AIM: <u>A</u>daptive <u>I</u>nference of <u>M</u>ulti-Modal LLMs
via Token Merging and Pruning

Yiwu Zhong[1], Zhuoming Liu[2], Yin Li[2], Liwei Wang[1*]

[1]The Chinese University of Hong Kong, [2]University of Wisconsin-Madison

Abstract

*Large language models (LLMs) have enabled the creation of multi-modal LLMs that exhibit strong comprehension of visual data such as images and videos. However, these models usually rely on extensive visual tokens from visual encoders, leading to high computational demands, which limits their applicability in resource-constrained environments and for long-context tasks. In this work, we propose a training-free adaptive inference method for multi-modal LLMs that can accommodate a broad range of efficiency requirements with a minimum performance drop. Our method consists of a) iterative token merging based on embedding similarity before LLMs, and b) progressive token pruning within LLM layers based on multi-modal importance. With a minimalist design, our method can be applied to both video and image LLMs. Extensive experiments on diverse video and image benchmarks demonstrate that, our method substantially reduces computation load (e.g., a **7-fold** reduction in FLOPs) while preserving the performance of video and image LLMs. Further, under a similar computational cost, our method outperforms the state-of-the-art methods in long video understanding (e.g., **+4.6** on MLVU). Additionally, our in-depth analysis provides insights into token redundancy and LLM layer behaviors, offering guidance for future research in designing efficient multi-modal LLMs. Our code will be available at* `https://github.com/LaVi-Lab/AIM`.

1. Introduction

Large language models (LLMs) [6, 52, 63, 80, 89] have been recently adapted for visual understanding, fostering the developments in both image [4, 7, 22, 38, 96] and video LLMs [33, 36, 44, 77, 91]. However, these multi-modal LLMs usually rely on a large number of visual tokens generated by visual encoders [56, 86], especially for video data where token counts can reach thousands per video. Such high token number requires extensive computational
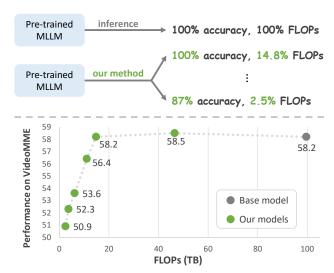
---

*Corresponding author.

Figure 1. Our proposed training-free method enables adaptive inference of pre-trained multi-modal LLMs, supporting a wide range of computational conditions. Compared with the base pre-trained model, our method substantially reduces FLOPs, without or with a manageable performance drop.

resources for both training and inference, restricting their use in real-world applications (*e.g.*, real-time processing on mobile devices). Further, as the number of video frames increases, the total number of tokens also grows, limiting the model's capacity to handle dense video frames. This often results in the loss of crucial temporal information, which is essential for comprehending long videos [60, 88, 94].

To bridge the gap, we propose leveraging the inherent redundancy present in visual data to develop *adaptive inference* for multi-modal LLMs. Adaptive inference dynamically adjusts a model's computational load during inference based on contextual factors [19], *e.g.*, image content, computational constraints, or desired accuracy levels. Since retaining all visual tokens during inference is often unnecessary, our key insight is to strategically select these tokens throughout the inference process. This approach allows for controlling the amount of computation necessary for inference, so as to optimize the model's efficiency while preserving its accuracy.

In this work, we propose a training-free method of adaptive inference tailored for multi-modal LLMs, consisting of token merging based on visual similarity and token pruning based on multi-modal importance. Specifically, the visual encoder transforms the input visual data into visual tokens, which are iteratively merged based on their embedding similarities. The merged tokens are then fed into the LLM, where the tokens considered less important for multi-modal reasoning are progressively pruned at each layer, resulting in a gradual reduction of tokens. By adjusting the key parameters in the token merging and token pruning process, our method enables adaptive inference that achieves various levels of computation reduction with manageable performance loss, shown in Figure 1. With this capability, we can select a configuration of token merging and pruning so that the model meets the resource constraint while the performance is optimized.

Additionally, during the development of our method, we have several findings that can be useful for designing efficient multi-modal LLMs in the future. **First**, the full set of video tokens is often unnecessary, as only 25% visual tokens fed into LLM can maintain close performance. **Second**, with fewer tokens per frame, LLMs can process a larger number of frames, thereby addressing information loss and improving performance, especially for long video understanding. **Third**, we observe that pruning text tokens across LLM layers or removing visual tokens in the early LLM layers significantly impacts performance. In contrast, pruning a large proportion of visual tokens in the later layers can maintain performance. These observations suggest that multi-modal LLMs focus on cross-modal fusion in the earlier layers while prioritizing text tokens in later layers.

We conduct extensive experiments on diverse video benchmarks and image benchmarks, with base models as LLaVA-OneVision [59] and LLaVA-1.5 [38], respectively. Without additional fine-tuning, our method achieves a substantial reduction in computational demands (*e.g.*, decreasing FLOPs and prefill time by a factor of **6.8** and **8.0**, respectively) while retaining performance close to the base video LLM and image LLM. Moreover, given the same computation resources, our method allows for involving more frames as inputs and even surpasses the state-of-the-art (SOTA) video LLM on long video understanding (*e.g.*, **+4.6** on the MLVU benchmark). We further provide an in-depth ablation study, offering insights into the redundancy of visual tokens and the behavior of individual LLM layers, which can guide future research on multi-modal LLMs. Altogether, these promising results are attributed to the capability of our adaptive inference that can accommodate diverse computational conditions (*e.g.*, **40-fold** reduction in FLOPs) with a manageable drop in performance.

Our contributions are summarized as follows:

• We introduce a training-free method that enables adaptive

inference of pre-trained multi-modal LLMs. It can reduce visual token redundancy and computational demand while preserving the base model's performance.
• Our approach adopts a minimalist design that merges visual tokens before the LLM and prunes visual tokens within the LLM progressively. This design is applicable to various multi-modal LLM models.
• Thanks to adaptive inference, our method achieves a substantial reduction in computational cost while preserving the performance of base image and video LLMs, and even outperforms SOTA video LLMs given similar FLOPs.

