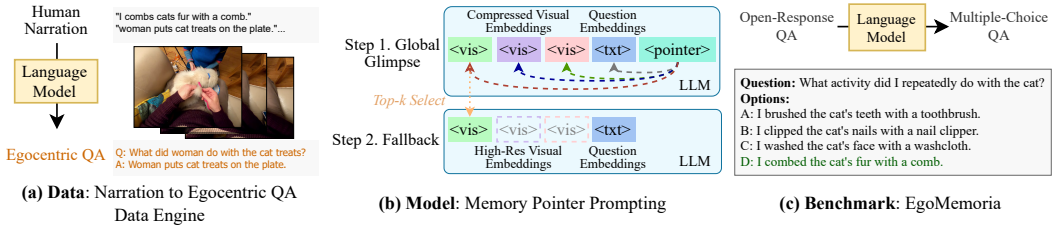
MM-EGO: TOWARDS BUILDING EGOCENTRIC MULTIMODAL LLMS

Anonymous authors
Paper under double-blind review



(a) Data: Narration to Egocentric QA Data Engine

(b) Model: Memory Pointer Prompting

(c) Benchmark: EgoMemoria

Figure 1: We introduce a foundation model for egocentric video understanding, contributing from three key perspectives: (a) a data engine that can automatically transform human narrations into 7 million egocentric QA samples, (b) a multimodal language model designed for egocentric video comprehension, and (c) the curation of a challenging egocentric video understanding benchmark.

ABSTRACT

This research aims to comprehensively explore building a multimodal foundation model for egocentric video understanding. To achieve this goal, we work on three fronts. First, as there is a lack of QA data for egocentric video understanding, we develop a data engine that efficiently generates 7M high-quality QA samples for egocentric videos ranging from 30 seconds to one hour long, based on human-annotated data. This is currently the largest egocentric QA dataset. Second, we contribute a challenging egocentric QA benchmark with 629 videos and 7,026 questions to evaluate the models' ability in recognizing and memorizing visual details across videos of varying lengths. We introduce a new de-biasing evaluation method to help mitigate the unavoidable language bias present in the models being evaluated. Third, we propose a specialized multimodal architecture featuring a novel "Memory Pointer Prompting" mechanism. This design includes a *global glimpse* step to gain an overarching understanding of the entire video and identify key visual information, followed by a *fallback* step that utilizes the key visual information to generate responses. This enables the model to more effectively comprehend extended video content. With the data, benchmark, and model, we successfully build MM-Ego, an egocentric multimodal LLM that shows powerful performance on egocentric video understanding.

1 INTRODUCTION

Study on egocentric videos explores how machines can see and understand the world from a first-person, self-centered perspective. Egocentric videos differ significantly from static-camera videos, such as movies or animations, both in terms of content and viewpoint. The content of egocentric videos primarily revolves around human daily activities. These videos typically share a perspective similar to human vision, where the camera and viewpoint frequently move. As a result of these characteristics, egocentric videos exhibit a distinct data distribution compared to static-camera videos, which has motivated a new area of research. In recent years, research interest in egocentric intelligence has been on the rise (Sigurdsson et al., 2018; Damen et al., 2018; Grauman et al., 2022; Mangalam et al., 2023; Plizzari et al., 2024). This growing interest is driven by the rapid advancements in AR/VR headsets and robotics, where cameras capture long-form egocentric videos in a

manner akin to human vision. Research on egocentric videos will allow these devices to understand their surroundings and human intentions, fostering more advanced machine intelligence and improving the human-machine interaction experience, with immeasurable research and application potential.

However, research on understanding egocentric videos remains in its early stages, with previous research primarily centered on specialized tasks such as story summarization (Lee et al., 2012), hand-object relationship understanding (Cai et al., 2016), action classification (Cartas et al., 2017; Li et al., 2021), and temporal or spatial grounding (Grauman et al., 2022). In contrast, works focusing on developing a more general egocentric video understanding model capable of complex understanding remain rare. Despite that video multimodal large language models (MLLMs) demonstrate strong video understanding and reasoning ability (Zhang et al., 2023; Wang et al., 2024; Lin et al., 2024; Zhang et al., 2024b), most of these works are unsuitable for egocentric video understanding from data, benchmark, and model design perspectives.

