Appendix

A. Limitations

First, our current performance still falls short of the SOTA models due to limited training data and insufficient model training. Second, while TransV enables processing thousands of frames, the model has not been trained on videos of such duration.

B. Experimental Setups

Implementation details. When training attention-based dropping in multi-turn dialogue scenarios, the attention distribution is computed using the last token of the final instruction as the query. For temporal video grounding data, we incorporate a time-aware prompt [13]: *"The video lasts for {} seconds, and {} frames are uniformly sampled from it."* During training, we randomly sample one instruction from a pool of 15 manually constructed task prompts, such as: *"From the video, locate the portion that aligns with the textual query , and output the start and end timestamps in seconds. The output format of the predicted timestamp should be like: 'start to end' seconds. A specific example is : 12.0 to 20.0 seconds"*. We do not use a system prompt, but we retain the BOS token to act as an attention sink [26].

Training data summary. Our training pipeline adopts a two-stage strategy. We summarize the training data in Section B. Specifically, in the first image-text alignment stage, we utilize 3 million images randomly sampled from the CC12M dataset [7], paired with captions sourced from PixelProse [20]. In the second video instruction tuning stage, we assemble a composite dataset to enhance MLLM's video understanding and timestamp prediction capabilities, comprising: (1) 1.3M samples from LLaVA-Video [31]; (2) 253K data from Kinetics400 [6] and WebVid [4] that are re-captioned with GPT-4o or Gemini by ShareGemini [19] and ShareGPT-4 [8]; (3) 100K samples from ET-Instruct [14]; (4) 112K samples from VideoGPT-Plus [15]; (5) 11K samples from LongVid [13] and MovieChat [21]; (6) 26K dense video captioning (DVC) samples aggregated from ActivityNet [5], COIN [22], HiREST [28], ViTT [11], and YouCook2 [32]; and (7) 250K temporal video grounding (TVG) samples [25] from YT-Temporal [27], DiDeMo [3], QuerYD [17], InternVid [24], and HowTo100M [16]. We obtain grounding data with annotations from VTG-IT [10], TimeIT [18], TimePro [29], HTStep [1], and LongVid [13]. This data collection process yields 339K temporal grounding samples. To ensure data quality, we apply a simple cleaning protocol to the TVG data. Specifically, we filter out coarse-grained samples where the ground truth duration exceeds 30 seconds or spans more than one-third of the total video length. We also discard invalid entries containing out-of-bound timestamps. Consequently, our TVG training data remains 250K.

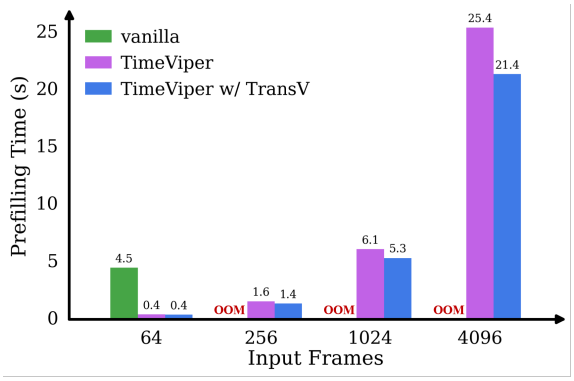


Figure 1. Comparison of prefilling time. TransV incurs no additional latency at low frame inputs (e.g., 64 frames) while significantly reducing prefilling time at high frame inputs. For instance, at 4,096 frames, TransV reduces prefilling time by 15.7% compared to the ToMe baseline.

C. Main Results

Impact of compression components on prefilling time. As shown in Figure 1,the vanilla model already incurs 4.5s latency at 64 frames. TimeViper drastically reduces this to 0.4s, and TransV further decreases prefilling time, with the effect becoming more pronounced as the number of frames increases. Notably, at 4,096 frames, TransV reduces prefilling time by 15.7% compared to TimeViper.

Effect of increasing the number of inference frames. Since the model is trained with 256 frames, we evaluate test-time scalability by varying the number of input frames. As shown in Figure 2, TimeViper scales robustly with longer contexts across four long video understanding benchmarks. For example, when increasing the input frames from 256 to 512 frames, MLVU improves from 65.64 to 69.00, and LVBench increases from 35.53 to 37.0.

