VScan: A Two-Stage Visual Token Reduction Framework for Accelerating Large Vision-Language Models

Ce Zhang [1 2] **Kaixin Ma** [2] **Tianqing Fang** [2] **Wenhao Yu** [2] **Hongming Zhang** [2]
Zhisong Zhang [2] **Yaqi Xie** [1] **Katia Sycara** [1] **Haitao Mi** [2] **Dong Yu** [2]

Abstract

Recent Large Vision-Language Models (LVLMs) have advanced multi-modal understanding by incorporating finer-grained visual perception and encoding. However, such methods incur significant computational costs due to longer visual token sequences, posing challenges for real-time deployment. To mitigate this, prior works have explored pruning unimportant visual tokens either at the output layer of the visual encoder or at the early layers of the language model. In this work, we revisit these design choices and reassess their effectiveness through comprehensive empirical studies of how visual tokens are processed throughout the visual encoding and language decoding stages. Guided by these insights, we propose VScan, a two-stage visual token reduction framework that addresses token redundancy by: (1) integrating complementary global and local scans with token merging during visual encoding, and (2) introducing pruning at intermediate layers of the language model. Extensive experimental results across four LVLMs validate the effectiveness of VScan in accelerating inference and demonstrate its superior performance over current state-of-the-arts on sixteen benchmarks.

1 Introduction

Large Vision-Language Models (LVLMs) have emerged as a transformative advancement in multi-modal learning, achieving remarkable proficiency across a broad range of vision-language tasks (Liu et al., 2023; Li et al., 2024; 2023a; Team et al., 2024). Recent advances in LVLMs (Liu et al., 2024b; Li et al., 2025) further enhance their capacity to process high-resolution images and multi-image/video inputs, enabling fine-grained perception in tasks such as video question answering (Fang et al., 2024), multi-image understanding (Fu et al., 2024), and referential grounding (Kazemzadeh et al., 2014). However, processing such rich visual inputs necessitates a substantial increase in the number of visual tokens, which often far exceeds the number of text tokens (Lin et al., 2024a; Li et al., 2025). This leads to significantly longer input sequences and, due to the quadratic complexity of self-attention (Vaswani et al., 2017), incurs substantial computational and memory overhead, thereby limiting real-time deployment of LVLMs in practical applications (Chen et al., 2024a; Yang et al., 2025).

Recognizing that not all visual tokens contribute meaningfully to the final LVLM response, recent works (Chen et al., 2024a; Xing et al., 2025; Zhang et al., 2024a) have proposed visual token reduction techniques aimed at improving computational efficiency by pruning visually redundant or textually irrelevant tokens. These methods generally fall into two categories: (1) *Text-agnostic pruning approaches* (Zhang et al., 2024a; Yang et al., 2025; Wang et al., 2025; Wen et al., 2025) (Figure 1(a)), which prune visually redundant tokens based on their significance and uniqueness during the visual encoding stage; and (2) *Text-aware pruning approaches* (Zhang et al., 2024b; Xing et al., 2025; Ye et al., 2025) (Figure 1(b)), which selectively remove tokens with low relevance to the text query during the early layers of language decoding stage. While these approaches have shown promising results, their performance is often constrained by their single-stage design and the lack of a systematic understanding of how visual tokens are processed and utilized throughout the *entire LVLM pipeline*.

In this work, we conduct an in-depth empirical analysis to reassess the effectiveness of these two prevailing pruning paradigms and distill insights that guide the design of more effective visual token reduction methods. Our study reveals two key observations: (1) In the visual encoding stage, the visual encoder attends to locally significant tokens in the shallow layers, focusing on fine-grained local details, while at deeper layers, it gradually shift its focus to a highly condensed set of tokens that encapsulate broader global context; (2) In the LLM decoding stage, early layers exhibit strong positional bias toward visual tokens appearing later in the sequence, neglecting their semantic relevance; as the layers deepen, cross-modal interactions begin to

[1]Robotics Institute, Carnegie Mellon University [2]Tencent AI Lab. Correspondence to: Katia Sycara <katia@cs.cmu.edu>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

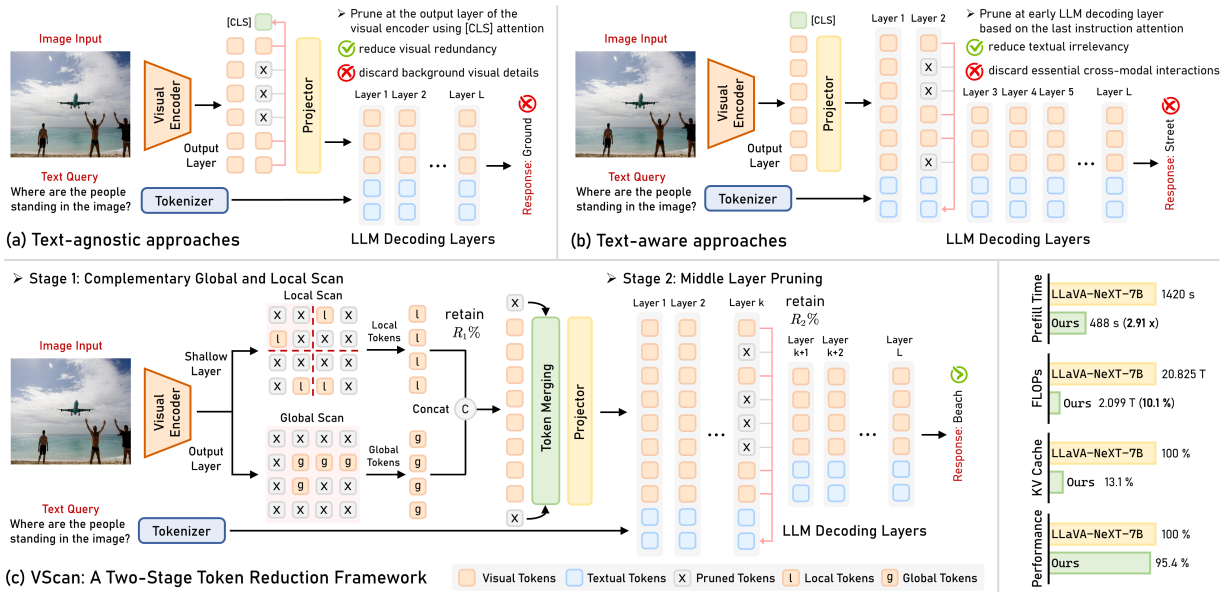

Figure 1: **Comparison of our VScan with existing approaches (*e.g.*, VisionZip (Yang et al., 2025) and FastV (Chen et al., 2024a)).** Our VScan delivers substantial inference acceleration for various LVLMs with minimal performance loss.

emerge, and output token probabilities typically converge in the mid-to-late layers where visual information is more effectively integrated into the language stream.

Building on these insights, we introduce VScan, a two-stage visual token reduction framework that enhances the efficiency of LVLMs by progressively pruning uninformative tokens during both visual encoding and language decoding, as shown in Figure 1(c). In the visual encoding stage, VScan employs a complementary global-local scan strategy to retain semantically important and spatially diverse tokens, followed by token merging to preserve comprehensive visual information. In the LLM decoding stage, VScan introduces middle layer pruning to further eliminate visual tokens with low relevance to the text query, while maintaining essential cross-modal interactions to minimize disruption to final task performance. We comprehensively evaluate the effectiveness of VScan on LLaVA-1.5, LLaVA-NeXT, Qwen-2.5-VL, and Video-LLaVA across sixteen image and video understanding benchmarks. Extensive experimental results demonstrate VScan's generalizable effectiveness across diverse LVLM architectures and LLM scales, highlighting its advantageous performance-efficiency trade-off.

2 Method

We introduce VScan, which progressively prunes uninformative tokens in visual encoding and LLM decoding stages to accelerate LVLM inference, as illustrated in Figure 1(c).

2.1 Empirical Analysis

We provide a comprehensive analysis of how LVLMs process visual tokens during both the visual encoding and language decoding in Appendix B, offering empirical guidance for designing more effective visual token reduction strategies. This analysis reveals two key insights: (1) Our findings highlight a gradual transition in the visual encoder from capturing low-level local details to modeling high-level, globally relevant semantics, suggesting that relying solely on the output layer may overlook the rich local information encoded in the shallow layers; (2) Our findings collectively suggest that early layers are suboptimal for pruning due to position bias and limited engagement with visual content. In contrast, pruning at middle layers is more appropriate as it better preserves critical cross-modal interactions and minimizes disruption to model predictions.

2.2 Complementary Global and Local Scans

Motivated by the observations in Appendix B, we design two complementary token selection schemes for the visual encoding stage, namely global and local scan, which select important tokens based on both local and global significance, enabling the capture of more comprehensive visual details.

Global Scan. Given that the final layers of visual encoders capture global information, we follow recent works (Zhang et al., 2024a; Yang et al., 2025) to select global tokens that receive the most attention from the [CLS] token $x_{\texttt{[CLS]}}$ in the output layer (*e.g.*, the penultimate layer in LLaVA-1.5 (Liu et al., 2024a)). Specifically, the [CLS] attention computation for each attention head can be represented by

$$Q_{\texttt{[CLS]}} = x_{\texttt{[CLS]}} W_Q^h, \quad K_V = \mathbf{x}_V W_K^h,$$
$$S^h_{\texttt{[CLS]}} = \texttt{Softmax}\left(\frac{Q_{\texttt{[CLS]}} K_V^\top}{\sqrt{D}}\right), \tag{1}$$

where W_Q^h and W_K^h represent the projections weights for

Table 1: **Performance comparisons on LLaVA-1.5-7B (Liu et al., 2024a) across 9 image understanding benchmarks**. The best results in each setting are **bolded**, and the second-best are underlined.