2. Related Work

Multi-modal LLMs. Large language models [46, 50, 51] trained on large-scale datasets exhibit strong performance in text understanding and generation tasks, laying the foundation for developing multi-modal LLMs with visual instruction tuning [7, 39, 96]. To further enhance model capabilities, recent work has focused on increasing visual input resolution [38, 40] and creating higher-quality instruction-tuning datasets [30]. Meanwhile, advancements in video-based LLMs have also been achieved by extending visual instruction tuning to video modality. For example, VideoChatGPT [44] pioneers video-based instruction tuning through the creation of video question-answer pairs, while Video-LLaVA [36] improves model performance by employing text-aligned video features [95]. In parallel, several models have extended their capabilities from static images to dynamic video inputs, such as LLaVA-OneVision [30, 92], Qwen2-VL [66], and mPlug-Owl3 [82].

Recent work has focused on improving video LLMs for long video understanding. Given the challenge posed by the large number of tokens in long videos, recent approaches [10, 29, 49, 68, 70, 73, 88] adopt Q-former [32] or state space models [18] to aggregate the information before feeding visual tokens into the LLMs. Others extend the context window of LLMs [21, 71, 88] or design sequence parallelism from a systems perspective [79] to process the long video sequence. Built on top of the pre-trained image and video LLMs, our adaptive inference method aims to reduce token redundancy during inference and adapt to diverse efficiency requirements while maintaining model performance.

Token Merging and Pruning. Transformers are widely used in deep learning models, yet they come with high computational demands. Token merging and pruning methods aim to reduce the number of tokens to reduce this load, commonly developed in both NLP models [15, 26, 27, 64, 81] and vision models [8, 28, 41, 57, 62, 72, 84]. These approaches typically require fine-tuning after token reduction. By contrast, some training-free methods have been developed for efficient vision Transformers [3, 11, 65]. Different from these works, we propose a training-free approach tailored for multi-modal LLMs.
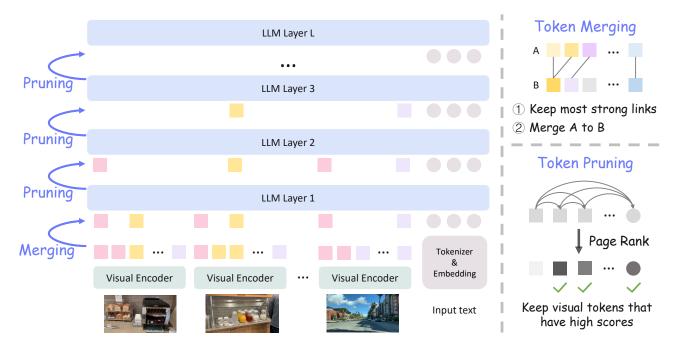
Figure 2. An overview of our training-free method for pre-trained multi-modal LLMs, optimized for accuracy-efficiency trade-off. During inference, we first merge the input visual tokens of LLM based on the cosine similarity between token embeddings, reducing their redundancy. The retained tokens are then fed into the LLM. Token pruning is applied within the LLM layers using the Page Rank algorithm, with a scheduler controlling the retention ratio of each layer. By adjusting the merging ratio and pruning scheduler parameters, our approach enables adaptive inference of multi-modal LLMs, accommodating various computational constraints with minimal performance loss.

For multi-modal LLMs, token pruning has been explored recently [5, 59]. FastV [5] prunes visual tokens at a particular selected LLM layer, while LLaVA-Prumerge [59] leverages the key-value pairs from the vision encoder to prune visual tokens before LLM. Unlike them, our method performs token reduction both before and across LLM layers, allowing for adaptive inference fitting with various computation constraints.

There are concurrent efforts on arXiv [76, 83, 90, 93] that also seek to reduce the computation of multi-modal LLMs. They are only applied to image LLMs, prune tokens exclusively within LLM, or rely on fine-tuning for an accuracy-efficiency balance. In comparison, our work is training-free, prunes visual tokens before and within LLM, generalizes to both video and image LLMs, and moreover, supports adaptive inference that accommodates a wide range of computation budgets with minimal performance loss.

Adaptive Inference. The ability to dynamically adjust the computational complexity of prediction models based on input data, latency constraints, or desired accuracy levels has received considerable interest in the vision and learning community [19]. Early attempts primarily focus on feature selection within multi-stage prediction pipelines [17, 25, 78]. More recent efforts have extended this concept to the adaptive inference of deep models. Methods have been developed for both convolutional networks and vi-

sion Transformers, enabling techniques such as input resampling, operation skipping, layer dropping, and early exiting [2, 12, 20, 24, 31, 47, 48, 54, 57, 67, 69, 74]. Similar ideas have also been explored for LLMs [9, 58]. In contrast to previous work, we propose an adaptive inference method tailored for multi-modal LLMs via merging similar tokens and pruning unimportant tokens in multi-modal reasoning.

3. Method

Multi-modal LLMs for images and videos usually require processing a high volume of visual tokens, resulting in substantial computational costs, especially for video tasks. Such computational load arises from the redundancy within visual tokens. To tackle this efficiency problem, we propose a training-free adaptive inference method by pruning redundant tokens in two stages, as shown in Figure 2. First, before visual tokens enter the LLM, we merge highly similar visual tokens, significantly reducing their count while preserving performance. Second, within the LLM, we further rank the visual tokens and remove less important ones at each layer, by applying the Page Rank algorithm to the self-attention weights. Together, our approach enables adaptive inference of multi-modal LLMs, reducing computational demands without compromising reasoning performance, and moreover, supports a broad range of computational requirements.

3.1. Multi-modal LLMs

Given an image, a typical image LLM [38] first transforms the image into visual tokens via a visual encoder, then projects visual tokens via an adapter (*e.g.*, multilayer perceptron), and finally concatenates the projected visual tokens and text tokens to perform LLM reasoning. Video LLMs [30] adopt a similar approach to image LLMs, except that the input visual data become uniformly sampled images (video frames), and adaptive pooling is oftentimes applied before LLMs to reduce the number of visual tokens. Despite this pooling, the visual token count remains high (*e.g.*, thousands of tokens per video), resulting in substantial computational demands for the LLM.

Our training-free method is designed to reduce the redundancy of visual tokens, by applying it to the input visual tokens of the LLM and to the visual tokens at each LLM layer. We denote the visual tokens right before LLM as $\mathbf{v^0} \in \mathcal{R}^{N_0 \times D}$ where N_0 is the initial number of visual tokens. Within LLM, we denote the input visual tokens and the input text tokens at the l-th layer as $\mathbf{v^l} \in \mathcal{R}^{N_l \times D}$ and $\mathbf{t^l} \in \mathcal{R}^{M_l \times D}$, respectively. $l \in [1, L]$ denotes the index of LLM layer and N_l (M_l) represents the number of visual (text) tokens at layer l.