Ablation of TransV. In Table 2, we compare the effect of introducing additional TransV parameters with pure token dropping (TD) on downstream tasks under the same setting (uni_7_0.5-attn_39_0.9). The results show that TransV effectively improves performance on most tasks.

Needle in a haystack evaluation. To evaluate TimeViper's ability to handle long videos, we conduct a "needle-in-a-haystack" test: within a video with sequence length varying from 1k to 5k frames, we select 20 VQA samples from MSCOCO and insert the image-based QA task into different depth position in the sequence, and assess whether the model can correctly identify the image content and answer the question. In Section D, the results show that TimeViper with hybrid model achieves an average accuracy of 81.3%, outperforming the Qwen2.5 baseline's 76.7% in the challenging task, demonstrating the effectiveness of hybrid model in long video understanding.

Stage 1: Projector Alignment	
Image caption data (3M)	CC12M (3M) [7] with PixelProse captions [20]
Stage 2: Video Instruction-Tuning	
Image instruction data (2.8M)	LLaVA-OneVision (2.8M) [12];
Video instruction data (1.8M)	LLaVA-Video (1.3M) [31]; Kinetics400 & WebVid (253K) [4, 6] (recaptioned via ShareGemini [19] & ShareGPT-4 [8]); VideoGPT-Plus (112K) [15]; ET-Instruct (100K) [14]; LongVid [13] & MovieChat (11K) [21]
Dense video captioning (26K)	ActivityNet [5], COIN [22], HiREST [28], ViTT [11], YouCook2 [32]
Temporal video grounding (250K)	YT-Temporal [27], DiDeMo [3], QuerYD [17], InternVid [24], HowTo100M [16] (Annotated by VTG-IT [10], TimeIT [18], TimePro [29], HTStep [1], LongVid [13])

Table 1. Data recipe. Overview of the datasets used in our two-stage training pipeline.

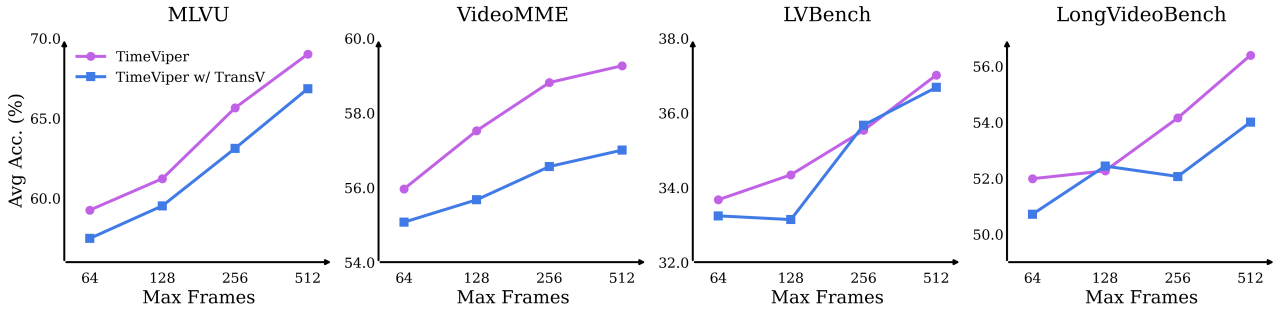


Figure 2. Comparison of performance as the number of input frames increases on long-video understanding benchmarks. We train our models with 256 frames as inputs, and sample 1 frame per second during evaluation. The x-axis here denotes the maximum number of frames. If a video exceeds this length, we take only the first max frames for inference.

Method	Param	MVB avg.acc	LVB val	MLVU avg.acc	VMME overall	LVBench avg.acc	Charades avg.acc	VDC avg.acc
TD	✗	53.7	51.8	63.0	56.8	35.2	37.0	39.4
TransV	✓	56.2	52.0	63.1	56.9	35.6	37.9	39.1

Table 2. Comparison between TransV and token dropping.

