

LLaVA-MINI: EFFICIENT IMAGE AND VIDEO LARGE MULTIMODAL MODELS WITH ONE VISION TOKEN

Anonymous authors

Paper under double-blind review

ABSTRACT

The advent of real-time large multimodal models (LMMs) like GPT-4o has sparked considerable interest in efficient LMMs. LMM frameworks typically encode visual inputs into vision tokens (continuous representations) and integrate them and textual instructions into the context of large language models (LLMs), where large-scale parameters and numerous context tokens (predominantly vision tokens) result in substantial computational overhead. Previous efforts towards efficient LMMs always focus on replacing the LLM backbone with smaller models, while neglecting the crucial issue of token quantity. In this paper, we introduce LLaVA-Mini, an efficient LMM with minimal vision tokens. To achieve a high compression ratio of vision tokens while preserving visual information, we first analyze how LMMs understand vision tokens and find that most vision tokens only play a crucial role in the early layers, where they fuse visual information into text tokens. Building on this finding, LLaVA-Mini introduces modality pre-fusion to fuse visual information into text tokens in advance, thereby facilitating the extreme compression of vision tokens fed to LLM backbone into one token. LLaVA-Mini can support the understanding of images, high-resolution images, and videos in an efficient manner. Experiments across 11 image-based and 7 video-based benchmarks demonstrate that LLaVA-Mini outperforms LLaVA-v1.5 with just 1 vision token instead of 576. Efficiency analyses reveal that LLaVA-Mini can reduce FLOPs by 77%, deliver low-latency responses within 40 milliseconds, and process over 10,000 frames of video on GPU hardware with 24GB of memory.¹

1 INTRODUCTION

Large multimodal models (LMMs), such as GPT-4o (OpenAI, 2024a), equip large language models (LLMs) (OpenAI, 2022; 2023) with the ability to understand visual information, exhibiting a common trend toward low-latency responses to enable real-time multimodal interactions. Recently, the most widely adopted LMMs (Liu et al., 2023b; 2024a; Zhu et al., 2024), exemplified by the LLaVA series (Liu et al., 2023b), involves embedding image patches into vision tokens through a vision encoder (Radford et al., 2021) and incorporating them into the LLM’s context to facilitate visual information comprehension, leading to strong performance in image and video understanding.

However, the substantial computational costs of LMMs present ongoing challenges. Unlike LLMs (Touvron et al., 2023a;b; Dubey et al., 2024), which only process textual inputs, LMMs must incorporate a large number of additional vision tokens into the LLM’s context to represent visual information (Liu et al., 2023b), significantly increasing computational complexity. For instance, in the widely used vision encoder CLIP ViT-L/336px, a single image is encoded into $24 \times 24 = 576$ vision tokens (Radford et al., 2021), where integrating such a large number of vision tokens into the context of parameter-heavy LLM results in significant computational overhead and higher inference latency. This issue becomes even more pronounced in high-resolution image modeling (which requires more vision tokens per image) (Liu et al., 2024b; Gen Luo, 2024) or video processing (which involves processing more images) (Maaz et al., 2024; Lin et al., 2023a). Therefore, developing efficient LLMs is essential for achieving GPT-4o-like low-latency multimodal interactions.

The computational demands of LMMs are primarily driven by model scale and the number of tokens in the input context. Existing approaches to improving LMM efficiency typically focus on model

¹Code is provided in supplementary materials. The model will be released after being de-anonymized.

downsizing (Chu et al., 2023; 2024; Yuan et al., 2024a; Zhou et al., 2024a) or quantization techniques (Yuan et al., 2024b), but often overlook another critical avenue: reducing the number of vision tokens to shorten the input context. Some token reduction methods rely on predefined rules to reduce the number of tokens output by the vision encoder (Bolya et al., 2023; Shang et al., 2024; Li et al., 2024e; Ye et al., 2024c; Hu et al., 2024), which leads to the loss of visual information and inevitably results in performance degradation (Wang et al., 2024; Fan et al., 2024).

In this paper, we aim to develop efficient LMMs by minimizing the number of vision tokens while maintaining comparable performance. To this end, we begin by exploring a foundational question: *How does the LMM (particularly the LLaVA architecture) understand vision tokens?* Through layer-wise analysis (refer to Sec.3), we observe that the importance of vision tokens changes across different layers of LLM. In the early layers, vision tokens play a crucial role, receiving considerable attention from the following text tokens (e.g., user input instructions and responses). However, as the layers deepen, the attention devoted to vision tokens decreases sharply, with most attention shifting towards the input instructions. Notably, even when we entirely remove vision tokens in some later layers, LMM keeps certain visual understanding capabilities. This finding suggests that vision tokens are more critical in early layers, where text tokens fuse visual information from vision tokens.

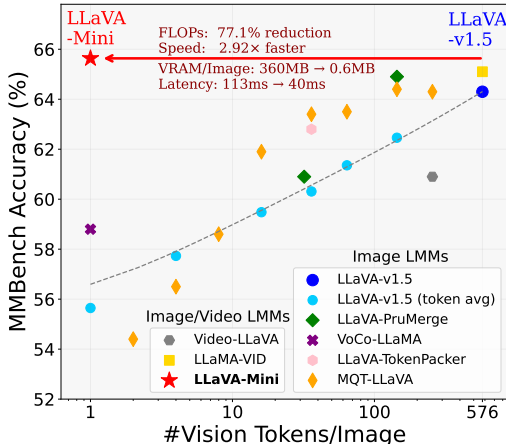


Figure 1: LLaVA-Mini achieves comparable performance to LLaVA-v1.5 using only 1 vision token instead of 576, yielding efficient computation, lower latency, and reduced VRAM usage.

Based on this finding, if the fusion process can be shifted from the early layers of LLM to perform before LLM, we can significantly reduce the number of vision tokens fed into the LLM without sacrificing performance. Along with this idea, we propose *LLaVA-Mini*, an efficient and high-quality LMM with minimal vision tokens. LLaVA-Mini introduces a modality pre-fusion module before LLM to fuse visual information into the instruction text in advance, and employs a compression module to highly compress the vision tokens before inputting them into LLM, thereby enhancing efficiency while preserving high-quality visual understanding. Under extreme settings, LLaVA-Mini requires only **one vision token per image fed into LLM backbone**, offering significant advantages in inference time and memory consumption for high-resolution image and long video processing.

Experiments across a wide range of 11 image-based and 7 video-based understanding benchmarks show that LLaVA-Mini achieves performance comparable to LLaVA-v1.5 (Liu et al., 2023b) while using only 1 vision token instead of 576 (compression rate of 0.17%). With minimal vision tokens, LLaVA-Mini offers substantial benefits in terms of computational efficiency (77% FLOPs reduction) and lowering GPU memory usage (360 MB \rightarrow 0.6 MB per image), as shown in Figure 1. As a result, LLaVA-Mini decreases inference latency of image understanding from 100 ms to 40 ms and also enables the processing of long videos exceeding 10,000 frames (over 3 hours) on an NVIDIA RTX 3090 with 24GB of memory, paving the way for low-latency multimodal interactions.

2 RELATED WORK

Large multimodal models (LMMs) (OpenAI, 2024b; Liu et al., 2023b; 2024a; Zhu et al., 2024) use vision encoder to convert the image into vision tokens, which are then processed by large language models (LLMs) (OpenAI, 2023; Chiang et al., 2023; Touvron et al., 2023a;b) to facilitate visual understanding. As LMMs are increasingly deployed in real-time applications (OpenAI, 2024a), enhancing their efficiency has become a critical concern. Recent efforts focus on either reducing the model size or the number of tokens that fed into LMM.

To reduce LMM’s model size, previous methods directly replace the LLM backbone with a smaller one (Chu et al., 2023; 2024; Yuan et al., 2024a; Zhou et al., 2024a), thereby lowering the total parameters. Quantization techniques (Yuan et al., 2024b) can also be applied to improve LMM

108 efficiency. However, reducing the scale and precision of parameters can impact the LLM backbone’s
 109 capabilities, resulting in performance declines in visual tasks (Shang et al., 2024).

110 Another efficiency determinant for LMMs is the context length provided to the LLM backbone,
 111 which includes both vision and text tokens. In practice, the number of vision tokens can be sub-
 112 stantial, particularly when processing high-resolution images and videos. For image-based LMMs,
 113 token merging (Bolya et al., 2023), PruMerge (Shang et al., 2024), and TokenPacker (Li et al.,
 114 2024e) aggregate vision tokens based on similarity, achieving compression rates of 20% to 50%.
 115 Qwen-VL (Bai et al., 2023) and MQT-LLaVA (Hu et al., 2024) utilize Q-former (Li et al., 2023a)
 116 or resampler to compress vision tokens into a fixed length. However, these methods directly reduce
 117 vision tokens, and inevitably result in the loss of visual information (Fan et al., 2024).

118 For video-based LMMs, Video-ChatGPT (Maaz et al., 2024), VideoChat (Li et al., 2024c), Video-
 119 LLaVA (Lin et al., 2023a), and Video-LLaMA (Zhang et al., 2023), select a fixed number of frames
 120 from videos of varying lengths. MovieChat (Song et al., 2024a) applies memory techniques to
 121 condense videos into a fixed-length representation. VideoLLM-online (Chen et al., 2024) process
 122 long video with extracting 11 token per frame. Such frame selection or merging methods may lose
 123 some key frames or misunderstand the temporal information of the video (Zhou et al., 2024b).

124 Previous methods have primarily focused on token reduction on the vision encoder. LLaVA-Mini
 125 takes this a step further by exploring how vision tokens and text tokens interact within the LLM
 126 backbone, and accordingly introduces a modality pre-fusion module, enabling an extreme compres-
 127 sion of vision tokens (1 vision token fed into LLM) while achieving comparable performance.

129 3 HOW DOES LLaVA UNDERSTAND VISION TOKENS?

130 To compress visual tokens while preserving visual understanding, we sought to figure out how
 131 LMMs understand visual tokens. Given the complexity of this issue, our preliminary analysis
 132 concentrated on the LLaVA architecture (Liu et al., 2023b), focusing on the role of visual tokens
 133 (particularly their quantity) in LMMs from an attention-based perspective (Xiao et al., 2024).

136 3.1 LLaVA ARCHITECTURE

137 LLaVA (Large Language and Vision Assistant) (Liu et al., 2023b) is an advanced multimodal archi-
 138 tecture that integrates vision and language processing capabilities. Building upon vision Trans-
 139 formers (ViT) (Dosovitskiy et al., 2021) for visual inputs and LLMs for text, LLaVA can generate
 140 language response \mathbf{X}^a based on the given language instruction \mathbf{X}^q and visual inputs \mathbf{X}^v .

141 Typically, a pre-trained CLIP ViT-L/14 (Radford et al., 2021) and a projection layer are employed
 142 to encode the visual inputs \mathbf{X}^v into vision tokens (i.e., continuous representations) \mathbf{H}^v . Then, vision
 143 tokens \mathbf{H}^v and language instruction’s embedding \mathbf{H}^q are fed into an LLM, such as Vicuna (Chiang
 144 et al., 2023) or Mistral, to generate the response \mathbf{X}^a . In practice, vision tokens are often inserted
 145 into the middle of the language instruction, so the inputs of LLM can be formally represented as:

$$147 \left\langle H_1^q, \dots, H_k^q, H_1^v, \dots, H_{l_v}^v, H_{k+1}^q, \dots, H_{l_q}^q \right\rangle, \quad (1)$$

148 where l_v and l_q denote the lengths of the vision tokens and language instruction, respectively. For
 149 instance, in LLaVA-v1.5, the system prompts are positioned before the image (i.e., H_1^q, \dots, H_k^q),
 150 while the user inputs follow the image (i.e., $H_{k+1}^q, \dots, H_{l_q}^q$) (Liu et al., 2023b).

153 3.2 PRELIMINARY ANALYSES

154 We begin by analyzing the significance of visual tokens in LMMs to guide the strategies for com-
 155 pressing vision tokens. Specifically, we evaluate the importance of visual tokens at each layer of
 156 LMMs from an attention-based perspective. Our analysis encompasses several LMMs, including
 157 LLaVA-v1.5-Vicuna-7B, LLaVA-v1.5-Vicuna-13B, LLaVA-v1.6-Mistral-7B, and LLaVA-NeXT-
 158 Vicuna-7B (Liu et al., 2023b; 2024b), to identify common characteristics across models of varying
 159 sizes and training datasets. **Appendix A gives the formal expression of the preliminary analyses.**

160 **Vision Tokens are More Important in Early Layers** To find out which layers in LMM the vision
 161 tokens play a more important role, we measure the attention weights assigned to different token

162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215

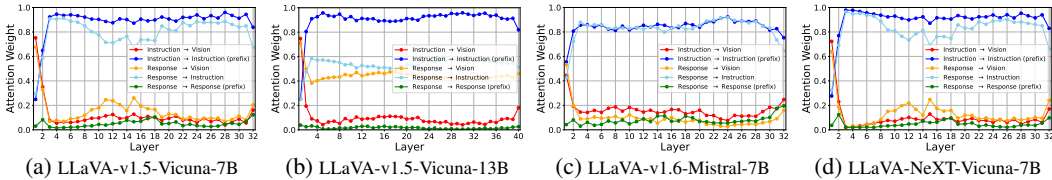


Figure 2: Layer-wise variation of attention weights assigned to different types of tokens (including instruction, vision, and response) in LLMs. “A→B” means the attention weights from A to B.