(a) From a data standpoint, although many MLLMs use some egocentric videos from ActivityNet (Yu et al., 2019), Ego4D (Grauman et al., 2022), and Charades (Sigurdsson et al., 2018) in their training, they have not been trained on *large-scale* egocentric video datasets, which inherently restricts their ability to comprehend lengthy first-person videos and accurately extract visual details. While Ego4D (Grauman et al., 2022) offers valuable human-annotated videos and labels for certain egocentric video understanding tasks, particularly episodic memory (which assesses a model's ability to retain visual details in such videos), its annotations are not structured for generating language responses, making them unsuitable for training MLLMs. Therefore, a large-scale egocentric video QA corpus is still needed. (b) In terms of benchmarking, exisiting video QA benchmarks either focus on shorter videos – such as EgoSchema (Mangalam et al., 2023) and QaEgo4D, which evaluate using around 3-minute and 8-minute videos, respectively – or concentrate on Internet video content (e.g., Video-MME (Fu et al., 2024)). This creates a notable gap in egocentric video understanding benchmarks that encompass videos ranging from seconds to an hour in length. (c) From a model design perspective, previous video MLLMs have primarily addressed long videos in two ways. The first approach involves uniformly sampling a limited number of video frames as visual input, as seen in Li et al. (2024a); Lin et al. (2024). Despite its simplicity, this approach achieves better performance among open-source models on public video benchmarks (Fu et al., 2024), largely because its design ensures high training efficiency and good scaling properties. The second approach involves feeding a large volume of visual tokens into the transformer backbone and employing engineering techniques, such as tensor parallelism and sequence parallelism (Xue et al., 2024; Zhang et al., 2024a), to facilitate training with millions of visual tokens in context. However, these long-context transformers suffer from slow training speeds and small overall batch sizes, which hinder performance improvements given the constraints of computational resources and training time. Intuitively, even humans cannot remember every detail of an hour-long video. We believe a more effective approach is to understand the video progressively: first get an overview of the entire video, then focus on specific details with particular questions in mind.

Building on the observations mentioned above, we introduce MM-Ego, an egocentric MLLM designed to process and understand long egocentric videos. Our contributions are threefold:

(i) **Data.** To scale training data for MLLMs with egocentric understanding ability, we develop an efficient data engine, using a "narration to egocentric QA" strategy, to automatically synthesize a large-scale egocentric QA dataset based on video narration data. Notably, rather than relying on existing vision-language models (VLMs) as labelers, we generate egocentric QAs based on the human-annotated fine-grained video clip narrations. This approach, conceptually related to (Di & Xie, 2024; Li et al., 2024b), ensures that our data quality is not constrained by the limitations of existing labeling VLMs. In this way, we create the first large-scale egocentric QA dataset, consisting of over 7 million egocentric QA samples that span video lengths from seconds to over an hour. This dataset enables the training of models to recognize and retain visual details from egocentric videos.

(ii) **Benchmark.** To evaluate the MLLMs' performance in understanding and memorizing visual details from egocentric videos, we propose the EgoMemoria benchmark. This challenging benchmark includes 7,026 multiple-choice questions for 629 egocentric videos ranging from 30 seconds to 1 hour. In the experiments on EgoMemoria, we further investigate the impact of inevitable language biases across different models during evaluation and introduce a debiased metric to more accurately assess the models' true egocentric understanding capabilities.