Evaluation Settings. We compare uniform sampling and first-frame sampling with 256 frames on VideoMME, where uniform sampling achieves 61.11 versus 58.67 for first-frame sampling on overall accuracy.

D. Visualization of Attention Mechanism

To better understand how hybrid MLLMs differ from Transformer-based MLLMs in processing multimodal inputs, we first formalize the definitions of attention scores used in both Mamba-2 and self-attention layers and then analyze attention behaviors across layers. For Mamba layers, we follow [2] to define the attention pattern, while for Transformer layers, we use the attention weights. Next, we define average attention scores used in both Mamba-2 and self-attention layers to analyze attentions received by different types of tokens.

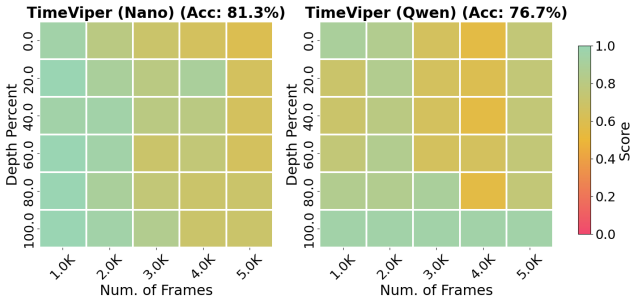


Figure 3. Needle in a haystack evaluation.

Mamba-2 Layer. A Mamba-2 layer is built around a core state-space model (SSM) block, which recurrently maintains a compact hidden state summarizing past information. Let x_t denote the input at step t, and $h_t \in \mathbb{R}^{N \times D}$ the hidden memory. The SSM update is defined as:

$$\begin{aligned} h_t &= A_t h_{t-1} + B_t x_t \\ y_t &= C_t^T h_t \end{aligned} \tag{1}$$

where A_t, B_t, and C_t are discretized SSM parameters [9]. This mechanism encodes temporal dependencies via learnable decay and gating dynamics, enabling efficient informa-

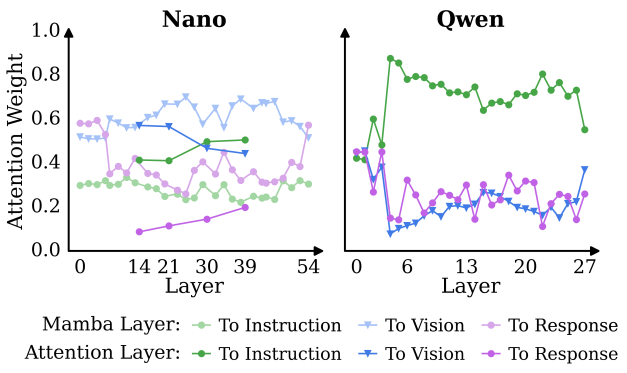


Figure 4. Comparison of average attention scores across all layers in Nanov2 and Qwen2.5. The visualization shows both attention and Mamba layers for Nano, and attention layers for Qwen. For Mamba layers, we normalize each row of the attention scores using the L_1 norm so that all values fall within the range $[0, 1]$.

tion propagation over long sequences.

Self-Attention Layer. In contrast, the self-attention layer directly models token interactions:

$$y = \text{Softmax}(L \odot \frac{QK^T}{\sqrt{D}}) \cdot V \tag{2}$$

where $[Q, K, V] = [W_Q, W_K, W_V]X$, and W_Q, W_K, W_V are learnable parameters. L is the causal attention mask.

Definition of attention score. For self-attention (Equation (2)), the attention score $M_{j,i} \in \mathbb{R}$ from x_j to x_i is:

$$y_i = \text{Softmax}(\frac{Q_i K_{\leq i}^T}{\sqrt{D}}) \cdot V_{\leq i} = \sum_{j=1}^{i} M_{i,j} V_j \tag{3}$$

For the SSM block, we rewrite Equation (1) to express its attention pattern as the weighted sum from inputs $[x_1, \ldots, x_i]$ to the output y_i:

$$y_i = \sum_{j=1}^{i} C_i^T (\prod_{k=j+1}^{i} A_k) B_j x_j = \sum_{j=1}^{i} M'_{i,j} x_j \tag{4}$$

Here, $|M'_{i,j}| \in \mathbb{R}^+$ serves as the "attention score" [2, 33] from x_j to x_i within the SSM block. Although both the self-attention and Mamba mechanisms employ a multi-head design [9, 23] along the hidden dimension, we omit this detail in the equations for simplicity.