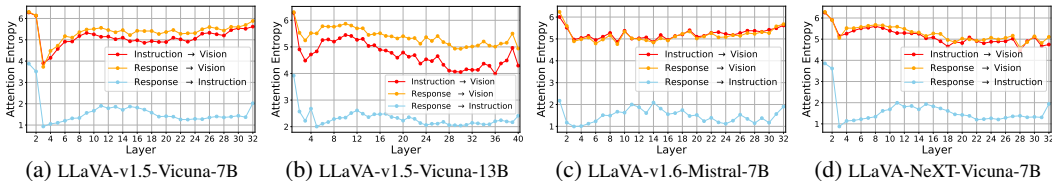


Figure 3: Attention entropy assigned to different types of tokens across different layers in LLMs.

types (including instruction, vision, and response) at each layer. As shown in Figure 2, the attention assigned to vision tokens varies significantly across layers. Visual tokens receive more attention in the earlier layers, but this attention sharply decreases in the deeper layers, with over 80% of the attention being directed towards instruction tokens. This change in attention suggests that vision tokens play a central role in the early layers, with the instruction tokens seeking relevant visual information from vision tokens through attention mechanisms. In the later layers, the model relies more on instructions that have already fused the visual data to generate responses.

Most Vision Tokens are Focused in Early Layers To further assess the importance of individual visual tokens, we calculate the entropy of the attention distribution at each layer. As shown in Figure 3, we find that the entropy of attention toward visual tokens is much higher in the earlier layers, indicating that most visual tokens are evenly attended to in the early layers.

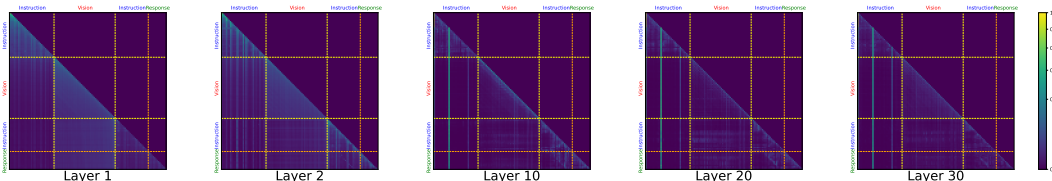


Figure 4: Attention visualization at different layers in LLaVA-v1.5 (color bar: logarithmic scale).

To intuitively illustrate the layer-wise variations in the importance of visual tokens, Figure 4 visualizes the attention distribution across each layer of LLaVA-v1.5. Almost all visual tokens receive broader attention in the early layers, while only some visual tokens are focused in the later layers. These observations suggest that all visual tokens are crucial in the early layers, and reducing their quantity inevitably results in a loss of visual information. This explains why previous methods of direct token reduction will compromise visual understanding capabilities (Shang et al., 2024; Ye et al., 2024c; Hu et al., 2024).

To further substantiate our finding that visual tokens are particularly critical in the early layers, we evaluated the visual understanding ability of LLMs when visual tokens were dropped at different layers. Specifically, we measured the performance of LLaVA-v1.5 on the GQA (Hudson & Manning, 2019) and MMBench (Liu et al., 2024c), with visual tokens being dropped at layers 1-4, 5-8, ..., 29-32, respectively. As shown in Figure 5, removing visual tokens in the early layers leads to a complete loss of visual understanding ability, while removing tokens in the higher layers has a minimal effect, with the model retaining much of its original performance. In conclusion, our analyses and ablation study reveal that vision tokens play a crucial role in the early layers of LLaVA, where text tokens fuse visual information from the vision tokens at this stage. This insight can inform our strategies for compressing vision tokens.

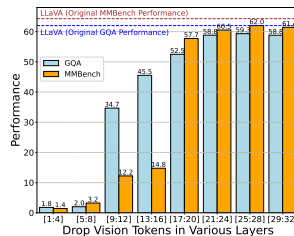


Figure 5: Performance of LLaVA-v1.5 when removing all vision tokens in various layers of LMM.

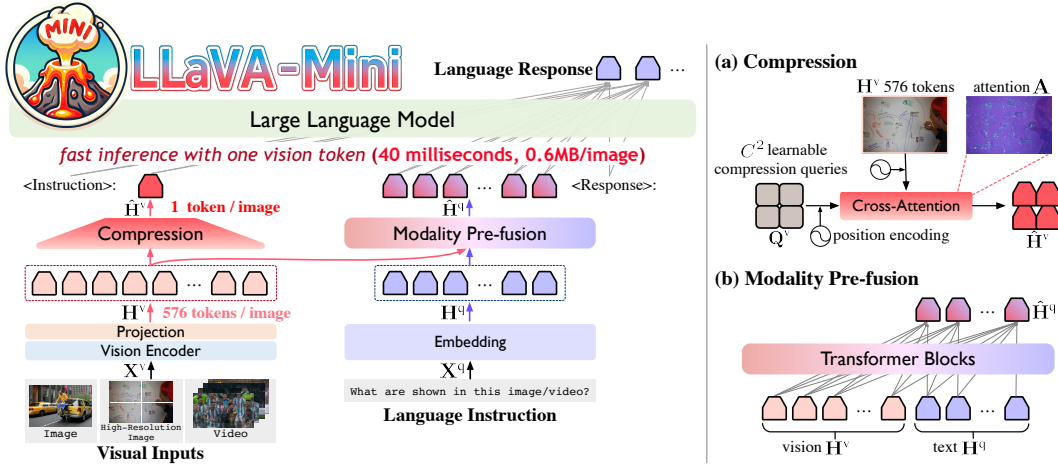


Figure 6: Architecture of LLaVA-Mini. **Left:** LLaVA-Mini represents each image with one vision token. **Right:** Detailed view of the proposed query-based compression and modality pre-fusion.

4 LLaVA-MINI

We introduce LLaVA-Mini, an efficient LMM with minimal vision tokens. Like previous work, LLaVA-Mini uses a vision encoder to encode an image into several vision tokens. To enhance the efficiency of LMMs, LLaVA-Mini significantly reduces the number of vision tokens fed into LLM backbone through a compression module. To retain visual information during compression, based on previous findings that vision tokens play a crucial role in the early layers for fusing visual information, LLaVA-Mini introduces a modality pre-fusion module before the LLM backbone to fuse the visual information into the text tokens. The detail of LLaVA-Mini is as follows.

4.1 ARCHITECTURE

The architecture of LLaVA-Mini is illustrated in Figure 6. For the visual inputs \mathbf{X}^v , a pre-trained CLIP vision encoder (Radford et al., 2021) is employed to extract visual features from each image. These features are then mapped into the word embedding space via a projection layer, producing vision tokens $\mathbf{H}^v \in \mathbb{R}^{N^2 \times d_h}$, where N^2 is the number of vision tokens and d_h is the LLM’s embedding dimension. For the language instruction \mathbf{X}^q , LLM’s embedding layer is used to generate text token representations $\mathbf{H}^q \in \mathbb{R}^{l_q \times d_h}$, where l_q is the number of text tokens.

Vision Token Compression To enhance the efficiency of LMMs, LLaVA-Mini reduces the number of vision tokens fed into the LLM backbone by utilizing a query-based compression module. To learn compression of the vision tokens, LLaVA-Mini introduces $C \times C$ learnable compression queries \mathbf{Q}^v . These queries interact with all vision tokens \mathbf{H}^v through cross-attention (Li et al., 2023a), selectively extracting the important visual information to produce $C \times C$ compressed vision tokens $\hat{\mathbf{H}}^v \in \mathbb{R}^{C^2 \times d_h}$. To preserve the spatial information in the image during compression, we introduce a 2D sinusoidal positional encoding $PE(\cdot)$ (He et al., 2021) on the learnable queries and original vision tokens. Formally, the compression can be expressed as:

$$\hat{\mathbf{H}}^v = \mathbf{A} \cdot \mathbf{H}^v, \text{ where } \mathbf{A} = \text{Softmax} \left(\left(\mathbf{Q}^v + PE(\mathbf{Q}^v) \right) \cdot \left(\mathbf{H}^v + PE(\mathbf{H}^v) \right)^\top \right), \quad (2)$$

where $\mathbf{A} \in \mathbb{R}^{C^2 \times N^2}$ is the similarity and $\hat{\mathbf{H}}^v$ are $C \times C$ compressed vision tokens.

Modality Pre-fusion The compression of vision tokens inevitably results in some loss of visual information. To retain as much visual information during compression as possible, LLaVA-Mini introduces a modality pre-fusion before the LLM backbone, enabling text tokens to fuse relevant visual information from all vision tokens in advance. Based on our previous observations, where this fusion stage occurs implicitly within the early layers of the LLM, the modality pre-fusion module $f(\cdot)$ consists of N_{fusion} Transformer blocks (Vaswani et al., 2017), where each Transformer block share the same structure and hyperparameters with LLM backbone. Vision tokens \mathbf{H}^v and text

tokens \mathbf{H}^q are concatenated and fed into the pre-fusion module, and the outputs corresponding to the text tokens are then extracted as fusion tokens, expressed as:

$$\hat{\mathbf{H}}^q = f(\text{Concat}(\mathbf{H}^v, \mathbf{H}^q))[-l_q:], \quad (3)$$

where $\hat{\mathbf{H}}^q \in \mathbb{R}^{l_q \times d_h}$ are fusion tokens of text representations with related visual information.

Finally, the compressed vision tokens $\hat{\mathbf{H}}^v$ and fusion tokens $\hat{\mathbf{H}}^q$ of text representations with related visual information ($C^2 + l_q$ tokens in total) are fed to LLM together to generate the response.

4.2 HIGH-RESOLUTION IMAGE AND VIDEO MODELING

LLaVA-Mini uses minimal vision tokens to represent visual information, making it possible to handle high-resolution images and videos much more efficiently.

High-Resolution Image The resolution of LMM is typically determined by the vision encoder, such as CLIP’s ViT-L, which encodes at a resolution of 336*336 pixels. To perceive images at a higher resolution, we divide each image into four sub-images by splitting it horizontally and vertically into two parts (Liu et al., 2024b). Each of these sub-images is processed by the vision encoder and projection individually, yielding $N^2 \times 4$ vision tokens with a high resolution of 672*672 pixels. The proposed compression module is then employed to reduce these $N^2 \times 4$ vision tokens into C^2 compressed vision tokens $\hat{\mathbf{H}}^v$. The modality pre-fusion module takes the 4 sub-images ($N^2 \times 4$ vision tokens), the original image (N^2 vision tokens), and the language instruction (l_q text tokens) as inputs, and then generates l_q fusion tokens $\hat{\mathbf{H}}^q$ with richer global and local visual information. Finally, the number of tokens input to the LLM is $C^2 + l_q$. Note that when handling high-resolution images, C is set slightly higher than in standard-resolution settings to preserve more details.

Video When handling videos, LMMs often extract multiple frames from the video (Li et al., 2023b), which incurs significant computational costs. For instance, in the case of LLaVA-v1.5, extracting frames at a rate of 1 frame per second (fps) from an 8-second video results in $576 \times 8 = 4608$ vision tokens, leading to substantial VRAM usage. LLaVA-Mini can represent each image with minimal vision tokens, providing a significant advantage in processing long videos. For a video consisting of M frames, LLaVA-Mini processes each frame individually, generating C^2 vision tokens and l_q fusion tokens per frame. C^2 vision tokens from each of M frames are sequentially concatenated to yield a total of $M \times C^2$ vision tokens, i.e., $\hat{\mathbf{H}}^v$. Then, l_q fusion tokens corresponding to M frames are aggregated through pooling operation to generate the video’s fusion tokens $\hat{\mathbf{H}}^q$. As a result, the number of tokens fed to the LLM is reduced from $MN^2 + l_q$ to $MC^2 + l_q$ for a video of M frames.

4.3 TRAINING

LLaVA-Mini follows the same training process as LLaVA, consisting of two stages.

