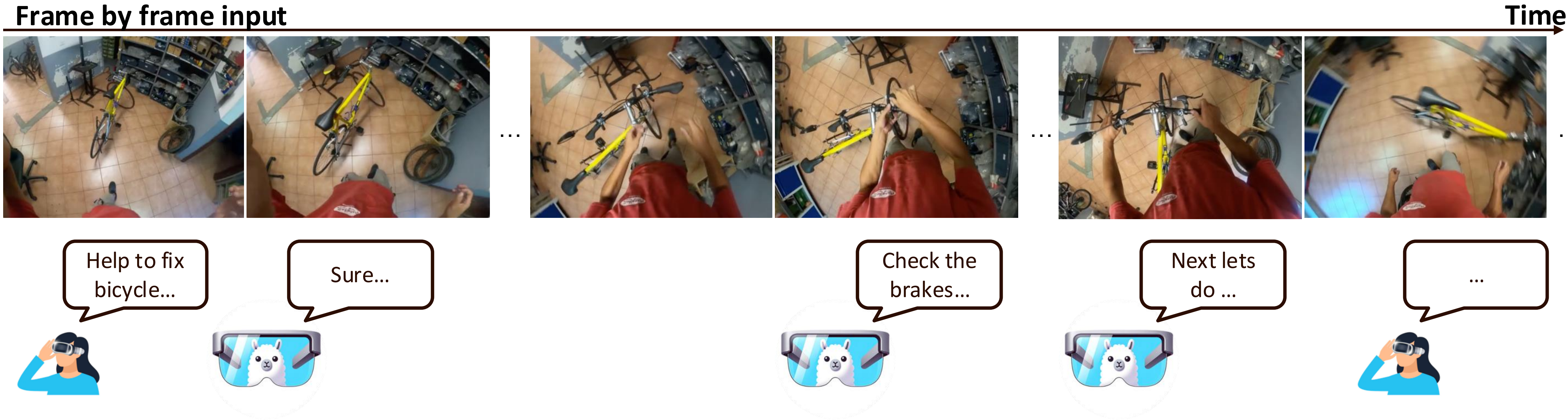
VideoLLM-online: Online Video Large Language Model for Streaming Video

陈卓 (Joya)

PhD Candidate @ Show Lab, National University of Singapore

Motivation

Idea in Meta Ego-Exo4D Data Collection: Assume we have an AI assistant on the glasses like J.A.R.V.I.S…



Frame by frame input

Time

…

…

…

Help to fix bicycle…

Sure…

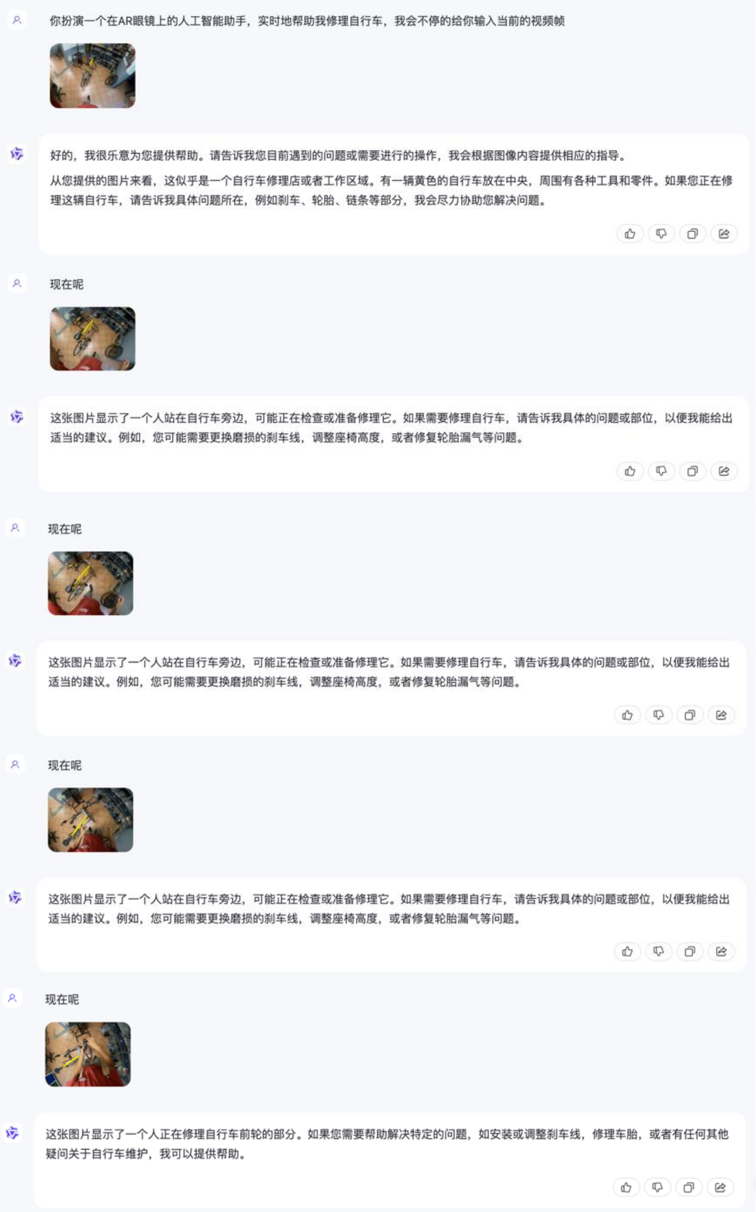
Check the brakes…

Next lets do …

…

Motivation

Popular LMMs work in an interleaved mode, not optimal for streaming input



你扮演一个在AR眼镜上的人工智能助手，实时地帮助我修理自行车，我会不停的给你输入当前的视频帧

好的，我很乐意为您提供帮助。请告诉我您目前遇到的问题或需要进行的操作，我会根据图像内容提供相应的指导。

从您提供的图片来看，这似乎是一个自行车修理店或者工作区域。有一辆黄色的自行车放在中央，周围有各种工具和零件。如果您正在修理这辆自行车，请告诉我具体问题所在，例如刹车、轮胎、链条等部分，我会尽力协助您解决问题。

现在呢

这张图片显示了一个人站在自行车旁边，可能正在检查或准备修理它。如果需要修理自行车，请告诉我具体的问题或部位，以便我能给出适当的建议。例如，您可能需要更换磨损的刹车线，调整座椅高度，或者修复轮胎漏气等问题。

现在呢

这张图片显示了一个人站在自行车旁边，可能正在检查或准备修理它。如果需要修理自行车，请告诉我具体的问题或部位，以便我能给出适当的建议。例如，您可能需要更换磨损的刹车线，调整座椅高度，或者修复轮胎漏气等问题。

现在呢

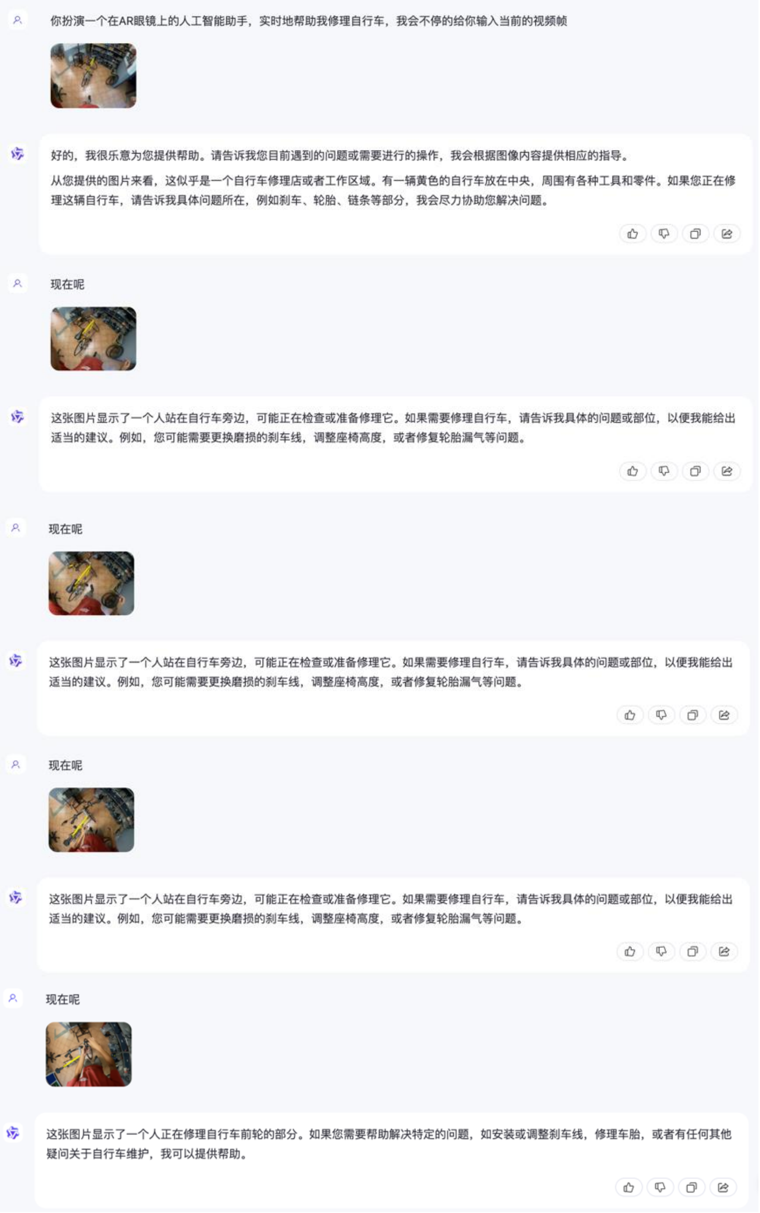
这张图片显示了一个人站在自行车旁边，可能正在检查或准备修理它。如果需要修理自行车，请告诉我具体的问题或部位，以便我能给出适当的建议。例如，您可能需要更换磨损的刹车线，调整座椅高度，或者修复轮胎漏气等问题。