3.2. Token Merging before LLM

The input to LLMs often consists of hundreds of visual tokens for an image or thousands for a video, resulting in substantial redundancy. Merging these redundant tokens before they enter the LLM can significantly reduce computational demands. Inspired by [3], we merge the visual tokens with high similarity, where similarity is quantified by the cosine similarity between token embeddings. As illustrated in Figure 2, given an initial set of visual tokens $\mathbf{v^0}$ at the first LLM layer, we divide adjacent tokens into sets A and B, calculate pairwise similarity scores between the sets, and identify the closest matching token in set B for each token in set A. We then merge the token pairs with the highest similarity scores by averaging their embeddings. This process reduces the number of tokens by half at most. We repeat this merging process iteratively (*e.g.*, twice) to achieve the desired retention ratio.

For videos, we merge the visual tokens within individual frames. Empirical studies suggest that merging tokens within individual frames has minimal impact on final reasoning performance in video tasks, while merging across frames can be harmful. We hypothesize that merging across frames disrupts temporal order of tokens, leading to a loss of crucial temporal information for video understanding.

3.3. Token Pruning within LLM

After merging similar visual tokens, we concatenate the merged visual tokens $\mathbf{v^1}$ with text tokens $\mathbf{t^1}$, forming $\mathbf{x^1} = [\mathbf{v^1}; \mathbf{t^1}]$, which is then passed to the LLM. At each LLM layer, we retain important tokens and prune the less important ones, based on the attention weights computed at each Transformer layer. Specifically, we compute the importance score of each token by applying the Page Rank algorithm [53, 65], using the attention weights as the adjacency matrix. The importance score s_i^l of token x_i^l at layer l is computed as follows:

$$s_i^l = \frac{1}{N^l + M^l} \sum_{j=1}^{N^l + M^l} \mathbf{A}_{i,j}^l \cdot s_j^l \tag{1}$$

where s_j^l is initialized uniformly over all tokens and $\mathbf{A}^l$ represents the softmax-normalized attention weights. Based on these importance scores, we only prune visual tokens and retain the most important visual tokens, leaving text tokens intact. Our rationale is that pruning text tokens substantially degrades performance, likely because multi-modal LLMs rely on text tokens to perform text-centric reasoning. As a result, the number of visual tokens input to the next layer is $N_0 \times r^l$, where r^l is the retention ratio at layer l.

Further, we design a scheduler to control the retention ratio r^l at the l-th layer. It determines the number of visual tokens retained at each LLM layer. Specifically, it is designed as a piece-wise function:

$$r_l = \begin{cases} 1, & \text{if } l < l_1 \\ 1 - k(l - l_1), & \text{if } l_1 \leq l \leq l_2 \\ 0, & \text{if } l > l_2 \end{cases} \tag{2}$$

where l denotes the layer index and $k = \frac{1}{l_2 - l_1}$ represents the slope. In this function, l_1 determines from which layer the token pruning starts, l_2 defines the layer where the visual tokens are pruned at all, and the difference of l_1 and l_2 controls the pruning progress. By adjusting the parameters l_1 and l_2, our approach allows a flexible balance between reasoning accuracy and computational efficiency.

This scheduler design is supported by our empirical findings: pruning visual tokens at the early layers negatively impacts the performance, while in contrast, pruning a large proportion of visual tokens at later layers can still maintain performance. We hypothesize that the LLM prioritizes cross-modal fusion in early layers and shifts focus toward text reasoning in later layers.

3.4. Adaptive Inference

By combining token merging and token pruning, our method enables adaptive inference that can meet diverse computational demands. Specifically, we can vary the retention ratio of token merging and the scheduler parameters (l_1 & l_2) of token pruning, to create a broad range of accuracy-efficiency trade-offs. The specific inference configuration can be determined by the computational requirements in real use cases (*e.g.*, FLOPs and prefill time). We demonstrate adaptive inference in our experiments and

showcase that our method achieves considerable computation reduction while preserving accuracy.

4. Experiments

In this section, we first introduce our implementation details and benchmarks, and then present our results, including the results of video benchmarks, image benchmarks, ablation studies, and adaptive inference.

Implementation Details. Our method is applied during the inference of pre-trained multi-modal LLMs. For video LLMs, we follow the hyperparamers as base model LLaVA-OV-7B [30], such as sampling 32 frames per video unless otherwise noted. It uses Qwen2 [80] as LLM with 28 layers in total. For our method, we set the retention ratio of token merging as 25%, l_1 as 14, and l_2 as 22. For image LLMs, we follow the hyperparamers as base model LLaVA-1.5-7B [38] which adopts Vicuna [6] as LLM with 32 layers in total. For our method, we set the retention ratio of token merging as 12.5%, l_1 as 13, and l_2 as 21. We compute FLOPs and prefill time of LLMs using the library from LLM-Viewer [85], and assume 100 (40) text tokens for video LLMs (image LLMs).

Benchmarks. We evaluate our method on both video and image benchmarks. For video LLMs, we consider the following widely adopted benchmarks. VideoMME [14] is a comprehensive video benchmark with diverse video durations and video domains. MLVU [94] highlights the reasoning for long videos. Egoschema [45] focuses on ego-centric video understanding. MVBench [34] and NextQA [75] focus on temporal action understanding. PercetionTest [55] is a comprehensive video benchmark that evaluates perception and reasoning skills. For image LLMs, we report results on 7 benchmarks. GQA [23] and VQA-v2 [16] are classic VQA datasets that assess visual reasoning ability, with GQA placing a particular emphasis on visual attributes. MME [13] serves as a broad benchmark for evaluating both perception and cognitive abilities. TextVQA [61] specifically measures OCR reasoning skills. SQA-IMG [43] covers questions across topics in natural sciences, language sciences, and social sciences. MMB [42] tests perception and reasoning capabilities, while POPE [35] addresses the problem of object hallucination.

4.1. Video Benchmarks

Table 1 and Table 2 show the results of our method applied to base video LLM and our method accepting more sampled video frames, respectively.

Setup. We choose LLaVA-OV-7B [30] as our base model and follow its evaluation protocol. Specifically, during inference of the pre-trained base model, we apply our token merging to the input visual tokens of Qwen2 LLM and perform token pruning at layers of Qwen2. We measure our

model performance on diverse video benchmarks and report efficiency metrics (FLOPs and prefill time) and accuracy metrics of each video benchmark.