Method	Venue	GQA	MMB	MMB^{CN}	MME	POPE	SQA^{IMG}	VQA^{V2}	VQA^{Text}	VizWiz	Average
Upper Bound, 576 Tokens (100%), 3.817 TFLOPs											
LLaVA-1.5-7B (Liu et al., 2024a)	*CVPR'24*	61.9	64.7	58.1	1862	85.9	69.5	78.5	58.2	50.0	100.0%
Retain 192 Tokens in Average (↓ 66.7%), ∼1.253 TFLOPs											
ToMe (Bolya et al., 2023)	*ICLR'23*	54.3	60.5	-	1563	72.4	65.2	68.0	52.1	-	88.5%
FastV (Chen et al., 2024a)	*ECCV'24*	52.7	61.2	57.0	1612	64.8	67.3	67.1	52.5	50.8	90.4%
SparseVLM (Zhang et al., 2024b)	*arXiv'24*	57.6	62.5	53.7	1721	83.6	**69.1**	75.6	56.1	50.5	96.1%
PyramidDrop (Xing et al., 2025)	*CVPR'25*	57.3	63.3	56.8	1797	82.3	69.0	75.1	56.5	**51.1**	97.2%
VisionZip (Yang et al., 2025)	*CVPR'25*	59.3	63.0	-	1783	85.3	68.9	77.4	57.3	-	97.8%
VScan (Ours)	-	**60.6**	**63.9**	**57.4**	**1806**	**86.2**	68.6	**77.8**	**57.7**	50.4	**99.0%**
Retain 64 Tokens in Average (↓ 88.9%), ∼0.415 TFLOPs											
ToMe (Bolya et al., 2023)	*ICLR'23*	48.6	43.7	-	1138	52.5	50.0	57.1	45.3	-	70.1%
FastV (Chen et al., 2024a)	*ECCV'24*	46.1	48.0	52.7	1256	48.0	51.1	55.0	47.8	50.8	76.7%
SparseVLM (Zhang et al., 2024b)	*arXiv'24*	52.7	56.2	46.1	1505	75.1	62.2	68.2	51.8	50.1	87.2%
PyramidDrop (Xing et al., 2025)	*CVPR'25*	47.5	58.8	50.5	1561	55.9	**69.2**	69.2	50.6	50.7	86.6%
VisionZip (Yang et al., 2025)	*CVPR'25*	55.1	60.1	-	1690	77.0	69.0	72.4	55.5	-	92.7%
VScan (Ours)	-	**58.3**	**62.1**	**55.7**	**1698**	**85.0**	69.1	**75.4**	**55.6**	**51.8**	**96.7%**

head $h \in [1, H]$, D is the hidden state size, and $S^h_{[\texttt{CLS}]}$ represents the [CLS] attention. The global tokens are selected by

$$\mathbf{x}^{\texttt{g}}_V = \left\{ x^i_V \in \mathbf{x}_V \,\middle|\, S^{\texttt{avg}}_{[\texttt{CLS}]} \geq \tau \right\},$$
$$\text{where } S^{\texttt{avg}}_{[\texttt{CLS}]} = \frac{1}{H} \sum_{h=1}^{H} S^h_{[\texttt{CLS}]}. \tag{2}$$

Here, τ is a soft threshold based on a top percentile of attention scores, set to retain a target number of tokens. Note that for LVLMs without a [CLS] token (*e.g.*, Qwen-2.5-VL (Bai et al., 2025)), we can similarly select the tokens using self-attention, *i.e.*, the average attention each visual token receives from others.

Local Scan. To complement the global tokens and capture finer local details, we divide the image into non-overlapping windows and select the locally important tokens with the highest [CLS] attention from the shallow layer l within each window. Specifically, we allocate token budgets uniformly across windows, and select local tokens from each window:

$$\mathbf{x}^{\texttt{l}}_V = \bigcup_{w=1}^{W} \left\{ x^j_V \in \mathbf{x}^w_V \,\middle|\, S^{\texttt{avg}}_{[\texttt{CLS}]} \geq \tau_w \right\} \tag{3}$$

where w denotes the window index, $\mathbf{x}^w_V$ represents the set of all tokens within the window, and τ_w is the soft threshold for window w. The final set of selected tokens is the union of global and local tokens, $\mathbf{x}^{\texttt{selected}}_V = \mathbf{x}^{\texttt{g}}_V \cup \mathbf{x}^{\texttt{l}}_V$, resulting in a retention rate of $R_1\%$. By default, we balance the selection such that $|\mathbf{x}^{\texttt{g}}_V| = |\mathbf{x}^{\texttt{l}}_V|$, *i.e.*, half of the retained tokens are global and half are local.

Token Merging. To alleviate information loss, we introduce a similarity-based token merging strategy that merges unselected visual tokens with their most similar selected counterparts. Specifically, for each unselected token x^u_V, we identify its most similar selected token $x^s_V \in \mathbf{x}^{\texttt{selected}}_V$ based on the highest cosine similarity. Once all unselected tokens are assigned to their closest selected tokens, we apply average merging (Bolya et al., 2023) within each group to obtain the final merged representation $\mathbf{x}^{\texttt{merged}}_V$.

2.3 Middle Layer Pruning

After selecting visually significant tokens, we further refine the token set based on their relevance to the text query. Building on the empirical insights from Appendix B, we design our approach to prune tokens at the mature middle layers of the LLM, aiming to avoid position bias, preserve cross-modal interactions, and minimize the impact on final predictions. Specifically, we compute the attention between all visual tokens and the last instruction token at middle layer k, denoted as

$$Q_T = x^{\texttt{last}}_T W^h_Q, \quad K_V = \mathbf{x}^{\texttt{merged}}_V W^h_K,$$
$$S^h_{\texttt{text}} = \texttt{Softmax}\left(\frac{Q_T K_V^\top}{\sqrt{D}} \right). \tag{4}$$

We similarly average the attention scores across attention heads and select $R_2\%$ textually relevant tokens with the highest average text attention. This allows us to retain a set of visual tokens that are both visually significant and textually relevant, contributing the most to an accurate response.

3 Experiments

In this section, we validate the effectiveness of our VScan on four LVLMs, evaluating its performance across various benchmarks and comparing it with other state-of-the-arts. Please see Appendix D.2 for detailed experimental settings.

3.1 Results and Discussions

Results on LLaVA-1.5. In Table 1, we apply our approach to LLaVA-1.5-7B and compare its performance with other baselines across 9 image understanding tasks. With only 128 and 192 tokens per image instead of the original 576,

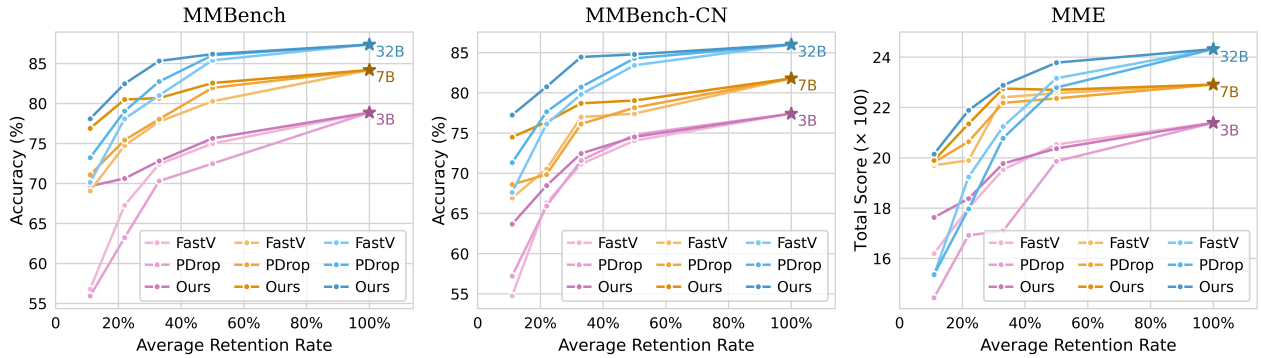

Figure 2: **Performance comparisons on Qwen-2.5-VL (Bai et al., 2025) with different LLM sizes (3B/7B/32B) across 3 image understanding benchmarks**. We present the performance of different approaches at 4 various retention rates.

our approach nearly retains the performance of the original LLaVA-1.5, with only negligible performance declines of 1.0% and 1.2%, respectively. Our approach becomes even more advantageous with higher reduction rates: With an aggressive 88.9% reduction rate, our approach results in only a 3.3% degradation in average performance across benchmarks, outperforming the second-best VisionZip by a substantial margin of 4.0%. These results suggest that our method effectively maintains high performance while reducing the number of visual tokens processed.

Results on LLaVA-NeXT. In Table C1, we further compare the performance achieved by our approach with other state-of-the-arts on the more advanced LLaVA-NeXT-7B. We present the performance of all approaches under a fixed budget of 320 tokens per image, corresponding to an 88.9% reduction rate. Our approach continues to achieve superior performance on LLaVA-NeXT-7B, attaining the best performance on 6 out of 9 benchmarks, and achieving 95.4% of the original LLaVA-NeXT-7B performance with only 11.1% of the token budget without additional training.