现在呢

这张图片显示了一个人正在修理自行车前轮的部分。如果您需要帮助解决特定的问题，如安装或调整刹车线，修理车胎，或者有任何其他疑问关于自行车维护，我可以提供帮助。

Motivation

Popular LMMs work in an interleaved mode, not optimal for streaming input



你扮演一个在AR眼镜上的人工智能助手，实时地帮助我修理自行车，我会不停的给你输入当前的视频帧

好的，我很乐意为您提供帮助。请告诉我您目前遇到的问题或需要进行的操作，我会根据图像内容提供相应的指导。

从您提供的图片来看，这似乎是一个自行车修理店或者工作区域。有一辆黄色的自行车放在中央，周围有各种工具和零件。如果您正在修理这辆自行车，请告诉我具体问题所在，例如刹车、轮胎、链条等部分，我会尽力协助您解决问题。

现在呢

这张图片显示了一个人站在自行车旁边，可能正在检查或准备修理它。如果需要修理自行车，请告诉我具体的问题或部位，以便我能给出适当的建议。例如，您可能需要更换磨损的刹车线，调整座椅高度，或者修复轮胎漏气等问题。

现在呢

这张图片显示了一个人站在自行车旁边，可能正在检查或准备修理它。如果需要修理自行车，请告诉我具体的问题或部位，以便我能给出适当的建议。例如，您可能需要更换磨损的刹车线，调整座椅高度，或者修复轮胎漏气等问题。

现在呢

这张图片显示了一个人站在自行车旁边，可能正在检查或准备修理它。如果需要修理自行车，请告诉我具体的问题或部位，以便我能给出适当的建议。例如，您可能需要更换磨损的刹车线，调整座椅高度，或者修复轮胎漏气等问题。

现在呢

这张图片显示了一个人正在修理自行车前轮的部分。如果您需要帮助解决特定的问题，如安装或调整刹车线，修理车胎，或者有任何其他疑问关于自行车维护，我可以提供帮助。

Why not optimal:

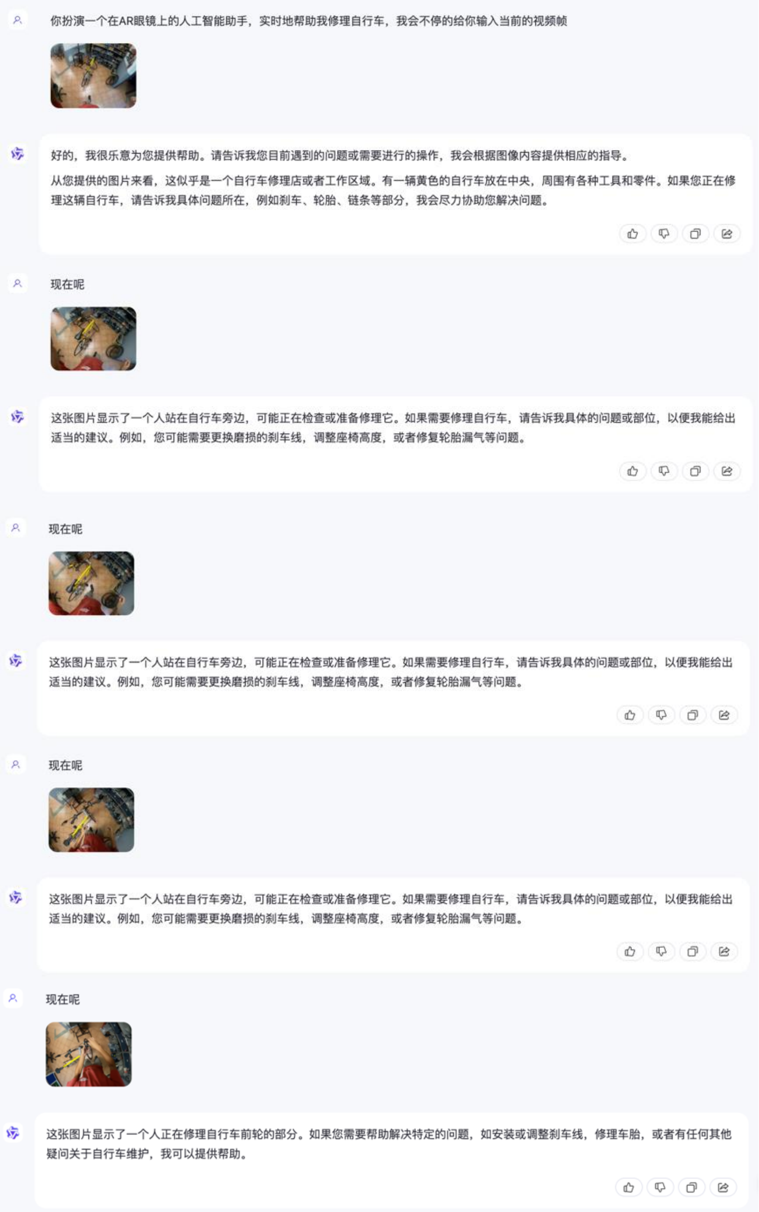
1. **Not active: Rely on user query input rather than actively assist**

2. **Not long-term: Dense per-frame response costs too many tokens**

3. **Not real-time: Per-frame language generation is slow**

Motivation

Popular LMMs work in an interleaved mode, not optimal for streaming input



你扮演一个在AR眼镜上的人工智能助手，实时地帮助我修理自行车，我会不停的给你输入当前的视频帧

好的，我很乐意为您提供帮助。请告诉我您目前遇到的问题或需要进行的操作，我会根据图像内容提供相应的指导。

从您提供的图片来看，这似乎是一个自行车修理店或者工作区域。有一辆黄色的自行车放在中央，周围有各种工具和零件。如果您正在修理这辆自行车，请告诉我具体问题所在，例如刹车、轮胎、链条等部分，我会尽力协助您解决问题。

现在呢

这张图片显示了一个人站在自行车旁边，可能正在检查或准备修理它。如果需要修理自行车，请告诉我具体的问题或部位，以便我能给出适当的建议。例如，您可能需要更换磨损的刹车线，调整座椅高度，或者修复轮胎漏气等问题。

现在呢

这张图片显示了一个人站在自行车旁边，可能正在检查或准备修理它。如果需要修理自行车，请告诉我具体的问题或部位，以便我能给出适当的建议。例如，您可能需要更换磨损的刹车线，调整座椅高度，或者修复轮胎漏气等问题。

现在呢

这张图片显示了一个人站在自行车旁边，可能正在检查或准备修理它。如果需要修理自行车，请告诉我具体的问题或部位，以便我能给出适当的建议。例如，您可能需要更换磨损的刹车线，调整座椅高度，或者修复轮胎漏气等问题。

现在呢

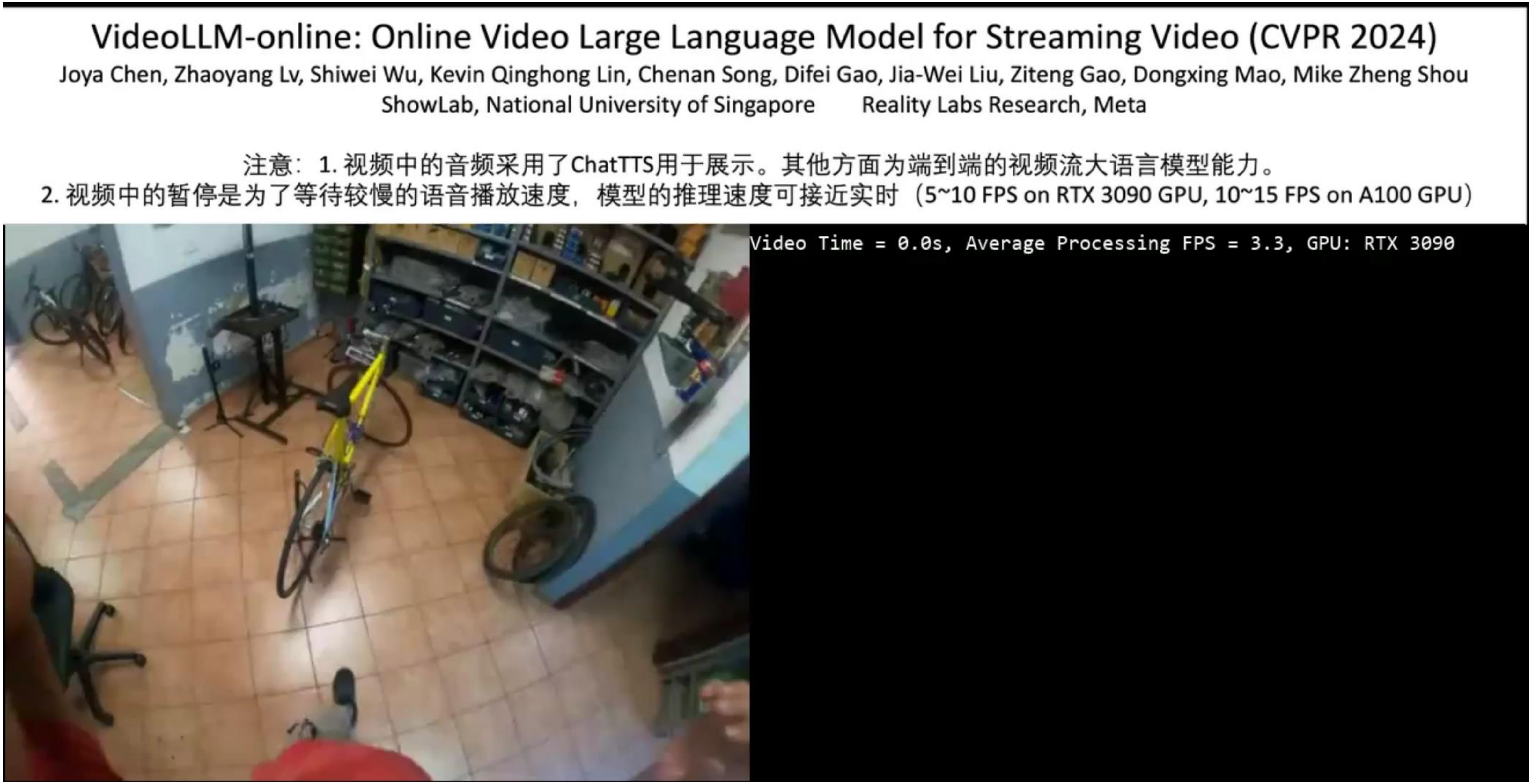
这张图片显示了一个人正在修理自行车前轮的部分。如果您需要帮助解决特定的问题，如安装或调整刹车线，修理车胎，或者有任何其他疑问关于自行车维护，我可以提供帮助。