Baselines. We choose baselines as FastV [5] which prunes visual tokens at a particular LLM layer, and LLaVA-1.5-Prumerge [59] which leverages the key-query pairs in the visual encoder to prune and merge visual tokens. We run their official implementation with the pre-trained LLaVA-OV-7B model in the training-free setting.

Accuracy-efficiency balance. As Table 1 shows, our model achieves substantial reductions in computational demands compared to the base model LLaVA-OV-7b (*e.g.*, 14.76 vs. 99.63 FLOPs), decreasing FLOPs by a factor of **6.8** and prefill time by **8.0**, with little or no performance drop across diverse benchmarks. Additionally, compared to the baseline models FastV and LLaVA-Prumerge, our model consistently achieves higher performance on most benchmarks, while only requiring **62.4**% FLOPs and **63.3**% prefill time at most. These results indicate that our method effectively reduces redundancy in visual tokens, providing an optimal balance between accuracy and efficiency.

More sampled frames improve long video understating with close efficiency. In Table 2, our model significantly reduces the FLOPs and prefill time of the base pretrained model, enabling the sampling of more frames in videos (*e.g.*, from 32 to 192 frames). With comparable total computation (*i.e.*, 99.27 vs. 99.63 FLOPs), our model improves performance on long video understanding benchmarks, especially on long video benchmark MLVU, with a notable gain of +**4.6**. This improvement is attributed to our method's ability to retain essential visual tokens with much less redundancy, while densely sampled frames capture additional information crucial for long video comprehension.

4.2. Image Benchmarks

The results are reported in Table 3, comparing our model with the base model and baseline method.

Setup. We choose LLaVA-1.5-7B [38] as our base model following its evaluation protocol. Again, during inference of the pre-trained base model, we merge the input visual tokens of Vicuna LLM and prune visual tokens at Vicuna's layers. We test our model on several image benchmarks and report efficiency metrics (FLOPs and prefill time) and accuracy metrics of each image benchmark.

Baselines. We again choose the baseline as FastV [5] and LLaVA-1.5-Prumerge [59] which provide training-free results on image benchmarks. We also compare with a variant LLaVA-1.5-Prumerge+ which trades computation for reasoning performance. Both our models and baseline models are evaluated in the training-free setting with the pre-trained LLaVA-1.5-7B as the base model.

Model	FLOPs (TB)	Prefill Time (ms)	VideoMME wo / w-subs	MVBench test	MLVU m-avg	EgoSchema test	NextQA mc	PerceptionTest val
Video LLMs								
VILA-40B [37]	-	-	60.1 / 61.1	-	-	58.0	67.9	54.0
PLLaVA-34B [77]	-	-	-	58.1	-	-	-	-
LLaVA-N-Video-32B [91]	-	-	60.2 / 63.0	-	65.5	60.9	77.3	59.4
IXC-2.5-7B [87]	-	-	55.8 / 58.8	69.1	37.3	-	71.0	34.4
LongVA-7B [88]	381.09	2186.04	52.6 / 54.3	-	56.3	-	68.3	-
LLaVA-OV-7B [30]	99.63	439.58	58.2 / 61.5	56.7	64.7	60.1	79.4	57.1
Training-free Method Applied during Inference								
FastV [5]	21.24	79.56	55.9 / 60.0	55.9	61.1	57.5	77.5	56.3
LLaVA-Prumerge [59]	23.65	86.89	57.0 / 59.9	56.5	60.6	**61.0**	77.6	55.8
Ours	**14.76**	**55.03**	**58.2 / 61.3**	**57.1**	**63.7**	59.6	**78.4**	**56.0**

Table 1. Performance on video benchmarks. Compared to the base model LLaVA-OV-7B, our method significantly reduces FLOPs and prefill time by a factor of **6.8** and **8.0**, respectively, with minimum or even without performance loss. Compared to baseline methods, our method outperforms them on most benchmarks with only **62.4**% FLOPs and **63.3**% prefill time.

Model	Number of Frames	FLOPs (TB)	Prefill Time (ms)	VideoMME wo / w-subs	MLVU m-avg	EgoSchema test
Video LLMs						
LLaVA-OV-7B [30]	32	99.63	439.58	58.2 / 61.5	64.7	60.1
Training-free Method Applied during Inference						
Ours	32	**14.76**	**55.03**	58.2 / 61.3	63.7	59.6
Ours	192	99.27	471.20	**59.2 / 62.3**	**69.3**	**60.8**

Table 2. Performance on long video benchmarks with more sampled frames as inputs. With comparable FLOPs and prefill time, our method can accommodate more sampled frames and thus improve the base model LLaVA-OV-7B for long video understanding.

Adaptive inference compared with the base model. As shown in Table 3, when compared to the base model LLaVA-1.5-7B, our model (with 1.00 FLOPs) significantly reduces the FLOPs and prefill time (*i.e.*, using only **12.5**% FLOPS and **30.6**% prefill time), with a noticeable performance drop. However, by adjusting the merging ratio and pruning scheduler, our model (with 2.22 FLOPs) largely mitigates this performance gap while still considerably reducing FLOPs and prefill time (*i.e.*, using only **27.1**% FLOPS and **37.3**% prefill time). This capability is supported by our design of adaptive inference that can accommodate various efficiency demands.

Comparison with baseline methods. In Table 3, with the same level of FLOPs and prefill time, our model consistently outperforms the baseline models on most benchmarks by a clear margin (*e.g.*, +2.3 on VQA-v2, +3.3 on GQA,

+33.2 on MME, +2.3 on MMB, +3.3 on POPE over LLaVA-Prumerge). These results suggest that our method can effectively retain visual tokens critical for image reasoning tasks, while reducing computation costs.

We also notice that our model performs less satisfactorily on TextVQA which includes text-intensive images and requires a model to heavily preserve the textual information. We conjecture that LLaVA-Prumerge leverages the self-attention weights from the visual encoder, while FastV retains tokens that are important to text tokens, thereby being more friendly to text-rich tasks.