Average attention score computation. We adopt the category-level attention score definition from LLaVA-Mini [30]. Tokens are grouped into instruction, vision, and response categories: $\mathcal{T}_{\text{ins}}$, $\mathcal{T}_{\text{vis}}$, and $\mathcal{T}_{\text{res}}$. Let a_{ij} denote the attention score from token t_i to token t_j, averaged over all attention heads. For two token categories $\mathcal{A}, \mathcal{B} \in \{\mathcal{T}_{\text{ins}}, \mathcal{T}_{\text{vis}}, \mathcal{T}_{\text{res}}\}$, we define their category-level attention score as:

$$\text{Attn}(\mathcal{A} \to \mathcal{B}) = \frac{\sum_{t_i \in \mathcal{A}} \sum_{t_j \in \mathcal{B}} a_{ij}}{\left| \left\{ t_i \in \mathcal{A} \mid \sum_{t_j \in \mathcal{B}} a_{ij} > 0 \right\} \right|}. \tag{5}$$

The denominator counts the number of tokens in $\mathcal{A}$ that attend to any token in $\mathcal{B}$ with non-zero weight, ensuring that tokens masked by the causal attention mask are excluded.

In Figure 4, we analyze the overall attention scores from the entire sequence $\mathcal{T} = \mathcal{T}_{\text{ins}} \cup \mathcal{T}_{\text{vis}} \cup \mathcal{T}_{\text{res}}$ to a target category $\mathcal{B}$. To ensure equal contribution, we compute their arithmetic mean:

$$\begin{aligned} \text{Attn}(\mathcal{T} \to \mathcal{B}) = \frac{1}{3} \big(& \text{Attn}(\mathcal{T}_{\text{ins}} \to \mathcal{B}) \\ & + \text{Attn}(\mathcal{T}_{\text{vis}} \to \mathcal{B}) \\ & + \text{Attn}(\mathcal{T}_{\text{res}} \to \mathcal{B}) \big). \end{aligned} \tag{6}$$

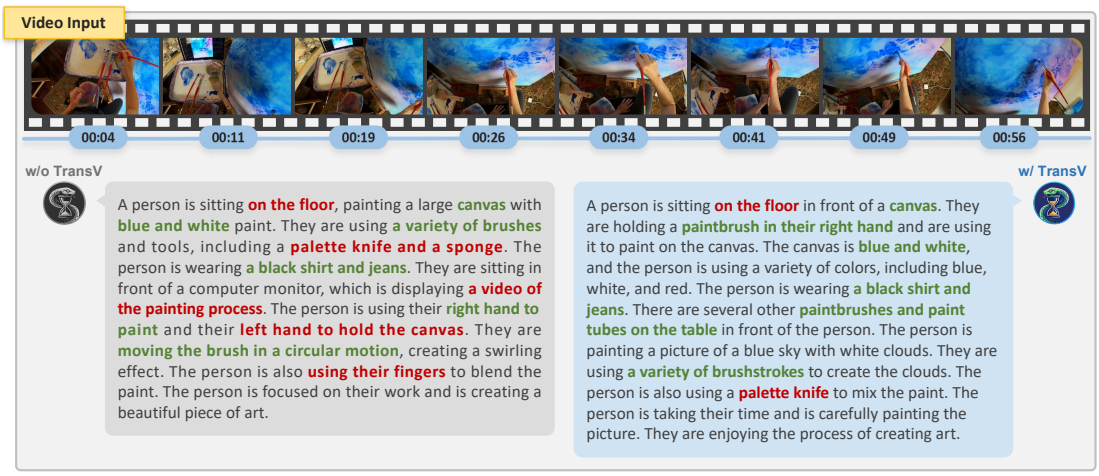
Hybrid MLLMs preserve stronger attention to vision tokens. To quantify model behavior, we compute the average attention received by instruction, vision, and response tokens across all layers. As shown in Figure 4, Qwen rapidly down-weights vision tokens after the early layers, instead favoring instruction and response tokens. In contrast, Nano maintains noticeably higher attention to vision tokens throughout the network. These findings suggest that the hybrid model is more effective at attending to visual information than the Transformer-based architecture.