Why not optimal:

1. **Not active: Rely on user query input rather than actively assist**
2. **Not long-term: Dense per-frame response costs too many tokens**
3. **Not real-time: Per-frame language generation is slow**

Per-frame ❌
When to speak ✅

Demo (audio supported by ChatTTS)



VideoLLM-online: Online Video Large Language Model for Streaming Video (CVPR 2024)

Joya Chen, Zhaoyang Lv, Shiwei Wu, Kevin Qinghong Lin, Chenan Song, Difei Gao, Jia-Wei Liu, Ziteng Gao, Dongxing Mao, Mike Zheng Shou

ShowLab, National University of Singapore　Reality Labs Research, Meta

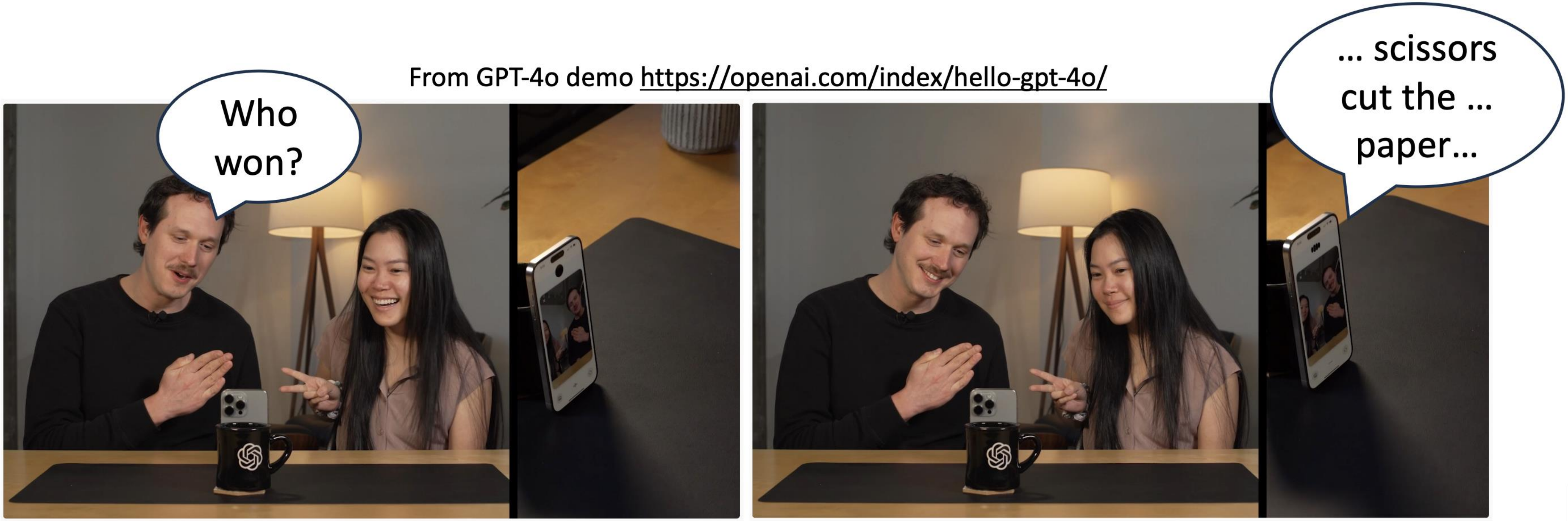
注意：1. 视频中的音频采用了ChatTTS用于展示。其他方面为端到端的视频流大语言模型能力。

2. 视频中的暂停是为了等待较慢的语音播放速度，模型的推理速度可接近实时（5~10 FPS on RTX 3090 GPU, 10~15 FPS on A100 GPU）

```
Video Time = 0.0s, Average Processing FPS = 3.3, GPU: RTX 3090
```

Streaming in GPT-4o?

GPT-4o also needs active audio input to assist vision problems

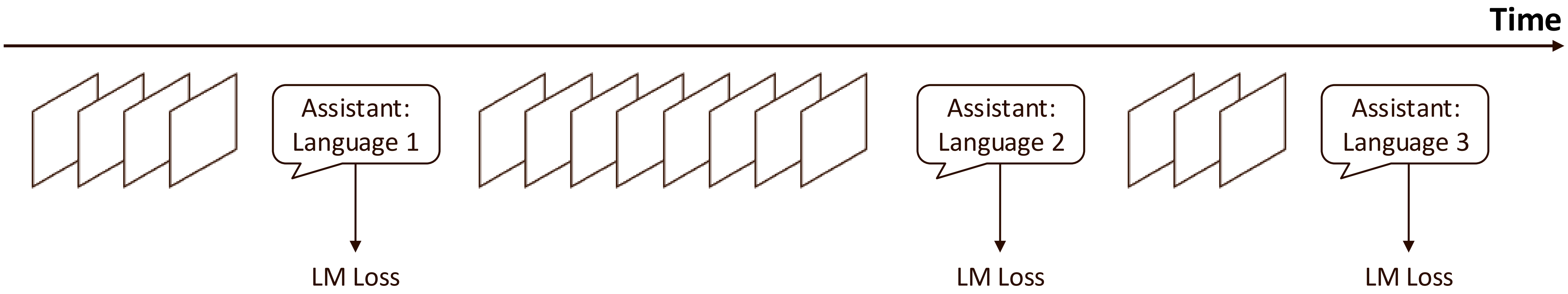


From GPT-4o demo https://openai.com/index/hello-gpt-4o/

Model Method

Review interleaved vision-language modeling (Ignore user query for simplicity)

1. **Training**



Time

Assistant: Language 1

Assistant: Language 2

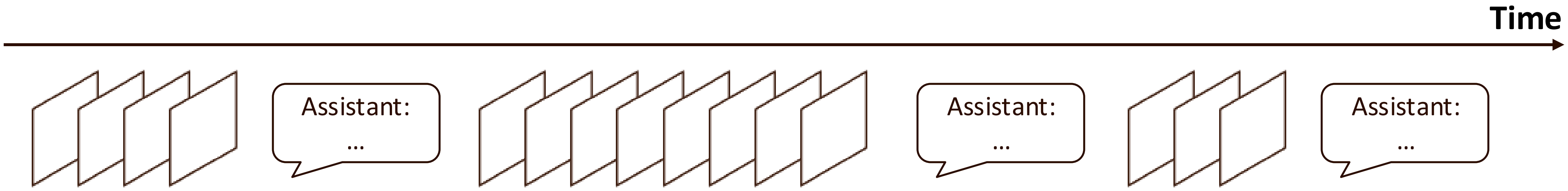
Assistant: Language 3

LM Loss

LM Loss

LM Loss

2. **Inference**



Time

Assistant: …

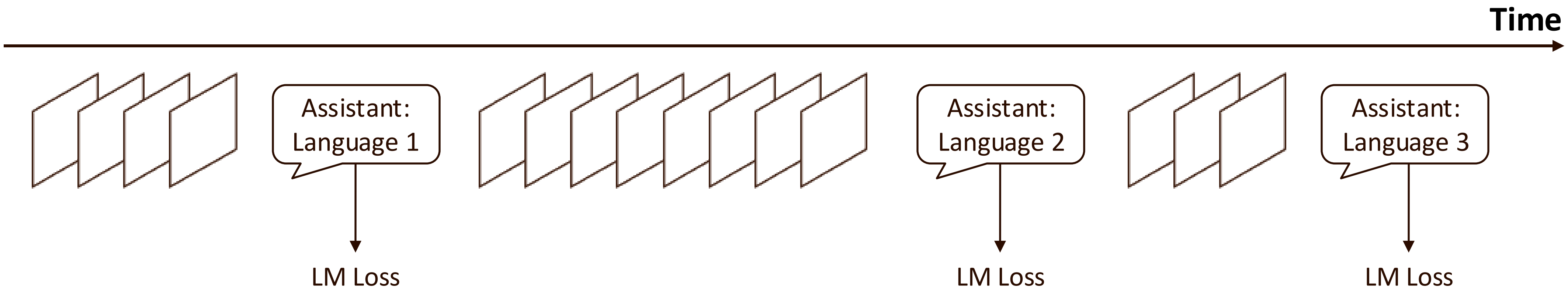
Assistant: …

Assistant: …

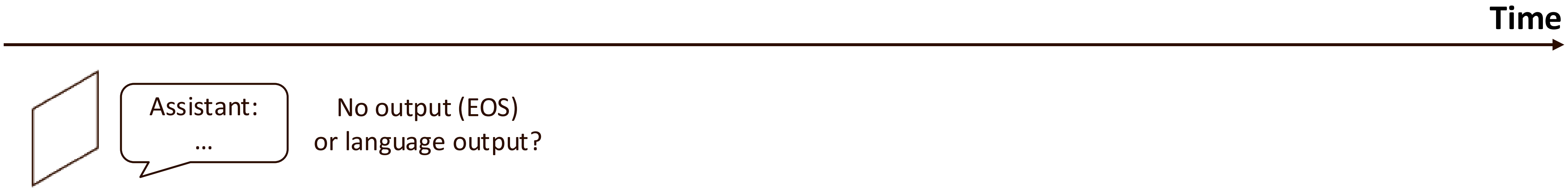
Model Method

Interleaved vision-language modeling for streaming input? (Ignore user query for simplicity)

