LION-FS: Fast & Slow Video-Language Thinker as Online Video Assistant

Supplementary Material

A. Experimental Detials

A.1. Architecture

We employ two visual encoders: SigLIP and EgoVLPv2. SigLIP represents the SigLIP-large-patch16-384 model, pre-trained on the WebLi dataset at a resolution of 384×384. EgoVLPv2 is the second generation of egocentric video-language pre-training models, trained on the EgoClip version of the Ego4D dataset at a resolution of 224×224. Due to the inconsistency in token dimensions produced by these visual encoders, each set of tokens is processed through an MLP to align with the dimensionality of text tokens. The Token Aggregation Router comprises an MLP and a SoftMax layer. At the frame level, it takes the `[VG]` token (i.e., the CLS token extracted by SigLIP) as input, assigning aggregation weights to all tokens derived from both visual encoders for each frame. Additionally, the Token Dropping Router employs an MLP and a SoftMax layer at the token level, assigning confidence scores to individual visual tokens. Tokens with confidence scores below a pre-defined discard threshold are deemed redundant and are subsequently dropped. For the large language model (LLM), we leverage Llama-3-8B-Instruct [4], an optimized variant tailored for conversational tasks. Our online video dialogue template adheres to the instruction-tuning paradigm, extending input to encompass a multimodal fusion of interleaved visual and textual elements. In the Slow Path of LION-FS, we adopt a Faster-RCNN object detection model to detect hands and hand-interacting objects, instead of using a DETR-based model [1], to ensure real-time processing of video frames.

A.2. Data Refinement

- **Grammar Correction.** For Ego-Exo4D, we reorganized and refined the annotations of short-term atomic descriptions by substituting third-person verb forms following "C" with the second-person pronoun "You" and the base verb form, e.g., modifying "C stands in a house" to "You stand in a house." Capital letters were preserved to denote other individuals (i.e., those not wearing the camera), such as in "Lady X and man M prepare concrete in a basin," to clearly differentiate among entities. For the narrations of Ego4D, we removed markers like "# C" (camera wearer), "# O" (other individuals), and "# Unsure" (uncertain narration) preceding each statement, maintaining consistency with the annotation strategy applied to Ego-Exo4D.
- **Dialogue Data Augmentation.** We employ modified data augmentation strategies inspired by VideoLLM-online and VideoLLM-MoD: **1)** Replace a learning message with incorrect content, then correct or leave it uncorrected to train the model's ability to identify errors and respond appropriately despite misinformation. **2)** Introduce temporal inconsistencies by inserting, deleting, or replacing frames to simulate real-world frame sequence variations. **3)** Use empty strings, None, or remove messages to emulate scenarios involving missing or incomplete responses.
- **Dual Visual Features Alignment.** We employ two encoders for visual feature extraction: SigLIP extracts image tokens from frames sampled at 2 FPS, while EgoVLPv2 processes frames sampled at 8 FPS by grouping them into sets of four and extracting video tokens at 2 groups per second. To achieve temporal alignment across different frame rates, we trim the final segment of the video shorter than one second and remove any annotations exceeding the new maximum duration.

A.3. Training Settings

All experiments are conducted using 8 × A800 80GB GPUs. We train the full MLP, Token Aggregation Router, Token Dropping Router, and LoRA [2] embedded in each linear layer of the LLM. The batch size is set to 1 per GPU, with training conducted for 10 epochs on the Ego-Exo4D dataset and 2 epochs on the Ego4D dataset. Gradient accumulation over 32 steps is used to achieve an effectively larger batch size. We employ the AdamW [3] optimizer with an initial learning rate of 0.0002, using cosine learning rate scheduling with a 5% warmup ratio.

A.4. Evaluation Metrics

- **Language Modeling Perplexity (LM-PPL)** measures the quality of a model's probabilistic distribution in language modeling. It provides an indirect evaluation by assessing the probabilities assigned to each generated token. A lower LM-PPL typically indicates a stronger language modeling capability of the model.
- **LM-Correctness** assesses the precision of the model's language generation by focusing on the degree of alignment between generated text and reference text. By calculating the proportion of correctly generated tokens within the language sequence, it reflects the model's actual performance in generative tasks.
- **Time Difference (TimeDiff)** evaluates the model's real-time processing and temporal alignment capabilities in

