

000	A	APPENDIX	
001			
002		CONTENTS	
003			
004			
005	A	Appendix	1
006			
007	Appendix		1
008			
009	A.1	Algorithm and Efficiency Analysis	1
010			
011	A.1.1	Pseudocode of Our Method	1
012	A.1.2	Runtime Comparison and Efficiency Analysis	2
013			
014	A.2	Additional Experiments	2
015			
016	A.2.1	Experiment Details	3
017	A.2.2	Evaluation on Larger-Scale Models	3
018	A.2.3	Evaluation on Advanced Vision-Language Architectures	4
019			
020			
021	A.3	Additional Ablation Studies	5
022			
023	A.3.1	Ablation Studies on Component Contributions	5
024	A.3.2	Ablation Studies on Information Preservation of both Importance and Diversity .	5
025	A.3.3	Ablation Studies on Sensitivity to NMS Threshold Scaling Factor λ	6
026			
027			
028	A.4	Visualization	7
029			
030	A.4.1	Visualization about Limitation	7
031	A.4.2	Visualization of PSCA Grouping and Retained Tokens	8
032			
033	A.1	ALGORITHM AND EFFICIENCY ANALYSIS	
034			
035	A.1.1	PSEUDOCODE OF OUR METHOD	
036			
037			
038		To provide a clearer understanding of our proposed visual token compression pipeline, we present	
039		the full pseudocode of our method. The algorithm consists of two key stages: (1) principal semantic	
040		component analysis (PSCA)-based token grouping in Alg. 1, and (2) intra-group non-maximum	
041		suppression (NMS) for redundancy removal in Alg. 2. This two-stage design enables both semantic	
042		preservation and token diversity under various compression ratios.	
043			
044		Algorithm 1 Semantic-Aware Token Grouping via PSCA	
045		Require: Visual tokens $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_T] \in \mathbb{R}^{T \times D}$, token budget N	
046		1: $\mathbf{V} = \text{pca_group}(\mathbf{X}, K = \lfloor \frac{N}{4} \rfloor)$, $\mathbf{V} \in \mathbb{R}^{T \times K}$	
047		2: for each group G_k do	
048		3: Initialize $G_k \leftarrow \emptyset$	
049		4: end for	
050		5: for $i = 1$ to T do	
051		6: $g(i) \leftarrow \arg \max_j \mathbf{V}_{i,j} $	
052		7: $G_{g(i)} \leftarrow G_{g(i)} \cup \mathbf{x}_i$	
053		8: end for	
		9: return Semantically coherent groups $\{G_1, \dots, G_K\}$	

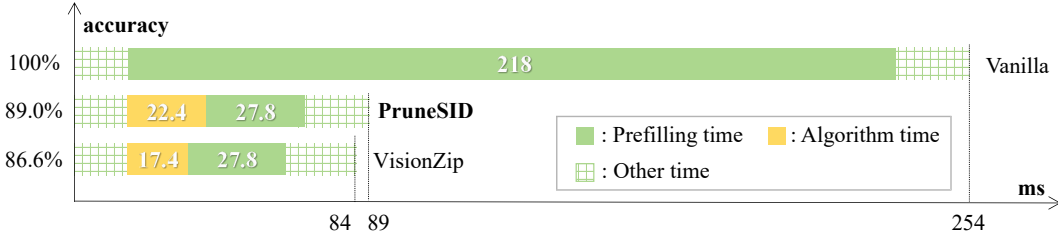


Figure 6: **Runtime comparison of different methods.** *Vanilla* denotes the average runtime of the uncompressed LLaVA-NeXT 7B model on the POPE dataset and the upper performance bound. VisionZip and **PruneSID** represent the runtimes after applying the corresponding compression methods. While maintaining comparable efficiency to VisionZip, our method achieves better performance.