Stage 1: Vision-Language Pretraining In this stage, compression and modality pre-fusion modules are not yet applied (i.e., the N^2 vision tokens remain unchanged). LLaVA-Mini learns to align vision and language representations using visual caption data. The training focuses solely on the projection module while the vision encoder and LLM remain frozen (Liu et al., 2023b).

Stage 2: Instruction Tuning In this stage, LLaVA-Mini is trained to perform various visual tasks based on minimal vision tokens, using instruction data. Compression and modality pre-fusion are introduced to LLaVA-Mini, and all modules except the frozen vision encoder (i.e., the projection, compression, modality pre-fusion, and LLM backbone) are trained in an end-to-end manner.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETTING

Benchmarks We evaluate LLaVA-Mini on image and video understanding tasks. Experiments are conducted on 11 image benchmarks and 7 video benchmarks. Refer to Appendix C for details.

Baselines LLaVA-Mini is an image/video LMM, so we compare it with several advanced image-based and video-based LMMs. Detailed description of baselines refer to Appendix D.

Table 1: Performance on 11 image-based benchmarks. ‘Res.’ is resolution and ‘#Vision Tokens’ is the number of vision tokens fed to LLM backbone. ‘*’ indicates that involving extra training data.

Methods	LLM	Res.	#Vision Tokens	VQA ^{v2}	GQA	VisWiz	SciQA	VQA ^T	POPE	MME	MMB	SEED	LLaVA ^v	MM-Vet	Avg. (%)
BLIP-2	Vicuna-13B	224	32	65.0	41.0	19.6	61.0	42.5	85.3	1293.8	-	46.4	38.1	22.4	-
InstructBLIP	Vicuna-7B	224	32	-	49.2	34.5	60.5	50.1	-	-	36.0	53.4	60.9	26.2	-
IDEFICS-9B	LLaMA-7B	224	64	50.9	38.4	35.5	-	25.9	-	-	48.2	-	-	-	-
IDEFICS-80B	LLaMA-65B	224	64	60.0	45.2	36.0	-	30.9	-	-	54.5	-	-	-	-
Qwen-VL	Qwen-7B	448	256	78.8	59.3	35.2	67.1	63.8	-	-	38.2	56.3	-	-	-
Qwen-VL-Chat	Qwen-7B	448	256	78.2	57.5	38.9	68.2	61.5	-	1487.5	60.6	58.2	-	-	-
SPHINX	LLaMA-13B	224	289	78.1	62.6	39.9	69.3	51.6	80.7	1476.1	66.9	56.2	73.5	36.0	56.0
SPHINX-2k	LLaMA-13B	762	2890	80.7	63.1	44.9	70.6	61.2	87.2	1470.6	65.9	57.9	76.9	40.2	59.0
mPLUG-Owl2	LLaMA-7B	448	1024	79.4	56.1	54.5	68.7	54.3	-	1450.2	64.5	57.8	-	36.2	-
Video-LLaVA	Vicuna-7B	224	256	74.7	60.3	48.1	66.4	51.8	84.4	-	60.9	-	73.1	32.0	-
LLaVA-v1.5	Vicuna-7B	336	576	78.5	62.0	50.0	66.8	58.2	85.9	1510.7	64.3	58.6	63.4	30.5	56.3
<i>LMMs with fewer vision tokens</i>															
MQT-LLaVA	Vicuna-7B	336	2	61.0	50.8	48.5	65.0	-	74.5	1144.0	54.4	-	41.7	19.5	-
MQT-LLaVA	Vicuna-7B	336	36	73.7	58.8	51.0	66.8	-	81.9	1416.3	63.4	-	59.6	27.8	-
MQT-LLaVA	Vicuna-7B	336	256	76.8	61.6	53.1	67.6	-	84.4	1434.5	64.3	-	64.6	29.8	-
PruMerge	Vicuna-7B	336	32	72.0	-	-	68.5	56.0	76.3	1350.3	60.9	-	-	-	-
PruMerge++	Vicuna-7B	336	144	76.8	-	-	68.3	57.1	84.0	1462.4	64.9	-	-	-	-
LLaMA-VID	Vicuna-7B	336	2	-	55.5	-	68.8	49.0	83.1	-	-	-	-	-	-
VoCo-LLaMA	Vicuna-7B	336	1	72.3	57.0	-	65.4	-	81.4	1323.3	58.8	53.7	-	-	-
TokenPacker	Vicuna-7B	336	36	75.0	59.6	50.2	-	-	86.2	-	62.8	-	-	29.6	-
<i>Ours</i>															
LLaVA-Mini	Vicuna-7B	336	1	77.6	60.9	56.2	70.4	57.0	84.4	1466.0	65.6	58.5	68.9	36.6	57.9
	Δ compare to LLaVA-v1.5			0.17%	-0.9	-1.1	+6.1	+3.6	-1.3	-1.5	-44.7	+1.3	-0.1	+5.5	+6.1
LLaVA-Mini-HD	Vicuna-7B	672	64	78.9	61.8	58.5	69.7	59.1	85.3	1476.8	67.5	60.2	69.3	33.9	58.6
	Δ compare to LLaVA-v1.5			11.1%	+0.4	-0.2	+8.5	+2.9	+0.9	-0.6	-33.9	+3.2	+1.6	+5.9	+3.4
LLaVA-Mini* (Image & Video)	LLaMA-3.1-8B-Instruct	336	1	79.0	61.3	57.4	83.1	58.5	85.3	1522.7	71.6	63.0	70.2	37.2	60.7

Table 2: Performance on video-based open-ended generative benchmarks. We report accuracy (%) for question-answer, and scores (1-5, higher is better) for question-answer and generative performance. Results marked with **bold** and underline indicate the best and second best, respectively.

Methods	#Frames	#Vision Tokens per Frame	Video-based Question-Answer						Video-based Generative Performance						
			MSVD-QA		MSRVTT-QA		ActivityNet-QA		Correctness	Detail	Contextual	Temporal	Consistency	Avg.	
			Acc.	Score	Acc.	Score	Acc.	Score							
LLaMA Adapter	5	256	54.9	3.1	43.8	2.7	34.2	2.7	2.03	2.32	2.30	1.98	2.15	2.19	
VideoChat	16	32	56.3	2.8	45.0	2.5	26.5	2.2	2.23	2.50	2.53	1.94	2.24	2.30	
Video-LLaMA	16	64	51.6	2.5	29.6	1.8	12.4	1.1	1.96	2.18	2.16	1.82	1.79	1.99	
Video-ChatGPT	100	~3.6	64.9	3.3	49.3	2.8	35.2	2.7	2.40	2.52	2.62	1.98	2.37	2.37	
BT-Adapter	100	~2.6	67.5	3.7	57.0	3.2	45.7	3.2	2.68	2.69	3.27	2.34	2.46	2.69	
MovieChat	2048	32	75.2	3.8	52.7	2.6	45.7	<u>3.4</u>	2.76	2.93	3.01	2.24	2.42	2.65	
LLaMA-VID	1fps	2	69.7	3.7	57.7	3.2	<u>47.4</u>	3.3	<u>2.96</u>	3.00	<u>3.53</u>	<u>2.46</u>	<u>2.51</u>	<u>2.88</u>	
Video-LLaVA	8	256	70.7	<u>3.9</u>	<u>59.2</u>	<u>3.5</u>	45.3	3.3	2.87	2.94	3.44	2.45	<u>2.51</u>	2.84	
LLaVA-Mini	1fps	1	<u>70.9</u>	4.0	59.5	3.6	53.5	3.5	2.97	<u>2.99</u>	3.61	2.48	2.67	2.94	

Configuration For a fair comparison, LLaVA-Mini employs the same configurations as LLaVA-v1.5 (Liu et al., 2023b), using the CLIP ViT-L/336px (Radford et al., 2021) as the vision encoder and Vicuna-v1.5-7B (Chiang et al., 2023) as the LLM backbone. The compressed hyperparameter C is set to 1, meaning vision tokens are compressed to one token. The number of modality pre-fusion layers N_{fusion} is set to 4. LLaVA-Mini uses the same training data as LLaVA-v1.5 (Liu et al., 2023b), using 558K caption data for pretraining and 665K instruction data for instruction tuning. The high-resolution version with 672*672 pixels (refer to Sec.4.2) is denoted as LLaVA-Mini-HD. To capture more visual details, the compressed hyperparameter C of LLaVA-Mini-HD is set to 8, i.e., compressing to 64 vision tokens. For video processing, LLaVA-Mini extracts 1 frame per second (1 fps) from the video and sets $C = 1$ to represent each frame with one vision token.

To further explore the potential of LLaVA-Mini, we introduce a variant that uses the CLIP ViT-L/336px (Radford et al., 2021) as vision encoder and the advanced LLaMA-3.1-8B-Instruct (Dubey et al., 2024) as LLM backbone. During instruction tuning, we combine 665K image instruction data from LLaVA (Liu et al., 2023b), 100K video instruction data from Video-ChatGPT (Maaz et al., 2024), and part of open-source data (Li et al., 2024a), resulting in 3 million training samples. LLaVA-Mini is trained using 8 NVIDIA A800 GPUs. Training details are provided in Appendix B.

5.2 MAIN RESULTS

Image-based Evaluation We compare LLaVA-Mini with LLaVA-v1.5 across 11 benchmarks to thoroughly assess the performance of LLaVA-Mini with minimal vision tokens. The results are

Table 3: Performance on MVBench (accuracy). Detailed scores are reported in Appendix I.

Methods	Action	Object	Position	Scene	Count	Attribute	Pose	Character	Cognition	Avg.
mPLUG-Owl	28.4	33.0	25.0	29.0	29.3	42.0	24.0	31.0	25.3	29.7
Video-ChatGPT	32.1	40.7	21.5	31.0	28.0	44.0	29.0	33.0	30.3	32.7
Video-LLaMA	34.4	42.2	22.5	43.0	28.3	39.0	32.5	40.0	29.3	34.1
VideoChat	38.0	41.2	26.3	48.5	27.8	44.3	26.5	41.0	27.7	35.5
LLaMA-VID	43.4	36.7	39.8	22.0	<u>36.5</u>	37.3	37.5	34.0	60.5	41.4
Video-LLaVA	48.0	46.5	27.8	<u>84.5</u>	35.5	45.8	<u>34.0</u>	<u>42.5</u>	34.2	<u>43.1</u>
LLaVA-Mini	52.1	<u>43.2</u>	<u>31.8</u>	85.5	37.5	<u>44.5</u>	29.5	52.0	<u>35.0</u>	44.5

Table 4: Results on MLVU (accuracy) of long video understanding. Evaluation includes Topic Reasoning (TR), Anomaly Recognition (AR), Needle QA (NQA), Ego Reasoning (ER), Plot QA (PQA), Action Order (AO), and Action Count (AC).

Methods	#Frames	Holistic		Single Detail			Multi Detail		Avg.
		TR	AR	NQA	ER	PQA	AO	AC	
Avg. Video Duration (minute)		7	10	14	10	8	16	13	11
Max Video Duration (minute)		20	543	139	20	13	137	130	143
Video-ChatGPT	100	26.9	24.0	40.3	42.0	29.9	25.1	<u>31.1</u>	31.3
MovieChat	2048	29.5	25.0	24.2	24.7	25.8	28.6	22.8	25.8
Movie-LLM	1fps	30.0	29.0	29.6	24.7	24.1	20.5	24.8	26.1
TimeChat	96	23.1	27.0	24.5	28.4	25.8	24.7	32.0	30.9
LLaMA-VID	1fps	50.8	34.5	30.1	32.7	32.5	23.9	27.8	33.2
MA-LMM	1000	<u>51.9</u>	<u>35.5</u>	<u>43.1</u>	38.9	<u>35.8</u>	<u>25.1</u>	24.3	<u>36.4</u>
LLaVA-Mini	1fps	76.0	50.0	44.5	37.5	49.0	24.3	18.4	42.8

Table 5: Results on EgoSchema (accuracy), a long-form video benchmark (~ 3 minutes) for first-person view temporal reasoning.

Methods	#Frames	EgoSchema
Random	-	20
mPLUG-Owl	16	31.1
InternVideo	16	32.1
Video-ChatGPT	100	36.2
VideoChat	16	<u>42.2</u>
TimeChat	96	33.0
LLaMA-VID	1fps	38.5
Video-LLaVA	8	38.4
LLaVA-Mini	1fps	51.2

reported in Table 1, where LLaVA-Mini achieves performance comparable to LLaVA-v1.5 while using only 1 vision token instead of 576. Previous efficient LMMs with fewer vision tokens often merged similar tokens directly after the vision encoder (Shang et al., 2024; Ye et al., 2024c), resulting in a significant loss of visual information and negatively impacting visual understanding of LMMs. For instance, LLaMA-VID, VoCo-LLaVA, and MQT-LLaVA, which reduce vision tokens to 1-2 tokens, lead to 5% performance drop on average. In contrast, LLaVA-Mini employs modality pre-fusion to integrate visual information into text tokens before compressing the vision tokens, achieving performance comparable to LLaVA-v1.5 at a token compression rate of 0.17%. Furthermore, LLaVA-Mini-HD shows an average performance improvement of 2.4% over LLaVA-v1.5 due to high-resolution image modeling. Note that LLaVA-Mini-HD has a computational load of 8.13 TFLOPs, which remains lower than LLaVA-v1.5’s 8.55 TFLOPs. More efficiency analyses refer to Sec.5.3. Overall, LLaVA-Mini preserves strong visual understanding capabilities while compressing vision tokens, enhancing the usability of efficient LMMs in visual scenarios.