Results on Qwen-2.5-VL. To further validate the general effectiveness of our approach, we apply it to the recent Qwen-2.5-VL with three different LLM scales and visualize the performance on three image understanding benchmarks in Figure 2. As shown, our approach consistently outperforms FastV and PDrop across all retention rates and model scales.

We extend our comparisons to more challenging grounding tasks and present the performance of different approaches in Table C2. Compared to image understanding tasks, these grounding tasks require higher token budgets to preserve visual information necessary for precise localization. A 75% reduction rate, for instance, halves the performance of FastV and PDrop. In this challenging scenario, our approach still robustly maintains 80.7% of the original performance.

Results on Video-LLaVA. In Appendix C.3, we further assess the effectiveness of our approach on video understanding tasks and compare its performance against other approaches on Video-LLaVA-7B. The results illustrate that with a 25% token budget, our approach maintains nearly 100% of the original Video-LLaVA-7B's performance, consistently outperforming other methods.

Table 2: **Efficiency comparisons on the POPE benchmark**. We report the theoretical FLOPs, actual runtime, KV cache compression rate (%), and the achieved accuracy.

Method	FLOPs ↓	Total Time ↓	Prefill Time ↓	KV Cache ↓	Accuracy ↑
LLaVA-1.5-7B	3.817 T	1113 s (1.00×)	416 s (1.00×)	100.0%	85.9
+ **Ours** (33%)	1.253 T	937 s (1.19×)	301 s (1.38×)	39.9%	**86.2**
+ **Ours** (11%)	**0.415** T	**812** s (1.37×)	**235** s (1.77×)	**19.9%**	85.0
LLaVA-NeXT-7B	20.825 T	2294 s (1.00×)	1420 s (1.00×)	100.0%	86.5
+ **Ours** (33%)	6.459 T	1701 s (1.35×)	994 s (1.43×)	34.8%	**86.1**
+ **Ours** (11%)	**2.099** T	**1120** s (2.05×)	**488** s (2.91×)	**13.1%**	85.1

3.2 Efficiency Analysis

In Table 2, we evaluate the practical acceleration effects of VScan. By retaining only 11% of the visual tokens, VScan achieves a 1.37× speedup in overall efficiency and a 1.77× speedup in prefilling efficiency on LLaVA-1.5-7B, while maintaining robust performance with only a 0.9% decline. Our approach achieves even more significant acceleration on LLaVA-NeXT-7B, where it delivers a 2.05× speedup in inference and a 2.91× speedup in the prefill stage. Additionally, our approach can also effectively compress KV cache storage across different backbones.

4 Conclusion

In this work, we present a comprehensive empirical study to understand how visual information is processed across both the visual encoding and LLM decoding stages. Building on these insights, we propose VScan—a two-stage, training-free visual token reduction framework—to accelerate LVLM inference while maintaining robust performance. Extensive experiments across 4 LVLM architectures and 16 image and video benchmarks demonstrate that our approach consistently outperforms existing state-of-the-art methods, achieving a superior trade-off between efficiency and accuracy.

References

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., Ring, R., Rutherford, E., Cabi, S., Han, T., Gong, Z., Samangooei, S., Monteiro, M., Menick, J. L., Borgeaud, S., Brock, A., Nematzadeh, A., Sharifzadeh, S., Bińkowski, M. a., Barreira, R., Vinyals, O., Zisserman, A., and Simonyan, K. Flamingo: a visual language model for few-shot learning. In *Advances in Neural Information Processing Systems*, volume 35, pp. 23716–23736, 2022.

Alexey, D., Fischer, P., Tobias, J., Springenberg, M. R., and Brox, T. Discriminative unsupervised feature learning with exemplar convolutional neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(9):1734–1747, 2015.

Arif, K. H. I., Yoon, J., Nikolopoulos, D. S., Vandierendonck, H., John, D., and Ji, B. Hired: Attention-guided token dropping for efficient inference of high-resolution vision-language models in resource-constrained environments. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 1773–1781, 2025.

Bai, J., Bai, S., Yang, S., Wang, S., Tan, S., Wang, P., Lin, J., Zhou, C., and Zhou, J. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023.

Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., Zhong, H., Zhu, Y., Yang, M., Li, Z., Wan, J., Wang, P., Ding, W., Fu, Z., Xu, Y., Ye, J., Zhang, X., Xie, T., Cheng, Z., Zhang, H., Yang, Z., Xu, H., and Lin, J. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Bolya, D., Fu, C.-Y., Dai, X., Zhang, P., Feichtenhofer, C., and Hoffman, J. Token merging: Your vit but faster. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=JroZRaRw7Eu.

Brauwers, G. and Frasincar, F. A general survey on attention mechanisms in deep learning. *IEEE Transactions on Knowledge and Data Engineering*, 35(4):3279–3298, 2021.

Chen, D. and Dolan, W. B. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pp. 190–200, 2011.

Chen, K., Zhang, Z., Zeng, W., Zhang, R., Zhu, F., and Zhao, R. Shikra: Unleashing multimodal llm's referential dialogue magic. *arXiv preprint arXiv:2306.15195*, 2023.

Chen, L., Zhao, H., Liu, T., Bai, S., Lin, J., Zhou, C., and Chang, B. An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models. In *European Conference on Computer Vision*, pp. 19–35. Springer, 2024a.

Chen, Z., Wu, J., Wang, W., Su, W., Chen, G., Xing, S., Zhong, M., Zhang, Q., Zhu, X., Lu, L., Li, B., Luo, P., Lu, T., Qiao, Y., and Dai, J. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24185–24198, 2024b.

Chung, H. W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., Li, Y., Wang, X., Dehghani, M., Brahma, S., Webson, A., Gu, S. S., Dai, Z., Suzgun, M., Chen, X., Chowdhery, A., Castro-Ros, A., Pellat, M., Robinson, K., Valter, D., Narang, S., Mishra, G., Yu, A., Zhao, V., Huang, Y., Dai, A., Yu, H., Petrov, S., Chi, E. H., Dean, J., Devlin, J., Roberts, A., Zhou, D., Le, Q. V., and Wei, J. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53, 2024.

Dai, W., Li, J., Li, D., Tiong, A. M. H., Zhao, J., Wang, W., Li, B., Fung, P., and Hoi, S. Instructblip: towards general-purpose vision-language models with instruction tuning. In *Advances in Neural Information Processing Systems*, pp. 49250–49267, 2023.

Dao, T. Flashattention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=mZn2Xyh9Ec.

Dao, T., Fu, D., Ermon, S., Rudra, A., and Ré, C. Flashattention: Fast and memory-efficient exact attention with io-awareness. In *Advances in Neural Information Processing Systems*, volume 35, pp. 16344–16359, 2022.

Fang, X., Mao, K., Duan, H., Zhao, X., Li, Y., Lin, D., and Chen, K. Mmbench-video: A long-form multi-shot benchmark for holistic video understanding. In *Advances in Neural Information Processing Systems*, volume 37, pp. 89098–89124, 2024.

Fu, C., Chen, P., Shen, Y., Qin, Y., Zhang, M., Lin, X., Yang, J., Zheng, X., Li, K., Sun, X., Wu, Y., and Ji, R. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023.

Fu, X., Hu, Y., Li, B., Feng, Y., Wang, H., Lin, X., Roth, D., Smith, N. A., Ma, W.-C., and Krishna, R. Blink: Multimodal large language models can see but not perceive. In *European Conference on Computer Vision*, pp. 148–166. Springer, 2024.

Gandelsman, Y., Efros, A. A., and Steinhardt, J. Interpreting CLIP's image representation via text-based decomposition. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=5Ca9sSzuDp.

Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., and Parikh, D. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6904–6913, 2017.

Gurari, D., Li, Q., Stangl, A. J., Guo, A., Lin, C., Grauman, K., Luo, J., and Bigham, J. P. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3608–3617, 2018.

Hudson, D. A. and Manning, C. D. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6700–6709, 2019.

Jaegle, A., Gimeno, F., Brock, A., Vinyals, O., Zisserman, A., and Carreira, J. Perceiver: General perception with iterative attention. In *International Conference on Machine Learning*, pp. 4651–4664. PMLR, 2021.

Jang, Y., Song, Y., Yu, Y., Kim, Y., and Kim, G. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2758–2766, 2017.

Kazemzadeh, S., Ordonez, V., Matten, M., and Berg, T. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 787–798, 2014.

Kondratyuk, D., Yuan, L., Li, Y., Zhang, L., Tan, M., Brown, M., and Gong, B. Movinets: Mobile video networks for efficient video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16020–16030, 2021.

Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Zhang, P., Li, Y., Liu, Z., and Li, C. LLaVA-onevision: Easy visual task transfer. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=zKv8qULV6n.

Li, J., Li, D., Savarese, S., and Hoi, S. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning*, pp. 19730–19742. PMLR, 2023a.

Li, Y., Du, Y., Zhou, K., Wang, J., Zhao, W. X., and Wen, J.-R. Evaluating object hallucination in large vision-language models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 292–305, 2023b.