Algorithm 2 Intra-Group Redundancy Removal via NMS

Require: Groups $\{G_1, \dots, G_K\}$, visual tokens \mathbf{X} , projection matrix \mathbf{V} , token budget N

- 1: Compute global redundancy score: $\rho \leftarrow \frac{2}{T(T-1)} \sum_{i < j} \text{sim}(\mathbf{x}_i, \mathbf{x}_j)$
- 2: Set threshold: $\tau \leftarrow \lambda \cdot \rho$, where $\lambda = \frac{N}{32}$
- 3: **for** each group G_k **do**
- 4: Initialize $\tilde{G}_k \leftarrow \emptyset$
- 5: Sort tokens in G_k by $s_i = |\mathbf{V}_{i,k}|$
- 6: **for** each token $\mathbf{x}_i \in G_k$ **do**
- 7: **if** $\text{sim}(\mathbf{x}_i, \mathbf{x}_j) < \tau$ for all $\mathbf{x}_j \in \tilde{G}_k$ **then**
- 8: $\tilde{G}_k \leftarrow \tilde{G}_k \cup \mathbf{x}_i$
- 9: **end if**
- 10: **end for**
- 11: **end for**
- 12: Allocate $n_k \leftarrow \left\lfloor \frac{|\tilde{G}_k|}{\sum_j |\tilde{G}_j|} \cdot N \right\rfloor$
- 13: $\tilde{\mathbf{X}} \leftarrow \bigcup_k \text{Top}_{n_k}(\tilde{G}_k)$
- 14: **return** Final compressed token set $\tilde{\mathbf{X}}$

A.1.2 RUNTIME COMPARISON AND EFFICIENCY ANALYSIS

While Alg. 1 and 2 are presented with sequential steps for clarity, our actual implementation leverages parallel processing to significantly accelerate execution. The full implementation details are available in our released codebase: <https://anonymous.4open.science/r/PruneSID-20352/>.

To further demonstrate the efficiency of our method, we benchmark its runtime under the same hardware and software configuration as used in the main paper’s Efficient Analysis section. A detailed comparison with VisionZip Yang et al. (2024) is presented in Fig. 6. On a per-sample basis, our method achieves a processing time of 22.4ms, which is comparable to VisionZip’s 17.4ms. Our *early compression* mechanism significantly reduces the total prefilling latency, decreasing it from 218ms to 27.8ms, by reducing the number of visual tokens before they are fed into the LLM.

Despite comparable preprocessing costs, our method consistently yields better performance. Under this configuration, our approach outperforms VisionZip by 2.4% on the POPE dataset, and achieves an average accuracy of 92.8% across multiple benchmarks (see Tab. 2 ↓ 94.4% in the main paper). These results highlight the practicality of our method, offering a favorable trade-off between speed and accuracy.

A.2 ADDITIONAL EXPERIMENTS

In this section, we present the extended experimental setup and results to validate the effectiveness and generalization ability of our method: (1) we begin by detailing the implementation of our experiments

Table 9: **Performance of PRUNESID on LLaVA-1.5 13B.** *Vanilla* refers to the uncompressed baseline model using all 576 visual tokens, serving as the upper performance bound.

Method	GQA	MMB	MME	POPE	SQA	VQA ^{v2}	VQA ^{Text}	MMMU	SEED ^I	VizWiz	Avg.
<i>Upper Bound, 576 Tokens (100%)</i>											
Vanilla CVPR24	63.2	67.7	1818	85.9	72.8	80.0	61.3	36.4	66.9	56.6	100%
<i>Retain 192 Tokens (↓ 66.7%)</i>											
VisionZip CVPR25	59.1	66.9	1754	85.1	73.5	78.1	59.5	36.4	65.2	54.9	97.8%
PRUNESID	59.6	65.9	1770	86.4	73.7	78.0	58.6	36.9	65.2	56.0	98.3%
<i>Retain 128 Tokens (↓ 77.8%)</i>											
VisionZip CVPR25	57.9	66.7	1743	85.2	74.0	76.8	58.7	36.1	63.8	55.0	97.0%
PRUNESID	58.9	65.5	1811	85.9	73.1	76.7	57.5	35.8	64.1	56.8	97.4%
<i>Retain 64 Tokens (↓ 88.9%)</i>											
VisionZip CVPR25	56.2	64.9	1676	76.0	74.4	73.7	57.4	36.4	60.4	55.9	94.2%
PRUNESID	57.8	63.8	1711	82.0	72.1	75.2	56.3	35.7	62.8	57.3	95.3%

in Sec. A.2.1. (2) To evaluate the generalization of our approach across different model scales, we report results on larger-scale model LLaVA-1.5 Liu et al. (2023) and LLaVA-NeXT Liu et al. (2024) in Sec. A.2.2. (3) To further assess the applicability of our method to more advanced and architecturally sophisticated vision-language models, we present results on Qwen2-VL Wang et al. (2024) in Sec. A.2.3.

