Filter, Correlate, Compress: Training-Free Token Reduction for MLLM Acceleration

Yuhang Han[1*], Xuyang Liu[2*], Zihan Zhang[3], Pengxiang Ding[1], Junjie Chen[2], Honggang Chen[2], Donglin Wang[1], Qingsen Yan[4,5], Siteng Huang[6†]

[1] Westlake University
[2] Sichuan University
[3] Johns Hopkins University
[4] Northwestern Polytechnical University
[5] Shenzhen Research Institute of Northwestern Polytechnical University
[6] Zhejiang University
yuhangh984@gmail.com, siteng.huang@gmail.com

Abstract

The quadratic complexity of Multimodal Large Language Models (MLLMs) with respect to context length poses significant computational and memory challenges, hindering their real-world deployment. In the paper, we devise a "***filter-correlate-compress***" framework to accelerate the MLLM by systematically optimizing multimodal context length during prefilling. The framework first implements ***FiCoCo-V***, a training-free method operating within the vision encoder. It employs a redundancy-based token discard mechanism that uses a novel integrated metric to accurately *filter* out redundant visual tokens. To mitigate information loss, the framework introduces a correlation-based information recycling mechanism that allows preserved tokens to selectively recycle information from *correlate*d discarded tokens with a self-preserving *compress*ion, thereby preventing the dilution of their own core content. The framework's ***FiCoCo-L*** variant further leverages task-aware textual priors to perform token reduction directly within the LLM decoder. Extensive experiments demonstrate that the *FiCoCo* series effectively accelerates a range of MLLMs, achieves up to **14.7×** FLOPs reduction with **93.6%** performance retention. Our methods consistently outperform state-of-the-art training-free approaches, showcasing effectiveness and generalizability across model architectures, sizes, and tasks without requiring retraining.

Code — https://github.com/kawhiiiileo/FiCoCo

1 Introduction

Multimodal Large Language Models (MLLMs) (Liu et al. 2023, 2024a; Zhang, Li, and Bing 2023; Chen et al. 2024b; Lin et al. 2024b) have effectively extended the impressive emergent capabilities of Large Language Model (LLM) (Touvron et al. 2023; OpenAI 2023; Bai et al. 2023a) decoders by integrating visual features with textual inputs. However, the substantial increase in context length when processing images and videos imposes quadratically scaling computational and memory demands. This, in turn, renders prefilling a critical bottleneck for MLLM response generation. As an empirical instance, the prefilling time for Qwen2-VL (Wang et al. 2024) in high-resolution visual question answering significantly outweighs its decoding time, constituting up to a remarkable **80%** of the total latency. In this paper, we introduce "***filter-correlate-compress***", a framework that systematically and progressively optimizes the length of multimodal context during prefilling. This enables MLLMs to minimize response latency while concurrently striving for the preservation of generation quality.

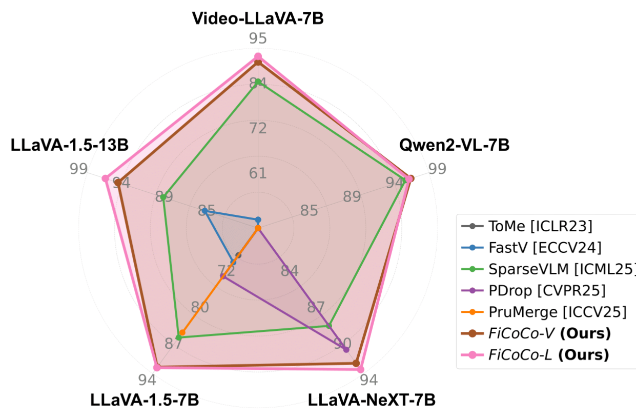

Figure 1: **The comparison to existing token reduction methods.** Our *FiCoCo* series achieves state-of-the-art results with five popular MLLMs across benchmarks.

Concurrently, we propose ***FiCoCo-V***, a training-free method that represents the framework's implementation within the vision encoder, primarily addressing data redundancy. Natural vision signals, such as images and videos, inherently possess a higher degree of information redundancy compared to human-generated languages (He et al. 2022; Feichtenhofer et al. 2022). However, in the constructed multimodal context, the number of visual tokens substantially

[*]Equal contributions.
[†]Corresponding Author.

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

exceeds that of textual tokens. Our framework thus initiates with a **redundancy-based token discard**, which reduces context length by measuring and ***filter***ing out redundant visual tokens at each layer. Specifically, unlike a potentially biased, single redundancy metric (Chen et al. 2024a; Zhang et al. 2024), *FiCoCo-V* integrates vision-aware and semantic-aware redundancy to accurately discard those more redundant tokens.

One common oversight is that tokens considered redundant may contain noise or still hold information beneficial to the task (Liang et al. 2022). Developing effective mechanisms to flexibly recover such information facilitates performance maintenance of the MLLM. However, preserved tokens must carefully select which information to receive from discarded ones, preventing excessive dilution of their core information. We highlight that inter-token correlation provides a principled metric to guide where and how information should be recycled. Moving forward, our framework designs a **correlation-based information recycling** mechanism that allows each redundant token to ***correlate*** a variable number of preserved tokens to adaptively retain its information, while the *FiCoCo-V* method models such correlation with direct attention. Subsequently, a self-preserving ***compress***ion operation ensures the prominence of the preserved tokens while allowing them to receive more information from highly correlated redundant tokens.

While *FiCoCo-V* strikes an appealing balance between efficiency and performance, executing token discard and information recycling in a task-agnostic manner can be short-sighted, as it fundamentally constrains the performance ceiling of the acceleration method. Consequently, we further propose ***FiCoCo-L***, which performs token reduction directly within the LLM decoder. By leveraging task-aware textual priors, *FiCoCo-L* more precisely pinpoints redundant tokens and recovers crucial information, thereby minimizing loss during the compression process. In Figure 1, when applied to LLaVA-1.5-7B (Liu et al. 2024a), both methods consistently outperform existing token reduction baselines across different FLOPs. In the most extreme case, our method can obtain a maximum improvement of **5.7**$\times$ in FLOPs while retaining **92.8%** performance. When applied to the more powerful LLaVA-NeXT-7B (Liu et al. 2024b), our methods even show stronger superiority, achieving a **14.7**$\times$ improvement in FLOPs while retaining at most **93.6%** performance. We also evaluate our methods on video understanding tasks, where our methods retain at most **92.8%** performance of vanilla Video-LLaVA (Lin et al. 2024a) with a **11.4**$\times$ improvement in FLOPs. As a conclusion, our success in token budget reduction and model acceleration can generalize across various MLLM architectures, sizes, and tasks.