Li, Y., Zhang, Y., Wang, C., Zhong, Z., Chen, Y., Chu, R., Liu, S., and Jia, J. Mini-gemini: Mining the potential of multi-modality vision language models. *arXiv preprint arXiv:2403.18814*, 2024.

Lin, B., Ye, Y., Zhu, B., Cui, J., Ning, M., Jin, P., and Yuan, L. Video-llava: Learning united visual representation by alignment before projection. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 5971–5984, 2024a.

Lin, Z., Lin, M., Lin, L., and Ji, R. Boosting multimodal large language models with visual tokens withdrawal for rapid inference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024b.

Liu, H., Li, C., Wu, Q., and Lee, Y. J. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916, 2023.

Liu, H., Li, C., Li, Y., and Lee, Y. J. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26296–26306, 2024a.

Liu, H., Li, C., Li, Y., Li, B., Zhang, Y., Shen, S., and Lee, Y. J. Llava-next: Improved reasoning, ocr, and world knowledge, January 2024b. URL https://llava-vl.github.io/blog/2024-01-30-llava-next/.

Liu, Y., Duan, H., Zhang, Y., Li, B., Zhang, S., Zhao, W., Yuan, Y., Wang, J., He, C., Liu, Z., Chen, K., and Lin, D. Mmbench: Is your multi-modal model an all-around player? In *European Conference on Computer Vision*, pp. 216–233. Springer, 2024c.

Lu, P., Mishra, S., Xia, T., Qiu, L., Chang, K.-W., Zhu, S.-C., Tafjord, O., Clark, P., and Kalyan, A. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *Advances in Neural Information Processing Systems*, volume 35, pp. 2507–2521, 2022.

Maaz, M., Rasheed, H., Khan, S., and Khan, F. Video-chatgpt: Towards detailed video understanding via large vision and language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pp. 12585–12602, 2024.

Mao, J., Huang, J., Toshev, A., Camburu, O., Yuille, A. L., and Murphy, K. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 11–20, 2016.

Mehta, S. and Rastegari, M. Mobilevit: Light-weight, general-purpose, and mobile-friendly vision transformer. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=vh-0sUt8HlG.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

Shang, Y., Cai, M., Xu, B., Lee, Y. J., and Yan, Y. Llava-prumerge: Adaptive token reduction for efficient large multimodal models. *arXiv preprint arXiv:2403.15388*, 2024.

Singh, A., Natarajan, V., Shah, M., Jiang, Y., Chen, X., Batra, D., Parikh, D., and Rohrbach, M. Towards vqa models that can read. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8317–8326, 2019.

Team, G., Georgiev, P., Lei, V. I., Burnell, R., Bai, L., Gulati, A., Tanzer, G., Vincent, D., Pan, Z., Wang, S., et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*, pp. 6000–6010, 2017.

Wang, H., Yu, Z., Spadaro, G., Ju, C., Quétu, V., and Tartaglione, E. Folder: Accelerating multi-modal large language models with enhanced performance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Wen, Z., Gao, Y., Wang, S., Zhang, J., Zhang, Q., Li, W., He, C., and Zhang, L. Stop looking for important tokens in multimodal language models: Duplication matters more. *arXiv preprint arXiv:2502.11494*, 2025.

Xing, L., Huang, Q., Dong, X., Lu, J., Zhang, P., Zang, Y., Cao, Y., He, C., Wang, J., Wu, F., et al. Pyramiddrop: Accelerating your large vision-language models via pyramid visual redundancy reduction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Xu, J., Mei, T., Yao, T., and Rui, Y. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5288–5296, 2016.

Yang, S., Chen, Y., Tian, Z., Wang, C., Li, J., Yu, B., and Jia, J. Visionzip: Longer is better but not necessary in vision language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Ye, X., Gan, Y., Ge, Y., Zhang, X.-P., and Tang, Y. Atp-llava: Adaptive token pruning for large vision language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Yu, Z., Xu, D., Yu, J., Yu, T., Zhao, Z., Zhuang, Y., and Tao, D. Activitynet-qa: A dataset for understanding complex web videos via question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 9127–9134, 2019.

Zhang, C., Wan, Z., Kan, Z., Ma, M. Q., Stepputtis, S., Ramanan, D., Salakhutdinov, R., Morency, L.-P., Sycara, K. P., and Xie, Y. Self-correcting decoding with generative feedback for mitigating hallucinations in large vision-language models. In *International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=tTBXePRKSx.

Zhang, J., Yao, D., Pi, R., Liang, P. P., and Fung, Y. R. VLM^2-Bench: A closer look at how well vlms implicitly link explicit matching visual cues. *arXiv preprint arXiv:2502.12084*, 2025b.

Zhang, Q., Cheng, A., Lu, M., Zhuo, Z., Wang, M., Cao, J., Guo, S., She, Q., and Zhang, S. [CLS] attention is all you need for training-free visual token pruning: Make vlm inference faster. *arXiv preprint arXiv:2412.01818*, 2024a.

Zhang, Y., Fan, C.-K., Ma, J., Zheng, W., Huang, T., Cheng, K., Gudovskiy, D., Okuno, T., Nakata, Y., Keutzer, K., and Zhang, S. Sparsevlm: Visual token sparsification for efficient vision-language model inference. *arXiv preprint arXiv:2410.04417*, 2024b.

Appendix

In the appendix, we provide additional details and experimental results to enhance understanding and insights into our method. The appendix is organized as follows:

- Section A reviews recent work that are related to our research, including recent advances in visual token pruning and efficient LVLMs.
- Section B provides a comprehensive analysis of how LVLMs process visual tokens during both the visual encoding and language decoding, offering empirical guidance for designing more effective visual token reduction strategies.
- Section C presents additional experimental results that further validate the effectiveness and robustness of our approach across various settings.
- Section D provides extended experimrntal details, including FLOPs calculation and full experimental configurations, to facilitate reproducibility.
- Section E lists the license information for all models, baselines, and benchmarks used in this work.
- Section F discusses the limitations of this work and explores its broader implications and impacts.

A Related Work

Efficient Large Vision-Language Models. Building on powerful auto-regressive LLMs (Achiam et al., 2023; Touvron et al., 2023; Chung et al., 2024), recent LVLMs typically adopt an encoder-projector-decoder architecture, where visual inputs are encoded into tokens and jointly processed with language sequences (Liu et al., 2023; Lin et al., 2024a; Bai et al., 2023; Chen et al., 2024b; Team et al., 2024; Chen et al., 2023). However, as image resolution increases or the input scales to multi-image/video, the number of visual tokens grows proportionally, leading to a quadratic increase in computation cost and runtime due to the self-attention mechanism (Vaswani et al., 2017; Brauwers & Frasincar, 2021), which limits the scalability of LVLMs in real-world applications (Bolya et al., 2023; Chen et al., 2024a; Mehta & Rastegari, 2022; Kondratyuk et al., 2021; Zhang et al., 2025b;a). To mitigate this issue, several LVLMs introduced specialized modules to enhance efficiency—such as the Q-Former in InstructBLIP (Dai et al., 2023) and the perceiver resampler (Jaegle et al., 2021) in OpenFlamingo (Alayrac et al., 2022)—that distill dense visual inputs into a compact set of features before LLM decoding. Orthogonal to these architectural strategies, FlashAttention (Dao et al., 2022; Dao, 2024) has emerged as a widely adopted, hardware-aware optimization that accelerates attention computation by minimizing redundant memory access, offering substantial speedups without compromising performance.

Vision Token Reduction in LVLMs. Another line of work aims to improve model efficiency on the sequence dimension—pioneering works such as ToMe (Bolya et al., 2023) and FastV (Chen et al., 2024a) have explored strategies like visual token merging and text-guided pruning to improve the efficiency of LVLMs. Building on these advances, subsequent approaches can be broadly divided into two main categories: (1) *Text-agnostic pruning approaches* (Shang et al., 2024; Arif et al., 2025; Zhang et al., 2024a; Yang et al., 2025), which identify and remove redundant or uninformative visual tokens during the visual encoding stage. For instance, VisionZip (Yang et al., 2025) selects dominant tokens based on `[CLS]` attention scores, while FOLDER (Wang et al., 2025) introduces token merging with reduction overflow in the final blocks of the visual encoder. (2) *Text-aware pruning approaches* (Zhang et al., 2024b; Xing et al., 2025), which aim to remove visual tokens that are irrelevant to the text query during the LLM decoding stage. For instance, SparseVLM (Zhang et al., 2024b) proposes an iterative sparsification strategy that selects visual-relevant text tokens to rate the significance of vision tokens, and PyramidDrop (Xing et al., 2025) performs progressive pruning at multiple decoding layers to balance efficiency and context preservation. In this work, we present a comprehensive analysis of how LVLMs process visual tokens during both the visual encoding and language decoding stages, and propose a corresponding two-stage approach, VScan, to effectively improve the inference efficiency of LVLMs while maintaining robust performance.

B Empirical Analysis

In this section, we provide a comprehensive analysis of how LVLMs process visual tokens during both the visual encoding and language decoding, offering empirical guidance for designing more effective visual token reduction strategies.