Video-based Evaluation We compare LLaVA-Mini with advanced video LMMs on 5 widely used video-based benchmarks. The results are reported in Table 2 and 3, where LLaVA-Mini demonstrates superior overall performance. Video LMMs such as VideoChat (Li et al., 2024c), Video-LLaVA (Lin et al., 2023a), and Video-LLaMA (Maaz et al., 2024) use much more vision tokens to represent each frame, and thereby can extract only 8-16 frames from a video due to the limited context length of LLMs, which may result in the loss of visual information in some frames. In contrast, LLaVA-Mini uses one vision token to represent each image and accordingly can extract frames from the video at a rate of 1 frame per second, thus performing better on video understanding.

Extrapolation to Long Videos Furthermore, we compare LLaVA-Mini with advanced long-video LMMs (can process video over 100 frames) on long-form video benchmarks, MLVU (Zhou et al., 2024b) and EgoSchema (Mangalam et al., 2023). Note that LLaVA-Mini is trained only on Video-ChatGPT instruction data and has not been exposed to any long video data, so its performance on long videos is entirely derived from the length extrapolation capabilities of its framework (Press et al., 2022). As shown in Table 4 and 5, LLaVA-Mini exhibits significant advantages in long video understanding. By representing each frame as one token, LLaVA-Mini facilitates straightforward extension to longer videos during inference. In particular, LLaVA-Mini is only trained on videos shorter than 1 minute (< 60 frames), and performs well on MLVU’s long-form video, which encompasses videos over 2 hours (> 7200 frames) during inference. Overall, with one vision token per frame, LLaVA-Mini demonstrates high-quality video understanding in a more efficient manner.

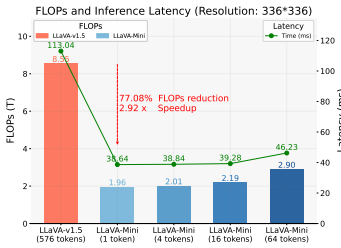


Figure 7: FLOPs and latency of LLaVA-Mini.

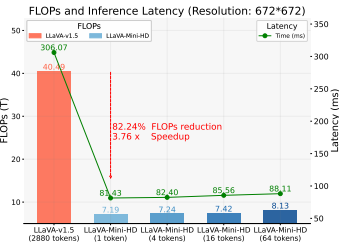


Figure 8: FLOPs and latency of LLaVA-Mini-HD.

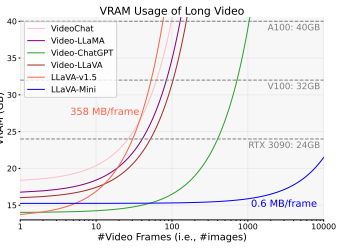


Figure 9: VRAM usage (3-hour video) of LLaVA-Mini.

Table 6: Performance of LLaVA-Mini with different numbers of modality pre-fusion layers N_{fusion} .

Methods	Pre-fusion #Layers	#Vision Tokens	FLOPs (T)	Performance		
				VQA ^{v2}	GQA	MMB
LLaVA-v1.5	-	576	8.55	78.5	62.0	64.3
LLaVA-Mini (w/o pre-fusion)	0	1	0.96	72.4	54.2	57.7
	0	16	1.16	74.1	55.4	59.2
	0	64	1.79	75.3	56.7	62.1
	0	144	2.85	76.9	58.9	64.9
LLaVA-Mini (w/ pre-fusion)	1	1	1.21	74.8	55.5	60.4
	2	1	1.46	76.0	57.6	63.1
	3	1	1.81	76.9	59.1	64.9
	4	1	1.96	77.6	60.9	65.6

Table 7: Performance of LLaVA-Mini with various vision tokens.

Methods	Res.	#Vision Tokens	Performance		
			VQA ^{v2}	GQA	MMB
LLaVA-v1.5	336	576	78.5	62.0	64.3
	336	1	77.6	60.9	65.6
	336	4	77.7	60.9	66.7
	336	16	78.1	61.3	66.6
LLaVA-Mini	336	64	78.5	61.6	67.5
	672	16	78.5	61.5	67.4
	672	64	78.9	61.8	67.5
	672	144	79.3	62.3	67.9
	672	576	80.0	62.9	68.1

5.3 EFFICIENCY

With the performance comparable to LLaVA-v1.5, we further explore the computational efficiency offered by LLaVA-Mini. Figures 7, 8, 9 illustrate the advantages of LLaVA-Mini in terms of computational load, inference latency, and memory usage, where FLOPs are calculated by `calcflops` (Ye, 2023), and latency is tested on the A100 without any engineering acceleration techniques.

FLOPs and Inference Latency As shown in Figure 7, LLaVA-Mini significantly reduces computational load by 77% compared to LLaVA-v1.5, achieving a speedup of 2.9 times. LLaVA-Mini achieves response latency lower than 40 ms, which is crucial for developing low-latency real-time LMMs. As shown in Figure 8, when modeling at high resolutions, the efficiency advantages of LLaVA-Mini are even more pronounced, yielding 82% FLOPs reduction and 3.76 times speedup.

Memory Usage Memory consumption poses another challenge for LMMs, particularly in video processing. Figure 9 demonstrates the memory requirements of LMMs when processing videos of varying lengths. In previous methods, each image requires approximately 200-358 MB memory (Liu et al., 2023b; Lin et al., 2023a), limiting them to handle only about 100 frames on a 40GB GPU. In contrast, LLaVA-Mini with one vision token requires just 0.6 MB per image, enabling it to theoretically support video processing of over 10,000 frames on RTX 3090 with 24 GB of memory.

6 ANALYSES

6.1 SUPERIORITY OF MODALITY PRE-FUSION

The proposed modality pre-fusion is central to LLaVA-Mini, as it integrates visual information into text tokens in advance, facilitating extreme compression of vision tokens. To investigate the effects of modality pre-fusion, we conduct an ablation study in Table 6. Without pre-fusion, token compression results in a performance drop of around 5%, even with 144 vision tokens retained, the performance of LMMs falls short of LLaVA-v1.5. This also explains why previous token merging methods often exhibit poor performance (Ye et al., 2024c) or can only achieve a compression rate of over 40% (Shang et al., 2024). Notably, under the same FLOPs, increasing the number of pre-fusion layers yields greater benefits than increasing the number of compression vision tokens. This sup-

486
487
488
489
490
491
492
493
494
495
496
497

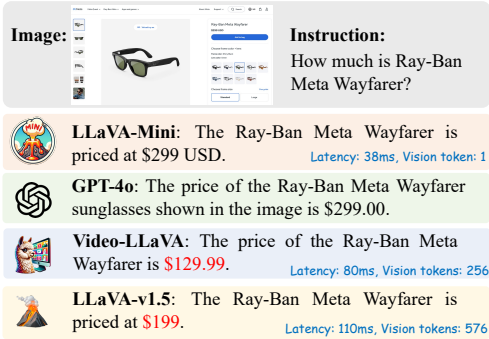


Figure 10: Case of image understanding.



Figure 11: Case of video understanding.

500
501
502
503
504
505
506
507
508
509
510
511
512
513

ports our preliminary analysis, which indicated that vision tokens exhibit varying importance across different layers and vision tokens are more critical in early layers. Investing more computational overhead in earlier stages where vision tokens are more important results in better performance.

6.2 EFFECT OF COMPRESSION

Table 8: Effect of query-based compression.

LLaVA-Mini employs query-based compression to achieve a high compression ratio for vision tokens. We compare the performance of query-based compression with direct average pooling in Table 8. Query-based compression can adaptively capture important features in the image while requiring only a minimal additional computational cost, demonstrating a significant advantage. Appendix F gives a visualization of the compression process and a more detailed analysis.

Compression	#Vision Tokens	FLOPs	Performance		
			VQA ^{v2}	GQA	MMB
Average Pooling	1	1.96T	76.1	59.8	64.0
Query-based		+2.42G	77.6	60.9	65.6
Average Pooling	4	2.01T	76.9	60.3	65.1
Query-based		+2.44G	77.7	60.9	66.7

6.3 PERFORMANCE WITH VARIOUS VISION TOKENS

514
515
516
517
518
519
520

LLaVA-Mini uses 1 vision token for standard images and 64 for high-resolution images. We explore the potential of LLaVA-Mini when further increasing the number of vision tokens (larger C) in Table 7. The results indicate that as the number of vision tokens increases, LLaVA-Mini continues to improve in performance. In particular, LLaVA-Mini outperforms LLaVA-v1.5 when both using 576 vision tokens, demonstrating its effectiveness when computational resources are plentiful.

6.4 CASE STUDY

521
522
523
524
525
526
527
528
529
530
531
532
533
534

Figures 10 and 11 present examples of LLaVA-Mini in image and video understanding tasks (refer to Appendix G for more cases). Despite using only one vision token, LLaVA-Mini performs effectively in capturing visual details, such as accurately identifying price information (OCR) in website screenshots. For video understanding, Video-LLaVA extracts 8 frames per video, regardless of video duration (Lin et al., 2023a). Training on only 8 frames (sometimes missing key frames) can cause hallucinations (Khattak et al., 2024), encouraging LMM to forge information beyond the extracted frames. For instance, given a celebration scene, Video-LLaVA mistakenly imagines “a group of men playing soccer on a field” before the celebration. This fixed-length frame extraction is a forced compromise due to the large number of vision tokens required per image while LLM’s context length is limited. In contrast, LLaVA-Mini, utilizing just one vision token per frame, can process videos at 1 fps, resulting in more robust video understanding. Overall, LLaVA-Mini ensures strong visual understanding while enhancing efficiency, making it a practical solution for multimodal interaction.

7 CONCLUSION

535
536
537
538
539

We introduce LLaVA-Mini, an efficient LMM with minimal vision tokens. LLaVA-Mini excels in image and video understanding while exhibiting superiority in computational efficiency, inference latency, and memory usage, facilitating the real-time multimodal interaction with efficient LMMs.

REFERENCES

- 540
541
542 Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang
543 Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, local-
544 ization, text reading, and beyond, 2023. URL <https://arxiv.org/abs/2308.12966>.
- 545 Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy
546 Hoffman. Token merging: Your vit but faster, 2023. URL <https://arxiv.org/abs/2210.09461>.
- 547
548 Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet:
549 A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE*
550 *Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.
- 551
552 David Chen and William Dolan. Collecting highly parallel data for paraphrase evaluation. In Dekang
553 Lin, Yuji Matsumoto, and Rada Mihalcea (eds.), *Proceedings of the 49th Annual Meeting of*
554 *the Association for Computational Linguistics: Human Language Technologies*, pp. 190–200,
555 Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL <https://aclanthology.org/P11-1020>.
- 556
557 Joya Chen, Zhaoyang Lv, Shiwei Wu, Kevin Qinghong Lin, Chenan Song, Difei Gao, Jia-Wei Liu,
558 Ziteng Gao, Dongxing Mao, and Mike Zheng Shou. Videollm-online: Online video large lan-
559 guage model for streaming video. In *CVPR*, 2024.
- 560
561 Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng,
562 Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An
563 open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023. URL <https://lmsys.org/blog/2023-03-30-vicuna/>.
- 564
565 Xiangxiang Chu, Limeng Qiao, Xinyang Lin, Shuang Xu, Yang Yang, Yiming Hu, Fei Wei, Xinyu
566 Zhang, Bo Zhang, Xiaolin Wei, et al. Mobilevlm: A fast, reproducible and strong vision language
567 assistant for mobile devices. *arXiv preprint arXiv:2312.16886*, 2023.
- 568
569 Xiangxiang Chu, Limeng Qiao, Xinyu Zhang, Shuang Xu, Fei Wei, Yang Yang, Xiaofei Sun, Yiming
570 Hu, Xinyang Lin, Bo Zhang, et al. Mobilevlm v2: Faster and stronger baseline for vision language
571 model. *arXiv preprint arXiv:2402.03766*, 2024.
- 572
573 Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas
574 Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszko-
575 reit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recogni-
576 tion at scale. In *International Conference on Learning Representations*, 2021. URL <https://openreview.net/forum?id=YicbFdNTTy>.
- 577
578 Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha
579 Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony
580 Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark,
581 Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere,
582 Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris
583 Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong,
584 Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny
585 Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino,
586 Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael
587 Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Ander-
588 son, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah
589 Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan
590 Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Ma-
591 hadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy
592 Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak,
593 Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Al-
wala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini,
Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yeary, Laurens van der
Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo,