2 Related Work

Multimodal Large Language Models (MLLMs). To acquire visual comprehension and reasoning capabilities, MLLMs (Dai et al. 2023; Bai et al. 2023b; Liu et al. 2023; Chen et al. 2024b) first use a pre-trained vision encoder (*e.g.*, from CLIP (Radford et al. 2021)) to extract visual features, which are then projected into the input embedding space of a pre-trained Large Language Model (LLM) (Touvron et al. 2023; OpenAI 2023; Bai et al. 2023a) decoder. The LLM then processes these visual embeddings alongside user instructions to understand the images and craft suitable responses. A key trend in the development of MLLMs is to leverage longer multimodal contexts to capture finer-grained visual details, thereby enabling a more profound comprehension of the visual content. For example, LLaVA-1.5 (Liu et al. 2024a) improves the vision encoder for higher resolutions, while LLaVA-NeXT (Liu et al. 2024b) quadruples input resolution with flexible aspect ratios to enhance fine-grained understanding. And Video-LLaVA (Zhang, Li, and Bing 2023) employs extended context windows and dynamic frame aggregation to accommodate longer input sequences for video-text tasks. However, increased context length introduces significant inference latency and storage overheads, which become major deployment bottlenecks.

Token Reduction for MLLMs. Token reduction approaches can be broadly unified as token compression, which aims to eliminate redundancy and condense visual information into a more compact representation while minimizing information loss (Rao et al. 2021; Liang et al. 2022; Bolya et al. 2023; Liu et al. 2025d,c,b,a). Our proposed methods adaptively adjust the number of tokens each discarded token is compressed into. This functions as an automatic, per-token switching mechanism between the two techniques, designed to maximize benefits.

Token reduction for MLLMs has gradually shifted from training-based methods (Cha et al. 2024; Li et al. 2024) to training-free approaches (Chen et al. 2024a; Zhang et al. 2024), as the latter enables direct application to off-the-shelf models without costly retraining overheads. For instance, FastV (Chen et al. 2024a) prunes unnecessary visual tokens based on the ranking of attention scores derived from the self-attention mechanism in the LLM. SparseVLM (Zhang et al. 2024) adaptively prunes visual tokens in the LLM based on their attention scores with text tokens. PDrop (Xing et al. 2025) drops visual tokens according to the attention between all the visual tokens and the last token of the instruction. In this study, our *FiCoCo* shows that more precise identification of redundant tokens and controlled recovery of discarded information can achieve superior performance while maintaining high efficiency.

3 Methodology of *FiCoCo-V*

3.1 Preliminaries: Revisiting MLLMs

Prefilling. Given the textual instruction, a MLLM generates responses according to the input image, where the critical prefilling phase involves two key steps: (1) Input tokenization, where the visual encoder extracts the visual features and projects them into a shared latent space with discrete textual tokens. (2) Causal self-attention computation, wherein the LLM decoder performs causal self-attention over this entire concatenated sequence to establish contextual dependencies, providing intermediate key-value pairs for efficient autoregressive decoding. This phase establishes the decoding context for subsequent token generation, directly impacting inference latency and throughput.

Self-Attention. The self-attention mechanism (Vaswani

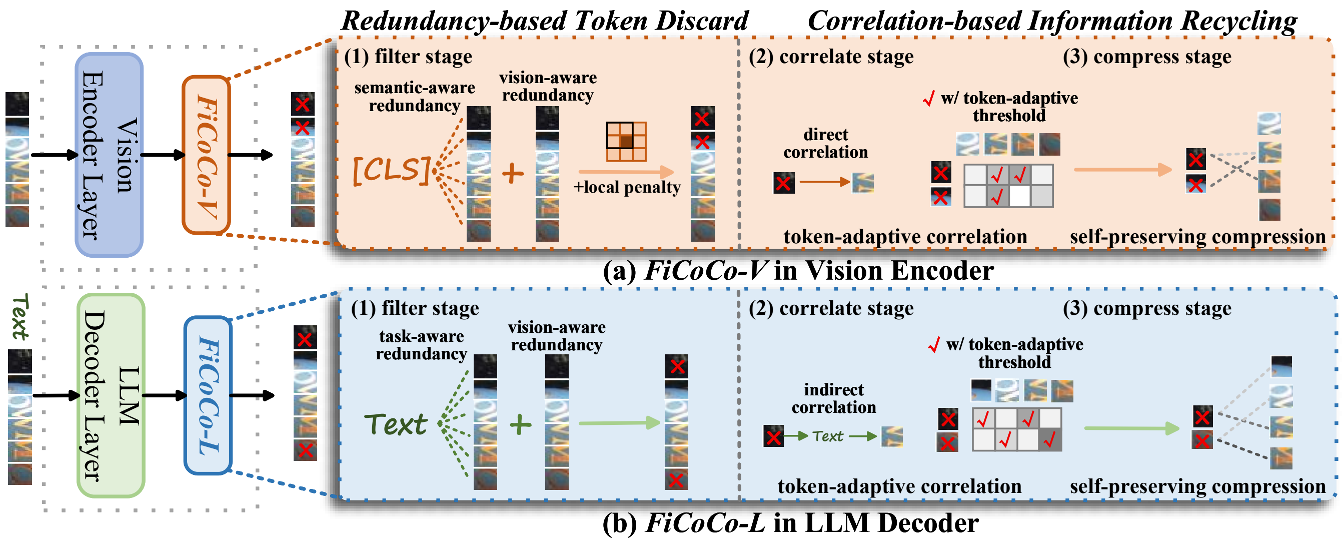

Figure 2: **Overview of *FiCoCo-V* and *FiCoCo-L*.** Due to the two methods being applied to different modules (vision encoder and LLM decoder), they have different implementations for summarized redundancy and correlation matrix in the filter and correlate stages. Simultaneously, the compression modules of *FiCoCo-V* and *FiCoCo-L* are identical, both employing self-preserving compression based on the correlation matrix.

