriptions in Section 8, further discussion on content-agnostic positional bias in Section 9, and detailed theoretical analysis in Section 10.

6. Detailed Experimental Settings

Benchmark Details. We evaluate V^2Drop on various multi-modal understanding benchmarks detailed as follows:

- **GQA** [16] comprises scene graphs, questions, and images, designed to test visual scene understanding and multi-aspect image reasoning capabilities.
- **MMBench** [31] evaluates models through a three-level hierarchical structure with 20 specific ability dimensions, enabling comprehensive assessment of perception and reasoning capabilities.
- **MME** [12] comprises 14 subtasks evaluating perceptual and cognitive abilities through manually constructed instruction-answer pairs, mitigating data leakage issues.
- **POPE** [22] evaluates object hallucination through binary questions about object presence, using accuracy, recall, precision, and F1 metrics across three sampling strategies.
- **ScienceQA** [32] spans natural, language, and social sciences with hierarchical categorization, evaluating multimodal understanding and multi-step reasoning capabilities.
- **TextVQA** [34] evaluates models' ability to read and reason about text within images through visual question-answering tasks requiring integrated textual understanding.
- **AI2D** [17]comprises 5,000 scientific diagrams with accompanying questions that test visual-spatial reasoning capabilities across educational content.
- **MMStar** [4]provides 12,000 high-resolution images designed to evaluate spatial, temporal, and commonsense reasoning across multimodal understanding tasks.
- **MVBench** [20] defines 20 video understanding tasks that require deep comprehension of temporal dimensions, beyond single-frame analysis.
- **VideoMME** [13] comprises 900 videos and 2,700 multiple-choice questions across six domains, with durations from 11 seconds to 1 hour, categorized into short, medium, and long subsets.

Baseline Models. The baseline LVLMs, as follows:

- **LLaVA-1.5** [25] enhances multimodal understanding by scaling visual instruction tuning with academic-task-oriented datasets and improved training recipes. It incorporates a two-stage training approach that first aligns vision and language representations, then fine-tunes on diverse instruction-following data, achieving strong performance on visual reasoning, OCR, and multimodal dialogue tasks across various benchmarks.
- **Qwen2-VL** [36] enhances multimodal perception by investigating scaling laws for vision-language models. By scaling model size (2B, 8B, and 72B parameters) and training data, it achieves competitive performance across diverse tasks. It supports any resolution input, enabling superior performance on document parsing, OCR, visual reasoning, and video understanding while maintaining strong text-image alignment.
- **LLaVA-OneVision** [18] unifies single-image, multi-image, and video tasks in a single model. It represents videos as long visual token sequences in the same "interleaved" format used for images, enabling smooth task transfer from images to videos and facilitating strong zero-shot video understanding capabilities.

Comparison Methods. We provide detailed introductions and comparisons of existing token compression methods mentioned in the main text, as follows:

- **ToMe** [3] merges similar tokens in visual transformer layers through lightweight matching techniques, achieving acceleration without requiring additional training.
- **LLaVA-PruMerge** [33] combines pruning and merging strategies by dynamically removing less important tokens using CLS-patch attention and clustering retained tokens based on key similarity.
- **FastV** [5] focuses on early-stage token pruning by leveraging attention maps, effectively reducing computational overhead in the initial layers.
- **DART** [39] introduces a duplication-aware token pruning approach that selects tokens based on their redundancy relative to pivot tokens rather than importance scores.
- **HiRED** [1] allocates token budgets across image partitions based on CLS token attention, followed by the selection of the most informative tokens within each partition, ensuring spatially aware token reduction.
- **PDrop** [43] adopts a progressive token-dropping strategy across model stages, forming a pyramid-like token structure that balances efficiency and performance.

Benchmark	Vanilla	Pruned Layers Selection						Other Methods		
		(4,14,30)	(3,14,29)	(3,15,27)	(3,16,24)	(3,17,22)	(2,16,21)	FastV	SparseVLM	PDrop
GQA	61.9	57.8	58.6	58.5	58.8	58.5	57.9	52.7	57.1	57.6
SQA	69.5	68.9	69.1	69.3	69.1	69.3	69.5	67.3	68.8	68.7
POPE	85.9	86.3	85.0	85.1	85.0	85.1	83.9	64.8	82.3	83.6
MME	1862	1753	1826	1847	1813	1826	1759	1612	1766	1721
MMB	64.6	63.2	63.4	63.7	63.5	63.7	62.8	61.2	63.2	62.5
TextVQA	58.2	53.1	54.0	54.8	55.2	55.6	54.8	52.5	56.1	56.1
Avg. (%)	100.0%	96.0%	97.0%	97.5%	97.3%	97.6%	96.2%	88.2%	96.0%	95.8%

Table 6. **Supplementary results on pruned layers selection.** Performance with 192 retained tokens on LLaVA-1.5-7B across datasets. The notation (a, b, c) represents a three-stage pruning strategy with token reduction applied at the a-th, b-th, and c-th layers, respectively.