Preliminary: Architecture of LVLMs. We consider an LVLM parameterized by θ, which consists of three major components: a visual encoder, a feature projector, and an LLM decoder. Given an image input, the visual encoder processes

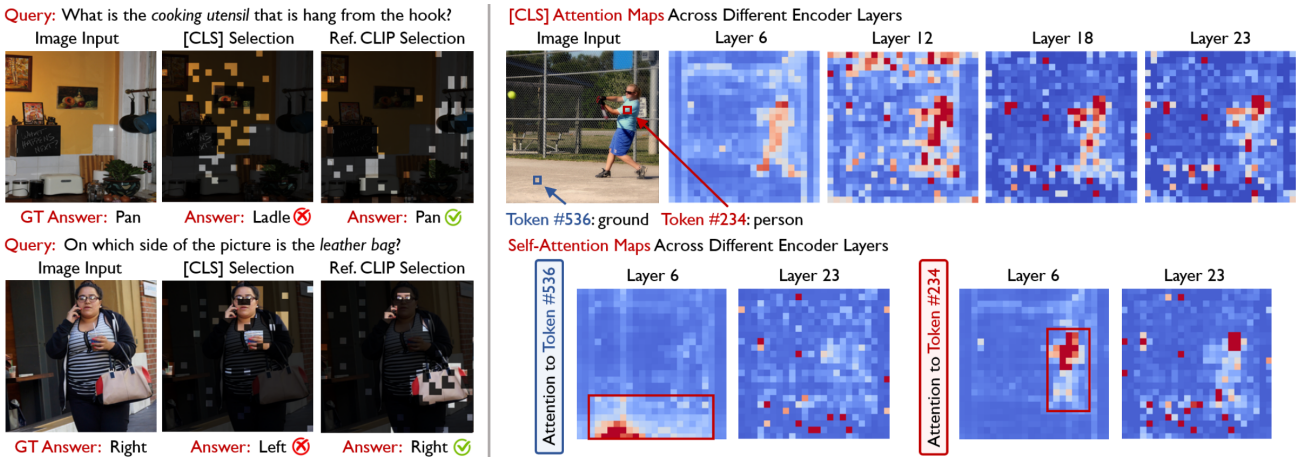

Figure B1: **Empirical study on visual redundancy reduction**. (*Left*) We illustrate two failure cases where relying solely on the output [CLS] attention leads to incorrect predictions. For comparison, we include reference token selections from CLIP-ViT-L-336px (Radford et al., 2021), following Gandelsman *et al.* (Gandelsman et al., 2024), which highlight regions of interest relevant to the text query. (*Right*) We visualize the [CLS] attention maps and self-attention maps of representative tokens (*e.g.*, #536: *ground*, #234: *person*) across different visual encoding layers, illustrating how attention patterns evolve from localized focus in shallow layers to broader global context in deeper layers.

the image patches, and the projector converts them into n visual tokens $\mathbf{x}_V = \{x_V^i\}_{i=1}^n$. These visual tokens are then concatenated with the tokenized textual query $\mathbf{x}_T$ and fed into the LLM decoder for auto-regressive next-token generation, represented as $y_t \sim p_\theta(y_t|\mathbf{x}_V, \mathbf{x}_T, \mathbf{y}_{<t})$, where the next token y_t is sampled from the output probability distribution $p_\theta(\cdot)$, and $\mathbf{y}_{<t}$ denotes the sequence of tokens generated prior to timestep t.

Rethinking Visual Redundancy Reduction. To address visual redundancy in token representations, recent works (Zhang et al., 2024a; Yang et al., 2025) have proposed *text-agonostic approaches* that retain visual tokens with high [CLS] attention at the output layer of the ViT-based visual encoder. However, this raises a critical question: *Is relying solely on output [CLS] attention truly sufficient to capture all task-relevant visual information?* Upon closer examination, we identify a clear yet often overlooked limitation of these approaches: they tend to favor tokens corresponding to visually salient objects, while aggressively discarding background visual details that may carry essential semantic information. As illustrated in Figure B1 (*Left*), output [CLS] attention is incorrectly directed to the *wall* and *person*, ignoring the actual targets—the *pan* and *leather bag*—leading to incorrect model responses.

To better understand and overcome this limitation, we analyze how visual information is processed across the visual encoding layers in LVLMs. Specifically, we visualize both the [CLS] attention and self-attention of representative tokens across different visual encoding layers, as illustrated in Figure B1 (*Right*). Our observations are as follows: (1) In the shallow layers, the [CLS] attention maps capture fine-grained local details across the image. In contrast, in the deeper layers, the attention becomes increasingly concentrated on the main entities, reflecting their global semantic relevance; (2) The self-attention maps for representative visual tokens reveal a similar local-to-global trend: in the shallow layers, these tokens primarily attend to nearby regions with similar semantic meaning, while in the deeper layers, their attention becomes more dispersed, integrating context from the entire image. These findings highlight a gradual transition in the visual encoder from capturing low-level local details to modeling high-level, globally relevant semantics, suggesting that relying solely on the output layer may overlook the rich local information encoded in the shallow layers.

Rethinking Textual Irrelevance Reduction. While prior works (Chen et al., 2024a; Zhang et al., 2024b; Lin et al., 2024b) have introduced effective *text-aware approaches* for pruning visual tokens at early layers during LLM decoding, a critical question remains: *Are early layers the optimal stage for pruning visual tokens to minimize their impact on the model's final response?* To investigate this, we conduct three empirical studies on POPE (Li et al., 2023b) and GQA (Hudson & Manning, 2019), analyzing how the model's knowledge and predictions evolve during the decoding process:

- **Study 1**: *How does position bias in token selection evolve across LLM layers?* Specifically, we visualize the distribution of the retained tokens selected by the attention score of the last instruction token (Chen et al., 2024a) across LLM layers

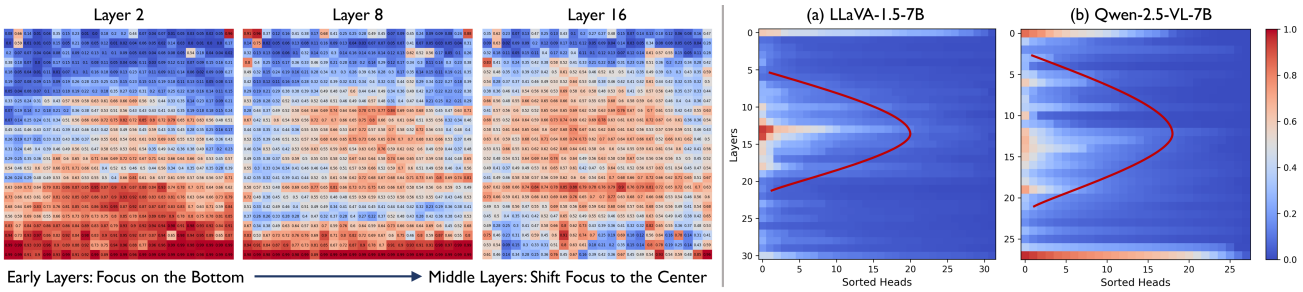

Figure B2: (*Left*) **Study 1**: Distribution of retained tokens at a 50% reduction rate in layers 2, 8, and 16 of LLaVA-1.5-7B on POPE (Li et al., 2023b); (*Right*) **Study 2**: Sum of visual attention across different attention heads and LLM layers using LLaVA-1.5-7B and Qwen-2.5-VL-7B on POPE (Li et al., 2023b).

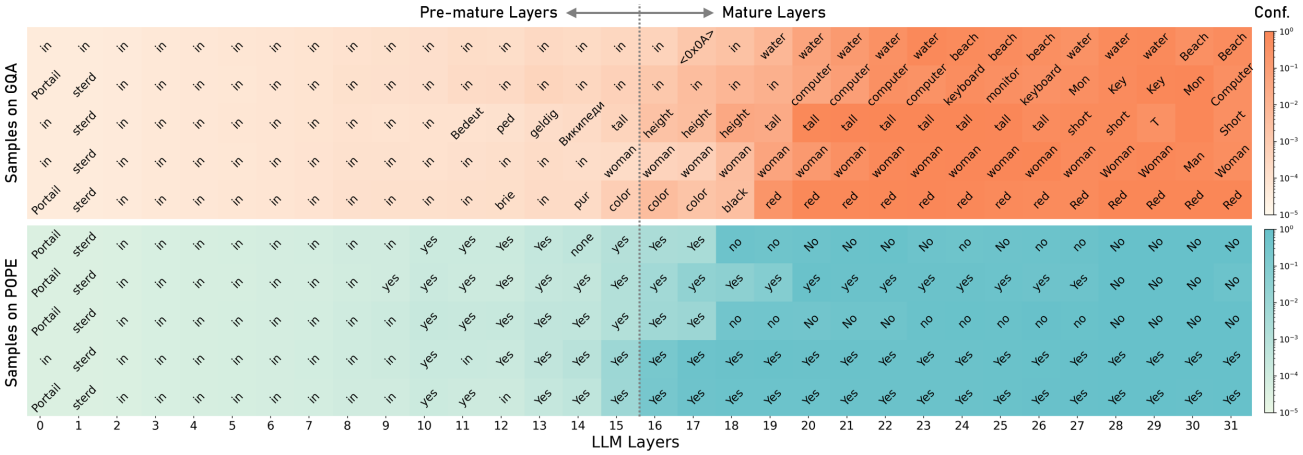

Figure B3: **Study 3**: Visualization of next-token predictions derived from the output hidden states of each LLM layer using LLaVA-1.5-7B. Darker colors indicate higher prediction confidence.

using LLaVA-1.5-7B. As shown in Figure B2 (*Left*), early layers (*e.g.*, layers 2 and 8) tend to select tokens at the bottom of the image, reflecting an *inherent LLM position bias*, as the last instruction token primarily attends to nearby tokens and focuses on local context (Vaswani et al., 2017), and flattened visual tokens from the bottom of the image are positioned closest to the instruction tokens in the sequence. As the LLM layers deepen, this undesirable position bias diminishes and the focus shifts toward the center of the image, which is more intuitive since the center of the image typically carries the most informative and task-relevant features (Alexey et al., 2015).