594 Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Man-
595 nat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova,
596 Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal,
597 Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Olivier Duchenne, Onur
598 Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhar-
599 gava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong,
600 Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic,
601 Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sum-
602 baly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa,
603 Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang,
604 Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende,
605 Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney
606 Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom,
607 Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta,
608 Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vladan Petro-
609 vic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang,
610 Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuwei Wang, Yaelle Goldschlag, Yashesh Gaur,
611 Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre
612 Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aaron Grattafiori, Abha
613 Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay
614 Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda
615 Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew
616 Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita
617 Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh
618 Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De
619 Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Bran-
620 don Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina
621 Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai,
622 Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li,
623 Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana
624 Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil,
625 Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Ar-
626 caute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco
627 Caggioni, Francisco Guzmán, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella
628 Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory
629 Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang,
630 Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Gold-
631 man, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman,
632 James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer
633 Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe
634 Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie
635 Wang, Kai Wu, Kam Hou U, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun
636 Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal
637 Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva,
638 Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian
639 Khabsa, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson,
640 Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Ke-
641 neally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel
642 Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mo-
643 hammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navy-
644 ata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong,
645 Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli,
646 Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux,
647 Piotr Dollár, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao,
Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li,
Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Ruty Rinott,
Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Sa-
tadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lind-
say, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Shuqiang

- 648 Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen
649 Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sungmin Cho,
650 Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser,
651 Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Tim-
652 othy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan,
653 Vinay Satish Kumar, Vishal Mangla, Vitor Albiero, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu
654 Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Con-
655 stable, Xiaocheng Tang, Xiaofang Wang, Xiaojuan Wu, Xiaolan Wang, Xide Xia, Xilun Wu,
656 Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi,
657 Youngjin Nam, Yu, Wang, Yuchen Hao, Yundi Qian, Yuze He, Zach Rait, Zachary DeVito, Zef
658 Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. The llama 3 herd of models, 2024.
659 URL <https://arxiv.org/abs/2407.21783>.
- 660 Xiaoran Fan, Tao Ji, Changhao Jiang, Shuo Li, Senjie Jin, Sirui Song, Junke Wang, Boyang Hong,
661 Lu Chen, Guodong Zheng, Ming Zhang, Caishuang Huang, Rui Zheng, Zhiheng Xi, Yuhao Zhou,
662 Shihan Dou, Junjie Ye, Hang Yan, Tao Gui, Qi Zhang, Xipeng Qiu, Xuanjing Huang, Zuxuan Wu,
663 and Yu-Gang Jiang. Mousi: Poly-visual-expert vision-language models, 2024. URL <https://arxiv.org/abs/2401.17221>.
- 665 Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu
666 Zheng, Ke Li, Xing Sun, Yunsheng Wu, and Rongrong Ji. Mme: A comprehensive evaluation
667 benchmark for multimodal large language models, 2024. URL <https://arxiv.org/abs/2306.13394>.
- 669 Yuxin Zhang Xiawu Zheng Xiaoshuai Sun Rongrong Ji Gen Luo, Yiyi Zhou. Feast your
670 eyes: Mixture-of-resolution adaptation for multimodal large language models. *arXiv preprint*
671 *arXiv:2403.03003*, 2024.
- 673 Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa
674 matter: Elevating the role of image understanding in visual question answering. In *Proceedings*
675 *of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.
- 676 Danna Gurari, Qing Li, Abigale J. Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and
677 Jeffrey P. Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In
678 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June
679 2018.
- 680 Bo He, Hengduo Li, Young Kyun Jang, Menglin Jia, Xuefei Cao, Ashish Shah, Abhinav Shrivastava,
681 and Ser-Nam Lim. Ma-Imm: Memory-augmented large multimodal model for long-term video
682 understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern*
683 *Recognition (CVPR)*, 2024.
- 685 Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked
686 autoencoders are scalable vision learners, 2021. URL <https://arxiv.org/abs/2111.06377>.
- 688 Wenbo Hu, Zi-Yi Dou, Liunian Harold Li, Amita Kamath, Nanyun Peng, and Kai-Wei Chang. Ma-
689 tryoshka query transformer for large vision-language models, 2024. URL <https://arxiv.org/abs/2405.19315>.
- 691 Drew A. Hudson and Christopher D. Manning. Gqa: A new dataset for real-world visual reasoning
692 and compositional question answering. In *Proceedings of the IEEE/CVF Conference on Computer*
693 *Vision and Pattern Recognition (CVPR)*, June 2019.
- 695 Muhammad Uzair Khattak, Muhammad Ferjad Naeem, Jameel Hassan, Muzammal Naseer, Fed-
696 erico Tombari, Fahad Shahbaz Khan, and Salman Khan. How good is my video Imm? com-
697 plex video reasoning and robustness evaluation suite for video-Imm, 2024. URL <https://arxiv.org/abs/2405.03690>.
- 699 H Laurençon, Daniel van Strien, Stas Bekman, Leo Tronchon, Lucile Saulnier, Thomas Wang,
700 Siddharth Karamcheti, Amanpreet Singh, Giada Pistilli, Yacine Jernite, et al. Introducing idefics:
701 An open reproduction of state-of-the-art visual language model, 2023. URL <https://huggingface.co/blog/idefics>. Accessed, pp. 09–18, 2023.

- 702 Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li,
703 Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer, 2024a. URL <https://arxiv.org/abs/2408.03326>.
704
- 705 Bohao Li, Yuying Ge, Yixiao Ge, Guangzhi Wang, Rui Wang, Ruimao Zhang, and Ying Shan.
706 Seed-bench: Benchmarking multimodal large language models. In *Proceedings of the IEEE/CVF*
707 *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13299–13308, June 2024b.
708
- 709 Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image
710 pre-training with frozen image encoders and large language models. In Andreas Krause, Emma
711 Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.),
712 *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Pro-*
713 *ceedings of Machine Learning Research*, pp. 19730–19742. PMLR, 23–29 Jul 2023a. URL
714 <https://proceedings.mlr.press/v202/li23q.html>.
- 715 KunChang Li, Yanan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang,
716 and Yu Qiao. Videochat: Chat-centric video understanding, 2024c. URL <https://arxiv.org/abs/2305.06355>.
717
- 718 Kunchang Li, Yali Wang, Yanan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen,
719 Ping Luo, Limin Wang, and Yu Qiao. Mvbench: A comprehensive multi-modal video understand-
720 ing benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern*
721 *Recognition (CVPR)*, pp. 22195–22206, June 2024d.
722
- 723 Wentong Li, Yuqian Yuan, Jian Liu, Dongqi Tang, Song Wang, Jie Qin, Jianke Zhu, and Lei Zhang.
724 Tokenpacker: Efficient visual projector for multimodal llm, 2024e. URL <https://arxiv.org/abs/2407.02392>.
725
- 726 Yanwei Li, Chengyao Wang, and Jiaya Jia. Llama-vid: An image is worth 2 tokens in large language
727 models, 2023b. URL <https://arxiv.org/abs/2311.17043>.
728
- 729 Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating object
730 hallucination in large vision-language models. In *The 2023 Conference on Empirical Methods*
731 *in Natural Language Processing*, 2023c. URL <https://openreview.net/forum?id=xozJw0kZXF>.
732
- 733 Bin Lin, Yang Ye, Bin Zhu, Jiayi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning
734 united visual representation by alignment before projection, 2023a. URL <https://arxiv.org/abs/2311.10122>.
735
- 736 Ziyi Lin, Chris Liu, Renrui Zhang, Peng Gao, Longtian Qiu, Han Xiao, Han Qiu, Chen Lin, Wenqi
737 Shao, Keqin Chen, Jiaming Han, Siyuan Huang, Yichi Zhang, Xuming He, Hongsheng Li, and
738 Yu Qiao. Sphinx: The joint mixing of weights, tasks, and visual embeddings for multi-modal
739 large language models, 2023b. URL <https://arxiv.org/abs/2311.07575>.
- 740 Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In
741 A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in*
742 *Neural Information Processing Systems*, volume 36, pp. 34892–34916. Curran Associates, Inc.,
743 2023a. URL [https://proceedings.neurips.cc/paper_files/paper/2023/](https://proceedings.neurips.cc/paper_files/paper/2023/file/6dcf277ea32ce3288914faf369fe6de0-Paper-Conference.pdf)
744 [file/6dcf277ea32ce3288914faf369fe6de0-Paper-Conference.pdf](https://proceedings.neurips.cc/paper_files/paper/2023/file/6dcf277ea32ce3288914faf369fe6de0-Paper-Conference.pdf).
- 745 Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In
746 A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in*
747 *Neural Information Processing Systems*, volume 36, pp. 34892–34916. Curran Associates, Inc.,
748 2023b. URL [https://proceedings.neurips.cc/paper_files/paper/2023/](https://proceedings.neurips.cc/paper_files/paper/2023/file/6dcf277ea32ce3288914faf369fe6de0-Paper-Conference.pdf)
749 [file/6dcf277ea32ce3288914faf369fe6de0-Paper-Conference.pdf](https://proceedings.neurips.cc/paper_files/paper/2023/file/6dcf277ea32ce3288914faf369fe6de0-Paper-Conference.pdf).
- 750 Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction
751 tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*
752 *(CVPR)*, pp. 26296–26306, June 2024a.
753
- 754 Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee.
755 Llava-next: Improved reasoning, ocr, and world knowledge, January 2024b. URL <https://llava-vl.github.io/blog/2024-01-30-llava-next/>.

- 756 Ruyang Liu, Chen Li, Yixiao Ge, Ying Shan, Thomas H Li, and Ge Li. One for all: Video conver-
757 sation is feasible without video instruction tuning. *arXiv preprint arXiv:2309.15785*, 2023c.
758
- 759 Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan,
760 Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. Mmbench: Is your multi-modal
761 model an all-around player?, 2024c. URL <https://arxiv.org/abs/2307.06281>.
- 762 Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu,
763 Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal rea-
764 soning via thought chains for science question answering. In S. Koyejo, S. Mo-
765 hamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural*
766 *Information Processing Systems*, volume 35, pp. 2507–2521. Curran Associates, Inc.,
767 2022. URL [https://proceedings.neurips.cc/paper_files/paper/2022/](https://proceedings.neurips.cc/paper_files/paper/2022/file/11332b6b6cf4485b84afadb1352d3a9a-Paper-Conference.pdf)
768 [file/11332b6b6cf4485b84afadb1352d3a9a-Paper-Conference.pdf](https://proceedings.neurips.cc/paper_files/paper/2022/file/11332b6b6cf4485b84afadb1352d3a9a-Paper-Conference.pdf).
- 769 Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-chatgpt:
770 Towards detailed video understanding via large vision and language models, 2024. URL <https://arxiv.org/abs/2306.05424>.
771
- 772 Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnos-
773 tic benchmark for very long-form video language understanding. In *Thirty-seventh Confer-*
774 *ence on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL
775 <https://openreview.net/forum?id=Jv1Wseddak>.
776
- 777 OpenAI. Introducing chatgpt, 2022. URL <https://openai.com/blog/chatgpt>.
778
- 779 OpenAI. Gpt-4 technical report, 2023.
780
- 781 OpenAI. Hello gpt-4o, 2024a. URL <https://openai.com/index/hello-gpt-4o/>.
782
- 783 OpenAI. Gpt-4v(ision) system card, 2024b. URL [https://cdn.openai.com/papers/](https://cdn.openai.com/papers/GPTV_System_Card.pdf)
784 [GPTV_System_Card.pdf](https://cdn.openai.com/papers/GPTV_System_Card.pdf).
785
- 786 Ofir Press, Noah Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables
787 input length extrapolation. In *International Conference on Learning Representations*, 2022. URL
788 <https://openreview.net/forum?id=R8sQPpGCv0>.
789
- 790 Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agar-
791 wal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya
792 Sutskever. Learning transferable visual models from natural language supervision. In Marina
793 Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine*
Learning, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR,
794 18–24 Jul 2021. URL [https://proceedings.mlr.press/v139/radford21a.](https://proceedings.mlr.press/v139/radford21a.html)
795 [html](https://proceedings.mlr.press/v139/radford21a.html).
- 796 Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. Timechat: A time-sensitive multimodal
797 large language model for long video understanding. *ArXiv*, abs/2312.02051, 2023.
798
- 799 Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. Llava-prumerge: Adaptive token
800 reduction for efficient large multimodal models, 2024. URL [https://arxiv.org/abs/](https://arxiv.org/abs/2403.15388)
801 [2403.15388](https://arxiv.org/abs/2403.15388).
802
- 803 Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh,
804 and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF*
805 *Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
806
- 807 Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe
808 Chi, Xun Guo, Tian Ye, Yanting Zhang, Yan Lu, Jenq-Neng Hwang, and Gaoang Wang.
809 Moviechat: From dense token to sparse memory for long video understanding. In *Proceedings*
of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 18221–
18232, June 2024a.
- Zhende Song, Chenchen Wang, Jiamu Sheng, Chi Zhang, Gang Yu, Jiayuan Fan, and Tao Chen.
Moviellm: Enhancing long video understanding with ai-generated movies, 2024b.