Methods	Throughput (item/s)			
	MME	GQA	MMBench	SQA
LLaVA-1.5-7B	8.02	7.5	7.13	6.9
FastV	9.46(1.18×)	8.68(1.16×)	8.65(1.21×)	8.14(1.18×)
Cosine Similarity	9.95(1.24×)	9.13(1.21×)	8.90(1.25×)	8.14(1.18×)
L1 Norm	10.16(1.27×)	9.23(1.23×)	9.01(1.26×)	8.49(1.23×)
L2 Norm	10.11(1.26×)	9.18(1.22×)	9.01(1.26×)	8.42(1.22×)

Table 7. **Supplementary results on variation metric selection.** Throughput with 128 retained tokens on LLaVA-1.5-7B across datasets. The notation (N×) represents an N-fold throughput improvement compared to the baseline model LLaVA-1.5-7B.

Methods	Performance				
	MME	POPE	MMBench	GQA	TextVQA
LLaVA-1.5-7B	1862	85.9	64.6	61.9	58.2
One-time dropping	1717	77.1	60.6	57.1	55.2
Progressive dropping	**1826**	**85.1**	**63.7**	**58.5**	**55.9**

Table 8. **Supplementary results on effects of progressive token dropping.** Performance with 192 retained tokens on LLaVA-1.5-7B across datasets.

- **SparseVLM** [54] ranks token importance using cross-modal attention and introduces adaptive sparsity ratios, complemented by a novel token recycling mechanism.
- **DyCoke** [35] is a two-stage VideoLLM method that prunes similar tokens temporally and compresses less-attended visual tokens in KV cache using LLM attention weights. Its reliance on frame-set division and similarity-based compression limits aggressive token compression, and while compatible with Flash Attention [10], it requires explicit attention weights making it incompatible with efficient attention operators.

Implementation Details. Our experiments are conducted on NVIDIA A100-PCIe-80GB GPUs. The implementation was carried out in Python 3.10, utilizing PyTorch 2.1.2 and CUDA 12.1. All baseline settings follow the original paper.

Experimental parameter details for V^{2}Drop. On LLaVA-1.5-7B, we conduct three-stage pruning at layers 3, 17, and 22. When retaining 192 tokens, we prune 50%, 70%, and 100% of Vision tokens at layers 3, 17, and 22. When retaining 128 tokens, we prune 72%, 75%, and 100% of Vision tokens at layers 3, 17, and 22, respectively. When retaining 64 tokens, we prune 95%, 95%, and 100% of Vision tokens at layers 3, 17, and 22, respectively.

Methods	Performance			
	MME	GQA	MMBench	SQA
LLaVA-1.5-7B	1862	61.9	64.6	69.5
FastV	1490	49.6	56.1	67.3
Cosine Similarity	1718	55.2	61.8	69.2
L1 Norm	1698	56.1	60.9	68.7
L2 Norm	1712	56.3	61.8	68.8

Table 9. **Supplementary results on variation metric selection.** Performance with 128 retained tokens on LLaVA-1.5-7B across datasets.

7. Additional Experimental Results

Supplementary Results on Variation Metric Selection. This section presents detailed results of V^{2}Drop from the ablation study on the Effects of Variation Metric. Table 7 and Table 9 respectively report the throughput and performance data of three variation metrics across multiple datasets. The experiments demonstrate that variation-based pruning strategies outperform attention-score-based pruning methods such as FastV in both performance and efficiency, validating the robustness of variation-based dropping strategies. For more intuitive visualizations, please refer to the main discussion in §4.3.

Supplementary Results on Pruned Layers Selection. This section presents comprehensive experimental results on LLaVA-1.5-7B, providing a detailed analysis of the pruning layer selection strategies of V^{2}Drop. Table 6 comprehensively lists performance metrics across multiple benchmarks, including GQA, SQA, POPE, MME, MMB, and TextVQA, with all experiments retaining 192 visual tokens. These findings further validate the robustness of V^{2}Drop under various pruning layer combinations. The table also includes comparisons with baseline methods such

as FastV, SparseVLM, and PDrop, highlighting the consistent superiority of V^{2}Drop across different configurations. For more intuitive visualizations, please refer to the main discussion in §4.3.