et al. 2017) stands as both the most essential and the most resource-intensive operation in transformer-based visual encoder and LLM decoder. Given the input 1D sequence $\mathbf{X}$ of length N, the self-attention layer produces a self-attention map $\mathbf{A} \in \mathbb{R}^{N\times N}$ to globally model the dependence relationships between tokens, formulated as $\mathbf{A} = \text{Attention}(\mathbf{Q}, \mathbf{K}) = \text{Softmax}\left(\mathbf{QK}^\top/\sqrt{D}\right)$, where $^\top$ denotes the transpose of the matrix, the query and key matrices $\mathbf{Q}, \mathbf{K} \in \mathbb{R}^{N\times D}$ are obtained by projecting $\mathbf{X}$ with learnable parameter matrices.

3.2 Redundancy-based Token Discard

Filter: What token should be discarded? When evaluating the redundancy of tokens within the vision encoder, we draw inspiration from the two natural principles of humans when quickly and comprehensively summarizing the content of a given image. Firstly, to accelerate the recognition, we tend to ignore those similar pixels as they commonly provide the same information. Similarly, within a self-attention layer, if a visual token requires substantial information from other visual tokens, it indicates that its own information is not unique, and the token can be replaced by other visual tokens. Formally, given the self-attention weight matrix $\mathbf{A}^v \in \mathbb{R}^{N\times N}$, where N is the number of the visual tokens, we can define the **vision-aware redundancy** of the i-th token by averaging its received attention, *i.e.*, $\frac{1}{N}\sum_{j=1}^{N}\mathbf{A}^v_{i,j}$. We emphasize that this design is significantly different from previous methods (Chen et al. 2024a), as they regard attention between visual tokens as a measure of importance.

Secondly, if provided with the overall concept of the image, we rapidly identify the area of interest based on the global semantic clue and ignore other regions. As typical vision encoders (Dosovitskiy et al. 2021; Radford et al. 2021) employ a `[CLS]` token to capture the global image representation, its attention weights $\mathbf{a}^{\texttt{CLS}}$ can quantify the global semantic content of patch tokens, which can be useful for multimodal understanding. And we can define the **semantic-aware redundancy** by applying a *negation* operation. We regard this general solution as the default due to its efficiency, and provide an alternative solution for a limited number of MLLMs without a `[CLS]` token (*e.g.*, SigLIP (Zhai et al. 2023)). Specifically, we average the keys of all visual tokens as an alternative of the `[CLS]` token, and regard its cosine similarity with visual tokens as a substitute for attention. Therefore, we calculate the overall redundancy score for each visual token as

$$\mathbf{s}^v_i = \lambda\frac{1}{N}\sum_{j=1}^{N}\mathbf{A}^v_{i,j} - (1-\lambda)\mathbf{a}^{\texttt{CLS}}_i, \tag{1}$$

where λ is a scalar hyperparameter that balances the factors. Since the visual tokens with *higher* redundancy scores are expected to be discarded, we filter out these tokens through a topK operation on the ranked scores, where the amount is related to the degree of reduction.

A concern is that tokens discarded in a layer might concentrate in a specific image area, potentially leading to spatial-centralized information loss. Therefore, we develop a **local penalty strategy** that encourages a more uniform spatial distribution of discarded tokens. Specifically, we represent the redundancy scoring vector $\mathbf{s}$ back to a 2D grid and partition it into non-overlapping windows of size W, using padding for previously discarded tokens to maintain the 2D information. Finally, we multiply the highest score within each window by a scaling coefficient, enhancing positive scores and diminishing negative ones. This effectively suppresses the global prominence of other large scores within the windows, reducing their likelihood of being discarded.

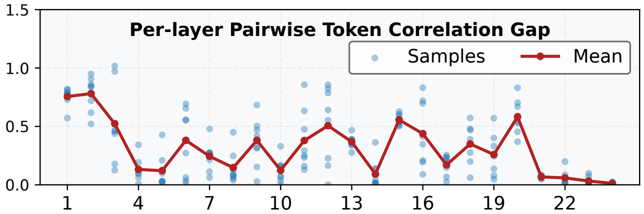

Figure 3: **Layer-wise distribution of Pairwise Token Correlation Gap.** We observe that the difference between the top-1 and top-2 correlation scores for each discarded token displays a high variance, highlighting the inadequacy of a fixed number of correlated preserved tokens.

As observed in the ablation study, this technique significantly enhances *FiCoCo-V*.

3.3 Correlation-based Information Recycling

Correlate: Where should discarded information be recycled? We conduct a matrix that evaluates the correlation between each discarded token and all the preserved visual tokens. Formally, given $N^{\mathbb{S}}$ discarded tokens, the matrix can be defined as $\mathbf{C} \in \mathbb{R}^{N^{\mathbb{S}} \times (N - N^{\mathbb{S}})}$. For *FiCoCo-V* in the vision encoder, attention weights inherently represent both the inter-token relationships and the flow of information, making them a measure of **direct** correlation. Therefore, the correlation matrix can be conducted as $\mathbf{C}^v_{i,j} = \mathbf{A}^v_{i,j}$.

To select the preserved tokens that receive information from each discarded token, a topK operation can be applied on each row of the correlation matrix $\mathbf{C}$. Here, K is the number of the selected preserved tokens, where $K = 0$ is equivalent to token pruning and $K > 0$ is token merging (*e.g.*, $K = 1$ for ToMe (Bolya et al. 2023)). To find an appropriate K, we measure the *Pairwise Token Correlation Gap*—the difference between the top-1 and top-2 correlation scores—for each discarded token. Our *FiCoCo-V* experiment computes this gap by discarding 8 tokens per layer across a 24-layer ViT. In Figure 3, we observe that the distribution of correlation gaps varies significantly. A few layers show a concentration of large gaps, indicating that discarded tokens can easily identify their single most correlated preserved token. In contrast, some layers have gaps concentrated near zero, suggesting each discarded token has multiple candidate preserved tokens with similarly high correlation values. And more layers display distributions with high variance, highlighting the inadequacy of a fixed K value.