- 810 Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha
811 Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, Austin Wang, Rob Fergus, Yann
812 LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal
813 llms, 2024.
- 814 Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée
815 Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Ar-
816 mand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation
817 language models, 2023a.
- 818 Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Niko-
819 lay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher,
820 Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy
821 Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn,
822 Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel
823 Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee,
824 Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra,
825 Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi,
826 Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh
827 Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen
828 Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic,
829 Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models,
830 2023b.
- 831 Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez,
832 Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von
833 Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Ad-
834 vances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.,
835 2017. URL [https://proceedings.neurips.cc/paper_files/paper/2017/
836 file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf).
- 837 An-Lan Wang, Bin Shan, Wei Shi, Kun-Yu Lin, Xiang Fei, Guozhi Tang, Lei Liao, Jingqun Tang,
838 Can Huang, and Wei-Shi Zheng. Pargo: Bridging vision-language with partial and global views,
839 2024. URL <https://arxiv.org/abs/2408.12928>.
- 840 Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan
841 Xu, Yi Liu, Zun Wang, Sen Xing, Guo Chen, Junting Pan, Jiashuo Yu, Yali Wang, Limin Wang,
842 and Yu Qiao. Internvideo: General video foundation models via generative and discriminative
843 learning, 2022. URL <https://arxiv.org/abs/2212.03191>.
- 844 Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming lan-
845 guage models with attention sinks, 2024. URL <https://arxiv.org/abs/2309.17453>.
- 846 Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging
847 video and language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern
848 Recognition (CVPR)*, June 2016.
- 849 Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen
850 Hu, Pengcheng Shi, Yaya Shi, Chenliang Li, Yuanhong Xu, Hehong Chen, Junfeng Tian, Qi Qian,
851 Ji Zhang, Fei Huang, and Jingren Zhou. mplug-owl: Modularization empowers large language
852 models with multimodality, 2024a. URL <https://arxiv.org/abs/2304.14178>.
- 853 Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, and Fei
854 Huang. mplug-owl2: Revolutionizing multi-modal large language model with modality collabor-
855 ation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition
856 (CVPR)*, pp. 13040–13051, June 2024b.
- 857 Xiaojun Ye. calcflops: a flops and params calculate tool for neural networks in pytorch framework,
858 2023. URL <https://github.com/MrYxJ/calculate-flops.pytorch>.
- 859 Xubing Ye, Yukang Gan, Xiaoke Huang, Yixiao Ge, Ying Shan, and Yansong Tang. Voco-llama:
860 Towards vision compression with large language models, 2024c. URL [https://arxiv.org/
861 abs/2406.12275](https://arxiv.org/abs/2406.12275).

864 Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang,
865 and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities, 2023.
866 URL <https://arxiv.org/abs/2308.02490>.
867

868 Zhengqing Yuan, Zhaoxu Li, Weiran Huang, Yanfang Ye, and Lichao Sun. TinyGPT-v: Ef-
869 ficient multimodal large language model via small backbones. In *2nd Workshop on Ad-*
870 *vaning Neural Network Training: Computational Efficiency, Scalability, and Resource Opti-*
871 *mization (WANT@ICML 2024)*, 2024a. URL <https://openreview.net/forum?id=lvmjTZQhRk>.
872

873 Zhihang Yuan, Yuzhang Shang, Yang Zhou, Zhen Dong, Zhe Zhou, Chenhao Xue, Bingzhe Wu,
874 Zhikai Li, Qingyi Gu, Yong Jae Lee, Yan Yan, Beidi Chen, Guangyu Sun, and Kurt Keutzer. Llm
875 inference unveiled: Survey and roofline model insights, 2024b. URL <https://arxiv.org/abs/2402.16363>.
876

877 Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language
878 model for video understanding, 2023. URL <https://arxiv.org/abs/2306.02858>.
879

880 Renrui Zhang, Jiaming Han, Chris Liu, Peng Gao, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu,
881 Hongsheng Li, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-
882 init attention, 2024. URL <https://arxiv.org/abs/2303.16199>.

883 Baichuan Zhou, Ying Hu, Xi Weng, Junlong Jia, Jie Luo, Xien Liu, Ji Wu, and Lei Huang. Tynyllava:
884 A framework of small-scale large multimodal models, 2024a. URL <https://arxiv.org/abs/2402.14289>.
885
886

887 Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Shitao Xiao, Xi Yang, Yongping Xiong, Bo Zhang,
888 Tiejun Huang, and Zheng Liu. Mlvu: A comprehensive benchmark for multi-task long video
889 understanding. *arXiv preprint arXiv:2406.04264*, 2024b.

890 Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. MiniGPT-4: Enhancing
891 vision-language understanding with advanced large language models. In *The Twelfth Interna-*
892 *tional Conference on Learning Representations*, 2024. URL [https://openreview.net/](https://openreview.net/forum?id=1tZbq88f27)
893 [forum?id=1tZbq88f27](https://openreview.net/forum?id=1tZbq88f27).
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917

A DETAILED SETTING OF PRELIMINARY ANALYSES

In Sec.3.2, we analyze the importance of visual tokens in LLMs from an attention-based perspective to inform strategies for compressing vision tokens. Here, we give the detailed introduction of the experimental setup for the attention analysis.

We focus on the LLaVA series architecture, where the input tokens to the LLM are composed of instruction tokens, vision tokens, and response tokens, as shown in Eq.(1). We compute the average attention received by each type of token to reveal how the importance of different token categories changes across layers.

Calculation of Attention Weights Formally, we denote the attention of the i^{th} token h_i to the j^{th} token h_j as a_{ij} , where a_{ij} is the average attention across all attention heads. All tokens fed to the LLM are divided into instruction tokens, vision tokens, and response tokens according to inputs type, denoted as sets $T_{instruction}$, T_{vision} , and $T_{response}$ respectively. Finally, denoted the target and source token types as $tgt_type, src_type \in \{\text{instruction, vision, response}\}$, the average attention weights from tgt_type type tokens to src_type type tokens in our analyses are calculated as:

$$\text{Attn}(tgt_type \rightarrow src_type) = \frac{\sum_{h_i \in T_{tgt_type}} \sum_{h_j \in T_{src_type}} a_{ij}}{\sum_{h_i \in T_{tgt_type}} \mathbb{1}_{\sum_{h_j \in T_{src_type}} a_{ij} > 0}}, \quad (4)$$

$$\text{where } \mathbb{1}_{\sum_{h_j \in T_{src_type}} a_{ij} > 0} = \begin{cases} 1 & \text{if } \sum_{h_j \in T_{src_type}} a_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

Specifically, $\sum_{h_i \in T_{tgt_type}} \sum_{h_j \in T_{src_type}} a_{ij}$ calculates the sum of attention weights from all tgt_type type tokens to all src_type type tokens, $\sum_{h_i \in T_{tgt_type}} \mathbb{1}_{\sum_{h_j \in T_{src_type}} a_{ij} > 0}$ counts the number of tgt_type type tokens, thus $\text{Attn}(tgt_type \rightarrow src_type)$ represents the average attention weight from tgt_type type tokens to src_type type tokens. $\text{Attn}(tgt_type \rightarrow src_type)$ is consistent with the legend in Figure 2.

Calculation of Attention Entropy The calculation of attention entropy is similar to that of attention weights, with the key difference being the addition of a normalization step. When computing the entropy of a specific type of token (e.g., vision tokens), the sum of attention weights for this token type may not equal 1. Thus, we perform a normalization on the attention of these tokens (e.g., vision tokens) to ensure the definition of entropy is satisfied.

In practice, for LLaVA-v1.5 (pad) (Liu et al., 2023b) and LLaVA-NeXT (anyres) (Liu et al., 2024b), which may involve different resolution vision inputs, we use their original settings. In our analysis, we do not further distinguish between different types of vision tokens (e.g., global or local), but treat them collectively as vision tokens.

B TRAINING DETAILS

Implementation Details The compression method of LLaVA-Mini can be easily plugged into existing multi-modal pipelines, as it only requires the addition of two extra modules (the compression module and the modality pre-fusion module) before the LLM, while the other components (such as the vision encoder, the LLM, and the training loss) remain unchanged. The pre-fusion module applies the same decoder-only architecture as the LLM, including both the structure and hyperparameters. The motivation behind this setting is to ensure flexible adaptation to existing LLM frameworks and other acceleration techniques.

Training The overall training process follows a two-stage paradigm similar to LLaVA, consisting of vision-language pretraining followed by instruction tuning. Table 9 reports the two-stage training details of LLaVA-Mini.

C BENCHMARKS

We conduct a comprehensive evaluation of LLaVA-Mini, including both image and video understanding benchmarks.

Table 9: Training details of LLaVA-Mini.

Settings	Stage1		Stage2
	Vision-Language Pretraining		Instruction Tuning
Modules	Vision Encoder	Frozen	Frozen
	Projection	Trainable	Trainable
	Large Language Model	Frozen	Trainable
	Compression	N/A	Trainable
	Modality Pre-fusion	N/A	Trainable
Hyperparameters	Batch Size	256	256
	Learning Rate	-	1e-4
	MM Learning Rate	1e-3	1e-5
	Schedule	Cosine decay	
	Warmup Ratio	0.03	
	Optimizer	AdamW	
	Epoch	1	2

C.1 IMAGE-BASED BENCHMARKS

Following the LLaVA framework (Liu et al., 2023b), we conduct experiments on 11 widely adopted benchmarks, including VQA-v2 (VQA^{v2}) (Goyal et al., 2017), GQA (Hudson & Manning, 2019), VisWiz (Gurari et al., 2018), ScienceQA-IMG (SciQA) (Lu et al., 2022), TextVQA (VQA^T) (Singh et al., 2019), POPE (Li et al., 2023c), MME (Fu et al., 2024), MMBench (MMB) (Liu et al., 2024c), SEED-Bench (SEED) (Li et al., 2024b), LLaVA-Bench-in-the-Wild (LLaVA^W) (Liu et al., 2023a), and MM-Vet (Yu et al., 2023), which cover a diverse range of visual tasks. The evaluation pipelines for all benchmarks are consistent with those used in LLaVA.

C.2 VIDEO-BASED BENCHMARKS

Video-based Generative Performance Benchmark For video-based evaluation, we conduct experiments on video open-ended question-answering benchmarks, including MSVD-QA (Chen & Dolan, 2011), MSRVTT-QA (Xu et al., 2016), and ActivityNet-QA (Caba Heilbron et al., 2015). Furthermore, we use the video-based generative performance benchmark (Maaz et al., 2024) to assess the performance of LLaVA-Mini across five dimensions: correctness, detail orientation, contextual understanding, temporal understanding, and consistency. The evaluation pipelines for both the open-ended question-answering and the generative performance benchmarks adhere to VideoChatGPT (Maaz et al., 2024), employing the GPT model (gpt-3.5-turbo version) to evaluate the accuracy of responses (True or False) and to assign a score ranging from 1 to 5 for response, where higher scores indicate superior performance.

MVBench (Li et al., 2024d) MVBench is a comprehensive benchmark for multimodal video understanding that encompasses 20 challenging tasks. The evaluation aspects of MVBench include Action (such as Action Sequence, Action Prediction, Action Antonym, Fine-grained Action, and Unexpected Action), Object (Object Existence, Object Interaction, Object Shuffle), Position (Moving Direction, Action Localization), Scene (Scene Transition), Count (Action Count, Moving Count), Attribute (Moving Attribute, State Change), Pose (Fine-grained Pose), Character (Character Order), and Cognition (Egocentric Navigation, Episodic Reasoning, Counterfactual Inference). The evaluation of MVBench employs a multiple-choice format, using accuracy as the metric.

MLVU (Zhou et al., 2024b) MLVU is a comprehensive benchmark for multi-task long video understanding. The evaluation aspects of MLVU include Topic Reasoning (TR), Anomaly Recognition (AR), Needle QA (NQA), Ego Reasoning (ER), Plot QA (PQA), Action Order (AO), and Action Count (AC). The evaluation of MLVU also employs a multiple-choice format, using accuracy as the metric.