Supplementary Results on Effects of Progressive Token Dropping. This section presents detailed results of V^{2}Drop from the ablation study on the Effects of Progressive Token Dropping. Table 8 shows the performance of two token pruning strategies in V^{2}Drop: progressive token dropping and one-time dropping, evaluated on LLaVA-1.5-7B. The experiments demonstrate that progressive token dropping significantly outperforms one-time dropping across all datasets, proving that progressive token dropping more effectively preserves critical visual information through its gradual selection mechanism. For more intuitive visualizations, please refer to the main discussion in §4.3.

More Visualizations of Token Compression. In Figure 10, we present additional token compression visualization results of V^{2}Drop across diverse scenarios. The visualizations demonstrate that by preserving key tokens based on visual token variation information, V^{2}Drop progressively selects core tokens from images and focuses on semantically critical regions. This indicates that our token variation metric can effectively localize important regions. As illustrated in the figure, across various real-world scenarios where critical regions are located at different positions within the image—such as the bottom-left (case 11), top-right (case 5), and top-left (case 1)—our method consistently establishes accurate correspondence between token importance and semantic relevance.

8. Algorithm Details of V^{2}Drop

Algorithm 1 presents the algorithm workflow of our V^{2}Drop method. This algorithm details the step-by-step process of our token compression approach, illustrating how V^{2}Drop dynamically compresses visual tokens based on variation analysis.

9. More Discussions about Content-agnostic Positional Bias.

Figures 2 and 3 reveal the inherent content-agnostic positional bias of LLM attention-guided methods such as SparseVLM and FastV. Figure 2 illustrates how these methods, despite assigning higher scores to critical regions, disproportionately favor later-positioned tokens regardless of content relevance, leading to the discarding of informative earlier tokens and triggering multimodal hallucinations. In contrast, measuring token-wise variation (*e.g.*, L2 Norm) intuitively reflects token importance and selectively retains semantically critical tokens. To quantify this bias, Figure 3 analyzes LLaVA-1.5-7B and Qwen2-VL-7B across three datasets (TextVQA, POPE, and MME), partitioning tokens into 10 equal intervals and calculating retention probabilities after pruning 50% of tokens at the third layer. Results demonstrate that attention-guided methods exhibit strong end-of-sequence bias, while variation-aware evaluation produces naturally uniform spatial distributions. Below, we provide a detailed theoretical analysis to establish the relationship between token variation and model output.

Algorithm 1 V^{2}Drop: Variation-aware Vision Token Dropping

Require: Vision tokens $\mathbf{F}^v \in \mathbb{R}^{M \times D'}$, Dropping layers $\mathcal{L} = \{l_1, l_2, \ldots, l_K\}$, Compression Targets $\{K_{l_1}, K_{l_2}, \ldots, K_{l_K}\}$
Ensure: Compressed vision tokens

Current token count $M_{\text{curr}} \leftarrow M$
for $l = 1, 2, \ldots, L$ **do**
  if $l \in \mathcal{L}$ **then**
    Step 1: Variation Computation
    for $i = 1$ to M_{curr} **do**
      $s_i^{(l)} \leftarrow \|\mathbf{f}_i^{(l)} - \mathbf{f}_i^{(l-1)}\|_2$
    end for
    $\mathbf{S}^{(l)} = \{s_1^{(l)}, s_2^{(l)}, \ldots, s_{M_{\text{curr}}}^{(l)}\}$
    Step 2: Token Ranking and Selection
    indices $\leftarrow \text{argsort}(\mathbf{S}^{(l)}, \text{descending})$
    $\hat{\mathbf{F}}_l^v \leftarrow \{\mathbf{f}_{\text{indices}[j]}^{(l)} : j = 1, \ldots, K_l\}$
    $\mathbf{F}_{\text{curr}}^v \leftarrow \hat{\mathbf{F}}_l^v$, $M_{\text{curr}} \leftarrow K_l$
  else
    $\mathbf{F}_{\text{curr}}^v \leftarrow \text{TransformerLayer}(\mathbf{F}_{\text{curr}}^v)$
  end if
end for
return $\mathbf{F}_{\text{curr}}^v$

10. Suppleymentary Theoretical Analysis

Here, we present the complete theoretical proof that rigorously establishes the connection between token variation and model output through first-order analysis.