A.2.1 EXPERIMENT DETAILS

Environments. All experiments are conducted using a single NVIDIA L20 GPU with 48GB memory. We build our evaluation pipeline upon the publicly available lmms-eval Zhang et al. (2024) framework, ensuring consistency with prior benchmarks. For fair comparison in the Efficient Analysis section, we directly benchmark both VisionZip Yang et al. (2024) and our method on the same hardware (L20-48GB) under identical settings. To align with previously reported results obtained using A800-80GB GPUs, we linearly scale our runtime measurements based on VisionZip’s performance differential across devices, following standard practice in prior work. This adjustment allows for a meaningful speed comparison while maintaining experimental reproducibility on accessible hardware.

Code. To facilitate reproducibility, we release the complete implementation of our method at: <https://anonymous.4open.science/r/PruneSID-12764/>.

A.2.2 EVALUATION ON LARGER-SCALE MODELS

In the main paper, we reported the compression results on 7B-scale models. To further evaluate its effectiveness across different model scales, we additionally present results on LLaVA-1.5 13B and LLaVA-NeXT 13B.

LLaVA-1.5 encodes each image into a fixed sequence of 576 visual tokens. Following prior work Yang et al. (2024), we evaluate our method by retaining 192, 128, and 64 tokens, respectively. As shown in Tab. 9, under all three settings, our method achieves average accuracy within 98.0%, 97.4%, and 95.2% of the uncompressed baseline across multiple benchmarks. Compared to the results on 7B-scale models, our method maintains similarly strong compression performance. We further compare our method against VisionZip. Our method outperforms VisionZip by 1% in average accuracy under the most aggressive compression setting of retaining only 64 tokens. When it comes to LLaVA-NeXT 13B in Tab. 10, our method demonstrates competitive performance against VisionZip across different token retention settings, and its advantage becomes particularly prominent under high compression rates. Specifically, under the largest compression rate (94.4% reduction, retaining only 160 tokens), our method’s average performance reaches 92.2%, which outperforms VisionZip by 2% (VisionZip: 90.2%). This result fully validates that our method maintains superior visual compression efficiency and information preservation capabilities, especially when facing extreme compression demands.

Table 10: **Performance of PRUNESID on LLaVA-NeXT 13B.** *Vanilla* refers to the uncompressed baseline model using all 576 visual tokens, serving as the upper performance bound.

Method	GQA	MMB	MME	POPE	SQA	VQA ^{v2}	VQA ^{Text}	MMMU	SEED ^I	VizWiz	Avg.
<i>Upper Bound, 2880 Tokens (100%)</i>											
Vanilla CVPR24	65.4	70.0	1858	86.2	73.5	81.8	64.3	36.2	71.9	64.0	100%
<i>Retain 640 Tokens (↓ 77.8%)</i>											
VisionZip CVPR25	63.0	68.6	1871	85.7	71.2	79.7	62.2	36.4	68.8	58.6	97.3%
PRUNESID	63.0	68.1	1839	85.6	71.6	79.1	61.2	36.2	68.5	60.2	96.9%
<i>Retain 320 Tokens (↓ 88.9%)</i>											
VisionZip CVPR25	60.7	67.2	1805	82.0	70.3	76.8	60.9	35.6	65.2	56.8	94.3%
PRUNESID	61.5	65.4	1810	82.7	71.8	76.9	59.5	36.8	66.8	58.5	95.2%
<i>Retain 160 Tokens (↓ 94.4%)</i>											
VisionZip CVPR25	57.8	64.9	1739	76.6	69.3	72.4	58.4	37.0	61.1	55.9	90.2%
PRUNESID	59.5	65.5	1715	77.8	69.1	73.6	56.7	36.4	64.1	56.7	92.2%

Table 11: **Performance of PRUNESID on Qwen2-VL 7B.** *baseline* refers to the uncompressed model using all the visual tokens, serving as the upper performance bound.