EgoSchema (Mangalam et al., 2023) EgoSchema is a long-form video question-answering dataset, which serves as a benchmark for assessing the long video understanding capabilities of first-person videos. The evaluation of EgoSchema also employs a multiple-choice format, using accuracy as the metric.

D INTRODUCTION TO BASELINES

LLaVA-Mini is an image and video LMM, so we compare it with several advanced image-based and video-based LMMs.

D.1 IMAGE-BASED LMMs

We compare LLaVA-Mini with LLaVA-v1.5 (Liu et al., 2023b) and other advanced LMMs of similar data and model scales, including BLIP-2 (Li et al., 2023a), InstructBLIP (Liu et al., 2024a), IDEFICS (Laurençon et al., 2023), Qwen-VL (Bai et al., 2023), Qwen-VL-Chat (Bai et al., 2023), SPHINX (Lin et al., 2023b), mPLUG-Owl2 (Ye et al., 2024b).

LMMs with Fewer Vision Tokens Additionally, we assess LLaVA-Mini against various efficient LMMs that utilize fewer vision tokens, showing advantages in compression rate and performance. Most of these models share the same architecture and training data as LLaVA, primarily focusing on the merging of vision tokens in the vision encoder. These efficient LMMs are introduced as follows.

MQT-LLaVA (Hu et al., 2024) introduces a flexible query transformer that allows encoding an image into a variable number of visual tokens (up to a predefined maximum) to adapt to different tasks and computational resources.

PruMerge (Shang et al., 2024) reduces visual tokens in LMMs by identifying and merging important tokens based on the attention sparsity in vision encoder. PruMerge has a variant, named PruMerge++, which enhances the original PruMerge method by evenly adding more vision tokens (about 144 vision tokens) to further improve performance.

LLaMA-VID (Li et al., 2023b) LLaMA-VID compresses the instruction and image into one token respectively, with a total of two tokens representing each image, thus facilitating the understanding of longer videos.

VoCo-LLaMA (Ye et al., 2024c) compresses all vision tokens using language models, significantly improving computational efficiency.

TokenPacker (Li et al., 2024e) is a visual projector that efficiently reduces visual tokens by 80% using a coarse-to-fine approach.

Previous methods have often focused on reducing the number of vision tokens output by the vision encoder. LLaVA-Mini takes this a step further by shifting attention to how vision tokens and text tokens interact within the LLM backbone. Based on this insight, we propose modality pre-fusion, which enables better performance even under the extreme compression of reducing vision tokens to just one token.

D.2 VIDEO-BASED LMMs

LLaVA-Mini can also perform high-quality video understanding, so we compare LLaVA-Mini with the current advanced video LMMs, including LLaMA-Adaptor (Zhang et al., 2024), InternVideo (Wang et al., 2022), VideoChat (Li et al., 2024c), Video-LLaMA (Zhang et al., 2023), mPLUG-Owl (Ye et al., 2024a), Video-ChatGPT (Maaz et al., 2024), BT-Adapor (Liu et al., 2023c), LLaMA-VID (Li et al., 2023b), and Video-LLaVA (Lin et al., 2023a).

We also compare LLaVA-Mini with several video LMMs specifically designed for long videos, including MovieChat (Song et al., 2024a), Movie-LLM (Song et al., 2024b), TimeChat (Ren et al., 2023), MA-LMM (He et al., 2024). Note that among these video LMMs, LLaVA-Mini and Video-LLaVA can complete image and video understanding with a unified model.

E EXTENDED EXPERIMENTAL RESULTS

E.1 RESULTS ON VISION-CENTRIC BENCHMARK

To further demonstrate its understanding capabilities of visual information, we evaluate LLaVA-Mini on vision-centric benchmarks CV-Bench, which is an widely-used benchmark for assessing

Table 10: Performance of LLaVA-Mini on CV-Bench, a vision-centric benchmarks of both 2D and 3D visual understanding.

Methods	#Vision Tokens	CVBench-2D	CVBench-3D	Avg.
LLaVA-v1.5	576	61.96	58.58	60.27
LLaVA-Mini	1	62.31	69.33	65.82
LLaVA-Mini	4	63.42	72.00	67.71
LLaVA-Mini	16	65.58	73.75	69.66

vision-centric capabilities, encompassing both 2D and 3D visual understanding (Tong et al., 2024). The results are shown in Table 10, where LLaVA-Mini demonstrates superior vision-centric understanding with fewer vision tokens on vision-centric understanding.

E.2 EFFECT OF COMPRESSION MODULE

Table 11: Comparison of LLaVA-Mini with previous token merging methods.

Methods	#Vision Tokens	Performance		
		VQA ^{v2}	GQA	MMB
MQT-LLaVA	2	61.0	50.8	54.4
MQT-LLaVA	36	73.7	58.8	63.4
MQT-LLaVA	256	76.8	61.6	64.3
PruMerge	32	72.0	-	60.9
PruMerge++	144	76.8	-	64.9
LLaVA-Mini	1	72.4	54.2	57.7
LLaVA-Mini	16	74.1	55.4	59.2
LLaVA-Mini	64	75.3	56.7	62.1
LLaVA-Mini	144	76.9	58.9	64.9

To verify the effectiveness of the compression module, we compared the compression module in LLaVA-Mini with previous advanced token merging methods. To ensure a fair comparison of token compression performance, we have removed the modality pre-fusion module from LLaVA-Mini for the comparison with SOTA token merging methods, including PruMerge (Shang et al., 2024), PruMerge++ (Shang et al., 2024), and MQT-LLaVA (Hu et al., 2024). Specifically, PruMerge applies the widely-used token merge (ToMe) technique (Bolya et al., 2023) on ViT, PruMerge++ improves upon PruMerge by uniformly sampling additional vision tokens, and MQT-LLaVA employs Matryoshka representation learning to compress vision tokens.

As shown in the Table 11, LLaVA-Mini’s compression module outperforms PruMerge, PruMerge++, and MQT-LLaVA at the same compression rate, showing the advantages of query-based compression.

E.3 EFFECT OF MODALITY PRE-FUSION

Table 12: Performance of LLaVA-Mini when using only pre-fusion module without compression.

Methods	#Vision Tokens	Performance		
		VQA ^{v2}	GQA	MMB
LLaVA-v1.5	576	78.5	62.0	64.3
LLaVA-Mini (w/o compression)	576	80.0	62.9	66.2

To validate the effect of the pre-fusion module, we remove the compression module and retained only the modality pre-fusion module, thereby comparing with LLaVA-v1.5 while both using 576 vision tokens. As shown in Table, when using only the pre-fusion module without compression, LLaVA-Mini achieves superior performance compared to LLaVA-v1.5 with both using 576 vision tokens, demonstrating the effectiveness of the pre-fusion module.

E.4 WHY PREFORMING COMPRESSION AND PRE-FUSION OUTSIDE LLM BACKBONE?

LLaVA-Mini performs compression and modality pre-fusion before to the LLM backbone. The motivation for conducting these processes outside the LLM backbone, rather than conducting at the L^{th} layer within the LLM, stems from two key considerations:

- Vision representations after the L^{th} layers contain contextual information, which hinders the compression module: After the vision tokens are fed into the LLM, the early layers cause the visual representations to carry contextual information. Applying query-based compression on top of these representations makes it difficult for the compression module to distinguish between different vision tokens.
- The inter-layer operations within the LLM may not be compatible with existing acceleration frameworks: One of the main motivations for placing the compression and pre-fusion modules outside the LLM backbone in LLaVA-Mini is to keep the LLM backbone unchanged. This design allows for compatibility with nearly all existing LLM acceleration technologies and frameworks, further enhancing efficiency.

Table 13: Comparison of performing compression and pre-fusion outside or within LLM backbone.

Methods	#Vision Tokens	FLOPs (T)	Performance		
			VQA ^{v2}	GQA	MMB
LLaVA-Mini	1	1.96	77.6	60.9	65.6
LLaVA-Mini (perform compression and pre-fusion within LLM)	1	1.84	76.3	60.1	64.5

We also conduct a comparison between LLaVA-Mini and LLaVA-Mini (compression and pre-fusion within LLM) in Table 13. The results demonstrate that the configuration of LLaVA-Mini is more advantageous. We will incorporate this result and the architectural motivation into the manuscript as per your recommendation.

E.5 EFFICIENCY ACROSS VARIOUS HARDWARE

Table 14: Inference latency (millisecond) of LLaVA-Mini on various hardware platforms.

Methods	#Vision Tokens	RTX 3090 (24G)	A100 (40G)	A800 (80G)
LLaVA-v1.5	576	198.75	113.04	87.43
LLaVA-Mini	1	64.52	38.64	27.43
	4	65.52	38.84	27.71
	16	68.97	39.28	28.92
	64	80.10	46.23	34.65

The efficiency improvements brought by LLaVA-Mini stem from reduced computational load (FLOPs), which is consistent across different hardware platforms. To demonstrate the scalability of model efficiency across different hardware platforms, we compute the inference latency of LLaVA-Mini on three hardware platforms: RTX 3090, A100, and A800. As shown in Table ??, the efficiency improvements brought by LLaVA-Mini are scalable across these hardware platforms.

E.6 COMPUTATIONAL OVERHEAD OF EACH COMPONENT

LLaVA-Mini significantly reduces the computational load of LLMs by decreasing the number of vision tokens. To further study the proportion of computational load contributed by each component in LLaVA-Mini, we compute the FLOPs of each module, as shown in the Table 15. The proposed compression module and pre-fusion module incur minimal computational cost, while the computation required by the LLM backbone is significantly reduced.

1188
 1189
 1190
 1191
 1192
 1193
 1194
 1195
 1196
 1197
 1198
 1199
 1200
 1201
 1202
 1203
 1204
 1205
 1206
 1207
 1208
 1209
 1210
 1211
 1212
 1213
 1214
 1215
 1216
 1217
 1218
 1219
 1220
 1221
 1222
 1223
 1224
 1225
 1226
 1227
 1228
 1229
 1230
 1231
 1232
 1233
 1234
 1235
 1236
 1237
 1238
 1239
 1240
 1241

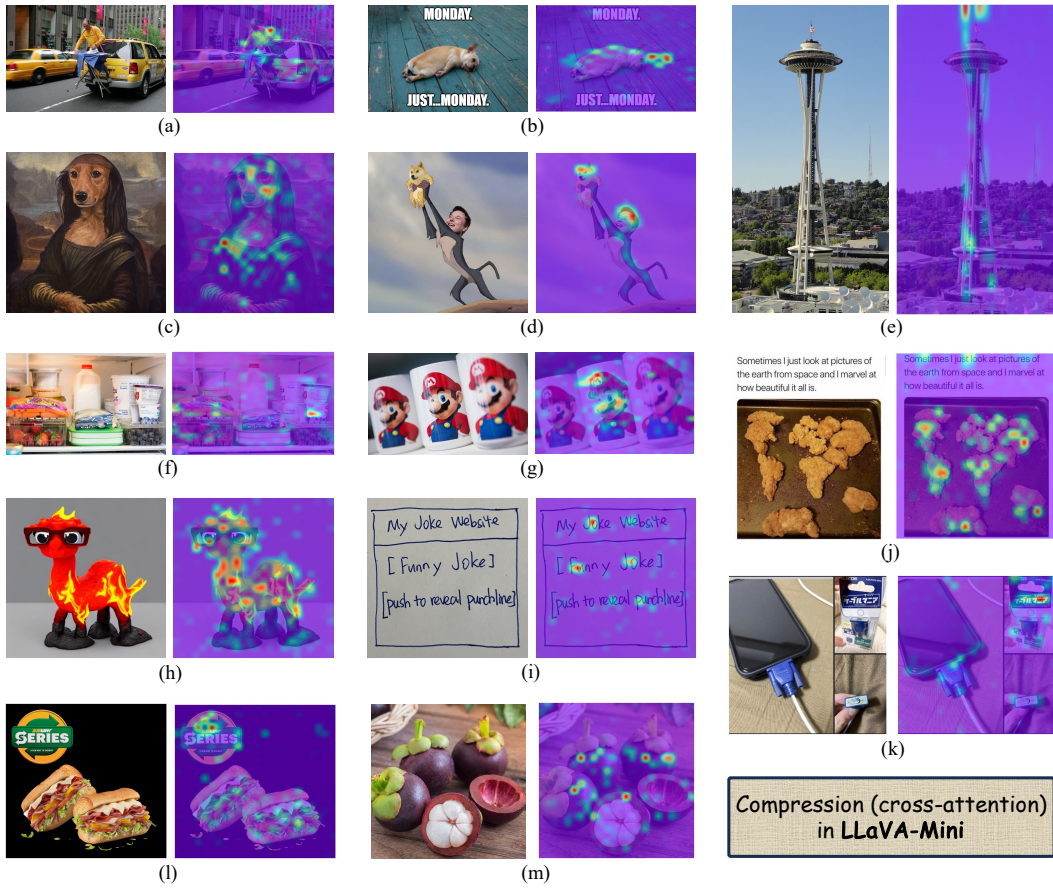


Figure 12: Visualization of the cross-attention in the compression module introduced in LLaVA-Mini. The left side is the original image, and the right side is the cross-attention distribution heatmap, where brighter areas are more heavily weighted during compression. The example images are all from the LLaVA-Bench-in-the-Wild benchmark.