E. Qualitative Results

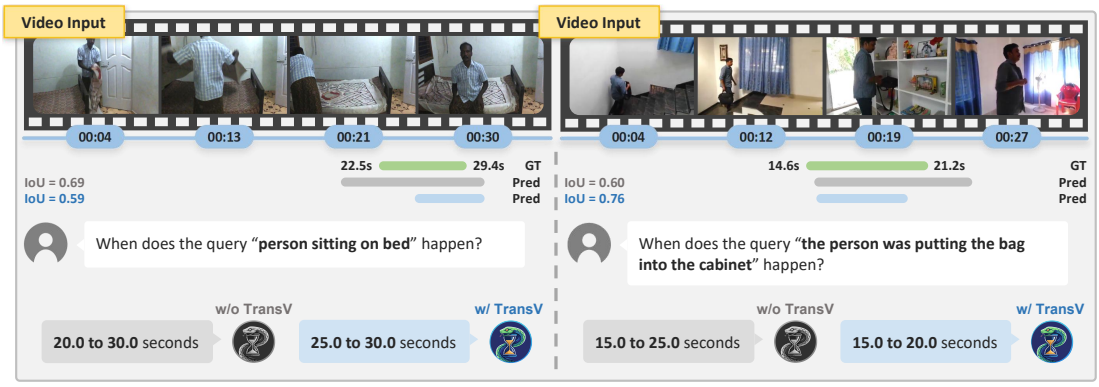
Qualitative results on VideoMME. Figure 5c illustrates TransV's effectiveness on MCQ tasks. In the first case that shows fine-grained retrieval, TimeViper w/ TransV successfully attends to a critical frame (03:36) to answer questions about the Berlin Wall. In the second case that requires long-term temporal reasoning, it correctly deduces the chronological order of a biology lecture by accurately aligning textual concepts with diverse temporal segments.

Qualitative results on Charades. For temporal video grounding in Figure 5b, we observe that incorporating the compression module yields only minimal changes. Both the original and compressed models accurately interpret timestamps and localize the corresponding video segments.

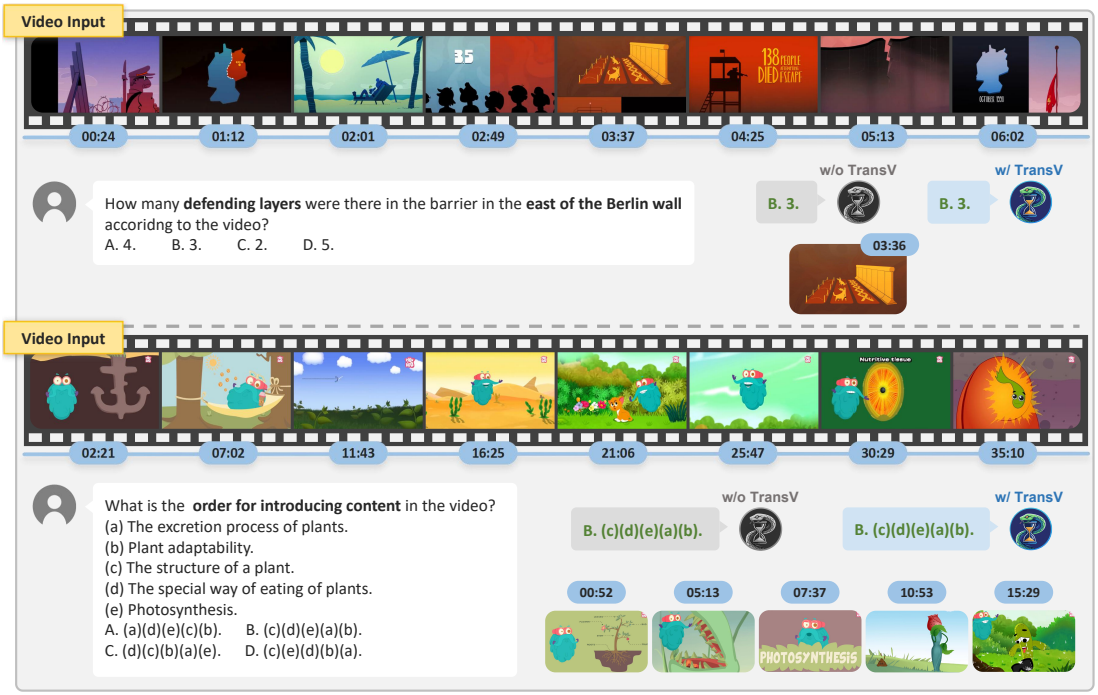
Qualitative results on VDC. Figure 5a presents detailed captioning results where green represents accurate details and red illustrates hallucinations. While the baseline suffers from object hallucination, e.g., fabricating a "sponge", TimeViper w/ TransV generates more faithful descriptions e.g., recognizing "paintbrushes". This suggests that compression may help reduce hallucination by filtering out irrelevant or misleading visual information, while largely maintaining the original model's descriptive behavior.



