RedVTP: Training-Free Acceleration of Diffusion Vision-Language Models Inference via Masked Token-Guided Visual Token Pruning

Supplementary Material

8. Appendix

8.1. Algorithm

The algorithmic details of our RedVTP are presented in Algorithm 1.

Algorithm 1: Response-driven Visual Token Pruning

Input: Attention maps in DLM $P_\theta(\cdot)$ at the 1-st inference step $\mathbf{A}_1^{(\mathrm{l},\mathrm{h})}$; Indices of masked response tokens after the 1-st inference step $\mathcal{I}_1(M)$; Visual tokens $V \in \mathbb{R}^{N\times d}$; Indices of visual tokens $\mathcal{I}(V)$

Output: Retained visual tokens $V^{\text{keep}} \in \mathbb{R}^{N_r\cdot d}$

/ Masked token-guided importance score */*

$\bar{\mathbf{A}} = \frac{1}{H}\frac{1}{L}\sum_{h=1}^{H}\sum_{l=1}^{L}\mathbf{A}_1^{(l,h)}$;

$\mathbf{S_1} = \frac{1}{|\mathcal{I}_1(M)|}\sum_{j\in\mathcal{I}_1(M)}\bar{\mathbf{A}}_{j,\mathcal{I}(V)}$;

/ Visual token pruning */*

$\mathcal{I}^{\text{keep}} = \texttt{Top}(\mathcal{I}(V), \mathbf{S}_1, r)$;

Retained visual tokens: $V^{\text{keep}} \leftarrow V[\mathcal{I}^{\text{keep}}_{\text{sorted}}]$;

return V^{keep}

8.2. Efficiency Comparison Between Ours and Progressive Pruning

As shown in Table 5, our pruning method consistently achieves better inference efficiency compared to Progressive Pruning (PP). On average across these benchmarks, our method reduces latency by 18.94% and improves throughput by 26.46% relative to PP.

Model & Method		Efficiency		
		RealworldQA [24]	DocVQA [19]	InfoVQA [18]
Ours	Latency (s/sample) ↓	**2.746**	**9.725**	**9.601**
	Throughput (tok/s) ↑	**0.364**	**3.187**	**3.229**
PP	Latency (s/sample) ↓	2.746	13.577	13.419
	Throughput (tok/s) ↑	0.364	2.283	2.310

Table 5. Comparison of latency and throughput between our method and PP when applied to LLaDA-V