4.3. Ablation Study

In this section, we conduct ablation studies for our method, by adjusting the retention ratio of token merging, the parameters of the token pruning scheduler, and the pruning

Model	FLOPs (TB)	Prefill Time (ms)	VQA-v2 (107,394)	GQA (12,578)	MME (2,374)	TextVQA (5,000)	SQA-IMG (2,017)	MMB (4,377)	POPE (8,910)
Image LLMs									
BLIP-2-13B [32]	-	-	41.0	41.0	1293.8	42.5	61.0	-	85.3
InstructBLIP-13B [7]	-	-	-	49.5	1212.8	50.7	63.1	-	78.9
Qwen-VL-Chat-7B [1]	6.44	22.51	78.2	57.5	1487.5	61.5	68.2	60.6	-
LLaVA-1.5-7B [38]	8.18	29.30	78.5	62.0	1510.7	58.2	66.8	73.7	85.9
Training-free Method Applied during Inference									
FastV [5]	2.58	10.34	74.1	56.6	1438.5	**57.3**	68.0	**72.7**	73.6
LLaVA-Prumerge+ [59]	2.41	9.73	74.6	57.4	1391.9	55.2	67.9	71.6	82.2
Ours	2.22	10.92	**75.4**	**58.6**	**1443.5**	53.8	**68.4**	72.5	**85.7**
FastV [5]	1.12	9.56	55.4	45.5	960.4	51.3	66.0	61.5	33.4
LLaVA-Prumerge [59]	1.04	8.99	66.7	51.3	1242.5	**53.8**	**68.0**	67.1	76.2
Ours	1.00	8.98	**69.0**	**54.6**	**1277.7**	48.4	67.1	**69.4**	**79.5**

Table 3. Performance on image benchmarks. VQA-v2 and GQA are the most stable benchmarks that have the most evaluation samples. Compared to the base model LLaVA-1.5-7B, our model significantly reduces FLOPs and prefill time (*e.g.*, by a factor of **3.7** and **2.7**, respectively), with a small performance loss. With the same level of FLOPs and prefill time, our method outperforms the baseline methods by a clear margin on most benchmarks.

Retention Ratio	FLOPs (TB)	Prefill Time (ms)	VideoMME wo-subs
100.0%	99.63	439.58	58.2
50.0%	46.48	182.65	58.5
25.0%	**22.90**	**83.94**	**58.0**
12.5%	11.64	41.22	56.6
6.3%	6.41	22.54	53.6
3.1%	3.85	13.68	52.3
1.6%	2.57	10.15	50.9

Table 4. Ablation study on token merging. We show the performance on the VideoMME benchmark by varying various retention ratios of token merging, with token pruning within LLMs disabled.

strategy. Their results are presented in Table 4, Table 5, and Table 6, respectively.

Setup. We perform experiments on the VideoMME benchmark as it evaluates models on the videos with various durations and diverse domains. Specifically, we vary the retention ratio of token merging (Table 4), scheduler parameters in token pruning (Table 5), and the strategy whether prune text tokens or not (Table 6), respectively. For token merging, we disable token pruning and investigate the effects of altering the retention ratios. For token pruning, we set the retention ratio of token merging as 25% and study the effects of various pruning schedulers or strategies. Following

our main experiments, the metrics include the efficiency aspect (FLOPs and prefill time) and the accuracy aspect.

Token merging substantially reduces redundancy. As shown in Table 4, video reasoning performance remains stable when the retention ratio is set at 25% or higher, while FLOPs and prefill time are significantly reduced (*e.g.*, to only **23**% of FLOPs and **19**% of prefill time relative to the base model). This suggests that most visual tokens are redundant, with only around one-quarter providing essential information for video understanding tasks. Moving forward, as the retention ratio is further reduced, there is a gradual decline in reasoning performance, accompanied by substantial savings in FLOPs and prefill time (*e.g.*, down to **3**% FLOPs and **2**% prefill time of the base model). These findings indicate that with retention ratios below 25%, the model will trade accuracy for efficiency, making it suitable for scenarios requiring high efficiency with less accuracy demands, such as mobile devices.

Accuracy-efficiency trade-offs controlled by token pruning scheduler. As shown in Table 5, by adjusting the pruning scheduler parameters l_1 and l_2, our method enables a broad range of accuracy-efficiency trade-offs. FLOPs decrease from 22.90 to 6.71, and prefill time drops from 83.94 to 28.18, with performance unaffected until FLOPs and prefill time are reduced to approximately half. Furthermore, in each block, when comparing the first row with subsequent rows, we observe a reduction in FLOPs and prefill time with little to no impact on performance. For example, Exp. 6 achieves a 16% reduction in both FLOPs and prefill

Exp.	l_1	l_2	FLOPs (TB)	Prefill Time (ms)	VideoMME wo-subs
1	28	29	22.90	83.94	58.0
2	21	29	20.15	73.61	58.0
3	14	29	17.41	63.34	57.7
4	7	29	14.66	53.08	57.4
5	21	22	17.50	65.35	58.1
6	**14**	**22**	**14.76**	**55.03**	**58.2**
7	7	22	12.01	44.75	56.8
8	14	15	12.10	46.77	54.3
9	7	15	9.36	36.44	52.9
10	7	8	6.71	28.18	41.9

Table 5. Ablation study on token pruning. We show the performance on the VideoMME benchmark by varying the parameters of our pruning scheduler (l_1 & l_2), with token merging set to 25% retention ratio.

Retention Ratio	(l_1,l_2)	Prune Text Tokens	VideoMME
25%	**(14,22)**	×	**58.2**
25%	(14,22)	✓	45.7

Table 6. Ablation study on pruning text tokens within LLM. We show the performance on the VideoMME benchmark, with the only difference as whether our method prunes text tokens or not.

Retention Ratio	l_1	l_2	FLOPs (TB)	Prefill Time (ms)	VideoMME wo-subs
100.0%	-	-	99.63	439.58	58.2
50.0%	-	-	46.48	182.65	58.5
25.0%	**14**	**22**	**14.76**	**55.03**	**58.2**
12.5%	14	22	11.14	39.41	56.4
6.3%	14	22	6.17	21.69	53.6
3.1%	14	22	3.72	13.26	52.3
1.6%	14	22	2.51	10.12	50.9

Table 7. Adaptive inference. By varying the retention ratio of token merging and the parameters of the pruning scheduler (l_1 & l_2), our method supports a broad range of computational conditions without or with a manageable performance drop.

time compared to Exp. 5, while maintaining performance. These findings validate our token pruning design, demonstrating its flexibility in controlling LLM computations and achieving optimal accuracy-efficiency trade-offs.

