ZipVL: Accelerating Vision-Language Models through Dynamic Token Sparsity

Supplementary Material

A. Efficient Approximation of Full Attention Scores

ZipVL requires accumulated attention scores to adaptively assign the ratio of important tokens and normalized attention scores to identify token importance. However, attention scores are not accessible in fast attention implementations such as FlashAttention. To integrate our method with FlashAttention, we follow prior literature [1] and select a subset of tokens, referred to as "**probe tokens**", and explicitly compute their attention scores:

$$\mathbf{A}_{probe} = \text{Softmax}\left(\frac{\mathbf{Q}_{probe}\mathbf{K}^T}{\sqrt{d_k}}\right). \tag{A}$$

Prior work [1] selects 10% of the tokens as probe tokens, which still yields quadratic complexity in Eq. A. In contrast, we select only 64 recent tokens and 64 randomly positioned tokens, which incurs negligible computation overhead in long-context scenarios.

B. Additional Experimental Results

B.1. Benchmarking Against More State-of-the-Art Methods

We compare the performance of ZipVL to state-of-the-art token merging [3, 4] and token pruning [2] approaches. The results are shown in Table A. ZipVL consistently outperforms VisionZip with a lower token budget. Compared to AIM, despite its fixed schedule aggressively compressing the token budget, it demonstrates significant accuracy drops in multiple benchmarks. ZipVL also outperforms PyramidDrop in two evaluated benchmarks with lower token budget.

Table A. Performance comparison with state-of-the-arts methods. Results are obtained over LLaVA-Next-13B.

Method	Token Budget (%)	VQAv2	TextVQA	GQA
VisionZip	33.3	78.5	63.6	62.5
AIM	6.4	71.5	61.8	57.5
PyramidDrop	46.8	77.5	**64.5**	63.3
ZipVL	30.6	**79.7**	63.6	**64.3**

B.2. Experiments on VideoChatGPT Benchmark

As shown in Table B, we further evaluate the performance of ZipVL on VideoChatGPT. The results show that ZipVL outperforms the baseline method, FastV, with notable margin while achieving higher sparsity.

B.3. Ablation on Probe Tokens

In this subsection, we conduct ablation studies on the selection of probe tokens, as shown in Table C. When 128 randomly positioned tokens are used as probe tokens, the accuracy drops significantly to 6.3%. While relying solely on recent tokens delivers reasonable performance, a hybrid approach that combines recent and randomly positioned tokens demonstrates superior performance (52.6% compared to 52.4%). Notably, this hybrid strategy achieves accuracy comparable to computing full attention scores.

Table B. Performance comparisons on VideoChatGPT with LongVA-7B model. The evaluation metrics include Correctness Information (CI), Detail Orientation (DO), and Contextual Understanding (CU).

Model	Method	Sparsity (%)	CI	DO	CU
LongVA-7B	Full	0	2.33	2.34	2.92
	FastV	46.42	2.27	2.15	2.84
	Ours	**54.65**	**2.38**	**2.34**	**2.96**

B.4. Ablation on Importance Metric

This subsection evaluates the impacts of different metrics on layer-wise adaptive ratio assignment and the identification of important tokens, as depicted in Table D. For adaptive ratio assignment, accumulated attention scores consistently outperform normalized attention scores in terms of overall performance. Conversely, for the identification of important tokens, employing normalized attention scores yields higher accuracy, which is consistent with the findings of prior studies [1] in LLMs.

B.5. Effect of the Threshold τ

This subsection investigates the impact of the attention retention threshold τ on the proportion of important tokens and model performance across different models and benchmarks, as shown in Figure A. When τ decreases but stays above 0.98, the proportion of important tokens drops significantly, while the accuracy declines only marginally. This suggests improved generation efficiency with minimal performance loss. However, when τ falls below 0.97, a noticeable drop in model performance occurs, accompanied by a continued decrease in the proportion of important tokens.

B.6. Overhead Analysis

As shown in Table E, we provide a detailed analysis of FLOPs and memory usage for each operation in ZipVL. The results indicate that the operations for evaluating token importance introduce minimal computational and memory overhead.

Table C. Performance comparisons across different probe token selection approaches on ChartQA benchmark. Here, "Ratio" denotes the proportion of tokens involved in attention computation. "All" denotes all tokens are used as probe tokens, requiring full attention score computation. To ensure a fair comparison, the threshold τ is adjusted to maintain a similar "Ratio" across approaches.