According to the above analysis, we devise a **token-adaptive** K. Specifically, for the i-th discarded token, we compute the ε-th quantiles of the i-th row in the correlation matrix to determine a **token-wise threshold** τ_i. Then this threshold is re-applied to the row to identify the target tokens correlated to the i-th discarded token. In other word, for the j-th preserved token, if $\mathbf{C}_{i,j} \geq \tau_i$, then this preserved token can be viewed as a correlated token for the i-th discarded token. And the number of correlated tokens for each discarded token is dynamic and adaptive. Therefore, we actually construct **"dense" information pathways**, where the correlation matrix facilitates the tracking of the information propagation from each discarded token to the candidate tokens. In contrast, a "convergent" correlation prompts all discarded tokens to merge into an additional token (Liang et al. 2022). Compared to that strategy, our "dense" correlations spread discarded information more widely among the remaining tokens and empirically demonstrate better performance.

Compress: How to effectively recycle information? After the correlate stage, each preserved token has a variable number of discarded tokens for updating itself. A straightforward update strategy involves averaging each preserved token with all discarded tokens that correlated to it (Bolya et al. 2023). However, as the number of discarded tokens increases, this strategy results in the preserved token having less information about itself after updates. And excessive integration of information from discarded tokens into preserved tokens leads to performance degradation through progressive dilution of their original semantic content. Therefore, our compression strategy must ensure the dominance of the preserved tokens. Moreover, naive averaging results in the amount of information received by a preserved token being independent of its correlation to the discarded tokens.

According to the above discussion, we update the preserved tokens with a **self-preserving compression**. Formally, we define the discarded tokens as a source set $\mathbb{S}$, and the preserved visual tokens as a target set $\mathbb{T}$. Therefore, given the correlation matrix $\mathbf{C}$, we formulate the compression as

$$\mathbf{X}^{\mathbb{T}}_j \leftarrow \frac{\mathbf{X}^{\mathbb{T}}_j + \sum_{i \in \mathbb{I}_j} \alpha_{ij} \mathbf{X}^{\mathbb{S}}_i}{1 + \sum_{i \in \mathbb{I}_j} \alpha_{ij}}, \text{where } \mathbb{I}_j = \{ i \in \mathbb{S} \mid \mathbf{C}_{i,j} \geq \tau_i \},$$

$$\alpha_{ij} = \frac{\mathbf{C}_{i,j}}{\sum_{j \in \mathbb{J}_i} \mathbf{C}_{i,j}}, \text{where } \mathbb{J}_i = \{ j \in \mathbb{T} \mid \mathbf{C}_{i,j} \geq \tau_i \}, \tag{2}$$

where the weight α_{ij} quantifies the proportion of information from the i-th discarded token that is allocated to the j-th correlated token. The strategy guarantees each preserved token preserves at least 50% of its original information. Moreover, the preserved token can receive more information from a discarded token with a strong correlation.

4 Task-Aware Improvements for *FiCoCo-L*

Despite its promising performance, *FiCoCo-V* identifies and removes visual tokens based solely on the visual content. Applied within the vision encoder, such a task-agnostic method fails to preserve the essential visual information based on task context. Therefore, we provide a task-aware solution ***FiCoCo-L*** applied in the LLM decoder, leveraging textual priors to reduce visual token redundancy and recycle task-related visual information. Specifically, *FiCoCo-L* updates the redundancy calculation in the ***filter*** stage, and the correlation calculation in the ***correlate*** stage.

Task-aware redundancy calculation. In the LLM decoder, since textual tokens directly encode task instructions, the attention weights that visual tokens received from textual tokens indicate their task relevance. Therefore, we can calculate a **task-aware redundancy** as - $\frac{1}{M}\sum_{k=N+1}^{N+M} \mathbf{A}^l_{i,k}$, where M denotes the number of textual tokens. As a result,

Method	Source	TFLOPs↓	SQA	VQA^{T}	POPE	GQA	MMB	VQAv2	Avg	Avg(%)
				TFLOPs=8.5						
LLaVA-1.5-7B	*NeurIPS23*	8.5	69.5	58.2	86.4	62.5	66.1	79.1	70.3	100
				TFLOPs=3.3$_{(\downarrow 61.2\%)}$						
ToMe	*ICLR23*	3.3	65.2	52.1	72.4	54.3	60.5	68.0	62.1	88.3
FastV	*ECCV24*	3.3	67.3	52.5	64.8	52.7	61.2	67.1	60.9	86.6
SparseVLM	*ICML25*	3.3	69.1	56.1	83.6	57.6	62.5	75.6	67.4	95.9
PDrop	*CVPR25*	3.3	68.8	56.1	82.3	57.1	63.2	75.1	67.1	95.4
PruMerge	*ICCV25*	3.3	67.9	54.3	71.3	54.3	59.6	70.6	63.0	89.6
FiCoCo-V	Ours	3.3	67.8	55.7	82.5	58.5	62.3	74.4	66.9	95.2
FiCoCo-L	Ours	3.3	**69.6**	**56.6**	**84.6**	**61.1**	**64.6**	**76.8**	**68.9**	**98.0**
				TFLOPs=2.4$_{(\downarrow 71.8\%)}$						
ToMe	*ICLR23*	2.5	59.6	49.1	62.8	52.4	53.3	63.0	56.7	80.7
FastV	*ECCV24*	2.5	60.2	50.6	59.6	49.6	56.1	61.8	56.3	80.1
SparseVLM	*ICML25*	2.5	67.1	54.9	80.5	56.0	60.0	**73.8**	65.4	93.0
PDrop	*CVPR25*	2.5	68.3	55.1	82.3	56.0	61.1	72.9	65.9	93.8
PruMerge	*ICCV25*	2.5	67.1	54.3	67.2	53.3	58.1	68.8	61.5	87.5
FiCoCo-V	Ours	2.4	68.3	55.6	82.2	57.6	61.1	73.1	66.3	94.3
FiCoCo-L	Ours	2.4	**69.4**	**56.3**	**84.4**	**60.6**	**61.9**	73.4	**67.7**	**96.3**
				TFLOPs=1.5$_{(\downarrow 82.4\%)}$						
ToMe	*ICLR23*	1.6	50.0	45.3	52.5	48.6	43.7	57.1	49.5	70.4
FastV	*ECCV24*	1.6	51.1	47.8	48.0	46.1	48.0	61.8	50.5	71.8
SparseVLM	*ICML25*	1.5	62.2	51.8	75.1	52.4	56.2	68.2	61.0	86.8
PDrop	*CVPR25*	1.6	68.6	45.9	55.9	41.9	33.3	69.2	52.5	74.6
PruMerge	*ICCV25*	1.5	68.1	54.0	65.3	51.9	55.3	67.4	60.3	85.8
FiCoCo-V	Ours	1.5	68.4	55.5	79.8	**54.9**	60.2	**72.1**	65.2	92.7
FiCoCo-L	Ours	1.5	**69.5**	**55.7**	**82.1**	53.2	**61.5**	69.7	**65.3**	**92.8**
				TFLOPs=24.9						
LLaVA-1.5-13B	*NeurIPS23*	24.9	71.4	61.3	86.2	63.4	68.0	80.0	71.7	100
				TFLOPs=15.4$_{(\downarrow 47.6\%)}$						
FastV	*ECCV24*	15.4	57.0	56.0	79.3	57.7	57.9	-	61.6	85.9
SparseVLM	*ICML25*	15.4	69.9	49.9	81.1	57.9	65.8	-	64.9	90.5
FiCoCo-V	Ours	15.4	72.1	57.2	82.3	59.2	63.1	76.8	68.5	95.5
FiCoCo-L	Ours	15.4	**72.4**	**58.3**	**83.1**	**60.1**	**65.2**	**77.6**	**69.5**	**96.9**