Method	GQA	MMB	MME	POPE	SQA	VQA ^{v2}	MMMU	SEED ^I	VizWiz	Avg.
<i>Upper Bound (100%)</i>										
baseline CVPR24	62.3	79.1	2318	87.9	84.7	81.2	51.1	76.5	68.2	100%
<i>Token retention ratio 33.3% (↓ 66.7%)</i>										
PACT CVPR25	59.1	75.8	2068	85.6	80.5	78.2	48.2	74.0	67.9	95.3%
PRUNESID	60.5	72.1	2146	87.0	81.8	79.8	48.3	75.5	66.1	96.1%
<i>Token retention ratio 22.2% (↓ 77.8%)</i>										
PACT CVPR25	55.8	72.2	2012	82.4	78.3	77.6	48.8	71.7	67.4	92.8%
PRUNESID	59.2	69.2	2119	86.0	79.5	78.1	47.3	74.8	64.8	94.1%
<i>Token retention ratio 11.1% (↓ 88.9%)</i>										
PACT CVPR25	50.1	63.1	1785	71.4	75.0	74.3	48.5	66.0	63.1	85.8%
PRUNESID	55.9	65.0	2039	82.9	76.1	73.6	46.9	71.8	64.1	90.4%

A.2.3 EVALUATION ON ADVANCED VISION-LANGUAGE ARCHITECTURES

To further assess the generalization of our method on recent, more sophisticated VLMs, we conduct experiments on Qwen-VL Wang et al. (2024), a family of models with notable architectural differences from LLaVA.

Unlike LLaVA’s fixed-resolution vision encoders, Qwen-VL employs a dynamically scalable vision encoder that supports arbitrary input resolutions. Moreover, the vision encoder is jointly trained with the language model, enabling tighter cross-modal alignment. After encoding, a lightweight MLP compresses every 2×2 group of neighboring tokens into a single token, resulting in a compact but expressive visual representation.

Importantly, this merging operation breaks the one-to-one correspondence between image patches and the attention scores in the ViT architecture, rendering attention-guided compression methods such as VisionZip inapplicable to the Qwen-VL series. In contrast, our method operates purely on the visual tokens themselves, independent of attention scores, making it directly applicable to such merged-token models.

We evaluate Qwen2-VL 7B under token retention ratios of 33.3%, 22.2%, and 11.1%. As shown in Tab. 11, compared with DivPrune Alvar et al. (2025), the pruning-acceleration approach provided for Qwen2-VL, our method remains more effective across all three token retention settings, showing strong compatibility with modern VLM architectures that feature dynamic input processing.

Table 12: Performance comparison of ablation variants on LLaVA-1.5 7B (relative to vanilla baseline, 100%).

Condition	GQA	MME	POPE	SQA	MMMU	SEED	MMB	Vizwiz	Avg.
<i>Upper Bound, 576 Tokens (100%)</i>									
Vanilla CVPR24	61.9	1862	85.9	69.5	36.3	60.5	64.7	54.3	100%
<i>Retain 192 Tokens (↓ 66.7%)</i>									
w/o PSCA	58.8	1743	84.9	68.2	35.7	58.8	62.7	54.1	97.2%
w/o NMS	59.1	1755	85.8	68.1	35.2	58.6	62.3	54.9	97.3%
Ours	60.2	1797	87.1	69.1	36.8	59.0	63.8	55.5	99.3%
<i>Retain 128 Tokens (↓ 77.8%)</i>									
w/o PSCA	57.8	1703	83.0	67.5	35.7	57.6	61.2	54.5	95.8%
w/o NMS	57.5	1722	84.9	67.4	36.1	57.4	61.4	54.3	96.3%
Ours	58.9	1760	86.9	68.8	35.8	57.9	62.6	56.0	98.0%
<i>Retain 64 Tokens (↓ 88.9%)</i>									
w/o PSCA	56.1	1670	78.5	67.3	36.1	54.5	59.3	55.2	93.9%
w/o NMS	55.9	1665	79.7	67.5	36.2	54.9	58.1	56.3	94.2%
Ours	57.2	1734	84.1	68.1	37.2	56.2	59.7	57.0	96.8%