(a) Qualitative results on VDC.



(b) Qualitative results on Charades.



(c) Qualitative results on VideoMME.

Figure 5. Qualitative results on three benchmarks.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (No. 62576347) and the Outstanding Innovative Talents Cultivation Funded Programs 2025 of Renmin University of China.

References

[1] Triantafyllos Afouras, Effrosyni Mavroudi, Tushar Nagarajan, Huiyu Wang, and Lorenzo Torresani. Ht-step: Aligning instructional articles with how-to videos. *Advances in Neural Information Processing Systems*, 36:50310–50326, 2023. 1, 2

[2] Ameen Ali Ali, Itamar Zimerman, and Lior Wolf. The hidden attention of mamba models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1516–1534, Vienna, Austria, 2025. Association for Computational Linguistics. 2, 3

[3] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *Proceedings of the IEEE international conference on computer vision*, pages 5803–5812, 2017. 1, 2

[4] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1728–1738, 2021. 1, 2

[5] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 961–970, 2015. 1, 2

[6] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 1, 2

[7] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3558–3568, 2021. 1, 2

[8] Lin Chen, Xilin Wei, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Zhenyu Tang, Li Yuan, et al. Sharegpt4video: Improving video understanding and generation with better captions. *Advances in Neural Information Processing Systems*, 37:19472–19495, 2024. 1, 2

[9] Tri Dao and Albert Gu. Transformers are ssms: generalized models and efficient algorithms through structured state space duality. In *Proceedings of the 41st International Conference on Machine Learning*. JMLR.org, 2024. 2, 3

[10] Yongxin Guo, Jingyu Liu, Mingda Li, Dingxin Cheng, Xiaoying Tang, Dianbo Sui, Qingbin Liu, Xi Chen, and Kevin Zhao. Vtg-llm: Integrating timestamp knowledge into video llms for enhanced video temporal grounding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3302–3310, 2025. 1, 2

[11] Gabriel Huang, Bo Pang, Zhenhai Zhu, Clara Rivera, and Radu Soricut. Multimodal pretraining for dense video captioning. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 470–490, Suzhou, China, 2020. Association for Computational Linguistics. 1, 2

[12] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *Transactions on Machine Learning Research*, 2024. 2

[13] Xinhao Li, Yi Wang, Jiashuo Yu, Xiangyu Zeng, Yuhan Zhu, Haian Huang, Jianfei Gao, Kunchang Li, Yinan He, Chenting Wang, et al. Videochat-flash: Hierarchical compression for long-context video modeling. *arXiv preprint arXiv:2501.00574*, 2024. 1, 2