10.1. Smoothness Assumption

We assume the model f has sufficient local smoothness in the representation space, such that the second-order remainder term in the Taylor expansion is bounded, satisfies:

$$\|R_j\| = \mathcal{O}(\|\Delta x_j^{(t)}\|^2). \tag{14}$$

This assumption is well- justified in Transformer-based LVLMs due to three architectural properties:

- **Residual connections** limit layer-wise changes, ensuring $\|\Delta x_j^{(t)}\|$ remains small relative to $\|x_j^{(t)}\|$;

- **Layer normalization** constrains the range of token representations, bounding higher-order derivatives;
- **Smooth activations** (e.g., GELU, SiLU) provide continuous second derivatives, ensuring Taylor expansion validity.

Under this assumption, for sufficiently small $\|\Delta x_j^{(t)}\|$, the quadratic term is negligible compared to the linear term, yielding:

$$\|\Delta f_j\| \approx \|J_j\|_{\text{op}} \cdot \|\Delta x_j^{(t)}\| \tag{15}$$

10.2. Justification of Bounded Jacobian Assumption

In the proof of Corollary, we assume that for all tokens j, the Jacobian operator norm is bounded below: $\|J_j\|_{\text{op}} \geq \mu > 0$ for some constant μ. Here is the proof for this assumption.

Assumption (Non-degenerate Gradients). The function f has non-degenerate gradients with respect to token representations, i.e., there exists $\mu > 0$ such that:

$$\|J_j\|_{\text{op}} = \left\| \frac{\partial f}{\partial x_j^{(t+1)}} \right\|_{\text{op}} \geq \mu, \quad \forall j \in [n] \tag{16}$$

This assumption is reasonable for the following reasons:

1. Information Flow in Transformers. In Transformer architectures, each token contributes to the final output through multi-head attention and feed-forward layers. The attention mechanism ensures that:

$$\frac{\partial \text{Output}}{\partial x_j} = \sum_{i=1}^{n} \frac{\partial \text{Output}}{\partial h_i} \cdot \frac{\partial h_i}{\partial x_j} \tag{17}$$

where h_i are intermediate representations. Due to the softmax normalization in attention, each token x_j receives non-zero attention weights from at least some positions, ensuring $\|\frac{\partial \text{Output}}{\partial x_j}\| > 0$.

2. Residual Connections Preserve Gradients. The residual structure $x^{(t+1)} = x^{(t)} + \text{Block}(x^{(t)})$ ensures that gradients flow directly through identity mappings:

$$\frac{\partial f}{\partial x_j^{(t)}} = \frac{\partial f}{\partial x_j^{(t+1)}} \cdot \left(I + \frac{\partial \text{Block}}{\partial x_j^{(t)}} \right) \tag{18}$$

The identity component I guarantees that gradients do not vanish, thus $\|J_j\|_{\text{op}} \geq \mu$ for some μ related to the minimum singular value of the identity component.

3. Layer Normalization Stabilizes Gradients. Layer normalization prevents gradient explosion and vanishing by maintaining bounded gradient norms across layers, ensuring $\|J_j\|_{\text{op}} \in [\mu, M]$ for constants $0 < \mu < M < \infty$.

Discussion: What if $\|J_j\|_{\text{op}} \to 0$?

What if some tokens have $\|J_j\|_{\text{op}} \approx 0$? This would indicate that these tokens have negligible influence on the output. In such cases:

- These tokens can be safely dropped regardless of their variation magnitude
- Our method naturally handles this case: if $\|J_j\|_{\text{op}} \approx 0$, then $\|\Delta f_j\| \approx 0$ regardless of $\|\Delta x_j^{(t)}\|$, so dropping them causes minimal performance degradation

Therefore, Assumption ($\|J_j\|_{\text{op}} \geq \mu > 0$) is theoretically justified for vast majority of vision tokens in LVLMs.