A.3 ADDITIONAL ABLATION STUDIES

A.3.1 ABLATION STUDIES ON COMPONENT CONTRIBUTIONS

This section presents detailed ablation studies to isolate and evaluate the individual contributions of two key components in our framework: the Principal Semantic Components Analysis (PSCA) module and the Non-Maximum Suppression (NMS) based redundancy removal mechanism. To accurately assess each component’s impact, we design two ablation variants of our model on LLaVA-1.5: 1) *w/o PSCA*: Replaces the PSCA-based grouping with random token grouping, while maintaining the original NMS algorithm. Within each randomly formed group, tokens are sorted by their vector ℓ_2 norm to determine importance. And 2) *w/o NMS*: Retains the original PSCA grouping and importance ranking, but replaces the NMS-based redundancy removal with a simple top- k selection that ignores token redundancy.

Tab. 12 summarizes the comparative performance across all configurations. The experimental results demonstrate that both components contribute significantly to the model’s overall performance. Removing either PSCA or NMS leads to consistent performance drops across most datasets and token retention settings. Our full model consistently outperforms both ablation variants across all token retention settings, with an average performance advantage of 1.7-2.9% over the ablation variants, validating the complementary nature of PSCA and NMS.

A.3.2 ABLATION STUDIES ON INFORMATION PRESERVATION OF BOTH IMPORTANCE AND DIVERSITY

This section presents ablation studies to validate that our method effectively preserves both information importance and diversity of visual tokens. To be specific, we design two ablation experiments on LLaVA-1.5 to isolate these two properties and demonstrate their individual contributions: 1) *Descend*: To test the importance preservation property, we retain tokens in descending order of their importance rankings (i.e., keeping the least important tokens first) while applying our standard NMS for redundancy reduction. This variant directly challenges the importance preservation mechanism by prioritizing less critical information. And 2) *Ascend*: To test the diversity preservation property, we retain tokens in ascending order of their importance rankings (i.e., selecting the most important tokens first) but without NMS to considering redundancy. This variant prioritizes importance while ignoring potential information redundancy and diversity.

Table 13: Performance comparison of importance and diversity preservation ablation variants on LLaVA-1.5 7B (relative to vanilla baseline, 100%).

Condition	GQA	MME	POPE	SQA	MMMU	SEED	MMB	Vizwiz	Avg (%)
<i>Upper Bound, 576 Tokens (100%)</i>									
Vanilla <small>CVPR24</small>	61.9	1862	85.9	69.5	36.3	60.5	64.7	54.3	100%
<i>Retain 192 Tokens (↓ 66.7%)</i>									
<i>Descend</i>	57.3	1667	83.9	66.8	35.8	56.4	58.8	54.8	96.1%
<i>Ascend</i>	59.1	1763	85.8	68.1	36.2	58.6	63.3	54.9	97.9%
Ours	60.1	1791	86.9	68.5	36.1	59.0	63.7	55.4	98.8%
<i>Retain 128 Tokens (↓ 77.8%)</i>									
<i>Descend</i>	56.0	1556	79.6	66.4	35.8	53.8	55.5	54.9	92.1%
<i>Ascend</i>	57.5	1722	84.9	68.4	36.1	57.4	62.4	55.3	96.9%
Ours	58.8	1749	86.5	68.3	35.8	57.8	62.1	55.8	97.6%
<i>Retain 64 Tokens (↓ 88.9%)</i>									
<i>Descend</i>	51.7	1361	67.5	64.8	35.1	49.3	43.6	54.1	84.2%
<i>Ascend</i>	55.9	1664	79.7	68.5	36.2	54.9	58.1	56.3	94.8%
Ours	57.1	1733	83.8	67.8	37.0	56.1	58.8	56.9	96.3%

Table 14: Performance sensitivity to NMS threshold scaling factor λ on LLaVA-1.5 7B (relative to vanilla baseline, 100%).