[14] Ye Liu, Zongyang Ma, Zhongang Qi, Yang Wu, Chang Wen Chen, and Ying Shan. E.t. bench: Towards open-ended event-level video-language understanding. In *Neural Information Processing Systems (NeurIPS)*, 2024. 1, 2

[15] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Videogpt+: Integrating image and video encoders for enhanced video understanding. *arxiv*, 2024. 1, 2

[16] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2630–2640, 2019. 1, 2

[17] Andreea-Maria Oncescu, Joao F Henriques, Yang Liu, Andrew Zisserman, and Samuel Albanie. Queryd: A video dataset with high-quality text and audio narrations. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2265–2269. IEEE, 2021. 1, 2

[18] Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and

Lu Hou. Timechat: A time-sensitive multimodal large language model for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14313–14323, 2024. 1, 2

[19] Share. Sharegemini: Scaling up video caption data for multimodal large language models, 2024. 1, 2

[20] Vasu Singla, Kaiyu Yue, Sukriti Paul, Reza Shirkavand, Mayuka Jayawardhana, Alireza Ganjdanesh, Heng Huang, Abhinav Bhatele, Gowthami Somepalli, and Tom Goldstein. From pixels to prose: A large dataset of dense image captions. *arXiv preprint arXiv:2406.10328*, 2024. 1, 2

[21] Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, et al. Moviechat: From dense token to sparse memory for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18221–18232, 2024. 1, 2

[22] Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. Coin: A large-scale dataset for comprehensive instructional video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1207–1216, 2019. 1, 2

[23] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3

[24] Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, Ping Luo, Ziwei Liu, Yali Wang, Limin Wang, and Yu Qiao. Internvid: A large-scale video-text dataset for multimodal understanding and generation. In *The Twelfth International Conference on Learning Representations*, 2024. 1, 2

[25] Ye Wang, Ziheng Wang, Boshen Xu, Yang Du, Kejun Lin, Zihan Xiao, Zihao Yue, Jianzhong Ju, Liang Zhang, Dingyi Yang, Xiangnan Fang, Zewen He, Zhenbo Luo, Wenxuan Wang, Junqi Lin, Jian Luan, and Qin Jin. Time-r1: Post-training large vision language model for temporal video grounding. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. 1

[26] Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *The Twelfth International Conference on Learning Representations*, 2024. 1

[27] Antoine Yang, Arsha Nagrani, Paul Hongsuck Seo, Antoine Miech, Jordi Pont-Tuset, Ivan Laptev, Josef Sivic, and Cordelia Schmid. Vid2seq: Large-scale pretraining of a visual language model for dense video captioning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10714–10726, 2023. 1, 2

[28] Abhay Zala, Jaemin Cho, Satwik Kottur, Xilun Chen, Barlas Oguz, Yashar Mehdad, and Mohit Bansal. Hierarchical video-moment retrieval and step-captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23056–23065, 2023. 1, 2

[29] Xiangyu Zeng, Kunchang Li, Chenting Wang, Xinhao Li, Tianxiang Jiang, Ziang Yan, Songze Li, Yansong Shi, Zhengrong Yue, Yi Wang, Yali Wang, Yu Qiao, and Limin Wang. Timesuite: Improving MLLMs for long video understanding via grounded tuning. In *The Thirteenth International Conference on Learning Representations*, 2025. 1, 2

[30] Shaolei Zhang, Qingkai Fang, Zhe Yang, and Yang Feng. LLaVA-mini: Efficient image and video large multimodal models with one vision token. In *The Thirteenth International Conference on Learning Representations*, 2025. 3

[31] Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun MA, Ziwei Liu, and Chunyuan Li. LLaVA-video: Video instruction tuning with synthetic data. *Transactions on Machine Learning Research*, 2025. 1, 2

[32] Luowei Zhou, Chenliang Xu, and Jason Corso. Towards automatic learning of procedures from web instructional videos. In *Proceedings of the AAAI conference on artificial intelligence*, 2018. 1, 2

[33] Itamar Zimerman, Ameen Ali, and Lior Wolf. Explaining modern gated-linear rnns via a unified implicit attention formulation. In *ICLR*, 2025. 3