Table 1: **Comparison results on LLaVA-1.5-7B/13B.** We evaluate *FiCoCo* variants under various computational budgets, compared to baselines. Only shared datasets across both model sizes are included here.

the overall redundancy for *FiCoCo-L* is summarized as

$$\mathbf{s}_i^l = \beta \frac{1}{N} \sum_{j=1}^{N} \mathbf{A}_{i,j}^l - (1-\beta) \frac{1}{M} \sum_{k=N+1}^{N+M} \mathbf{A}_{i,k}^l, \qquad (3)$$

where the scalar hyperparameter β balances the factors.

Empirically, we observe that the "*local penalty*" strategy slightly degrades the performance of *FiCoCo-L*. We believe the reason is that this strategy weakens the task prior when encouraging spatial-uniform preservation of visual information. Consequently, we remove the strategy in *FiCoCo-L*.

Task-aware correlation calculation. We explore an additional form of **indirect** semantic correlation for *FiCoCo-L*, which leverages textual tokens as a bridge. Specifically, when measuring the association between the i-th token and the j-th token, we sum the products of the attention weights from the i-th token to all textual tokens and from all textual tokens to the j-th token. If the peak attention weights of the i-th token and the j-th token are concentrated on the same textual tokens, then the computed correlation between them is higher. In summary, we have

$$\mathbf{C}_{i,j}^l = \gamma \mathbf{A}_{i,j}^l + (1-\gamma) \frac{1}{M} \sum_{k=N+1}^{N+M} \mathbf{A}_{i,k}^l \cdot \mathbf{A}_{k,j}^l, \qquad (4)$$

where γ is the scalar hyperparameter for factor balance. Note that for clarity, our formula calculations are designed to target individual elements within vectors or matrices. However, these operations can be efficiently tensorized in the practical implementation to facilitate batched inference. And the implementation can be plug and play with less than 10 lines of additional code.

5 Experiments

5.1 Experimental Setups

We evaluate *FiCoCo* on multiple MLLMs: LLaVA-1.5 (Liu et al. 2024a), LLaVA-NeXT (Liu et al. 2024b), and Qwen2-VL (Wang et al. 2024) for image understanding, and Video-LLaVA (Lin et al. 2024a) for video understanding. *FiCoCo* is benchmarked against mainstream token reduction methods: ToMe (Bolya et al. 2023), FastV (Chen et al. 2024a), SparseVLM (Zhang et al. 2024), PDrop (Xing et al. 2025), and PruMerge (Shang et al. 2025).

5.2 Main Comparisons

Results on LLaVA-1.5-7B/13B. Table 1 presents the performance of *FiCoCo* across benchmarks based on LLaVA-1.5-7B/13B. The LLaVA-1.5-7B results yield two key findings: **(1)** Both *FiCoCo-V* and *FiCoCo-L* consistently outperform existing training-free methods across different computational budgets. Under extreme compression (TFLOPs = 1.5, 10% visual tokens), both variants achieve **>92%** average accuracy, surpassing the second-best SparseVLM

Method	MMB	SQA	VQA^T	MMMU	Avg	Avg (%)
			TFLOPs=42.7			
LLaVA-NeXT-7B	67.9	70.2	61.3	35.1	58.6	100.0
			TFLOPs=5.0$_{(\downarrow 88.3\%)}$			
PDrop	63.4	67.5	54.4	29.8	53.8	91.7
			TFLOPs=2.9$_{(\downarrow 93.2\%)}$			
SparseVLM	63.1	67.5	46.3	32.8	52.4	89.4
FiCoCo-V	60.5	**68.1**	**55.3**	34.1	54.5	93.0
FiCoCo-L	**63.6**	67.9	53.1	**34.8**	**54.9**	**93.6**

Table 2: Comparison with LLaVA-NeXT-7B on cross-image understanding benchmarks.