Model	Method	Probe Tokens	τ	Ratio (%)	Acc. (%)
LLaVA-Next-7B	Original	-	-	100	54.8
	ZipVL	64 recent	0.980	50.5	52.3
		128 recent	0.987	50.5	52.4
		128 random	0.975	51.3	6.3
		64 recent & 64 random	0.975	**50.6**	**52.6**
		All	0.960	53.6	52.6

Table D. Performance comparisons across different metric for adaptive ratio assignment and important token identification. Data is collected over LLaVA-Next-7B model on ChartQA benchmark. Here, "Ratio" denotes the proportion of tokens involved in attention computation. To ensure a fair comparison, the threshold τ is adjusted to maintain similar "Ratio" values across approaches.

Metric for Adaptive Ratio Assignment	Metric for Important Token Identification	τ	Ratio (%)	Acc. (%)
Accumulated Attention Scores	Accumulated Attention Scores	0.975	50.28	52.40
Accumulated Attention Scores	Normalized Attention Scores	0.975	50.55	**52.64**
Normalized Attention Scores	Accumulated Attention Scores	0.995	51.84	51.56
Normalized Attention Scores	Normalized Attention Scores	0.995	51.45	51.84

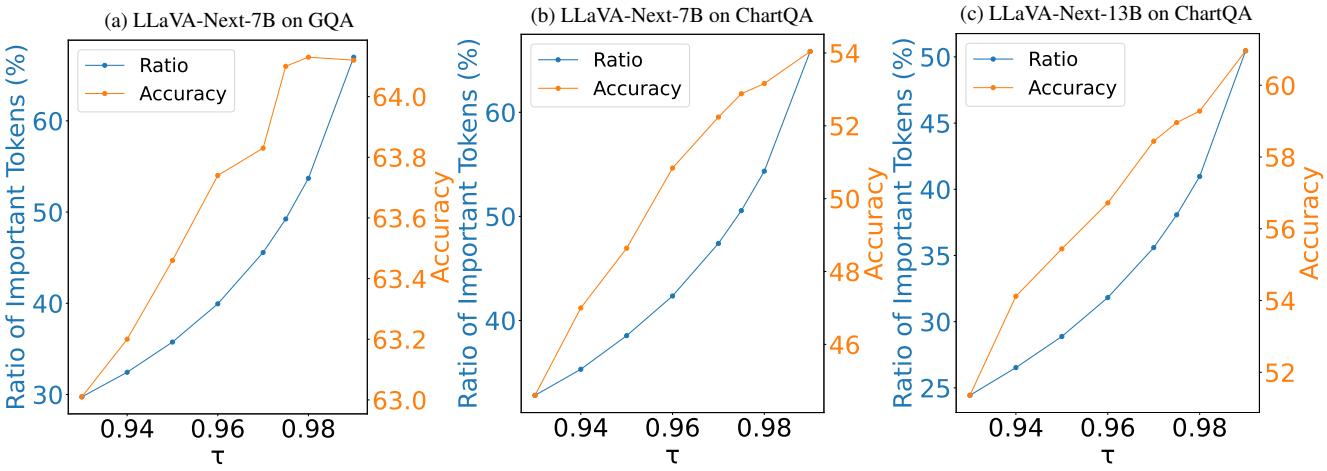

Figure A. The effect of attention scores retention threshold τ on the ratio of important tokens and the model performance.

Table E. The overhead of each operation in ZipVL over LongVA-7B. Data is collected with a sequence length of 32K.

Method	TTFT (s)	FLOPs (T)	Memory (MB)
Original	3.28	774.47	29474.44
+ approximate attention (Eq. 6)	3.73	776.15	29474.44
+ sort & cumsum (Eq. 7)	3.74	776.15	29474.44
+ normalize & top-k (Eqs. 8-9)	3.74	776.15	29477.04
+ sparse attention	**2.60**	**446.90**	**28031.60**

References

[1] Yefei He, Luoming Zhang, Weijia Wu, Jing Liu, Hong Zhou, and Bohan Zhuang. Zipcache: Accurate and efficient kv cache quantization with salient token identification. In *NeurIPS*, 2024. 1

[2] Long Xing, Qidong Huang, Xiaoyi Dong, Jiajie Lu, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, Jiaqi Wang, Feng Wu, et al. Pyramiddrop: Accelerating your large vision-language models via pyramid visual redundancy reduction. *arXiv preprint arXiv:2410.17247*, 2024. 1

[3] Senqiao Yang, Yukang Chen, Zhuotao Tian, Chengyao Wang, Jingyao Li, Bei Yu, and Jiaya Jia. Visionzip: Longer is better but not necessary in vision language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19792–19802, 2025. 1

[4] Yiwu Zhong, Zhuoming Liu, Yin Li, and Liwei Wang. Aim: Adaptive inference of multi-modal llms via token merging and pruning. *arXiv preprint arXiv:2412.03248*, 2024. 1