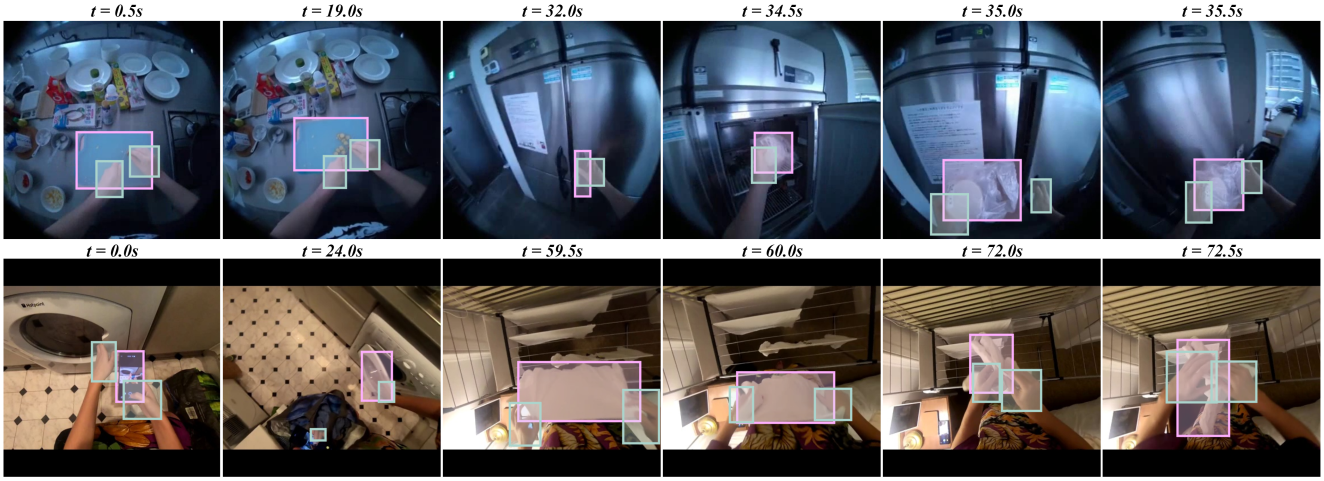

Figure 1. Visualization of Local Adaptive Augmentation in the Slow Path on Ego-Exo4D and Ego4D datasets. Green bounding boxes highlight the user's hands in the first-person view, while pink bounding boxes indicate the objects interacting with the hands. Experiments demonstrate that the object interacting with the user's hands is often the same, and retaining only a single interaction object's bounding box effectively reduces false detections.

handling video stream inputs. It is computed as the difference between the timestamp of the response occurrence and the expected timestamp. The average TimeDiff across each dialogue turn is used as the metric.

- **Fluency** assesses the integration of visual information and the naturalness and coherence of generated text. It calculates the proportion of successfully predicted tokens throughout the dialogue turns, including both response determination and response generation predictions, providing a comprehensive evaluation of language and temporal efficiency in video online dialogue.

B. Additional Experimental Analysis

B.1. Additional Efficiency Evaluation

The Fast Path of LION-FS incorporates two distinct visual encoders, enabling the aggregation of different visual tokens (using the Token Aggregation Router) and dropping redundant ones (using the Token Dropping Router), thereby enhancing the efficiency of processing high framerate videos. The Slow Path distinguishes keyframes (determined as responsive frames) from ordinary frames (determined as silent frames), and applies augmentation only to keyframes, minimizing efficiency impact. As shown in Table 1, VideoLLM-online and VideoLLM-MoD **1)** merely add tokens to every frame without optimizing specifically for keyframes, and **2)** as the input frame rate increases to 8 FPS, their inability to aggregate tokens leads to cumulative token growth, resulting in a significant rise in FLOPs. Additionally, when no frame augmentation is performed across all methods, LION-FS achieves FLOPs comparable to VideoLLM-MoD while supporting input at 8 FPS.

Table 1. **FLOPs evaluation under the same test sample in Ego-Exo4D.** Since the code for VideoLLM-MoD has not been released, VideoLLM-MoD* is reproduced based on its paper. Both LION-FS and VideoLLM-Mod* adopt a strategy of dropping interleaved layers with a dropping ratio of $\beta = 0.5$. Leveraging the two customized routers and the keyframe augmentation strategy, LION-FS significantly reduces FLOPs even at an input frame rate of 8 FPS. Without frame augmentation, LION-FS maintains nearly constant FLOPs while supporting 8 FPS input.

Method	Aug. Strategy	Input FPS	FLOPs
VideoLLM-online	All frames	2	55.65T
VideoLLM-online	None	8	59.49T
VideoLLM-MoD*	All frames	2	47.00T
VideoLLM-MoD*	None	8	50.29T
LION-FS	Keyframes	8	**21.53T**
VideoLLM-online	None	2	15.45T
VideoLLM-MoD*	None	2	**12.28T**
LION-FS	None	8	12.40T

B.2. Local Adaptive Augmentation for Box Tokens

The primary task of an online assistant in first-person scenes is to engage in real-time dialogue with the user regarding their current actions, focusing on the interaction between the user and the environment. However, these interaction areas often constitute a small portion of the total frame in first-person videos (especially from the Ego-Exo4D & Ego4D datasets), leading to attention dispersion in MLLMs. We address this by using bounding boxes, as shown in Figure 1, to highlight the user's hands and their interaction with the environment, guiding the LLM to focus on these areas and improving response precision. We first filter out the patch tokens covered by the bounding boxes in each frame, and then apply global pooling to the tokens within each bounding box to obtain a single representation per box, termed as the Box Token. We define up to three Box Tokens per frame, corresponding to the user's two hands and an interacting object. When hands are absent or missed in detection, we replace them with a global pooling token of the frame to maintain a consistent number of tokens. In fact, when interaction regions are absent, focusing more on the global scene is often necessary.

D. Limitations

The design of loss functions for online video dialogue modeling is hindered by the negative impact of long-tailed distributions, as the frequency of `[EOS]` occurrences significantly exceeds that of `[Assistant]` when determining response generation. Consequently, the model tends to favor predicting silence during training. To address this, we propose transforming the binary decision task into a multi-class prediction (using discrete special tokens to predict varying response probabilities) or a response probability regression task, thereby mitigating the effects of the long-tailed distribution. Additionally, we aim to introduce a fixed-length memory mechanism in LION-FS to replace the Key-Value Cache, prioritizing recent context while retaining essential historical information. This approach ensures computational efficiency, thereby enabling the handling of unlimited video lengths in online video dialogue.

E. Societal Impact and Potential Risk

LION-FS is fine-tuned on large language models (LLMs) using the Ego-Exo4D and Ego4D datasets. Given the potential for LLMs to generate hallucinations or biased responses, and the inherent unreliability of annotations in these datasets, caution is advised when deploying LION-FS as an online video assistant. Its responses should be critically evaluated, and comprehensive safety and fairness assessments are essential before practical deployment.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 1

[2] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 1

[3] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 1

[4] meta. Introducing meta llama 3: The most capable openly available llm to date. In *https://ai.meta.com/blog/meta-llama-3/*, 2024. 1