Token	Method	MMB	POPE	VQA^T	Avg	Avg (%)
		Base. ≈1300				
	Qwen2-VL-7B	80.5	86.4	84.3	83.7	100.00
600	SparseVLM	79.6	86.5	80.3	82.1	98.09
	FiCoCo-V	79.9	86.5	81.2	82.5	98.57
	FiCoCo-L	80.1	86.3	81.4	**82.6**	**98.69**
500	SparseVLM	78.8	86.3	79.0	81.4	97.25
	FiCoCo-V	79.1	86.6	79.8	**81.8**	**97.73**
	FiCoCo-L	78.9	86.1	79.7	81.6	97.49
400	SparseVLM	79.0	85.8	77.1	80.6	96.29
	FiCoCo-V	79.1	86.0	78.3	**81.1**	**96.89**
	FiCoCo-L	78.6	85.9	78.3	80.9	96.66

Table 3: **Comparison with Qwen2-VL-7B under different token budgets.** *FiCoCo* is compared against SparseVLM under 600/500/400 tokens on MMB, POPE, and VQA^T.

by approximately **6%**, demonstrating the effectiveness of our information recovery mechanism. **(2)** *FiCoCo-L* outperforms *FiCoCo-V* when computational budgets are generous, as it captures task-relevant visual tokens and maximally **focuses** on question-related regions. However, under constrained budgets, both variants achieve similar performance (with an average difference of only **0.1%**). This convergence arises because, under severe token constraints, the distinction between preserving visual saliency (*FiCoCo-V*) and task-specific relevance (*FiCoCo-L*) diminishes—both are compelled to retain only the most essential visual elements for maintaining core model functionality. Moreover, under the LLaVA-1.5-13B setting, *FiCoCo-V/L* achieves superior performance with only ∼22% visual tokens (15.4 TFLOPs), outperforming the strongest baseline SparseVLM by **5.0%** and **6.4%** in average accuracy.

Results on LLaVA-NeXT-7B. We impose two computational constraints: TFLOPs are set to 5.0 for PDrop to match reported results, and 2.9 for SparseVLM and *FiCoCo*. Table 2 shows that, under TFLOPs = 2.9, *FiCoCo-V* and *FiCoCo-L* outperform SparseVLM by **3.6%** and **4.2%**, respectively. Moreover, despite operating under a lower TFLOPs budget than PDrop, our methods *FiCoCo-V* and *FiCoCo-L* consistently outperform PDrop by **1.9%** in terms of average accuracy, highlighting their superior efficiency and robustness in handling dense visual token scenarios within resource-constrained settings.

Results on Qwen2-VL. Following SparseVLM's settings on Qwen2-VL, we compress about 54.5% of visual tokens. Since Qwen2-VL lacks a `[CLS]` token, an equivalent token averaging scheme is used in *FiCoCo-V*. As shown in Table 3, under this ratio, *FiCoCo* maintains over **98%** accuracy and surpasses SparseVLM. Further, compressing each additional 100 tokens yields only a 0.8% drop, highlighting *FiCoCo*'s robustness and the effectiveness of the averaging scheme.

Method	TGIF	MSVD	MSRVTT	ActivityNet	Avg	Avg(%)
			TFLOPs=29.7			
Video-LLaVA	47.1	69.8	56.7	43.1	54.2	100.0
			TFLOPs=2.6$_{(\downarrow 91.2\%)}$			
FastV	23.1	38.0	19.3	30.6	27.8	52.1
SparseVLM	44.7	**68.2**	31.0	42.6	46.9	86.5
FiCoCo-V	**43.1**	67.4	47.8	**42.8**	**50.3**	**92.8**
FiCoCo-L	44.3	64.5	**49.2**	40.1	49.5	91.4

Table 4: Comparison results on video understanding benchmarks with Video-LLaVA.

Stage	Method	SQA	TextVQA
	FiCoCo-V	**68.37**	**55.46**
Filter	w/o vision-aware redundancy	67.81	52.51
	w/o semantic-aware redundancy	64.67	48.74
	w/o local penalty	68.12	53.24
Correlate	fixed K=0	67.82	53.56
	fixed K=1	67.43	46.97
	fixed K=2	67.21	51.36
	convergent correlation	67.60	54.38
Compress	average compression	67.92	53.34

Table 5: Ablation results of *FiCoCo-V*.

Results on Video-LLaVA. For fair evaluation, the number of video tokens is limited to 136 (about 6.6% of all visual tokens). As shown in Table 4, both *FiCoCo-V* and *FiCoCo-L* reach over **90%** of Video-LLaVA's performance, while *FiCoCo-V* surpasses FastV and SparseVLM by **40.7%** and **6.3%**, respectively, demonstrating its superiority in video understanding. The stronger performance of *FiCoCo-V* over *FiCoCo-L* likely stems from higher visual redundancy in videos, where compression aids attention to salient content.

5.3 Ablation Study

To further validate the effectiveness of our design at each stage, we conduct extensive ablation studies on the SQA and TextVQA benchmarks under a fixed computational budget of 1.5 TFLOPs. In Table 5, we ablate all three stages of ***FiCoCo-V*** to analyze their individual contributions.

• **Filter.** Both vision-aware and semantic-aware redundancy improve the identification of discarded tokens. Notably, semantic-aware redundancy has a more significant impact on the final performance. This indicates that token reduction within the vision encoder should prioritize the retention of tokens rich in global semantic information. Additionally, we observe that by promoting a spatially uniform distribution of discarded tokens, the local penalty strategy aids in preserving visual information.

• **Correlate.** We evaluate fixed K values of 0 (pruning, *i.e.*, no correlation-based recycling), 1 (single-token recycling), and 2 (multi-token recycling). While the token-adaptive K strategy performs best, an intriguing result is that K=0 surpasses the other two. This likely occurs because small fixed K values limit information sources for correlated tokens, causing over-dilution and noise. Thus, pruning yields bet-

Stage	Method	SQA	TextVQA
	FiCoCo-L	**69.46**	**55.72**
Filter	w/o vision-aware redundancy	69.16	55.43
	w/o task-aware redundancy	68.22	55.64
	w/ local penalty	68.79	55.38
Correlate	w/o indirect correlation	68.89	54.78
	w/o direct correlation	68.45	55.45
	fixed K=0	68.96	50.33
	fixed K=1	68.57	50.11
	fixed K=2	68.32	50.18
	convergent correlation	67.80	54.89
Compress	average compression	68.32	54.66

Table 6: Ablation results of *FiCoCo-L*.

ter performance. Moreover, our "dense" correlation outperforms the "convergent" variant, which compresses discarded tokens into one; retrieving information while preserving token integrity proves more effective.

• **Compress.** Our self-preserving compression outperforms directly averaging the features, indicating that the calculated weights can effectively regulate the contribution of information sources in the updates of correlated tokens.