1. Training



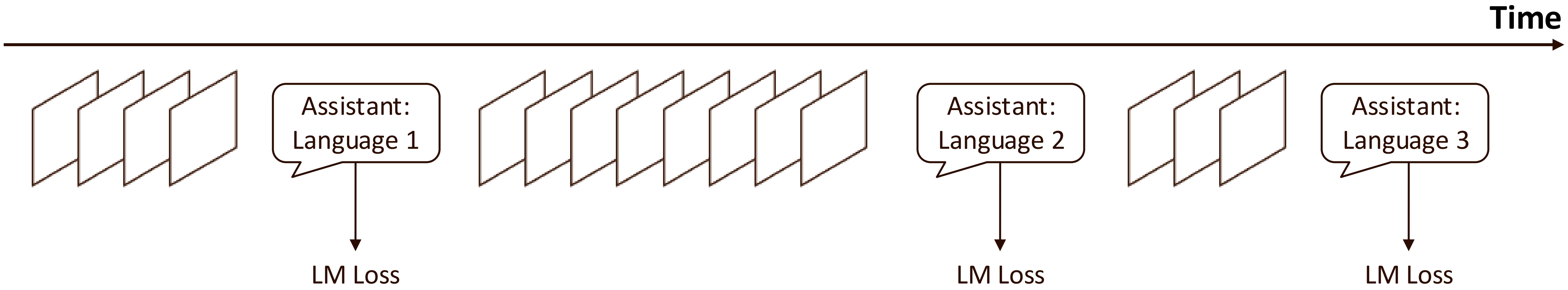
2. Streaming Inference



Model Method

The naïve streaming vision-language modeling as multi-turn conversation

1. Interleaved Training



Time

Assistant: Language 1

LM Loss

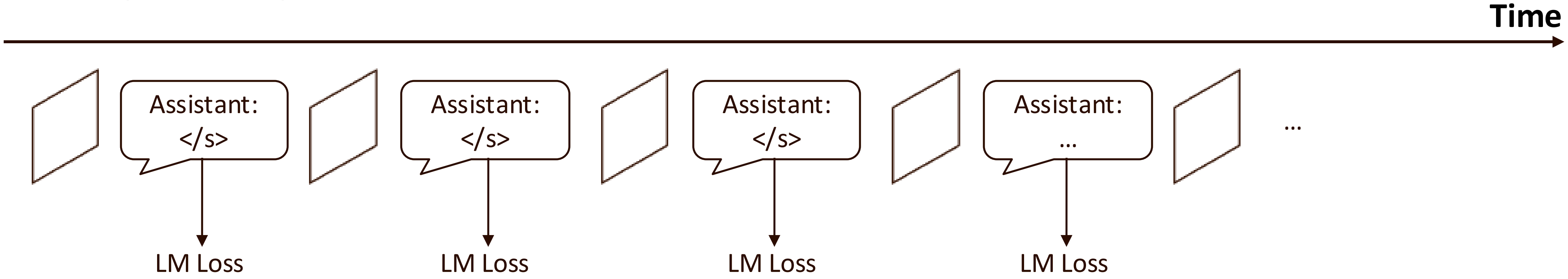
Assistant: Language 2

LM Loss

Assistant: Language 3

LM Loss

2. Streaming Training



Time

Assistant: </s>

LM Loss

Assistant: </s>

LM Loss

Assistant: </s>

LM Loss

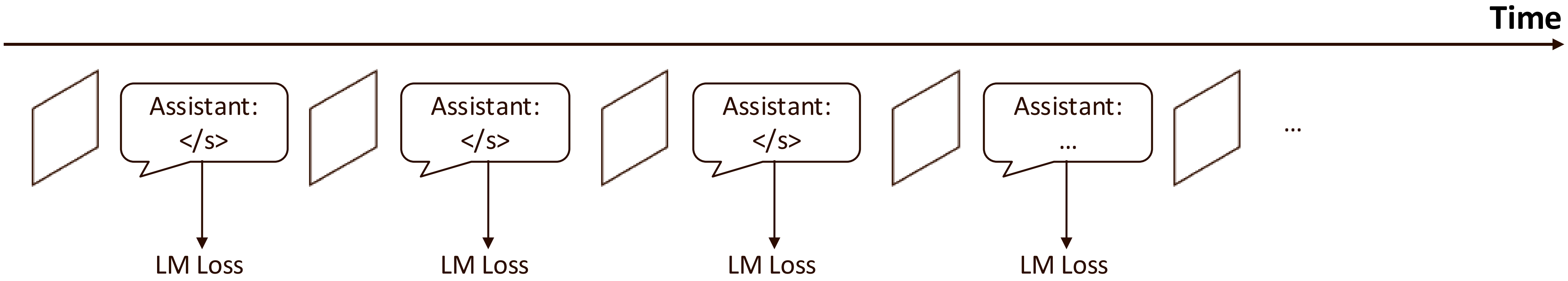
Assistant: …

LM Loss

…

Model Method

Dense streaming frames cost too many unnecessary tokens from conversation template



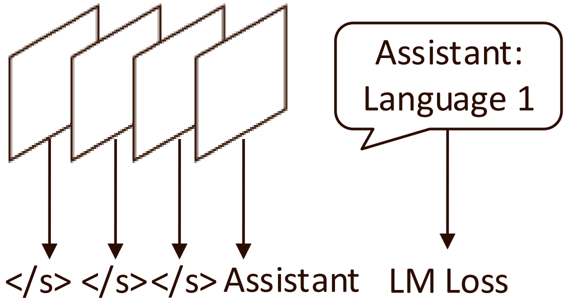
<|start_header_id|>assistant<|end_header_id|>\n\n<|eot_id|>

5 extra tokens by Llama-3 Tokenizer for every frame

30 FPS, 1min -> 9000 extra tokens!

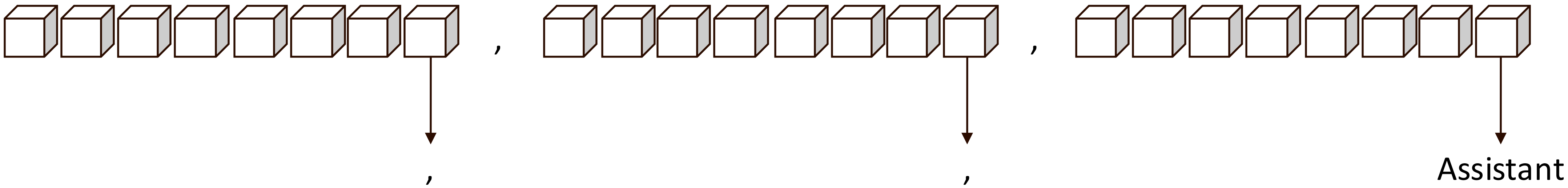
Model Method

Directly learn EOS on each frame



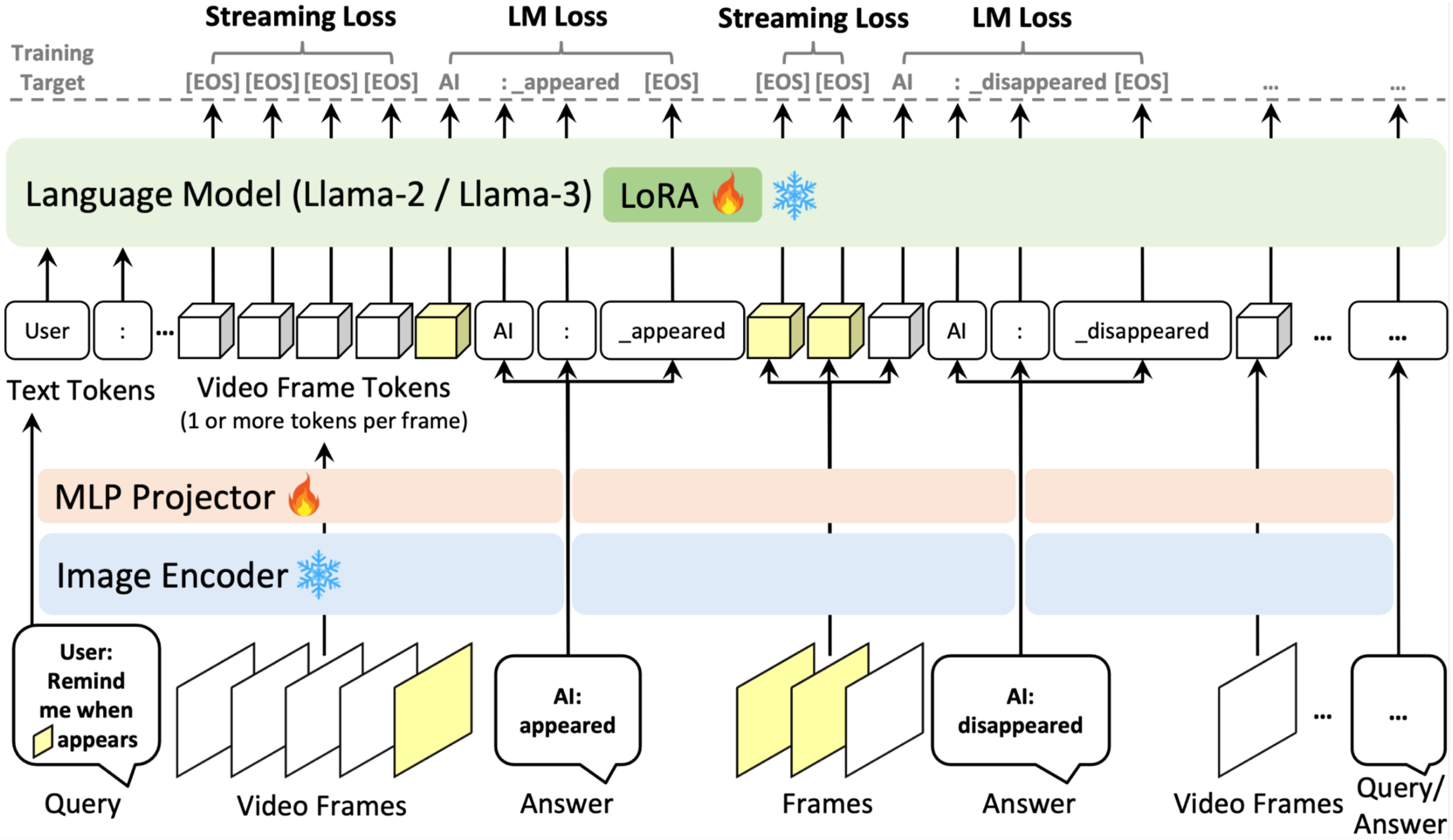
0 extra tokens!