Early LLM layers vs. later LLM layers. We find that LLM emphasizes multi-modal fusion at early layers and shifts the focus to text-centric reasoning at later layers. For example, experiments 5, 8, and 10 in Table 5 remove all visual tokens after layer 22, layer 15, and layer 8, respectively. Performance is maintained when visual tokens are removed starting from layer 22 (Exp. 5), but it declines when tokens are removed from layer 15 (Exp. 8) and more significantly from layer 8 (Exp. 10). Notably, removing tokens as early as layer 8 causes a substantial performance drop (*e.g.*, from 58.0 to 41.9). Based on these findings, we adopt a strategy (Exp. 16) that retains visual tokens in early layers, gradually prunes them in middle layers, and completely removes them in later layers.

Text tokens matter in LLM layers. In Table 6, involving text tokens in the pruning process results in a substantial performance drop (*e.g.*, from 58.2 to 45.7). This finding aligns with the understanding that LLMs primarily perform text-based reasoning, making it essential to retain text tokens throughout inference.

4.4. Adaptive Inference

As shown in Table 7, by combining token merging and token pruning, our method supports a broad range of accuracy-efficiency trade-offs. For instance, compared to the base model (99.63 FLOPs and 58.2 accuracy), our default configuration achieves the same accuracy with significantly reduced FLOPs (14.76), while an efficient configuration reduces FLOPs even further (2.51) with an acceptable performance decrease (50.9 accuracy). Overall, our approach spans a **40-fold** reduction in FLOPs with less than a 13% drop in accuracy. This flexibility is enabled by adjusting the key parameters, including the retention ratio in token merging and l_1 & l_2 in token pruning. These configurations make our adaptive inference method suitable for various devices and efficiency requirements, such as AR glasses, mobile phones, personal computers, and robots.

5. Conclusion

In this work, we present a training-free approach of adaptive inference for multi-modal LLMs, through token merging based on visual similarity and token pruning based on multi-modal importance. Extensive experiments demonstrate that our method significantly reduces computational demands while maintaining reasoning performance, such as a 6.8/3.7-fold reduction in FLOPs across video/image benchmarks and improved SOTA performance on long video benchmarks. Additionally, our key findings reveal that only a small fraction of visual tokens are necessary for multi-modal understanding, and progressive token pruning across LLM layers further optimizes computational efficiency. We hope that our work will provide a foundation for future advancements in adaptive multi-modal LLMs, in which the accuracy-efficiency tradeoffs can be dynamically adjusted in response to varying computing environments.

References

[1] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023. 7

[2] Emmanuel Bengio, Pierre-Luc Bacon, Joelle Pineau, and Doina Precup. Conditional computation in neural networks for faster models. *arXiv preprint arXiv:1511.06297*, 2015. 3

[3] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your ViT but faster. In *International Conference on Learning Representations*, 2023. 2, 4

[4] Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multi-modal llm's referential dialogue magic. *arXiv preprint arXiv:2306.15195*, 2023. 1

[5] Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. 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*, pages 19–35. Springer, 2025. 3, 5, 6, 7

[6] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhang-hao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yong-hao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, 2023. 1, 5

[7] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *arXiv preprint arXiv:2305.06500*, 2023. 1, 2, 7

[8] Peiyan Dong, Mengshu Sun, Alec Lu, Yanyue Xie, Kenneth Liu, Zhenglun Kong, Xin Meng, Zhengang Li, Xue Lin, Zhenman Fang, et al. HeatViT: Hardware-Efficient Adaptive Token Pruning for Vision Transformers. In *Proceedings of the IEEE International Symposium on High-Performance Computer Architecture*, pages 442–455, 2023. 2

[9] Nan Du, Yanping Huang, Andrew M Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, et al. Glam: Efficient scaling of language models with mixture-of-experts. In *International Conference on Machine Learning*, pages 5547–5569. PMLR, 2022. 3

[10] Gueter Josmy Faure, Jia-Fong Yeh, Min-Hung Chen, Hung-Ting Su, Winston H. Hsu, and Shang-Hong Lai. Hermes: temporal-coherent long-form understanding with episodes and semantics, 2024. 2

[11] Mohsen Fayyaz, Soroush Abbasi Koohpayegani, Farnoush Rezaei Jafari, Sunando Sengupta, Hamid Reza Vaezi Joze, Eric Sommerlade, Hamed Pirsiavash, and Jürgen Gall. Adaptive Token Sampling for Efficient Vision Transformers. In *Proceedings of the European Conference on Computer Vision*, pages 396–414. Springer, 2022. 2

[12] Michael Figurnov, Maxwell D Collins, Yukun Zhu, Li Zhang, Jonathan Huang, Dmitry Vetrov, and Ruslan Salakhutdinov. Spatially adaptive computation time for residual networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1039–1048, 2017. 3

[13] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, and Rongrong Ji. Mme: A comprehensive evaluation benchmark for multimodal large language models, 2024. 5

[14] Chaoyou Fu, Yuhan Dai, Yondong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. *arXiv preprint arXiv:2405.21075*, 2024. 5

[15] Saurabh Goyal, Anamitra Roy Choudhury, Saurabh Raje, Venkatesan Chakaravarthy, Yogish Sabharwal, and Ashish Verma. PoWER-BERT: Accelerating BERT Inference via Progressive Word-Vector Elimination. In *Proceedings of the International Conference on Machine Learning*, pages 3690–3699. PMLR, 2020. 2

[16] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering, 2017. 5

[17] Alex Grubb and Drew Bagnell. Speedboost: Anytime prediction with uniform near-optimality. In *Artificial Intelligence and Statistics*, pages 458–466. PMLR, 2012. 3

[18] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces, 2024. 2

[19] Yizeng Han, Gao Huang, Shiji Song, Le Yang, Honghui Wang, and Yulin Wang. Dynamic neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7436–7456, 2021. 1, 3

[20] Hanzhang Hu, Debadeepta Dey, Martial Hebert, and J Andrew Bagnell. Learning anytime predictions in neural networks via adaptive loss balancing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3812–3821, 2019. 3

[21] Zhiyuan Hu, Yuliang Liu, Jinman Zhao, Suyuchen Wang, Yan Wang, Wei Shen, Qing Gu, Anh Tuan Luu, See-Kiong Ng, Zhiwei Jiang, and Bryan Hooi. Longrecipe: Recipe for efficient long context generalization in large language models, 2024. 2

[22] Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Barun Patra, et al. Language is not all you need: Aligning perception with language models. *Advances in Neural Information Processing Systems*, 36, 2024. 1

[23] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*, 2019. 5

[24] Zequn Jie, Peng Sun, Xin Li, Jiashi Feng, and Wei Liu. Anytime recognition with routing convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 43(6):1875–1886, 2019. 3