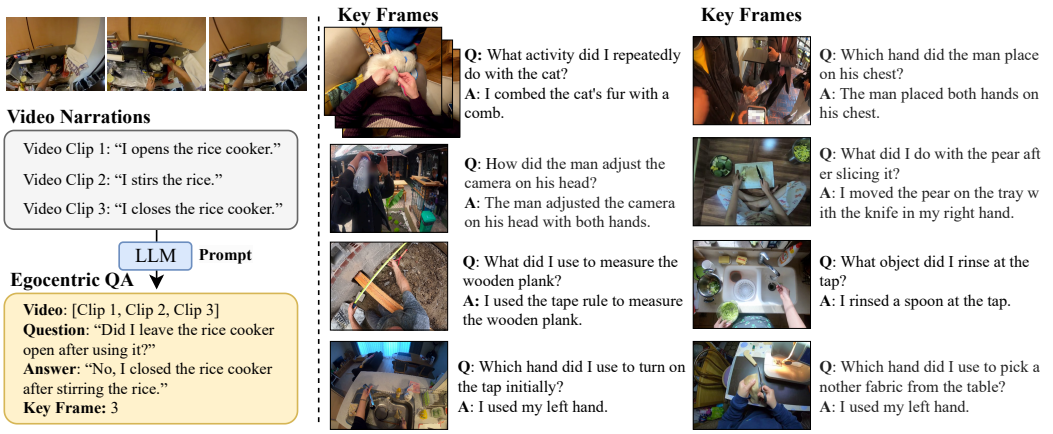


Figure 2: "Narration to Egocentric QA" data engine. Given a sequence of human-annotated video narrations, we instruct a language model (GPT-4o) to generate egocentric understanding-related questions and answers, along with identifying the key frames necessary to answer those questions.

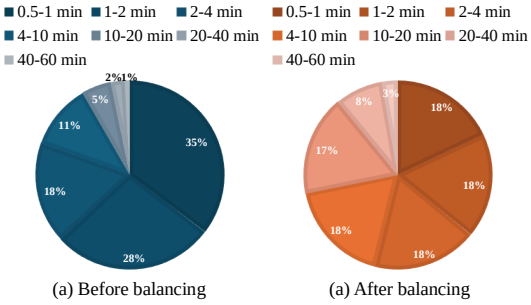
(iii) **Model.** For our MM-Ego model, we develop a progressive approach to handle egocentric videos by introducing a Memory Pointer Prompting method. It consists of two steps: "global glimpse" and "fallback". In the *global glimpse* step, we extract compressed frame-level visual embeddings from the entire video to get a global understanding. Then, we employ a memory pointer embedding, designed to examine all compressed frame-level visual embeddings along with the question embeddings, to aid in identifying key visual embeddings in a question-aware manner. In the following *fallback* step, the selected key visual embeddings, in a higher-resolution form, are then used as final input to the LLM for processing and generation. This approach allows us to achieve a global understanding of the entire video while also identifying and utilizing key visual information to answer questions related to visual details.

2 METHOD

2.1 "NARRATION TO EGOCENTRIC QA" DATA ENGINE

As outlined in Section 1, high-quality egocentric QA pairs are lacking for training an MLLM with egocentric video understanding ability. To address this gap, we develop an innovative "narration to egocentric QA" data engine that automatically generates episodic memory-related QA samples based on human-annotated video clip narrations from the Ego4D dataset (Grauman et al., 2022) without the need for additional manual annotations.

Our approach leverages over 3,000 hours of privacy-protected, de-identified egocentric videos accompanied by more than 3 million high-quality, human-created narrations. These fine-grained language descriptions provide a rich resource for generating QA pairs.



(a) Before balancing (a) After balancing

Figure 3: Video length distribution in our egocentric QA dataset.

The workflow of the data engine is illustrated in Figure 2. By organizing sequential video clips {Clip 1, Clip 2, ..., Clip N} and their corresponding narrations {Narration 1, Narration 2, ..., Narration N} in proper chronological order, we create comprehensive narration paragraphs that describe entire video sequences. We then employ a powerful **text-only language model**, i.e., GPT-4o, to generate diverse and confident QA pairs related to episodic memory based on these narration paragraphs. The language model is instructed to attach the index of the narration sentence upon which each QA pair is based. This indexing allows us to map each QA pair back to the corresponding time frames in the original videos, enabling the extraction of key frame information crucial for subsequent model training.