α	Retain 64					Retain 128					Retain 192				
	GQA	MME	POPE	SQA	Avg.	GQA	MME	POPE	SQA	Avg.	GQA	MME	POPE	SQA	Avg.
24	56.7	1726	83.4	68.0	94.8%	58.4	1718	85.5	68.3	96.1%	59.7	1770	86.3	68.4	97.6%
28	57.0	1718	84.1	67.8	94.9%	58.8	1723	86.4	68.2	96.6%	59.9	1783	86.8	68.4	98.0%
32	57.1	1733	83.8	67.8	95.1%	58.8	1749	86.5	68.3	97.0%	60.1	1791	86.9	68.5	98.3%
36	57.3	1705	84.7	67.9	95.1%	58.7	1733	85.9	68.3	96.5%	59.7	1789	86.6	68.5	98.0%
40	57.3	1675	84.7	67.9	94.7%	58.6	1711	85.7	68.1	96.2%	59.6	1772	86.3	68.4	97.6%

Tab. 13 summarizes the performance of all configurations. The *Descend* variant, which retains less informative tokens, suffers severe degradation (e.g., 84.2% at 64 tokens), highlighting the necessity of preserving importance. The *Ascend* variant, which emphasizes importance but ignores redundancy, performs better but notably drops on POPE, underscoring the need for diversity. Our full model consistently surpasses both variants (0.9–1.5% over *Ascend*, 2.7–12.1% over *Descend*), demonstrating its effectiveness in jointly preserving importance and diversity without trade-offs.

A.3.3 ABLATION STUDIES ON SENSITIVITY TO NMS THRESHOLD SCALING FACTOR λ

This section presents a comprehensive ablation study to analyze the sensitivity of our method to the NMS threshold scaling factor λ , and its impact on both model performance and the distribution of retained tokens across semantic groups. As background, our NMS threshold is defined as $\tau = \lambda \cdot \rho$ (where ρ denotes the redundancy metric between tokens), and λ is a scaling factor adaptively determined by the global token budget N (with a default setting of $\lambda = \frac{N}{32}$ in our base model). We design two complementary experiments to evaluate the role of α with $\lambda = \frac{N}{\alpha}$:

Performance Sensitivity. We test distinct values of α (24, 28, 32, 36, 40) under three token retention settings (192, 128 and 64 tokens) on the LLaVA-1.5 7B model. We report performance across four key benchmarks (GQA, MME, POPE, SQA) to quantify how λ impacts multimodal understanding. Tab. 14 summarizes the performance across different α values. The results demonstrate that performance remains highly stable across different values of α (24–40), indicating that the method is not sensitive to fine-tuning of this parameter. The optimal performance is consistently achieved at $\alpha = 32$, yielding the highest average scores across multiple token retention settings.

Token Distribution Sensitivity. For the 64-token retention setting (on the MME benchmark), analyze how α affects the number of retained tokens across 16 semantic groups (Group 0 to Group 15) generated by the PSCA module. We also include a “Before pruning” baseline to show the initial

Table 15: Token distribution across PSCA groups (Group 0–15) for different λ (retain 64 tokens, MME benchmark).

$\alpha \backslash$ Group id	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Before pruning	153.6	28.1	37.5	36.3	34.4	31.1	28.1	26.8	25.1	23.9	23.3	23.1	23.9	25.5	27.1	28.4
24	1.0	4.2	4.1	3.8	3.8	4.0	4.1	4.1	4.1	4.0	4.0	4.0	4.2	4.4	4.6	4.8
28	1.1	4.3	4.3	4.1	4.0	4.2	4.1	4.1	4.0	3.8	3.8	3.8	4.0	4.2	4.4	4.6
32	1.1	3.8	3.8	3.6	3.7	4.0	4.1	4.1	4.2	4.1	4.1	4.1	4.3	4.5	4.7	4.9
36	1.1	3.4	3.5	3.5	3.6	3.9	4.1	4.1	4.2	4.2	4.2	4.3	4.4	4.6	4.8	5.0
40	1.3	3.0	3.3	3.4	3.6	3.8	4.0	4.1	4.2	4.2	4.3	4.4	4.6	4.8	5.0	5.2