Multiple vision tokens for a frame? We learn the interval token on the last vision token



Model Method

- CLIP ViT ⟶ CLS w. Pooled 3x3 Tokens per frame ⟶ MLP (learn) ⟶ LLM w. LoRA (learn)
- Organize a long sequences of **interleaved visual-language tokens in temporal order**
- **Streaming EOS (end-of-sequence) prediction loss** + Standard language modeling loss



Model Method

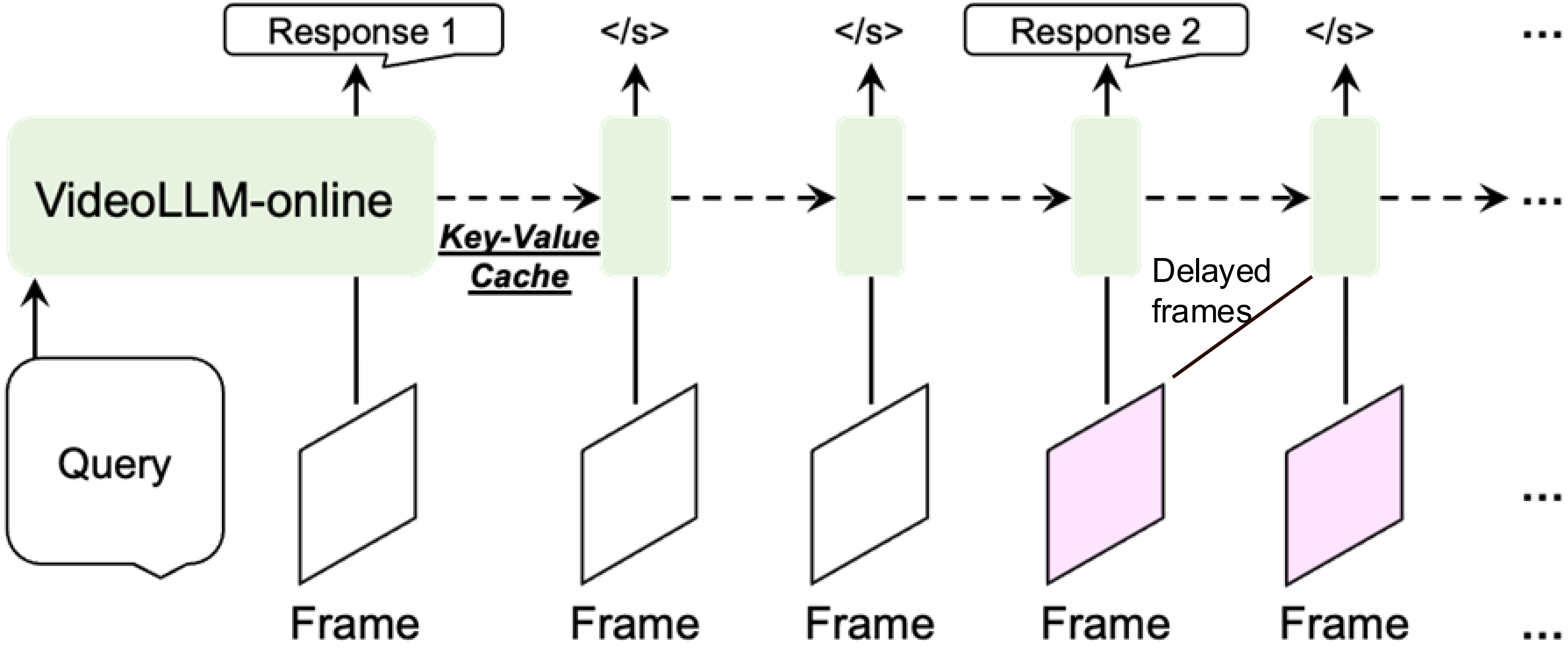
Inference pipeline

1. Adjust EOS threshold on streaming video frames

2. Continuous KV Cache during streaming inference

3. Parallelize the fast frame encoder and the slow language model

4. Common acceleration tricks, flash-attention, bf-16



Data Method

For data collected in a streaming way

Ego4D narration data collection can be regarded as a streaming dialogue by data annotators.

1. 5-minute video narration to interleaved dialogue;
2. Use Llama-3-8B-Instruct to refine narration, e.g. C does … -> You are …
3. Learning EOS on intermediate video frames.

A chat between a curious user and an artificial intelligence assistant. The assistant gives helpful, detailed, and polite answers to the user's questions.

<frame>

USER: You are my AR glass assistant in streaming mode. Help me to narrate my view.

ASSISTANT: You are looking at the rope on the floor.</s>

<frame>,<frame>,<frame>,<frame>,<frame>,<frame>,<frame>,<frame>,<frame>,<frame>,<frame>,<frame>,<frame>,<frame>,<frame>

ASSISTANT: You tie the shoe lace on the floor.</s>

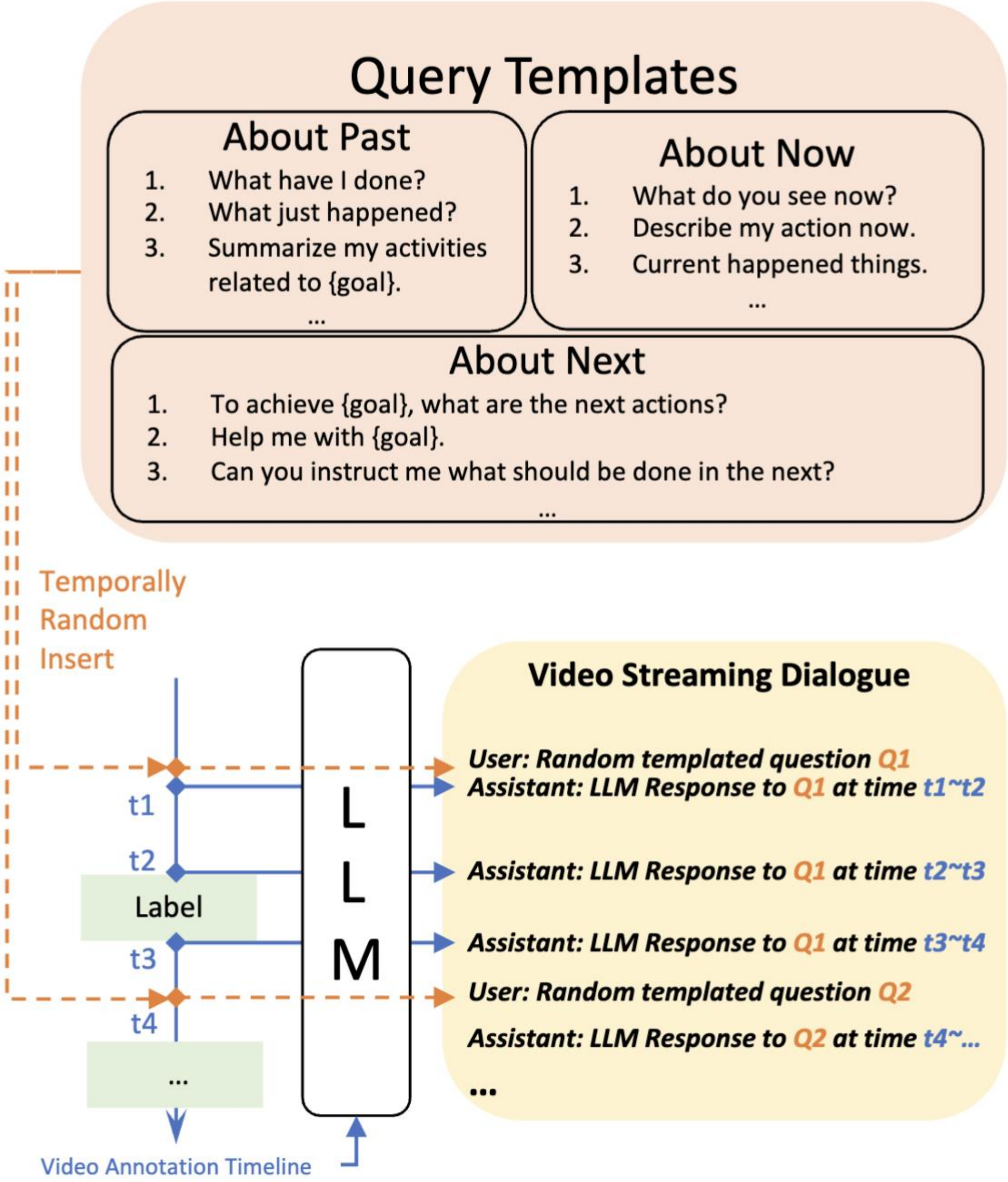
<frame>,<frame>,<frame>,<frame>,<frame>,<frame>,<frame>,<frame>,<frame>,<frame>,<frame>,<frame>,<frame>