Applying this data engine to the extensive Ego4D dataset allows us to efficiently scale the creation of egocentric QA data. We partition the dataset into training and testing sets according to the official

Ego4D episodic memory task. The egocentric QA dataset provides more than 7 million QA samples in 938K multi-turn conversations. The data encompasses videos of varying durations, ranging from 30 seconds to 1 hour, as illustrated in Figure 3. To ensure comprehensive coverage and prevent bias towards shorter videos, we balance the number of conversations across different video lengths in training. This is the first large-scale egocentric QA dataset featuring videos of such extended ranges of duration.

Through these steps, our "narration to egocentric QA" data engine addresses the scarcity of large-scale, high-quality egocentric QA data for egocentric scenes, and sets a solid foundation for building MM-Ego, a sophisticated egocentric MLLM, which we introduce in the following section.

2.2 MM-EGO MODEL

Our modeling goal is to develop an MLLM for handling egocentric videos, which are lengthy and rich in visual details. On the one hand, frame-level information is necessary to capture the full content of the video, as skipping frames during sampling could result in a significant loss of visual details. On the other hand, processing all visual tokens generated by the visual encoder is computationally challenging for the transformer model. For instance, if each image is encoded into 729 visual embeddings (tokens), the total number of visual embeddings for a 300-frame video would be 218,700. However, most MLLMs are trained with a context length of less than 10,000 tokens (Li et al., 2024a). Taking these factors into account, we introduce the MM-Ego model, which is built for handling a large volume of egocentric video frames while maintaining manageable computational costs within the transformer backbone. MM-Ego introduces an innovative Memory Pointer Prompting mechanism, which operates in two main steps: global glimpse and fallback. We will introduce the details of MM-Ego in the following sections.

2.2.1 VISUAL AND TEXTUAL EMBEDDING

Given an input video and the question, the first step is to embed them into visual and textual embeddings separately for later processing. We begin by uniformly sampling the video into up to N frames, where N can be in the range of hundreds. Then, we extract per-frame visual feature maps from these frames using a robust vision encoder, SigLIP-so400m (Zhai et al., 2023). Following the method outlined by Li et al. (2024a), we apply a 2-layer MLP to project the visual feature maps to the LLM embedding space and use average pooling to reduce the height and width of the visual feature maps by a factor of two and flatten the height and width dimension, resulting in N relatively high-resolution visual embeddings $\{\mathbf{V}^i \in \mathbb{R}^{T\times C}, i \in [1, N]\}$ where T is the embedding length and C is the embedding dimension. For the textual embedding, since we use Qwen2 (Yang et al., 2024) as the LLM, we use its tokenizer and embedding layer to transform the input text into textual embeddings. For question q, we denote the corresponding textual question embedding as $\{\mathbf{E}^q_{\text{que}} \in \mathbb{R}^{T_q\times C}, q \in [1, Q]\}$ where Q is the total number of questions and T_q is the embedding length of question q.

2.2.2 MEMORY POINTER PROMPTING

As processing all N high-resolution visual embeddings with the LLM is computationally difficult, we propose to identify key visual embeddings in a question-aware manner and only send those selected embeddings to the subsequent LLM. Inspired by previous works on Pointer Networks (Vinyals et al., 2015; Merity et al., 2016), we propose a Memory Pointer Prompting mechanism, which is illustrated in Figure 4. Memory Pointer Prompting consists of two steps during inference: global glimpse and fallback. In the global glimpse step, key visual embeddings are identified from all frame-level embeddings, guided by the context of the question. During the subsequent fallback step, the important visual embeddings are selected, and their higher-resolution versions are provided to the LLM transformer backbone for further processing and language response generation.