In Table 6, we ablate all three stages for ***FiCoCo-L***:

• **Filter.** Although both vision-aware and task-aware redundancies contribute to redundancy estimation, neither dominates. This may be because the attention mechanism in LLMs captures stable token dependencies, reducing the need for redundancy measurement to rely heavily on semantic cues. Moreover, applying the local penalty strategy in *FiCoCo-L* slightly degrades performance, likely because enforcing spatial uniformity of token retention disrupts the redundancy assessments already established by attention.

• **Correlate.** It is observed that both correlations enhance the identification of relevant tokens, improving performance across datasets. Similar to *FiCoCo-V*, adopting a token-adaptive K with "dense" correlations proves optimal.

• **Compress.** Updating the preserved tokens with a self-preserving compression still achieves better performance.

5.4 Qualitative Analysis

We visualize the discarded tokens of *FiCoCo-V* (Figure 4a) and *FiCoCo-L* (Figure 4b) under varying compression levels across VQA scenarios. Tokens highly relevant to answers are highlighted with red boxes to assess information preservation. A token linked to "2" is traced in Figure 4a, and one linked to "*GAMES*" in Figure 4b. In both cases, as compression increases (TFLOPs from 4.2 to 1.5), more tokens—including key ones—are discarded, reducing critical information. We trace information recycling from discarded tokens (red arrows) and highlight correlated tokens (green boxes), where transparency reflects retained information. These correlated tokens aggregate essential cues and support question answering. Notably, discarded information is distributed across multiple correlated tokens, enhancing comprehension of salient regions (Figure 4b), qualitatively validating our method.

Figure 4: **Visualizations of token discard and information recycling by (a) *FiCoCo-V* and (b) *FiCoCo-L*.** Red: traced patch token; Green: recycling destination.

Method	LLaVA-NeXT-7B TFLOPs	LLaVA-NeXT-7B Throughput	LLaVA-1.5-7B TFLOPs	LLaVA-1.5-7B Throughput
Vanilla	42.7	3.8	8.5	8.99
FiCoCo-V	2.9 (↓ **93.2%**)	7.9 (↑ **107.9%**)	1.5 (↓ **82.4%**)	12.9 (↑ **43.5%**)
FiCoCo-L	2.9 (↓ **93.2%**)	6.5 (↑ **71.1%**)	1.5 (↓ **82.4%**)	11.6 (↑ **29.1%**)

Table 7: **Efficiency analysis of *FiCoCo*.** Lower TFLOPs and higher throughput (img/s) indicate better efficiency.

5.5 Efficiency Analysis

As shown in Table 7, we present the trends of throughput and TFLOPs changes after applying *FiCoCo* in the LLaVA-NeXT and LLaVA-1.5 architectures. Introducing *FiCoCo* into LLaVA-NeXT reduces TFLOPs by 93.2%, increasing throughput by $2.08\times$ (*FiCoCo-V*) and $1.71\times$ (*FiCoCo-L*). In LLaVA, a TFLOPs reduction of 82.4% yields throughput gains of $1.43\times$ and $1.29\times$, respectively. These results highlight *FiCoCo*'s ability to substantially lower computational overhead while enhancing throughput.

6 Conclusion

In this paper, we propose a "*filter-correlate-compress*" acceleration framework to systematically eliminate visual redundancy in MLLMs through a principled three-stage pipeline. The *filter* stage performs dynamic redundancy-based token discard using variation-aware thresholds, while the *correlate* stage identifies semantic relationships between tokens and the *compress* stage jointly enables correlation-based information recycling, thereby significantly reducing computational complexity while preserving critical multimodal information. The effectiveness of our framework is rigorously demonstrated through specialized variants for vision encoders (*FiCoCo-V*) and LLM decoders (*FiCoCo-L*), achieving consistent acceleration benefits across diverse MLLM architectures for both image and video understanding tasks with minimal accuracy degradation.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (No. 62301432), the Natural Science Basic Research Program of Shaanxi Province (Grant No. QCYRCXM-2023-057), the Fundamental Research Funds for the Central Universities, and the Guangdong Basic and Applied Basic Research Foundation (Grant No. 2025A1515011119). It was also supported by the Chengdu Science and Technology Program (Grant No. 2025-YF12-00006-RC), the Police Integration Computing Key Laboratory of Sichuan Province (Grant No. JWRH202502002), and the Open Fund of the Key Laboratory of the Ministry of Education on Artificial Intelligence in Equipment (Grant No. 2024-AAIE-KF04-03). This work was supported by the National Science and Technology Innovation 2030 - Major Project (Grant No. 2022ZD0208800), and NSFC General Program (Grant No. 62176215)

References

Bai, J.; Bai, S.; Chu, Y.; Cui, Z.; Dang, K.; Deng, X.; Fan, Y.; Ge, W.; Han, Y.; Huang, F.; Hui, B.; Ji, L.; Li, M.; Lin, J.; Lin, R.; Liu, D.; Liu, G.; Lu, C.; Lu, K.; Ma, J.; Men, R.; Ren, X.; Ren, X.; Tan, C.; Tan, S.; Tu, J.; Wang, P.; Wang, S.; Wang, W.; Wu, S.; Xu, B.; Xu, J.; Yang, A.; Yang, H.; Yang, J.; Yang, S.; Yao, Y.; Yu, B.; Yuan, H.; Yuan, Z.; Zhang, J.; Zhang, X.; Zhang, Y.; Zhang, Z.; Zhou, C.; Zhou, J.; Zhou, X.; and Zhu, T. 2023a. Qwen Technical Report. *arXiv preprint arXiv:2309.16609*.

Bai, J.; Bai, S.; Yang, S.; Wang, S.; Tan, S.; Wang, P.; Lin, J.; Zhou, C.; and Zhou, J. 2023b. Qwen-VL: A Frontier Large Vision-Language Model with Versatile Abilities. *arXiv preprint arXiv:2308.12966*.

Bolya, D.; Fu, C.; Dai, X.; Zhang, P.; Feichtenhofer, C.; and Hoffman, J. 2023. Token Merging: Your ViT But Faster. In *Proceedings of the International Conference on Learning Representations*.

Cha, J.; Kang, W.; Mun, J.; and Roh, B. 2024. Honeybee: Locality-Enhanced Projector for Multimodal LLM. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13817–13827.