ASSISTANT: You are looking around the room.</s>

…

Data Method

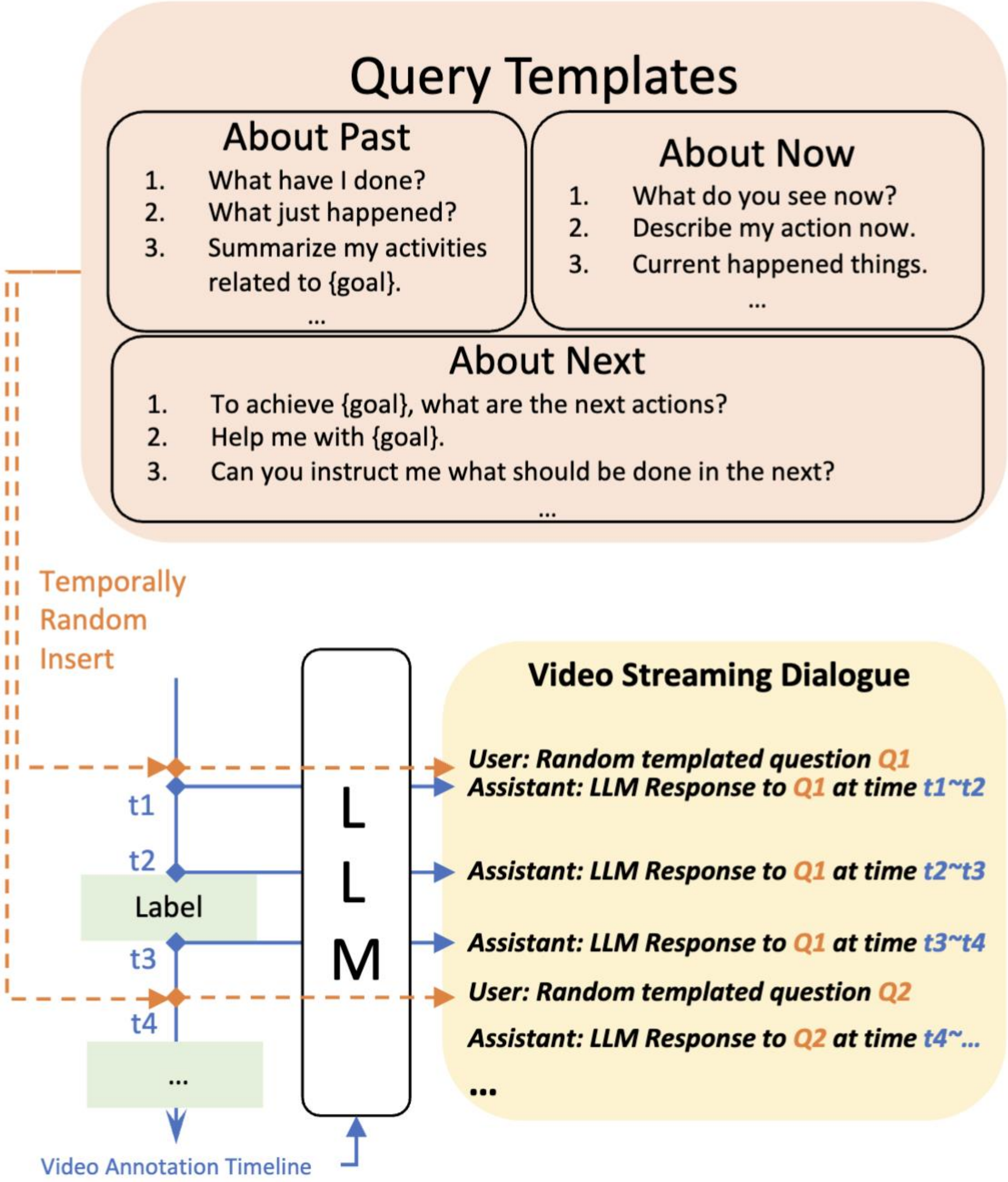
For more common offline video dataset, we prompt LLM to generate streaming dialogue


Query Templates
About Past
1. What have I done?
2. What just happened?
3. Summarize my activities related to {goal}.
...
About Now
1. What do you see now?
2. Describe my action now.
3. Current happened things.
...
About Next
1. To achieve {goal}, what are the next actions?
2. Help me with {goal}.
3. Can you instruct me what should be done in the next?
...
Temporally Random Insert
t1
t2
Label
t3
t4
...
Video Annotation Timeline
LLM
Video Streaming Dialogue
User: Random templated question Q1
Assistant: LLM Response to Q1 at time t1~t2
Assistant: LLM Response to Q1 at time t2~t3
Assistant: LLM Response to Q1 at time t3~t4
User: Random templated question Q2
Assistant: LLM Response to Q2 at time t4~...
...

1. Randomly insert templated or LLM generated questions into the video timeline
2. Prompting LLM to simulate responses at the timestamps in annotations

Data Method

For more common offline video dataset, we prompt LLM to generate streaming dialogue



Query Templates

About Past

1. What have I done?
2. What just happened?
3. Summarize my activities related to {goal}.

...

About Now

1. What do you see now?
2. Describe my action now.
3. Current happened things.

...

About Next

1. To achieve {goal}, what are the next actions?
2. Help me with {goal}.
3. Can you instruct me what should be done in the next?

...

Video Streaming Dialogue

User: Random templated question Q1
Assistant: LLM Response to Q1 at time t1~t2
Assistant: LLM Response to Q1 at time t2~t3
Assistant: LLM Response to Q1 at time t3~t4
User: Random templated question Q2
Assistant: LLM Response to Q2 at time t4~...
...

Generated Dialogue Example:

- **Annotation:**

0.0s – 135.9s: Slice tomatoes into smaller pieces
136.7s – 149.4s: Add pepper to the sliced tomatoes
149.5s – 166.0s: Add salt to the sliced tomatoes
166.2s – 174.8s: Add sauce to the sliced tomatoes
175.0s - 212.4s: Mix ingredients.
212.4s – 228.1s: Clean hands...

- **Generated Dialogue Data:**

(6.3s) User: What am I doing right now?
(6.3s) Assistant: You're currently slicing tomatoes into smaller pieces.
(135.9s) Assistant: You've finished slicing the tomatoes. What's next?
(136.7s) Assistant: Now, you're adding pepper to the sliced tomatoes.
(149.4s) Assistant: You've added pepper. Next, you should add salt to the sliced tomatoes.
(149.5s) User: Is the pepper important for the dish?
(149.5s) Assistant: Yes, pepper enhances the flavor of the dish. But now, let's focus on adding salt to the tomatoes.
(166.0s) Assistant: You've added salt. Now, it's time to add sauce to the sliced tomatoes.
(166.1s) User: What kind of sauce should I use?
(166.1s) Assistant: For this dish, a tomato-based sauce would work well. But feel free to experiment with other options if you prefer...

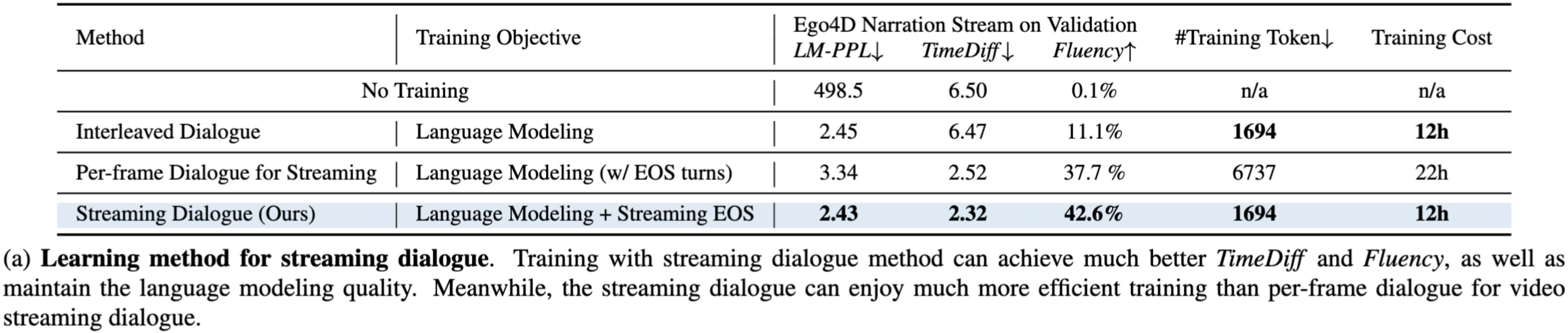
Experiments

Define metrics for streaming

1. **(Offline) LM-PPL: language modeling perplexity given context**
2. **(Online) TimeDiff: the time difference between expected response time and actual response time**
3. **(Online) Fluency: the token causal matching ratio with ground-truth tokens**

Experiments

Ego4D Streaming Narration

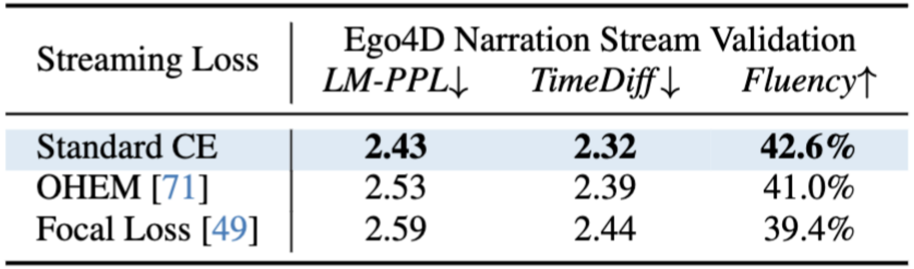


Method	Training Objective	Ego4D Narration Stream on Validation			#Training Token↓	Training Cost
		LM-PPL↓	*TimeDiff*↓	*Fluency*↑		
No Training		498.5	6.50	0.1%	n/a	n/a
Interleaved Dialogue	Language Modeling	2.45	6.47	11.1%	**1694**	**12h**
Per-frame Dialogue for Streaming	Language Modeling (w/ EOS turns)	3.34	2.52	37.7 %	6737	22h
Streaming Dialogue (Ours)	Language Modeling + Streaming EOS	**2.43**	**2.32**	**42.6%**	**1694**	**12h**

(a) **Learning method for streaming dialogue**. Training with streaming dialogue method can achieve much better *TimeDiff* and *Fluency*, as well as maintain the language modeling quality. Meanwhile, the streaming dialogue can enjoy much more efficient training than per-frame dialogue for video streaming dialogue.