Global Glimpse Step. We begin by compressing the visual embeddings through average pooling along the embedding length dimension, resulting in a set of compressed visual embeddings $\{\mathbf{E}^i_{\text{vis}} \in \mathbb{R}^{1\times C}, i \in [1, N]\}$. Next, we introduce a learnable memory pointer prompt embedding $\mathbf{P} \in \mathbb{R}^{1\times C}$, duplicate it Q times, yielding $\{\mathbf{P}^i \in \mathbb{R}^{1\times C}, i \in [1, Q]\}$, and concatenate the embeddings as follows:

$$[\mathbf{E}^1_{\text{vis}}, \mathbf{E}^2_{\text{vis}}, ..., \mathbf{E}^N_{\text{vis}}, \mathbf{E}^1_{\text{que}}, \mathbf{P}^1].$$

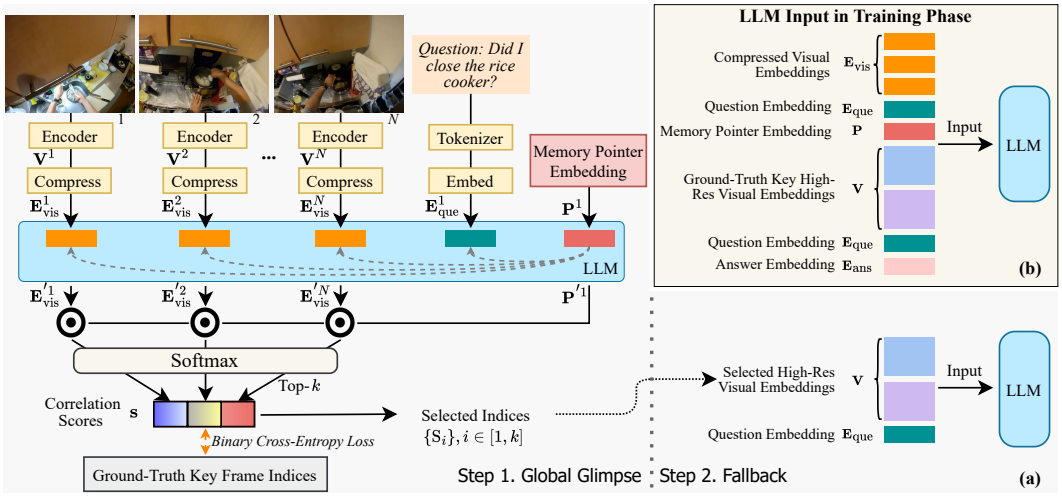


Figure 4: **(a)** Overview of the proposed Memory Pointer Prompting mechanism. Its inference consists of two steps: (1) *Global Glimpse*: We concatenate the compressed visual embeddings from all frames, denoted as $\mathbf{E}^i_{\text{vis}}$ for $i \in [1, N]$, with the question embeddings $\mathbf{E}^1_{\text{que}}$ and the memory pointer embedding $\mathbf{P}^1$. This combined embedding sequence is then input into the LLM. From the last layer, we extract embeddings and compute the dot product between the memory pointer embedding and all compressed visual embeddings to generate the correlation scores. The indices of the frames with the top k scores are selected. During training, the correlation scores are supervised by ground-truth key frame indices via a binary cross-entropy loss. (2) *Fallback*: The high-resolution visual embeddings corresponding to the selected indices are fed into the LLM along with the question embeddings for final processing and response generation. **(b)** Illustration of LLM input sequence during training.