10.3. Connection to V^{2}Drop Algorithm

Proposition 1 (Dropping Strategy Justification). Given n tokens at layer t, we aim to select $|\mathcal{S}_{\text{drop}}| = \alpha n$ tokens to drop while minimizing total output perturbation:

$$\mathcal{S}^*_{\text{drop}} = \underset{\substack{\mathcal{S} \subseteq [n] \\ |\mathcal{S}| = \alpha n}}{\arg\min} \sum_{j \in \mathcal{S}} \|\Delta f_j\| \tag{19}$$

Proof. By Theorem 1, $\|\Delta f_j\| \approx \|J_j\|_{\text{op}} \cdot \|\Delta x_j^{(t)}\|$. Under Assumption 2 ($\mu \leq \|J_j\|_{\text{op}} \leq M$), we have:

$$\begin{aligned} \sum_{j \in \mathcal{S}} \|\Delta f_j\| &\approx \sum_{j \in \mathcal{S}} \|J_j\|_{\text{op}} \cdot \|\Delta x_j^{(t)}\| \\ &\in \left[\mu \sum_{j \in \mathcal{S}} \|\Delta x_j^{(t)}\|, M \sum_{j \in \mathcal{S}} \|\Delta x_j^{(t)}\| \right] \end{aligned} \tag{20}$$

Since $\|J_j\|_{\text{op}}$ varies within a bounded range, minimizing $\sum_{j \in \mathcal{S}} \|\Delta f_j\|$ is approximately equivalent to:

$$\mathcal{S}^*_{\text{drop}} \approx \underset{\substack{\mathcal{S} \subseteq [n] \\ |\mathcal{S}| = \alpha n}}{\arg\min} \sum_{j \in \mathcal{S}} \|\Delta x_j^{(t)}\| \tag{21}$$

Therefore, V^{2}Drop's strategy of selecting tokens with minimal variation $\|\Delta x_j^{(t)}\|$ for dropping approximately minimizes total output perturbation, while computationally efficient (only requiring simple L2 norm computation). □

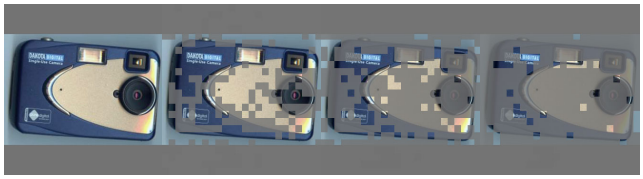
10.4. Connection to information flow

In Transformer layers with residual connections:

$$x_j^{(t+1)} = x_j^{(t)} + \text{Attn}(x_j^{(t)}) + \text{FFN}(x_j^{(t)}), \tag{22}$$

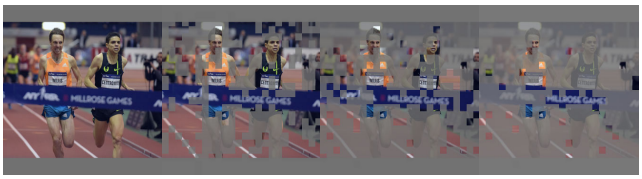
the variation $\Delta x_j^{(t)} = \text{Attn}(x_j^{(t)}) + \text{FFN}(x_j^{(t)})$ represents the *effective update* applied by the layer. Tokens with large $\|\Delta x_j^{(t)}\|$ are those being actively refined by the network, indicating they carry task-relevant information being extracted and propagated to subsequent layers.

1： Q: What is the brand of this camera? A: Dakota

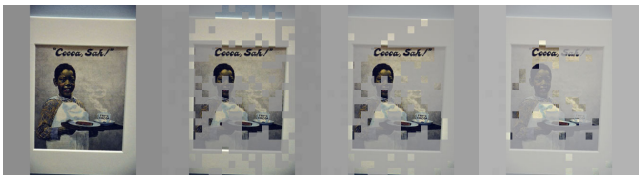


2： Q: What brand liquor is on the right? A: Bowmore

3： Q: What event is this from? A: Millrose games



4： Q: What is being served? A: Cocoa



5： Q: Is this a reference book? A: Yes

6： Q: What is the title of the book? A: Revoltez vous!

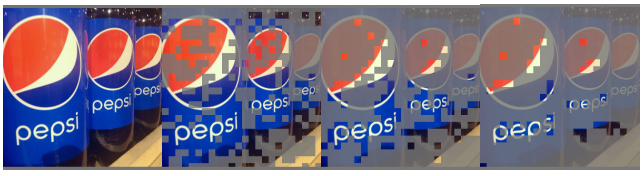
7： Q: What country does he play for? A: Holland

8： Q: Who must survive? A: YAAM

9： Q: Is that their lunchbox favorite? A: Yes

10： Q: What is the album name? A: CODE:Veronica

11： Q: Are these bottles of pepsi? A: Yes



12： Q: What brand is this drink? A: Red hook

Figure 10. **More visualization of token compression by V^{2}Drop.** The presented examples are from TextVQA, where grey masks indicate discarded visual tokens.