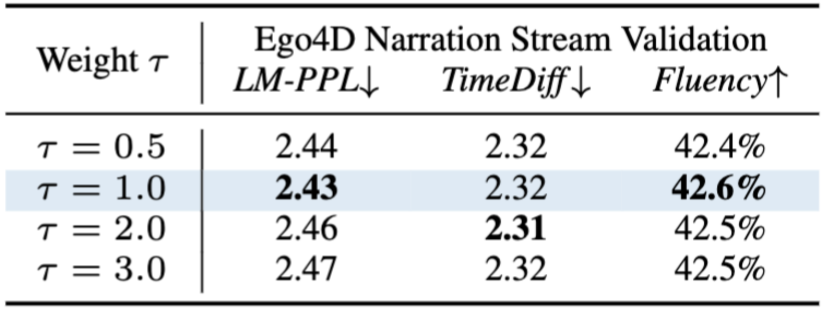
Experiments

Ego4D Streaming Narration



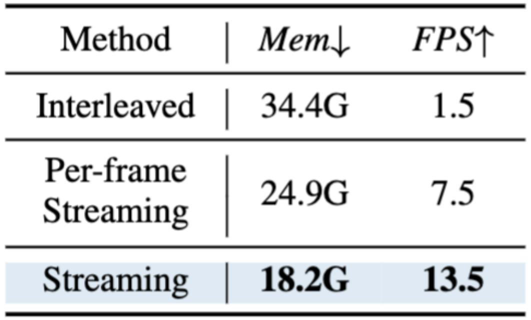
Streaming Loss	Ego4D Narration Stream Validation		
	LM-PPL↓	*TimeDiff*↓	*Fluency*↑
Standard CE	**2.43**	**2.32**	**42.6%**
OHEM [71]	2.53	2.39	41.0%
Focal Loss [49]	2.59	2.44	39.4%

(b) **Streaming loss function**. Standard CE (cross-entropy) is enough for training streaming dialogue; there is no need to specifically to address the class imbalance on EOS token.



Weight τ	Ego4D Narration Stream Validation		
	LM-PPL↓	*TimeDiff*↓	*Fluency*↑
$\tau = 0.5$	2.44	2.32	42.4%
$\tau = 1.0$	**2.43**	2.32	**42.6%**
$\tau = 2.0$	2.46	**2.31**	42.5%
$\tau = 3.0$	2.47	2.32	42.5%

(c) **Streaming loss weight**. Using slightly higher streaming loss weight ($\tau = 2.0$) achieves the best trade-off among various metrics.

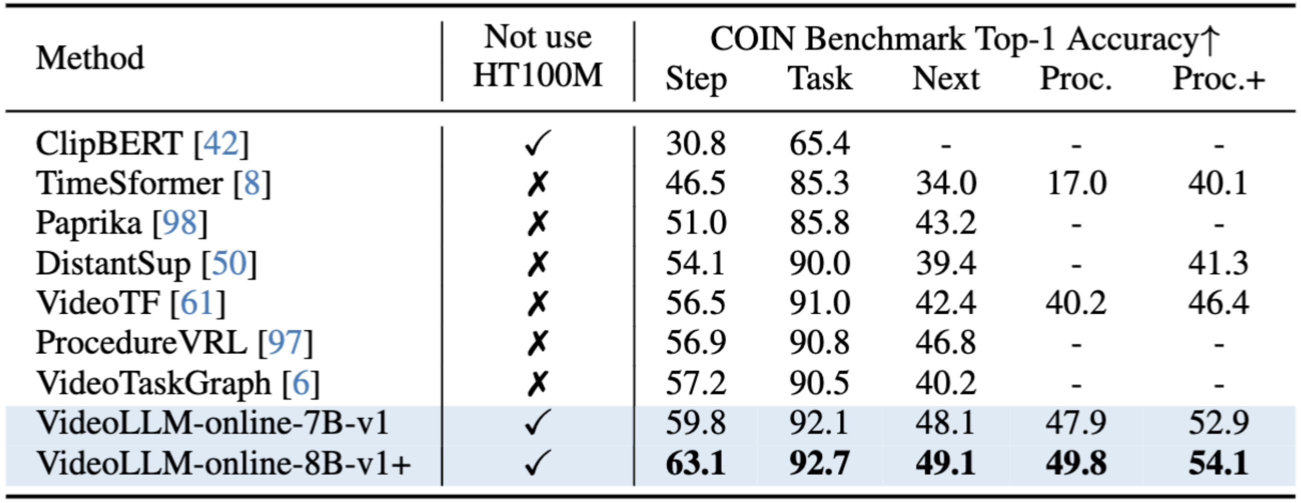


Method	*Mem*↓	*FPS*↑
Interleaved	34.4G	1.5
Per-frame Streaming	24.9G	7.5
Streaming	**18.2G**	**13.5**

(d) **Generation memory/speed.** Streaming dialogue method has much better efficiency.

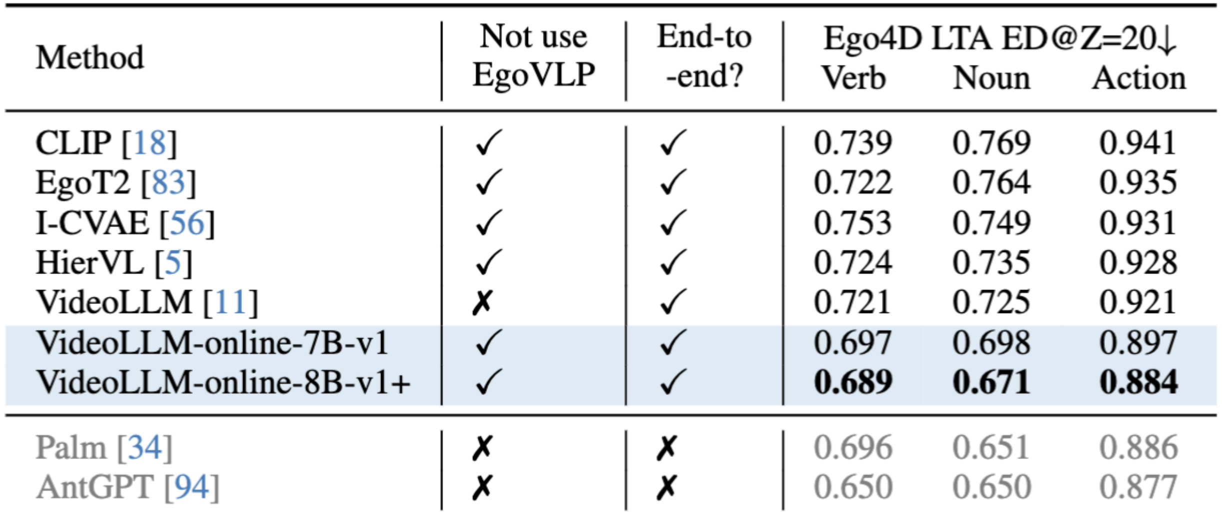
Experiments

Downstreaming Fine-tuning



Method	Not use HT100M	COIN Benchmark Top-1 Accuracy↑ Step	Task	Next	Proc.	Proc.+
ClipBERT [42]	✓	30.8	65.4	-	-	-
TimeSformer [8]	✗	46.5	85.3	34.0	17.0	40.1
Paprika [98]	✗	51.0	85.8	43.2	-	-
DistantSup [50]	✗	54.1	90.0	39.4	-	41.3
VideoTF [61]	✗	56.5	91.0	42.4	40.2	46.4
ProcedureVRL [97]	✗	56.9	90.8	46.8	-	-
VideoTaskGraph [6]	✗	57.2	90.5	40.2	-	-
VideoLLM-online-7B-v1	✓	59.8	92.1	48.1	47.9	52.9
VideoLLM-online-8B-v1+	✓	**63.1**	**92.7**	**49.1**	**49.8**	**54.1**

(a) Results on COIN benchmarks (left to right): step recognition, task recognition, next forecasting, procedure forecasting, procedure forecasting with a goal.