Here $Q = 1$ as MLLMs generate answers for only one question at a time. In this way, the question embedding is followed by a pointer embedding, which will be used to identify key visual embeddings with knowledge of the question embedding. The entire embedding sequence is then fed into the LLM, from which we obtain the output embedding sequence of the final layer:

$$[\mathbf{E}'^1_{\text{vis}}, \mathbf{E}'^2_{\text{vis}}, ..., \mathbf{E}'^N_{\text{vis}}, \mathbf{E}'^1_{\text{que}}, \mathbf{P}'^1].$$

We extract and stack the processed visual embeddings $\{\mathbf{E}'^i_{\text{vis}} \in \mathbb{R}^{1\times C}, i \in [1, N]\}$ to obtain the matrix $\boldsymbol{E}_{\text{vis}} \in \mathbb{R}^{N\times C}$. We conduct a softmax dot product operation between $\boldsymbol{E}_{\text{vis}}$ and $\mathbf{P}'^1$:

$$\mathbf{s} = \text{Softmax}(\boldsymbol{E}_{\text{vis}} \cdot \mathbf{P}'^{1\mathsf{T}}) \in \mathbb{R}^N. \tag{1}$$

Here $\mathbf{s}$ is a correlation score vector indicating the correlation between the question and each frame.

Balancing Exploration and Exploitation. Our approach to selecting key visual embeddings parallels the principles of Bayesian Optimization (Frazier, 2018), where the objective function is expensive to evaluate. In such cases, it's important to balance exploration (sampling in areas where the uncertainty is high) and exploitation (sampling in areas where the surrogate model predicts high performance). However, relying solely on the aforementioned Memory Pointer Prompting may lead to overemphasizing certain areas of interest, potentially undermining the exploration process. To mitigate this issue, we introduce perturbations into the score distribution by incorporating a uniform sampling distribution. The probability vector of uniform sampling can be written as:

$$\mathbf{u}^i = \begin{cases} \alpha & \text{if } i \in \text{linspace}(0, N, k), \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Here α is an explore-exploit balancing parameter to adjust the probability distribution. We overlap the probability vector of uniform sampling and score matrix $\mathbf{s}$:

$$\mathbf{s} \leftarrow \mathbf{s} + \mathbf{u}. \tag{3}$$

We then identify the top-k indices as the set $\{\text{S}_i, i \in [1, k]\}$. In this way, we find the key visual embeddings in a question-aware manner.

Fallback Step. During inference, as shown in Figure 4, with the set of indices $\{\text{S}_i, i \in [1, k]\}$ for the selected visual embeddings, we now assemble the LLM input sequence as follows:

$$[\underbrace{\mathbf{V}^{\text{S}_1}, \mathbf{V}^{\text{S}_2}, ..., \mathbf{V}^{\text{S}_k}}_{\text{Selected Top-}k\text{ Visual Embeedings}}, \mathbf{E}^1_{\text{que}}].$$

Table 1: Distribution of videos and QA samples with different lengths.

Class	Short		Medium			Long		Sum
Minutes	0.5-1	1-2	2-4	4-10	10-20	20-40	40-60	-
Videos	100	100	100	100	100	100	29	629
QAs	500	498	987	997	1715	1792	537	7026

Table 2: Distribution of correct options in MCQs.

Option	A	B	C	D
Counts	1776	1751	1770	1729

As previously introduced, $\mathbf{V}^{S_1}, \mathbf{V}^{S_2}, ..., \mathbf{V}^{S_k}$ denote the selected top-k high-resolution visual embeddings, which provide more visual details than the compressed visual embeddings. This new embedding sequence is fed into the LLM to generate the final language response. In summary, the proposed Memory Pointer Prompting approach allows us to consider the full scope of video information while filtering out redundant data in the LLM transformer, ensuring computational efficiency. The new input serves as the final input of the LLM to generate the language response given the visual and textual information.