[25] Sergey Karayev, Mario Fritz, and Trevor Darrell. Anytime recognition of objects and scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 572–579, 2014. 3

[26] Gyuwan Kim and Kyunghyun Cho. Length-Adaptive Transformer: Train Once with Length Drop, Use Anytime with Search. *arXiv preprint arXiv:2010.07003*, 2020. 2

[27] Sehoon Kim, Sheng Shen, David Thorsley, Amir Gholami, Woosuk Kwon, Joseph Hassoun, and Kurt Keutzer. Learned Token Pruning for Transformers. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 784–794, 2022. 2

[28] Zhenglun Kong, Peiyan Dong, Xiaolong Ma, Xin Meng, Wei Niu, Mengshu Sun, Xuan Shen, Geng Yuan, Bin Ren, Hao Tang, et al. SPViT: Enabling Faster Vision Transformers Latency-Aware Soft Token Pruning. In *Proceedings of the European Conference on Computer Vision*, pages 620–640. Springer, 2022. 2

[29] Bruno Korbar, Yongqin Xian, Alessio Tonioni, Andrew Zisserman, and Federico Tombari. Text-conditioned resampler for long form video understanding, 2024. 2

[30] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. 2, 4, 5, 6

[31] Hengduo Li, Zuxuan Wu, Abhinav Shrivastava, and Larry S Davis. 2d or not 2d? adaptive 3d convolution selection for efficient video recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6155–6164, 2021. 3

[32] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 2, 7

[33] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, Limin Wang, and Yu Qiao. Mvbench: A comprehensive multimodal video understanding benchmark, 2023. 1

[34] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, Limin Wang, and Yu Qiao. Mvbench: A comprehensive multimodal video understanding benchmark, 2024. 5

[35] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models, 2023. 5

[36] Bin Lin, Bin Zhu, Yang Ye, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*, 2023. 1, 2

[37] Ji Lin, Hongxu Yin, Wei Ping, Pavlo Molchanov, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26689–26699, 2024. 6

[38] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2023. 1, 2, 4, 5, 7

[39] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023. 2

[40] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, 2024. 2

[41] Xiangcheng Liu, Tianyi Wu, and Guodong Guo. Adaptive Sparse ViT: Towards Learnable Adaptive Token Pruning by Fully Exploiting Self-Attention. *arXiv preprint arXiv:2209.13802*, 2022. 2

[42] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. Mmbench: Is your multi-modal model an all-around player?, 2024. 5

[43] Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering, 2022. 5

[44] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. 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 (ACL 2024)*, 2024. 1, 2

[45] Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding, 2023. 5

[46] Meat. Llama3.2. https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/, 2024. 2

[47] Lingchen Meng, Hengduo Li, Bor-Chun Chen, Shiyi Lan, Zuxuan Wu, Yu-Gang Jiang, and Ser-Nam Lim. Adavit: Adaptive vision transformers for efficient image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12309–12318, 2022. 3

[48] Yue Meng, Chung-Ching Lin, Rameswar Panda, Prasanna Sattigeri, Leonid Karlinsky, Aude Oliva, Kate Saenko, and Rogerio Feris. Ar-net: Adaptive frame resolution for efficient action recognition. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pages 86–104. Springer, 2020. 3

[49] Thong Thanh Nguyen, Zhiyuan Hu, Xiaobao Wu, Cong-Duy T Nguyen, See-Kiong Ng, and Anh Tuan Luu. Encoding and controlling global semantics for long-form video question answering, 2024. 2

[50] OpenAI. ChatGPT. https://openai.com/blog/chatgpt/, 2023. 2

[51] OpenAI. Gpt-4 technical report, 2023. 2

[52] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744, 2022. 1

[53] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, 1999. 4

[54] Bowen Pan, Rameswar Panda, Yifan Jiang, Zhangyang Wang, Rogerio Feris, and Aude Oliva. Ia-red2: Interpretability-aware redundancy reduction for vision transformers. *Advances in Neural Information Processing Systems*, 34:24898–24911, 2021. 3

[55] Viorica Pătrăucean, Lucas Smaira, Ankush Gupta, Adrià Recasens Continente, Larisa Markeeva, Dylan Banarse, Skanda Koppula, Joseph Heyward, Mateusz Malinowski, Yi Yang, Carl Doersch, Tatiana Matejovicova, Yury Sulsky, Antoine Miech, Alex Frechette, Hanna Klimczak, Raphael Koster, Junlin Zhang, Stephanie Winkler, Yusuf Aytar, Simon Osindero, Dima Damen, Andrew Zisserman, and João Carreira. Perception test: A diagnostic benchmark for multimodal video models. In *Advances in Neural Information Processing Systems*, 2023. 5

[56] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, and Jack Clark. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1

[57] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. DynamicViT: Efficient Vision Transformers with Dynamic Token Sparsification. *Advances in Neural Information Processing Systems*, 34:13937–13949, 2021. 2, 3

[58] Daniel Rotem, Michael Hassid, Jonathan Mamou, and Roy Schwartz. Finding the sweet spot: Analysis and improvement of adaptive inference in low resource settings. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14836–14851, 2023. 3

[59] Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. Llava-prumerge: Adaptive token reduction for efficient large multimodal models. *arXiv preprint arXiv:2403.15388*, 2024. 2, 3, 5, 6, 7

[60] Yuzhang Shang, Bingxin Xu, Weitai Kang, Mu Cai, Yuheng Li, Zehao Wen, Zhen Dong, Kurt Keutzer, Yong Jae Lee, and Yan Yan. Interpolating video-llms: Toward longer-sequence lmms in a training-free manner, 2024. 1

[61] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read, 2019. 5

[62] Quan Tang, Bowen Zhang, Jiajun Liu, Fagui Liu, and Yifan Liu. Dynamic Token Pruning in Plain Vision Transformers for Semantic Segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 777–786, 2023. 2

[63] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 1

[64] Hanrui Wang, Zhekai Zhang, and Song Han. SpAtten: Efficient Sparse Attention Architecture with Cascade Token and Head Pruning. In *Proceedings of the IEEE International Symposium on High-Performance Computer Architecture*, pages 97–110, 2021. 2

[65] Hongjie Wang, Bhishma Dedhia, and Niraj K Jha. Zerotprune: Zero-shot token pruning through leveraging of the attention graph in pre-trained transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16070–16079, 2024. 2, 4