- **Study 2**: *From which layer does the LLM begin to gather and process visual information?* We visualize the sum of attention received by all visual tokens from the last instruction token across different LLM layers using LLaVA-1.5-7B and Qwen-2.5-VL-7B in Figure B2 (*Right*). The red curve in each plot highlights the layer-wise attention patterns directed towards visual information. We observe that the middle LLM layers are primarily responsible for interacting with the visual tokens, whereas the early and deep layers focus predominantly on processing textual information.

- **Study 3**: *At which LLM layer do next-token predictions begin to converge?* In Figure B3, we provide an interpretation of the hidden states across different LLM layers in LLaVA-1.5-7B. Specifically, we feed the hidden states from each LLM decoding layer into the final linear projection layer to obtain vocabulary logits and intermediate next-token predictions. We observe that in more challenging open-ended tasks like GQA, the next-token predictions stabilize around LLM layer 20, whereas in simpler yes/no tasks such as POPE, the predictions converge earlier, around LLM layer 16.

These findings collectively suggest that early layers are suboptimal for pruning due to position bias and limited engagement with visual content. In contrast, pruning at middle layers is more appropriate as it better preserves critical cross-modal interactions and minimizes disruption to model predictions.

Table C1: **Performance comparisons on LLaVA-NeXT-7B (Liu et al., 2024b) across 9 image understanding benchmarks**. The best results in each setting are **bolded**, and the second-best are underlined.

Method	Venue	GQA	MMB	MMB^{CN}	MME	POPE	SQA^{IMG}	VQA^{V2}	VQA^{Text}	VizWiz	Average
Upper Bound, 2,880 Tokens (100%), ∼20.825 TFLOPs											
LLaVA-NeXT-7B (Liu et al., 2024b)	*CVPR'24*	64.2	67.4	60.6	1851	86.5	70.1	81.8	61.3	57.6	100.0%
Retain 320 Tokens in Average (↓ 88.9%), ∼2.099 TFLOPs											
FastV (Chen et al., 2024a)	*ECCV'24*	55.9	61.6	51.9	1661	71.7	62.8	72.9	55.7	53.1	88.7%
HiRED (Arif et al., 2025)	*AAAI'25*	59.3	64.2	55.9	1690	83.3	66.7	75.7	58.8	**54.2**	93.9%
PyramidDrop (Xing et al., 2025)	*CVPR'25*	56.4	63.4	56.2	1663	77.6	**67.5**	73.5	54.4	54.1	91.4%
VisionZip (Yang et al., 2025)	*CVPR'25*	59.3	63.1	-	1702	82.1	67.3	76.2	**58.9**	-	94.0%
VScan (Ours)	-	**60.7**	**65.3**	**57.8**	**1767**	**85.1**	66.9	**77.1**	58.0	53.8	**95.4%**

Table C2: **Performance comparisons on Qwen-2.5-VL-7B (Bai et al., 2025) across 3 referring expression comprehension benchmarks: RefCOCO, RefCOCO+, and RefCOCOg**. The best results in each setting are **bolded**, and the second-best are underlined. †Evaluation is based on our re-implementation.

Method	Venue	RefCOCO			RefCOCO+			RefCOCOg		Average
		val	testA	testB	val	testA	testB	val	test	
Upper Bound, 4∼16384 Tokens (100%)										
Qwen-2.5-VL-7B (Bai et al., 2025)	*arXiv'25*	89.45	92.56	85.16	83.50	89.02	79.15	86.76	87.24	100.0%
Retain 75% Tokens in Average (↓ 25%)										
FastV† (Chen et al., 2024a)	*ECCV'24*	85.27	87.84	82.28	79.02	82.95	72.86	82.95	83.32	94.8%
PyramidDrop† (Xing et al., 2025)	*CVPR'25*	87.79	91.00	83.22	81.48	86.55	74.02	84.62	85.10	97.2%
VScan (Ours)	-	**88.75**	**91.94**	**83.96**	**82.39**	**87.90**	**74.15**	**85.54**	**86.55**	**98.3%**
Retain 50% Tokens in Average (↓ 50%)										
FastV† (Chen et al., 2024a)	*ECCV'24*	73.85	73.38	74.21	66.75	68.88	62.65	71.06	71.86	81.2%
PyramidDrop† (Xing et al., 2025)	*CVPR'25*	77.52	80.82	72.07	70.27	75.48	63.33	74.86	75.65	85.1%
VScan (Ours)	-	**86.78**	**90.74**	**82.37**	**79.99**	**86.12**	**71.67**	**84.03**	**84.44**	**96.1%**
Retain 25% Tokens in Average (↓ 75%)										
FastV† (Chen et al., 2024a)	*ECCV'24*	43.57	46.81	40.86	39.47	43.78	36.02	43.04	42.69	48.5%
PyramidDrop† (Xing et al., 2025)	*CVPR'25*	46.46	53.83	37.23	42.29	47.76	32.81	45.32	44.91	50.4%
VScan (Ours)	-	**74.32**	**79.05**	**68.22**	**67.22**	**73.72**	**58.95**	**69.42**	**69.43**	**80.7%**

C Additional Experimental Results

C.1 Results on LLaVA-NeXT

In Table C1, we further compare the performance achieved by our approach with other state-of-the-arts on the more advanced LLaVA-NeXT-7B (Liu et al., 2024b). We present the performance of all approaches under a fixed budget of 320 tokens per image, corresponding to an 88.9% reduction rate. Our approach continues to achieve superior performance on LLaVA-NeXT-7B (Liu et al., 2024b), attaining the best performance on 6 out of 9 benchmarks, and achieving 95.4% of the original LLaVA-NeXT-7B (Liu et al., 2024b) performance with only 11.1% of the token budget without additional training.

C.2 Results on Qwen-2.5-VL

We extend our comparisons to more challenging grounding tasks and present the performance of different approaches in Table C2. Compared to image understanding tasks, these grounding tasks require higher token budgets to preserve visual information necessary for precise localization. A 75% reduction rate, for instance, halves the performance of FastV (Chen et al., 2024a) and PDrop (Xing et al., 2025). In this challenging scenario, our approach still robustly maintains 80.7% of the original performance. Additionally, our approach achieves 96.1% of the original performance with only 50% of the visual tokens. These results demonstrate that our approach is versatile and can be effectively applied to various vision-language tasks, offering an excellent performance-efficiency trade-off.

C.3 Results on Video-LLaVA

Finally, we validate the effectiveness of our approach on video understanding tasks and compare its performance against other approaches on Video-LLaVA-7B (Lin et al., 2024a), as shown in Table C3. Specifically, we report the accuracy and GPT-evaluated scores for each benchmark to assess the quality of the responses. The results illustrate that with a 25% token budget, our approach maintains nearly 100% of the original Video-LLaVA-7B's performance, consistently outperforming other methods.

Table C3: **Performance comparisons on Video-LLaVA-7B (Lin et al., 2024a) across 4 video understanding tasks with a 75% reduction rate**. The best results are **bolded**, and the second-best are underlined. †Evaluation is based on our re-implementation.

Method	TGIF		MSVD		MSRVTT		ActivityNet	
	Acc.	Score	Acc.	Score	Acc.	Score	Acc.	Score
Video-LLaVA-7B (Lin et al., 2024a)	47.0	3.40	70.5	3.92	58.3	3.51	42.2	3.37
FastV† (Chen et al., 2024a)	42.7	3.19	67.4	3.83	53.6	3.40	36.1	3.15
PyramidDrop† (Xing et al., 2025)	44.1	3.26	66.7	3.81	56.1	3.45	37.4	3.15
VScan (Ours)	**46.9**	**3.35**	**69.8**	**3.93**	**57.1**	**3.48**	**42.6**	**3.34**

C.4 Empirical Validation of Global and Local Scan

To validate the effectiveness of our global and local scan schemes, we construct adversarial subsets for GQA and POPE, namely AdvGQA and AdvPOPE, which contains failure cases similar to those shown in Figure B1 (*Left*), where the text queries play an important role and relying solely on the global scan to select visual tokens leads to errors. To accurately select these samples, we follow Gandelsman *et al*. (Gandelsman et al., 2024) to decompose the image representations and pinpoint the tokens or regions most relevant to the query. Specifically, we utilize both the text and visual encoders of CLIP-ViT-L-336px (Radford et al., 2021) to identify the visual tokens relevant to the text query as a reference. Two examples of the visual tokens selected by CLIP are shown in Figure B1 (*Left*). A sample is included in the adversarial set if the response is correct when using the 64 tokens selected by CLIP, but becomes incorrect when using the 64 tokens selected solely by the global scan. Following this, we collected 886 and 515 adversarial samples in AdvGQA and AdvPOPE, respectively.