Training Procedure. Given the novel design of MM-Ego, its training procedure is different from popular MLLMs (Liu et al., 2023). Specifically, let the answer embedding for question $q \in [1, Q]$ be denoted as $\mathbf{E}^q_{\text{ans}}$, then the input embedding sequence during the training process is represented as:

$$[\underbrace{\mathbf{E}^1_{\text{vis}}, \mathbf{E}^2_{\text{vis}}, ..., \mathbf{E}^N_{\text{vis}}}_{\text{Compressed Visual Embeddings}}, \mathbf{E}^1_{\text{que}}, \mathbf{P}^1, ..., \mathbf{E}^Q_{\text{que}}, \mathbf{P}^Q, \underbrace{\mathbf{V}^{S_1}, \mathbf{V}^{S_2}, ..., \mathbf{V}^{S_k}}_{\text{Selected High-Res Visual Embeddings}}, \mathbf{E}^1_{\text{que}}, \mathbf{E}^1_{\text{ans}}, ..., \mathbf{E}^Q_{\text{que}}, \mathbf{E}^Q_{\text{ans}}].$$

We also provide a simplified illustration (where $Q = 1$) of the input embedding sequence structure during training in Figure 4. Here, we begin by inputting the compressed visual embeddings for all N frames, followed by the question embedding and memory pointer embedding. Next, we integrate the k selected high-resolution visual embeddings (based on the ground-truth key frame labels), and finally, incorporate both the question and answer embeddings. Once the input sequence is prepared as outlined above, we can train MM-Ego similarly to traditional large language models. The compressed visual embeddings, question embedding, and memory pointer embeddings used as prefixes do not contribute to the language cross-entropy loss.

When training on samples from our curated egocentric QA dataset where there are ground-truth key frame labels for each question, we compute the correlation score vector $\mathbf{s}$ in the global glimpse step, and supervise it using a binary cross-entropy loss. For training samples that lack ground-truth key frame labels, we omit the prefixes, which results in the traditional MLLM training process.

3 Experiments

In the experiment section, we will first present a new egocentric video understanding benchmark, specifically designed to assess episodic memory capabilities. Following this, we will perform comprehensive experiments to evaluate MM-Ego, utilizing both the newly introduced benchmark and existing public benchmarks.

3.1 EgoMemoria Benchmark

To evaluate the performance of egocentric MLLMs, especially in terms of episodic memory ability, we propose a new benchmark called EgoMemoria. Specifically, we generate memory-related questions and answers from human-annotated narrations in the validation set of the Ego4D dataset. To ensure diversity, for each video we only generate a limited number of questions. We divide the videos into seven different length ranges: 0.5 to 1 min, 1 to 2 min, 2 to 4 min, 4 to 10 min, 10 to 20 min, 20 to 40 min, and 40 to 60 min. We aim to balance the number of samples in different video lengths. The distribution of videos and corresponding question-answer pairs (QAs) for each category is shown in Table 1. Furthermore, we group these video lengths into three broader categories: short (0.5 to 2 min), medium (2 to 20 min), and long (20 to 60 min). In total, we collect 629 videos with 7,026 questions. The most frequently occurring verbs and nouns in the questions are visualized in Figure 5.

Figure 5: The most frequently occurring verbs and nouns in EgoMemoria.