Table 15: Computational overhead (FLOPs) of each component in LLaVA-Mini.

Methods	Res.	FLOPs(T)					
		Vision Encoder	Projection	Compression	Pre-fusion	LLM	Total
LLaVA-v1.5	336	0.349	0.024	-	-	8.177	8.55
LLaVA-Mini	336	0.349	0.024	0.001	0.125	1.460	1.96
LLaVA-v1.5	672	1.745	0.121	-	-	38.623	40.49
LLaVA-Mini	672	1.745	0.121	0.009	1.183	4.131	7.19

F VISUALIZATION OF COMPRESSION

LLaVA-Mini introduces query-based compression to adaptively compress vision tokens while preserving essential information. The learnable queries in compression module interact with all vision tokens through cross-attention to capture key visual information. To verify the effectiveness of the proposed compression, Figure 12 visualizes the cross-attention during the compression process. Across various image types and styles (e.g., photographs, text, screenshots, and cartoons), LLaVA-Mini’s compression exhibits strong interpretability, effectively extracting key visual information from images. In cases where critical information is concentrated (such as (b), (d), (h), (i) in Figure 12), LLaVA-Mini focuses on these key locations. Conversely, in cases where the main object is unclear. (such as (f), (j), (l), (m) in Figure 12), LLaVA-Mini exhibits a more dispersed attention pattern during the compression process, thereby preserving a broader range of visual information.

In particular, for complex image like Figure 12(k), which contain multiple sub-figures with logical relationships, the proposed compression module adaptively pay attention to the VGA-shaped charger, the product name on the charger packaging, and the charging port of the charger, demonstrating the superiority of the proposed compression. Overall, compared to compression based on average pooling, query-based compression allows LLaVA-Mini to adaptively assign greater weight to key information, effectively retaining important visual details after compression.

G MORE CASES

Image Understanding Figure 13 illustrates an example of LLaVA-Mini’s capabilities in more complex image reasoning. The image in Figure 13 incorporates features such as metaphor and counterfactual reasoning, requiring LMMs to accurately interpret the visual information and reason about the humorous aspects based on the entities present in the scene. The results demonstrate that only LLaVA-Mini and GPT-4o successfully capture the phrases “*the musicians are performing on a giant violin*” and “*The violin, being a large and buoyant object, is placed on the water*”, while both Video-LLaVA and LLaVA-v1.5 fail to understand this image. In terms of the perception of entities in the picture, both Video-LLaVA and LLaVA-v1.5 exhibit hallucinations in their descriptions. Specifically, Video-LLaVA erroneously interprets the image as “*The image is a panel of four pictures*” and “*the man is playing a guitar*”, while LLaVA-v1.5 fails to recognize the presence of the violin entirely.

Figure 14 illustrates an example of LLaVA-Mini’s capabilities in a more challenging OCR task. The text in the image is presented in an unusual cursive handwriting style, which can significantly hinder the recognition quality of LMMs. For this challenging OCR case, both LLaVA-Mini and GPT-4o accurately identify the text in the image, particularly with LLaVA-Mini using only one vision token. In contrast, Video-LLaVA and LLaVA-v1.5 incorrectly recognize “*Duke*” and “*wedding*”, and erroneously add an exclamation mark “*!*” at the end. Overall, LLaVA-Mini demonstrates superior performance in perceiving and reasoning about visual information.

Video Understanding Figure 15 illustrates an example of LLaVA-Mini’s capabilities in processing longer first-person video. The results show that LLaVA-Mini exhibits a more comprehensive and detailed understanding of the video, effectively capturing entities in the room, such as the yoga mat. In contrast, Video-LLaVA mistakenly imagines “*he takes a shower*” due to its limitation of extracting only 8 frames from the video. Video-ChatGPT provides much shorter responses, lacking some detailed information. Overall, LLaVA-Mini exhibits a superior understanding of the video.

1296
1297
1298
1299
1300
1301
1302
1303
1304
1305

Image:



Instruction:

What is funny about this image? Describe it panel by panel.

1306
1307
1308
1309
1310
1311
1312
1313
1314



LLaVA-Mini: The image is a creative and humorous take on a traditional concert setting, where the musicians are performing on a giant violin, which is a whimsical and unexpected sight. The violin, being a large and buoyant object, is placed on the water, and the musicians are seated on it as if it were a stage. The presence of the gondolas and the water setting adds to the humor, as it's not a typical environment for a concert. The musicians' serious expressions and the formal attire of the performers contrast with the absurdity of the situation, which is likely intended to be a playful and entertaining spectacle. The image plays on the concept of a concert in a unique and imaginative way, using the visual pun of the violin's shape and the water setting to create a humorous and memorable scene.

1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328



GPT-4o: This image shows a humorous and creative scene of a boat shaped like a giant violin floating on a canal, surrounded by traditional gondolas. Here's a panel-by-panel description:
Central Focus: The Violin Boat – The main object of humor is the boat designed to look like a massive violin or stringed instrument. On top of it, there are people standing or possibly playing music. This design turns an ordinary boat into a whimsical, oversized musical instrument.
Surrounding Gondolas – Around the violin-shaped boat, regular gondolas can be seen with people rowing and sitting in them. These gondolas look tiny compared to the giant violin boat, adding to the comedic contrast. The gondoliers and passengers in these smaller boats seem to be watching or escorting the violin boat, enhancing the spectacle.
Water and Context – The canal is typical of places like Venice, known for its waterways and gondolas. The idea of transforming a boat into a musical instrument on such a canal adds to the absurdity and charm of the scene.
 The humor comes from the unusual sight of a giant floating violin on a canal, combining the elegance of music with a quirky, oversized version of an everyday object in a whimsical, real-world setting.

1329
1330
1331
1332



Video-LLaVA: The image is a panel of four pictures that showcase a humorous scene of a boat with a man playing a musical instrument on it. The boat is floating in a body of water, and the man is playing a guitar. The funny aspect of the scene is that the boat is shaped like a guitar, making it a unique and amusing sight. The four pictures capture different moments of the man playing the guitar on the boat, showcasing the creativity and humor of the scene.

1333
1334
1335
1336
1337



LLaVA-v1.5: The image is a panel of a comic strip featuring a man in a boat, paddling with a paddle. The boat is surrounded by other boats, and the man is wearing a tie. The funny aspect of this image is that the man is wearing a tie while paddling, which is an unusual combination of attire for a boat ride. The scene is set in a river, and there are several people in the boats around him. The man's attire and the presence of multiple boats create a humorous and unexpected visual.

1338
1339
1340
1341

Figure 13: Example demonstrating LLaVA-Mini’s image understanding capability on more complex image reasoning. Output marked in red indicates factual errors.

1342
1343


H LIMITATIONS

1344
1345
1346
1347
1348
1349

The limitation of LLaVA-Mini lies in the trade-off between the number of vision tokens (efficiency) and performance. As LLaVA-Mini uses a single vision token, it compresses visual information, which may leave room for improvement in image understanding tasks that involve complex visual content. A potential solution is to slightly increase the number of vision tokens, such as using 16 vision tokens, to achieve a better balance between efficiency and performance. Query-based compression offers flexibility in adjusting the number of vision tokens, as we can simply modify

1350
1351
1352
1353
1354
1355
1356
1357
1358
1359


Image:




Instruction:

What's the text on the cake?


1360

 **LLaVA-Mini:** Congratulations Kate & Luke on your upcoming arrival


1361

 **GPT-4o:** The text on the cake says: "Congratulations Kate & Luke on your upcoming arrival"

1362

 **Video-LLaVA:** The text on the cake reads "Congratulations Kate & **Duke** on your upcoming arrival!"


1363

 **LLaVA-v1.5:** The text on the cake reads "Congratulations Kate & Luke on your upcoming **wedding!**"

1364
1365
1366
1367
1368
1369

1370 Figure 14: Example demonstrating LLaVA-Mini’s image understanding capability on more chal-
1371 lenging OCR task. Output marked in red indicates recognition errors.
1372

1373
1374
1375
1376
1377
1378




3min video


1379

Instruction: Describe what happen in this first person video?


1380

 **LLaVA-Mini:** The first-person video captures a moment of solitude and relaxation, likely in a private space such as a bedroom or a hotel room. The individual is seated comfortably on a chair, engrossed in reading a book, which suggests a moment of leisure or study. The presence of the yoga mat indicates that the person may have been or plans to engage in physical activity, possibly as part of a routine or for relaxation. The room's minimalistic decor and the person's casual attire suggest a setting that is informal and personal. The video does not provide any explicit narrative or context beyond these observations, so any further interpretation would be speculative.

1381
1382
1383
1384
1385
1386
1387
1388

 **Video-LLaVA:** In this video, a man is seen sitting on a chair and reading a book. He then stands up and **walks towards the bathroom where he takes a shower. After that, he is seen sitting on the floor and reading the book again.** The video seems to be a simple yet peaceful depiction of a man's daily routine. The man appears to be calm and relaxed throughout the video, and the setting is minimalistic, with no other characters or objects present. The video's focus is solely on the man and his actions, and the viewer is left to interpret the meaning behind the scenes. Overall, the video is a simple yet intriguing portrayal of a man's daily routine, and it is sure to leave the viewer with a sense of calm and relaxation.

1389
1390
1391
1392
1393
1394
1395
1396
1397

 **Video-ChatGPT:** In the video, a man is seen sitting on a chair and reading a book. He then puts the book down and stands up. The video then shows him walking down a hallway and entering a room.

1398
1399
1400
1401

1402 Figure 15: Example demonstrating LLaVA-Mini’s video understanding capability on first-person
1403 view video. Output marked in red indicates factual errors.

the parameter C to control the number of tokens. In practice, the number of vision tokens can be adjusted based on the specific efficiency requirements of different scenarios, allowing for a trade-off between efficiency and performance. We appreciate your feedback, and will incorporate the limitation and possible solutions in the next version.

I DETAILED RESULTS ON MVBENCH

Table 16 reports the detailed results on each subset of MVBench, corresponding to Table 3.

Table 16: Detailed results on 20 subsets of MVBench.

Spatial	Temporal	mPLUG-Owl	Video-ChatGPT	Video-LLaMA	VideoChat	LLaMA-VID	Video-LLaVA	LLaVA-Mini
Average		29.7	32.7	34.1	35.5	41.4	43.1	44.5
Action	Action Sequence	22.0	23.5	27.5	33.5	63.5	44.5	44.5
	Action Prediction	28.0	26.0	25.5	26.5	42.0	50.0	44.5
	Action Antonym	34.0	62.0	51.0	56.0	26.5	49.0	76.0
	Fine-grained Action	29.0	22.5	29.0	33.5	43.0	42.0	37.0
	Unexpected Action	29.0	26.5	39.0	40.5	42.0	54.5	58.5
Object	Object Existence	40.5	54.0	48.0	53.0	39.0	52.5	50.0
	Object Interaction	27.0	28.0	40.5	40.5	34.5	46.5	50.0
	Object Shuffle	31.5	40.0	38.0	30.0	36.5	40.5	29.5
Position	Moving Direction	27.0	23.0	22.5	25.5	44.0	27.0	31.0
	Action Localization	23.0	20.0	22.5	27.0	35.5	28.5	32.5
Scene	Scene Transition	29.0	31.0	43.0	48.5	22.0	84.5	85.5
Count	Action Count	31.5	30.5	34.0	35.0	44.5	44.5	35.0
	Moving Count	27.0	25.5	22.5	20.5	28.5	26.5	40.0
Attribute	Moving Attribute	40.0	39.5	32.5	42.5	19.0	53.0	48.0
Pose	State Change	44.0	48.5	45.5	46.0	55.6	38.5	41.0
	Fine-grained Pose	24.0	29.0	32.5	26.5	37.5	34.0	29.5
Character	Character Order	31.0	33.0	40.0	41.0	34.0	42.5	52.0
Cognition	Egocentric Navigation	26.0	29.5	30.0	23.5	84.5	32.5	31.0
	Episodic Reasoning	20.5	26.0	21.0	23.5	40.5	38.0	38.0
	Counterfactual Inference	29.5	35.5	37.0	36.0	56.5	32.0	36.0