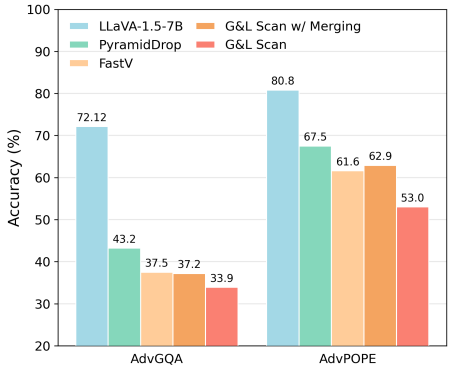

Figure C4: **Performance comparisons on AdvGQA and AdvPOPE**. We report the results for each approach with 64 visual tokens retained using LLaVA-1.5-7B (Liu et al., 2024a).

In Figure C4, we present a performance comparison of incorporating both global and local scans with FastV (Chen et al., 2024a) and PyramidDrop (Xing et al., 2025), which select visual tokens based on text attention and are expected to handle samples in the adversarial set effectively. We observe that incorporating both global and local scans achieves performance comparable to these text-guided approaches, despite being text-agnostic and selecting important tokens solely based on visual significance. These results validate that combining both scanning strategies and token merging helps preserve the maximum amount of visual information, effectively preventing information loss.

C.5 Empirical Validation of Middle Layer Pruning

We conduct a comparative analysis on the GQA benchmark using LLaVA-1.5-7B (Liu et al., 2024a) to examine the effect of pruning tokens at different LLM layers, while keeping the average reduction rate consistent across settings. To better highlight the impact of pruning depth, we first apply a global scan to reduce the visual tokens to 288/144 (*i.e.*, R_1 = 50%/25%), and then perform pruning at various LLM layers to reach an average retention rate of 75% during the LLM decoding stage. As shown in Table C4, pruning at middle LLM layers (*e.g.*, layers 16 or 20) yields the best performance, whereas pruning at earlier layers (*e.g.*, layer 2) leads to up to a 1.9% drop in accuracy. These results align with our empirical insights in Section B and validate the effectiveness of pruning at middle layers to remove textual irrelevance in LVLMs.

Table C4: **Comparative study of pruning visual tokens at different LLM layers**. R_1 denotes the retention rate in the visual encoding stage, while k and R_2 indicate the pruning layer and retention rate in the LLM decoding stage, respectively.

Settings	$R_1 = 50\%$	$R_1 = 25\%$
$k = 2$, $R_2 = 73.3\%$	59.6	56.8
$k = 8$, $R_2 = 66.7\%$	59.6	57.2
$k = 12$, $R_2 = 60.0\%$	59.6	58.6
$k = 16$, $R_2 = 50.0\%$	**60.7**	**58.7**
$k = 20$, $R_2 = 33.3\%$	60.6	**58.7**
$k = 24$, $R_2 = 0.0\%$	60.2	58.4

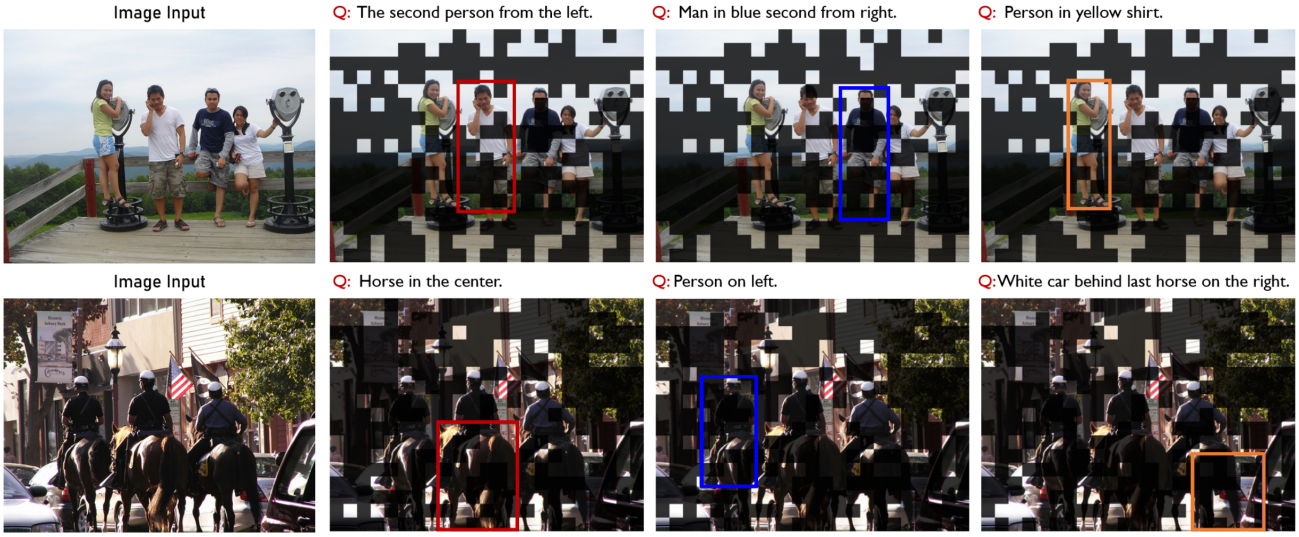

Figure C5: **Qualitative results on RefCOCO benchmark using Qwen-2.5-VL (Bai et al., 2025)**. We present the predicted boxes for 6 different queries on 2 images, along with visualizations of the retained tokens.

Table C5: **Ablation experiment using LLaVA-1.5-7B with an average reduction rate of 11.1% on GQA and MME**. We ablate the effects of (a) retention rates R_1 and R_2; (b) the proportion of global and local tokens; and (c) encoding layer l for the local scan, while keep all other settings fixed.

(a) Retention rates R_1 and R_2.

R_1	R_2	GQA	MME
11.1%	100.0%	56.7	1651
13.3%	66.7%	57.1	1683
14.8%	50.0%	57.5	1676
16.7%	33.3%	**58.3**	1698
22.2%	0.0%	52.7	**1720**

(b) Global & local tokens.

Global	Local	GQA	MME
0%	100%	58.0	1681
25%	75%	58.1	1689
50%	50%	**58.3**	**1698**
75%	25%	57.4	1688
100%	0%	57.5	1665

(c) Local scan encoding layer l.

Encoding Layer	GQA	MME
$l = 2$ (*shallow*)	57.1	**1712**
$l = 6$ (*shallow*)	**58.3**	1698
$l = 12$ (*middle*)	57.8	1692
$l = 18$ (*deep*)	57.5	1678
$l = 23$ (*output*)	57.3	1683

C.6 Qualitative Results

In Figure C5, we present qualitative examples from the RefCOCO benchmark using Qwen-2.5-VL (Bai et al., 2025). For each image, we show the model's predicted bounding boxes in response to different referring expressions, along with visualizations of the retained visual tokens after token pruning. These examples illustrate that our method can accurately localize the target objects described in queries, while significantly reducing the number of visual tokens used for inference. This demonstrates the model's ability to preserve semantically important information under token-efficient settings.

C.7 Ablation Studies

Varying Retention Rates R_1 and R_2. We analyze how different retention rate configurations affect performance by varying the retention rates R_1 and R_2 during the visual encoding and LLM decoding stages, respectively. As shown in Table C5 (a), we observe that relying solely on token selection in the visual encoder or applying overly aggressive pruning during LLM decoding leads to suboptimal performance. Instead, a more balanced and gradual two-stage pruning strategy, *i.e.*, $R_1 = 16.7\%$ and $R_2 = 33.3\%$, achieves the best performance. These results also validate the effectiveness of combining both visual token reduction strategies to jointly improve efficiency and maintain accuracy.

Mixing Global and Local Tokens. In Table C5 (b), we examine the impact of mixing global and local token selections by varying their proportions. We find that selecting an equal ratio of global and local tokens yields the best performance, achieving 58.3% on GQA and 1698 on MME. In contrast, retaining only local or global tokens results in 0.3% and 0.8% performance drop on GQA, respectively. These results highlight the complementary roles of global and local tokens, which together capture rich visual information and help preserve the model's visual reasoning capabilities.

Different Encoding Layers for Local Scan. In Table C5 (c), we explore the effect of performing local token selection at different layers of the visual encoder. Consistent with our empirical findings in Section B, we find that applying the local scan at a shallow layer ($l = 6$) yields the best performance. However, performing the local scan at very early ($l = 2$) or the output layer ($l = 23$) leads to a noticeable performance drop, with performance on GQA falling to 57.1 and 57.3, respectively.

D More Experimental Details

D.1 Computational Complexity

We follow PyramidDrop (Xing et al., 2025) to compute the theoretical floating-point operations (FLOPs) introduced in the LLM decoding layers during the pre-filling stage for processing the visual tokens. Specifically, in each of the K decoding layers, self-attention calculation with a causal mask is applied, followed by multiple feed-forward network (FFN) layers. The total FLOPs can thus be computed as:

$$\text{Total FLOPs} = \sum_{k=1}^{K} \left(4n_k d^2 + 2n_k^2 d + 3n_k dm\right), \tag{5}$$

where K is the number of transformer layers, n_k is the number of visual tokens at LLM layer k, d is the hidden state size, and m is the intermediate size of the FFN. This calculation suggests that reducing the number of visual tokens can significantly decrease the FLOPs required during inference.