[66] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 2

[67] Xin Wang, Fisher Yu, Zi-Yi Dou, Trevor Darrell, and Joseph E Gonzalez. Skipnet: Learning dynamic routing in convolutional networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 409–424, 2018. 3

[68] Xidong Wang, Dingjie Song, Shunian Chen, Chen Zhang, and Benyou Wang. Longllava: Scaling multi-modal llms to 1000 images efficiently via a hybrid architecture, 2024. 2

[69] Yulin Wang, Rui Huang, Shiji Song, Zeyi Huang, and Gao Huang. Not all images are worth 16x16 words: Dynamic transformers for efficient image recognition. In *Advances in Neural Information Processing Systems*, 2021. 3

[70] Yuxuan Wang, Cihang Xie, Yang Liu, and Zilong Zheng. Videollamb: Long-context video understanding with recurrent memory bridges, 2024. 2

[71] Hongchen Wei and Zhenzhong Chen. Visual context window extension: A new perspective for long video understanding, 2024. 2

[72] Siyuan Wei, Tianzhu Ye, Shen Zhang, Yao Tang, and Jiajun Liang. Joint Token Pruning and Squeezing Towards More Aggressive Compression of Vision Transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2092–2101, 2023. 2

[73] Yuetian Weng, Mingfei Han, Haoyu He, Xiaojun Chang, and Bohan Zhuang. Longvlm: Efficient long video understanding via large language models, 2024. 2

[74] Zuxuan Wu, Tushar Nagarajan, Abhishek Kumar, Steven Rennie, Larry S Davis, Kristen Grauman, and Rogerio Feris. Blockdrop: Dynamic inference paths in residual networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8817–8826, 2018. 3

[75] Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa:next phase of question-answering to explaining temporal actions, 2021. 5

[76] Long Xing, Qidong Huang, Xiaoyi Dong, Jiajie Lu, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, Jiaqi Wang, Feng Wu, et al. Pyramiddrop: Accelerating your large vision-language models via pyramid visual redundancy reduction. *arXiv preprint arXiv:2410.17247*, 2024. 3

[77] Lin Xu, Yilin Zhao, Daquan Zhou, Zhijie Lin, See Kiong Ng, and Jiashi Feng. Pllava: Parameter-free llava extension from images to videos for video dense captioning. *arXiv preprint arXiv:2404.16994*, 2024. 1, 6

[78] Zhixiang Xu, Kilian Q Weinberger, and Olivier Chapelle. The greedy miser: learning under test-time budgets. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pages 1299–1306, 2012. 3

[79] Fuzhao Xue, Yukang Chen, Dacheng Li, Qinghao Hu, Ligeng Zhu, Xiuyu Li, Yunhao Fang, Haotian Tang, Shang

Yang, Zhijian Liu, et al. Longvila: Scaling long-context visual language models for long videos. *arXiv preprint arXiv:2408.10188*, 2024. 2

[80] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024. 1, 5

[81] Deming Ye, Yankai Lin, Yufei Huang, and Maosong Sun. Tr-BERT: Dynamic Token Reduction for Accelerating BERT Inference. *arXiv preprint arXiv:2105.11618*, 2021. 2

[82] Jiabo Ye, Haiyang Xu, Haowei Liu, Anwen Hu, Ming Yan, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. mplug-owl3: Towards long image-sequence understanding in multi-modal large language models, 2024. 2

[83] Weihao Ye, Qiong Wu, Wenhao Lin, and Yiyi Zhou. Fit and prune: Fast and training-free visual token pruning for multi-modal large language models. *arXiv preprint arXiv:2409.10197*, 2024. 3

[84] Hongxu Yin, Arash Vahdat, Jose M. Alvarez, Arun Mallya, Jan Kautz, and Pavlo Molchanov. A-ViT: Adaptive Tokens for Efficient Vision Transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10809–10818, 2022. 2

[85] Zhihang Yuan, Yuzhang Shang, Yang Zhou, Zhen Dong, Chenhao Xue, Bingzhe Wu, Zhikai Li, Qingyi Gu, Yong Jae Lee, Yan Yan, Beidi Chen, Guangyu Sun, and Kurt Keutzer. Llm inference unveiled: Survey and roofline model insights, 2024. 5

[86] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023. 1

[87] Pan Zhang, Xiaoyi Dong, Yuhang Zang, Yuhang Cao, Rui Qian, Lin Chen, Qipeng Guo, Haodong Duan, Bin Wang, Linke Ouyang, et al. Internlm-xcomposer-2.5: A versatile large vision language model supporting long-contextual input and output. *arXiv preprint arXiv:2407.03320*, 2024. 6

[88] Peiyuan Zhang, Kaichen Zhang, Bo Li, Guangtao Zeng, Jingkang Yang, Yuanhan Zhang, Ziyue Wang, Haoran Tan, Chunyuan Li, and Ziwei Liu. Long context transfer from language to vision. *arXiv preprint arXiv:2406.16852*, 2024. 1, 2, 6

[89] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022. 1

[90] Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, et al. Sparsevlm: Visual token sparsification for efficient vision-language model inference. *arXiv preprint arXiv:2410.04417*, 2024. 3

[91] Yuanhan Zhang, Bo Li, haotian Liu, Yong jae Lee, Liangke Gui, Di Fu, Jiashi Feng, Ziwei Liu, and Chunyuan Li. Llava-next: A strong zero-shot video understanding model, 2024. 1, 6

[92] Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Video instruction tuning with synthetic data, 2024. 2

[93] Zeliang Zhang, Phu Pham, Wentian Zhao, Kun Wan, Yu-Jhe Li, Jianing Zhou, Daniel Miranda, Ajinkya Kale, and Chenliang Xu. Treat visual tokens as text? but your mllm only needs fewer efforts to see. *arXiv preprint arXiv:2410.06169*, 2024. 3

[94] Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Shitao Xiao, Xi Yang, Yongping Xiong, Bo Zhang, Tiejun Huang, and Zheng Liu. Mlvu: A comprehensive benchmark for multi-task long video understanding, 2024. 1, 5

[95] Bin Zhu, Bin Lin, Munan Ning, Yang Yan, Jiaxi Cui, HongFa Wang, Yatian Pang, Wenhao Jiang, Junwu Zhang, Zongwei Li, et al. Languagebind: Extending video-language pretraining to n-modality by language-based semantic alignment. *arXiv preprint arXiv:2310.01852*, 2023. 2

[96] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 1, 2