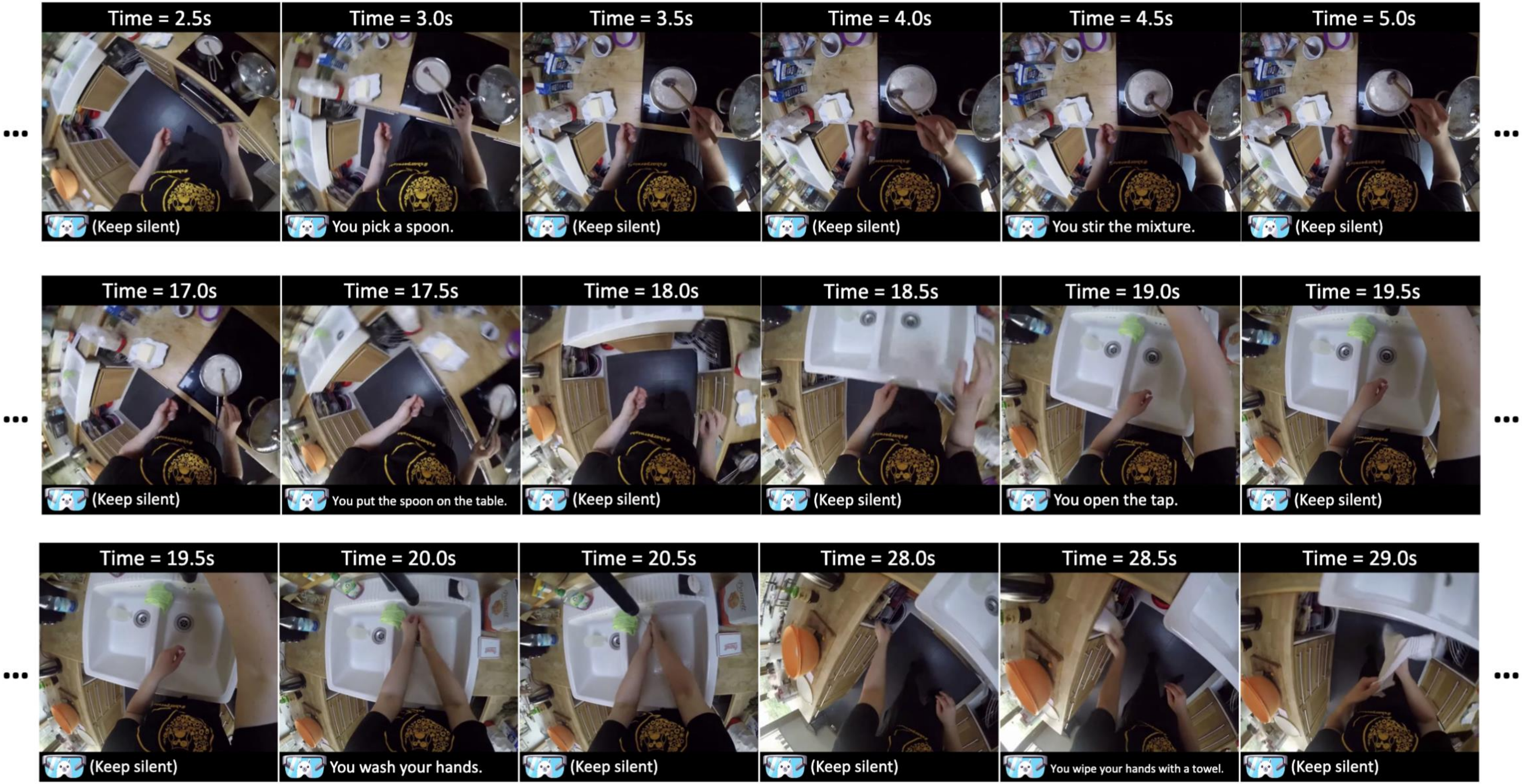


Method	Not use EgoVLP	End-to -end?	Ego4D LTA ED@Z=20↓ Verb	Noun	Action
CLIP [18]	✓	✓	0.739	0.769	0.941
EgoT2 [83]	✓	✓	0.722	0.764	0.935
I-CVAE [56]	✓	✓	0.753	0.749	0.931
HierVL [5]	✓	✓	0.724	0.735	0.928
VideoLLM [11]	✗	✓	0.721	0.725	0.921
VideoLLM-online-7B-v1	✓	✓	0.697	0.698	0.897
VideoLLM-online-8B-v1+	✓	✓	**0.689**	**0.671**	**0.884**
Palm [34]	✗	✗	0.696	0.651	0.886
AntGPT [94]	✗	✗	0.650	0.650	0.877

(b) Results on Ego4D LTA benchmark, evaluated on public server. ED@Z=20 denotes editing distance for future 20 actions.

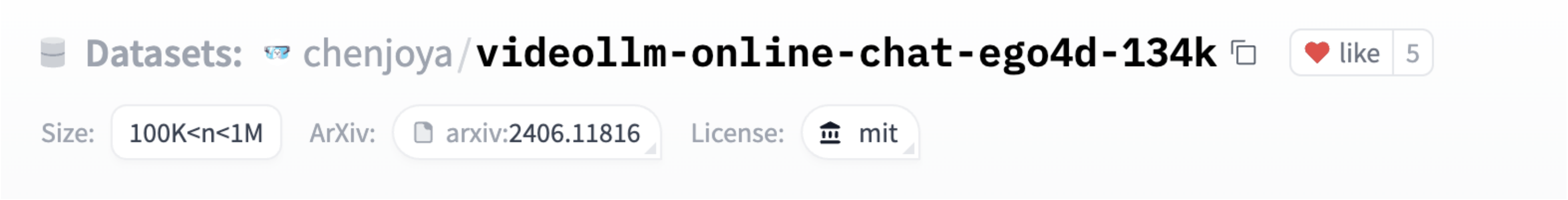
Experiments

Online Temporal Alignment



Open source

Demo model used data: Ego4D 113k streaming narration and 21k (generated) streaming dialogue



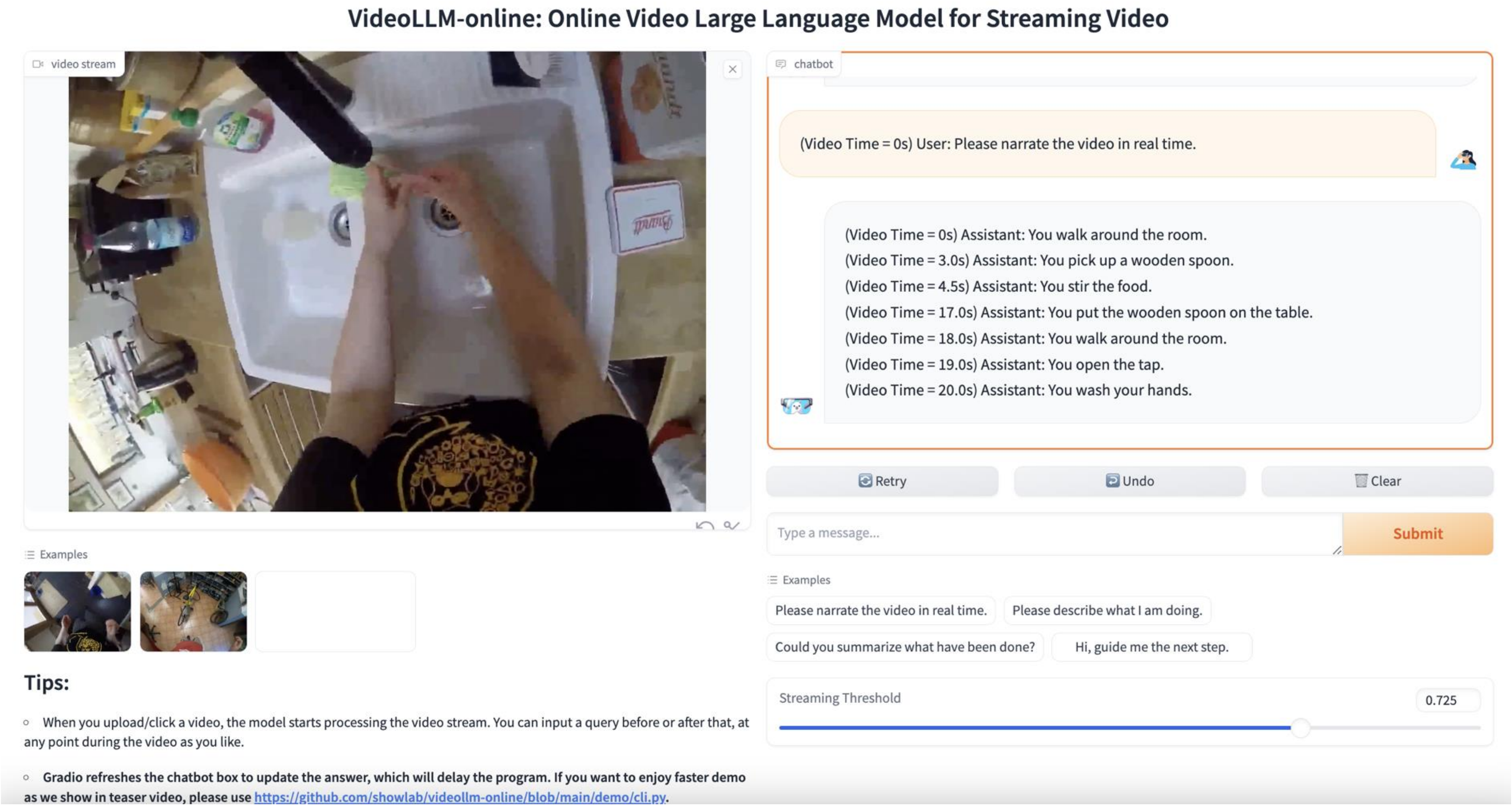
Datasets: chenjoya/**videollm-online-chat-ego4d-134k**

Size: 100K<n<1M

ArXiv: arxiv:2406.11816

License: mit

Gradio Demo (CLI is faster, 5~10 FPS for RTX 3090, 10~15 FPS for A100 GPU)



VideoLLM-online: Online Video Large Language Model for Streaming Video

(Video Time = 0s) User: Please narrate the video in real time.

(Video Time = 0s) Assistant: You walk around the room.
(Video Time = 3.0s) Assistant: You pick up a wooden spoon.
(Video Time = 4.5s) Assistant: You stir the food.
(Video Time = 17.0s) Assistant: You put the wooden spoon on the table.
(Video Time = 18.0s) Assistant: You walk around the room.
(Video Time = 19.0s) Assistant: You open the tap.
(Video Time = 20.0s) Assistant: You wash your hands.

Please narrate the video in real time.
Please describe what I am doing.
Could you summarize what have been done?
Hi, guide me the next step.

Streaming Threshold: 0.725

Tips:

- When you upload/click a video, the model starts processing the video stream. You can input a query before or after that, at any point during the video as you like.
- Gradio refreshes the chatbot box to update the answer, which will delay the program. If you want to enjoy faster demo as we show in teaser video, please use https://github.com/showlab/videollm-online/blob/main/demo/cli.py.

VideoLLM-online: Online Video Large Language Model for Streaming Video

Paper, Code, Demo, Data

https://showlab.github.io/videollm-online/

感谢聆听！Q&A