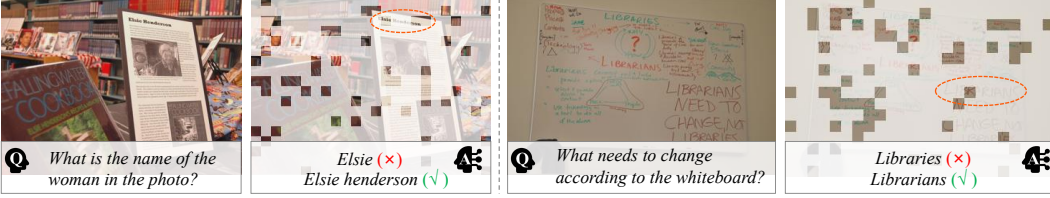


Figure 7: **Visualization of a limitation of our method.** Left: input image and the corresponding question. Right: tokens retained by our method and the VLM’s response after compression. In fine-grained scenarios, our method may discard important local details due to redundancy in the relevant region, leading to incomplete answers.

token count per group prior to NMS-based redundancy removal. As shown in Tab. 15, we can observe that the distribution of group token counts remains relatively stable across different values of α . Additionally, we note the following observations and explanations:

- *The number of tokens retained in Group 0 is small:* Group ids are related to the principal component ordering in PSCA. Group 0 often corresponds to the background regions of the image that exhibit higher redundancy and contain less important semantic information, resulting in a smaller number of retained tokens.
- *The number of tokens retained increases as group id increases:* According to the nature of PCA, later principal component groups are associated with tokens that have greater variability and lower redundancy. Under the same NMS threshold, more tokens are retained after redundancy removal.
- *As α increases, tokens are more likely to be distributed in higher-numbered groups:* As α increases, λ decreases, and the NMS redundancy removal threshold $\tau = \lambda \cdot \rho$ also lowers. Tokens in lower-ranked, more redundant groups are more likely to be pruned, leading to a smaller number of retained tokens.

These findings provide further insight into how α influences the distribution of tokens across different groups, contributing to the overall balance between redundancy removal and token retention.

A.4 VISUALIZATION

A.4.1 VISUALIZATION ABOUT LIMITATION

While the task-agnostic nature of our method provides strong generalization ability, it may be less effective in tasks requiring fine-grained or instruction-specific reasoning. As shown in Fig. 7, when a query focuses on specific visual details, our method may overlook relevant tokens under extreme compression settings (e.g., retaining only 11.1% of tokens). To address this limitation, future work will explore incorporating task-adaptive cues or instruction-aware filtering mechanisms to better align token selection with downstream task requirements.

A.4.2 VISUALIZATION OF PSCA GROUPING AND RETAINED TOKENS

We present additional visualizations of PSCA-based token grouping and the final retained tokens to demonstrate the effectiveness of our method. Fig. 8 shows partial results of our semantic token grouping and compares the retained tokens between our method and the attention-guided baseline, VisionZip, under an extreme compression ratio (retaining 32 tokens, 5.6%). Fig. 9 shows the grouping and selection results when retaining 64 tokens (11.1%). As observed, our method selectively preserves representative tokens from highly redundant groups with shared semantics, enabling broader semantic diversity within a limited token budget. In contrast, VisionZip tends to retain tokens with high attention scores that are often semantically similar, leading to redundancy and insufficient coverage of diverse visual concepts.

432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485



Figure 8: Visualization of PSCA-based semantic token grouping and final retained tokens under an extreme compression ratio (retaining 32 tokens, 5.6%). *Semantically Coherent Groups* show partial grouping results from our PSCA stage, and *Retained* compares the selected tokens from our method and VisionZip. Our approach preserves representative tokens from semantically redundant groups, enabling broader information coverage than attention-only methods like VisionZip.

486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539



Figure 9: Visualization of PSCA-based token grouping and final retained tokens when retaining 64 tokens (11.1%). Similar to Fig. 8, our method effectively selects representative tokens across diverse groups while filtering redundancy. This helps maintain semantic diversity under tight token budgets, outperforming redundancy-prone baselines like VisionZip.

REFERENCES

- Saeed Ranjbar Alvar, Gursimran Singh, Mohammad Akbari, and Yong Zhang. Divprune: Diversity-based visual token pruning for large multimodal models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 9392–9401, 2025.
- Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.
- Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Lllavanext: Improved reasoning, ocr, and world knowledge, 2024.
- Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.
- 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. *arXiv preprint arXiv:2412.04467*, 2024.
- Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, et al. Lmms-eval: Reality check on the evaluation of large multimodal models. *arXiv preprint arXiv:2407.12772*, 2024.