D.2 Full Experimental Details

Models. We applied our approach to four popular LVLMs with different architectures to evaluate its general effectiveness. Specifically, we follow previous work in this field to compare performance on LLaVA-1.5-7B (Liu et al., 2024a), which is widely recognized for academic use and maps an image input to 576 tokens, and LLaVA-NeXT-7B (Liu et al., 2024b), which offers enhanced high-resolution visual understanding by representing an image input using up to 2,880 visual tokens. We also include evaluations on Video-LLaVA-7B (Lin et al., 2024a), which extends the framework to handle video input, processing up to 8 frames with 2,048 visual tokens. Furthermore, we are among the first to present experimental results on the recent Qwen-2.5-VL (Bai et al., 2025) model with various LLM sizes (3B, 7B, 32B), which incorporates dynamic resolution processing to handle images of varying sizes, supporting token counts ranging from 4 to 16,384.

Benchmarks and Metrics. We conduct extensive experiments on 9 standard image understanding benchmarks, including visual question answering benchmarks such as GQA (Hudson & Manning, 2019), ScienceQA (Lu et al., 2022), VQAv2 (Goyal et al., 2017), TextVQA (Singh et al., 2019) and VizWiz (Gurari et al., 2018); multi-modal reasoning benchmarks such as MMBench (Liu et al., 2024c), MMBench-CN (Liu et al., 2024c), MME (Fu et al., 2023), and POPE (Li et al., 2023b). We also include evaluations on 3 more challenging referring grounding tasks using RefCOCO (Kazemzadeh et al., 2014), RefCOCO+ (Kazemzadeh et al., 2014), and RefCOCOg (Mao et al., 2016), and report the accuracy achieved by different approaches. In these grounding tasks, a localization is considered correct if the predicted bounding box has an IoU score of at least 0.5 with the ground truth. Additionally, we evaluate our approach on 4 video question answering benchmarks: TGIF (Jang et al., 2017), MSVD (Chen & Dolan, 2011), MSRVTT (Xu et al., 2016), and ActivityNet (Yu et al., 2019). We follow previous work (Maaz et al., 2024; Xing et al., 2025) to utilize both accuracy and the ChatGPT score[1] as key performance metrics for these video-based benchmarks.

Baselines. We compare the performance of our approach with 6 state-of-the-art visual token pruning methods: ToMe (Bolya et al., 2023), FastV (Chen et al., 2024a), SparseVLM (Zhang et al., 2024b), HiRED (Arif et al., 2025), PyramidDrop (Xing et al., 2025), and VisionZip (Yang et al., 2025). (1) ToMe (Bolya et al., 2023), which uses bipartite soft matching to iteratively merge similar tokens within ViT layers; (2) FastV (Chen et al., 2024a), which drops visual tokens in early layers of the LLM, guided by text-oriented attention score; (3) SparseVLM (Zhang et al., 2024b), which selects vision-relevant text tokens to evaluate the significance of visual tokens; (4) HiRED (Arif et al., 2025), which dynamically assigns token budgets to sub-images for high-resolution image inputs; (5) PyramidDrop (Xing et al., 2025), which divides the LLM into stages and drops a portion of visual tokens at the end of each stage; (6) VisionZip (Yang et al., 2025), which selects a set of dominant tokens and merges unselected tokens contextually. To ensure a fair comparison, we directly report the results of these baselines from their respective original papers unless stated otherwise.

Implementation Details. We adhere to the default inference settings for each evaluated LVLM as specified in their

[1]Evaluated using *gpt-3.5-turbo*: https://platform.openai.com/docs/models/gpt-3.5-turbo.

Instructions:
You are an intelligent chatbot designed for evaluating the correctness of generative outputs for question-answer pairs. Your task is to compare the predicted answer with the correct answer and determine if they match meaningfully. Here's how you can accomplish the task:

##INSTRUCTIONS:
- Focus on the meaningful match between the predicted answer and the correct answer.
- Consider synonyms or paraphrases as valid matches.
- Evaluate the correctness of the prediction compared to the answer.

User Input:
Please evaluate the following video-based question-answer pair:

Question: {`question`}
Correct Answer: {`answer`}
Predicted Answer: {`pred`}

Provide your evaluation only as a yes/no and score where the score is an integer value between 0 and 5, with 5 indicating the highest meaningful match. Please generate the response in the form of a Python dictionary string with keys 'pred' and 'score', where value of 'pred' is a string of 'yes' or 'no' and value of 'score' is in INTEGER, not STRING. DO NOT PROVIDE ANY OTHER OUTPUT TEXT OR EXPLANATION. Only provide the Python dictionary string. For example, your response should look like this: {'pred': 'yes', 'score': 4}.

Table D6: **GPT-aided evaluation setup**. We present the prompt and user input format for evaluating the LVLM responses in video understanding tasks.

respective codebases. Additionally, we perform local scan at a shallow layer, specifically at $l = 6$ for LLaVA-series models and $l = 8$ for Qwen-2.5-VL. For LLM-stage pruning, we select the middle layer as $k = 16$ for LLaVA-series models and $k = 14$ for Qwen-2.5-VL. By default, we set the retention rate at the LLM middle layer to $R_2 = 33.3\%$, and adjust R_1 accordingly to achieve the target average reduction rate. Note that these design choices are grounded in our analysis in Section B, and we also provide comprehensive ablation studies in Section C.7.

Remarks on FlashAttention. Our proposed VScan is compatible with FlashAttention (Dao, 2024; Dao et al., 2022), as we recompute the attention scores for the last instruction token using vanilla attention calculation (Vaswani et al., 2017) outside the standard LLM layers. A detailed efficiency analysis can be found in Section 3.2.

D.3 GPT-Aided Evaluation on Video Understanding Tasks

Following the evaluation protocol of Video-LLaVA, we employ GPT-3.5-Turbo to assess model responses on video understanding tasks, evaluating them based on both accuracy and quality score. Specifically, we adopt the prompt shown in Table D6 to guide GPT in rating each response:

- **Accuracy**: A binary yes/no judgment indicating whether the response is correct.
- **Score**: An integer ranging from 0 to 5, where 5 represents the highest degree of relevance and informativeness and indicates the highest meaningful match.

E License Information

We list the license information for all the used assets as follows.

Benchmarks. We evaluate on a comprehensive set of 16 benchmarks spanning image QA, video QA, multimodal reasoning, and referential understanding tasks:

- **GQA** (Hudson & Manning, 2019): compositional visual question answering. This benchmark is released under the CC BY 4.0 license.
- **ScienceQA** (Lu et al., 2022): multimodal science questions with diagrams and text. This benchmark is released under the MIT license.

- **VQAv2** (Goyal et al., 2017): real-world image-based QA with balanced answers. This benchmark is released under the CC BY 4.0 license.
- **TextVQA** (Singh et al., 2019): reading and reasoning over scene text in images. This benchmark is released under the CC BY 4.0 license.
- **VizWiz** (Gurari et al., 2018): real-world visual questions from blind users. This benchmark is released under the CC BY 4.0 license.
- **MMBench** (Liu et al., 2024c) / **MMBench-CN** (Liu et al., 2024c): multilingual multimodal reasoning. These benchmarks are released under Apache License 2.0.
- **MME** (Fu et al., 2023): fine-grained multimodal evaluation on object, OCR, and commonsense. This benchmark is released under Apache License 2.0.
- **POPE** (Li et al., 2023b): probing object hallucinations in vision-language models. This benchmark is released under the MIT license.
- **RefCOCO / RefCOCO+** (Kazemzadeh et al., 2014), **RefCOCOg** (Mao et al., 2016): referential expression grounding. These benchmarks are released under Apache License 2.0.
- **TGIF** (Jang et al., 2017): video QA with spatiotemporal reasoning. The license of this work is not specified.
- **MSVD** (Chen & Dolan, 2011): short video captioning and QA. This benchmark is released under the MIT license.
- **MSRVTT** (Xu et al., 2016): large-scale video-text retrieval and QA. This benchmark is released under the MIT license.
- **ActivityNet-QA** (Yu et al., 2019): complex event-centric video QA. This benchmark is released under Apache License 2.0.

Models. We apply our approach to four widely used LVLMs.

- **LLaVA-1.5** (Liu et al., 2024a) is released under the LLaMA 2 Community License.
- **LLaVA-NeXT** (Liu et al., 2024b) is released under Apache License 2.0.
- **Qwen-2.5-VL** (Bai et al., 2025) is released under Apache License 2.0.
- **VideoLLaVA** (Lin et al., 2024a) is released under Apache License 2.0.

Code. Our codebase builds upon PyramidDrop (Xing et al., 2025), licensed under MIT, and FasterVLM (Zhang et al., 2024a), licensed under Apache 2.0.

F Limitations and Broader Impacts

Limitations. One key limitation of this work is the inherent trade-off between efficiency and accuracy: while the proposed VScan significantly reduces inference cost of LVLMs, aggressive token pruning may still distort visual information and lead to degraded performance, particularly on challenging tasks that demand fine-grained understanding or compositional reasoning.

Broader Impacts. The development of efficient LVLMs has significant potential to influence a wide range of applications, from autonomous systems and robotics to healthcare, education, and accessibility. By optimizing visual token reduction with VScan, we are addressing the computational overheads associated with processing large visual inputs, enabling faster and more efficient inference in real-time applications. This can lead to more widespread adoption of LVLMs in settings where rapid decision-making is crucial, such as autonomous vehicles, real-time video analysis, and interactive AI systems.