Chen, L.; Zhao, H.; Liu, T.; Bai, S.; Lin, J.; Zhou, C.; and Chang, B. 2024a. An Image is Worth 1/2 Tokens After Layer 2: Plug-and-Play Inference Acceleration for Large Vision-Language Models. In *Proceedings of the European Conference on Computer Vision*, 19–35.

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. 2024b. 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*, 24185–24198.

Dai, W.; Li, J.; Li, D.; Tiong, A. M. H.; Zhao, J.; Wang, W.; Li, B.; Fung, P.; and Hoi, S. C. H. 2023. InstructBLIP: Towards General-purpose Vision-Language Models with Instruction Tuning. In *Proceedings of the Advances in Neural Information Processing Systems*, 49250–49267.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *Proceedings of the International Conference on Learning Representations*.

Feichtenhofer, C.; Fan, H.; Li, Y.; and He, K. 2022. Masked Autoencoders As Spatiotemporal Learners. In *Proceedings of the Advances in Neural Information Processing Systems*, 35946–35958.

He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. B. 2022. Masked Autoencoders Are Scalable Vision Learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15979–15988.

Li, W.; Yuan, Y.; Liu, J.; Tang, D.; Wang, S.; Zhu, J.; and Zhang, L. 2024. TokenPacker: Efficient Visual Projector for Multimodal LLM. *arXiv preprint arXiv:2407.02392*.

Liang, Y.; Ge, C.; Tong, Z.; Song, Y.; Wang, J.; and Xie, P. 2022. Not All Patches are What You Need: Expediting Vision Transformers via Token Reorganizations. In *Proceedings of the International Conference on Learning Representations*.

Lin, B.; Ye, Y.; Zhu, B.; Cui, J.; Ning, M.; Jin, P.; and Yuan, L. 2024a. Video-LLaVA: Learning United Visual Representation by Alignment Before Projection. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 5971–5984.

Lin, J.; Yin, H.; Ping, W.; Molchanov, P.; Shoeybi, M.; and Han, S. 2024b. VILA: On Pre-training for Visual Language Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 26679–26689.

Liu, H.; Li, C.; Li, Y.; and Lee, Y. J. 2024a. Improved Baselines with Visual Instruction Tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 26286–26296.

Liu, H.; Li, C.; Li, Y.; Li, B.; Zhang, Y.; Shen, S.; and Lee, Y. J. 2024b. LLaVA-NeXT: Improved reasoning, OCR, and world knowledge.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023. Visual Instruction Tuning. In *Proceedings of the Advances in Neural Information Processing Systems*, 34892–34916.

Liu, X.; Gui, X.; Zhang, Y.; and Zhang, L. 2025a. Mixing Importance with Diversity: Joint Optimization for KV Cache Compression in Large Vision-Language Models. *arXiv preprint arXiv:2510.20707*.

Liu, X.; Wang, Y.; Ma, J.; and Zhang, L. 2025b. Video Compression Commander: Plug-and-Play Inference Acceleration for Video Large Language Models. *arXiv preprint arXiv:2505.14454*.

Liu, X.; Wang, Z.; Han, Y.; Wang, Y.; Yuan, J.; Song, J.; Zheng, B.; Zhang, L.; Huang, S.; and Chen, H. 2025c. Global Compression Commander: Plug-and-Play Inference Acceleration for High-Resolution Large Vision-Language Models. *arXiv preprint arXiv:2501.05179*.

Liu, X.; Wen, Z.; Wang, S.; Chen, J.; Tao, Z.; Wang, Y.; Jin, X.; Zou, C.; Wang, Y.; Liao, C.; et al. 2025d. Shifting ai efficiency from model-centric to data-centric compression. *arXiv preprint arXiv:2505.19147*.

OpenAI. 2023. GPT-4 Technical Report. *arXiv preprint arXiv:2303.08774*.

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. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the International Conference on Machine Learning*, 8748–8763.

Rao, Y.; Zhao, W.; Liu, B.; Lu, J.; Zhou, J.; and Hsieh, C. 2021. DynamicViT: Efficient Vision Transformers with Dynamic Token Sparsification. In Ranzato, M.; Beygelzimer, A.; Dauphin, Y. N.; Liang, P.; and Vaughan, J. W., eds., *Proceedings of the Advances in Neural Information Processing Systems*, 13937–13949.

Shang, Y.; Cai, M.; Xu, B.; Lee, Y. J.; and Yan, Y. 2025. LLaVA-PruMerge: Adaptive Token Reduction for Efficient Large Multimodal Models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; Rodriguez, A.; Joulin, A.; Grave, E.; and Lample, G. 2023. LLaMA: Open and Efficient Foundation Language Models. *arXiv preprint arXiv:2302.13971*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All you Need. In *Proceedings of the Advances in Neural Information Processing Systems*, 5998–6008.

Wang, P.; Bai, S.; Tan, S.; Wang, S.; Fan, Z.; Bai, J.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Fan, Y.; Dang, K.; Du, M.; Ren, X.; Men, R.; Liu, D.; Zhou, C.; Zhou, J.; and Lin, J. 2024. Qwen2-VL: Enhancing Vision-Language Model's Perception of the World at Any Resolution. *arXiv preprint arXiv:2409.12191*.

Xing, L.; Huang, Q.; Dong, X.; Lu, J.; Zhang, P.; Zang, Y.; Cao, Y.; He, C.; Wang, J.; Wu, F.; and Lin, D. 2025. 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*.

Zhai, X.; Mustafa, B.; Kolesnikov, A.; and Beyer, L. 2023. Sigmoid Loss for Language Image Pre-Training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 11941–11952.

Zhang, H.; Li, X.; and Bing, L. 2023. Video-LLaMA: An Instruction-tuned Audio-Visual Language Model for Video Understanding. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 543–553.

Zhang, Y.; Fan, C.-K.; Ma, J.; Zheng, W.; Huang, T.; Cheng, K.; Gudovskiy, D.; Okuno, T.; Nakata, Y.; Keutzer, K.; and Zhang, S. 2024. SparseVLM: Visual Token Sparsification for Efficient Vision-Language Model Inference. *arXiv preprint arXiv:2410.04417*.