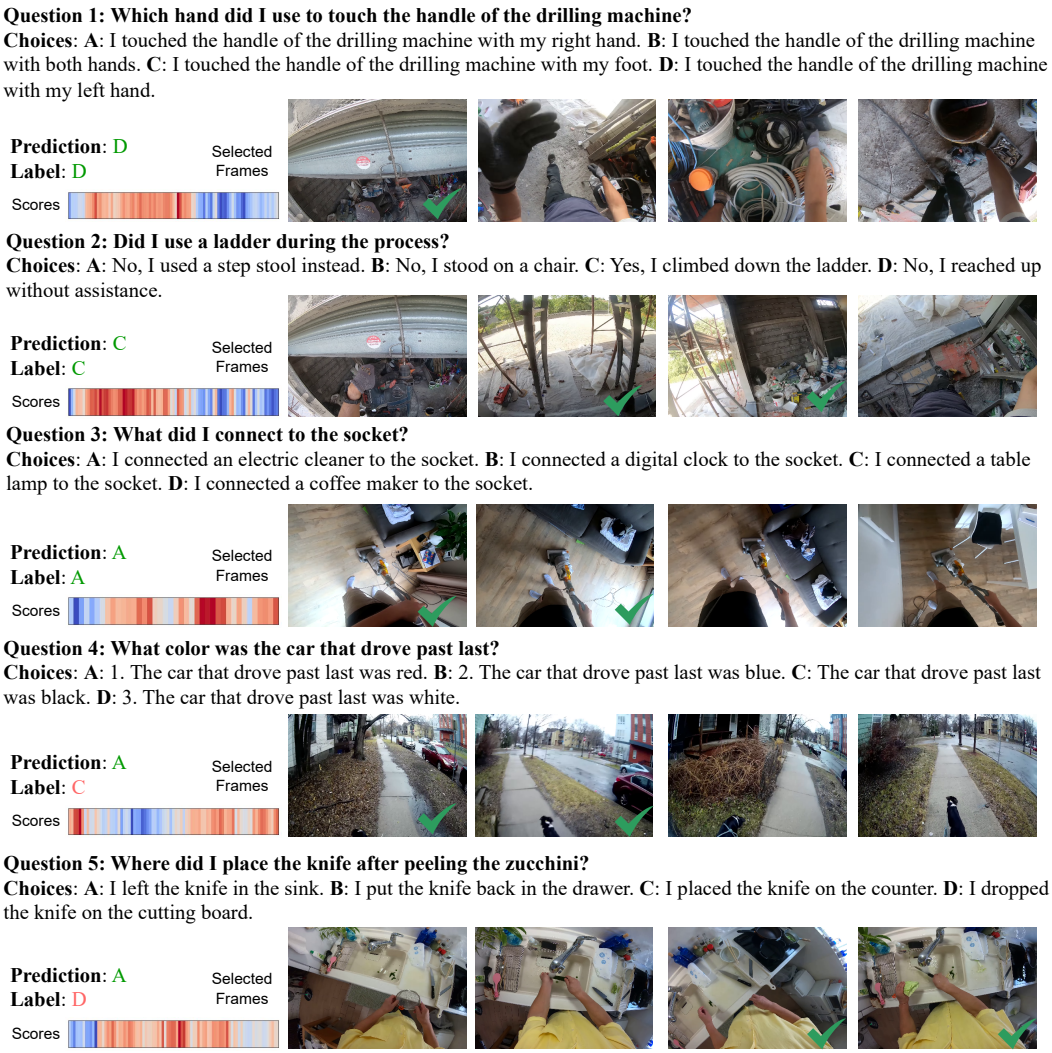


Figure 6: EgoMemoria QAs visualization and prediction analysis of the global glimpse step. We find high consistency between the identified key frames and the questions, demonstrating the effectiveness of the proposed Memory Pointer Prompting method. The visualized correlation scores show distinct distributions for different questions given the same video, indicating its question-specific nature. The ✓ indicates that the selected frames are relevant to the questions.

Since free-form answers are typically evaluated using a closed-source LLM as a judge, the evaluation can be inconsistent and subject to significant variance, especially due to model version updates. To ensure more reliable, standardized, and consistent performance evaluation, we convert the free-form answers into multiple-choice questions (MCQs), which helps reduce score instability. In practice, based on the free-form answer, we instruct ChatGPT to generate three additional choices that are plausible but incorrect, considering the original question and answer. We then randomize the order of these choices to achieve a uniform distribution of correct options, as shown in Table 2, to minimize bias in option placement. We visualize some randomly sampled examples in Figure 6.

3.2 Experimental Setup

Training Data. We employ a joint image-video supervised fine-tuning (SFT) strategy. To enhance the model's capability in understanding a broader range of visual data, we combine our egocentric QA dataset with a variety of multimodal datasets. We curate an SFT dataset mixture consisting of our egocentric QA dataset, Ego4D narration dataset (Grauman et al., 2022), LLaVA-NeXT SFT collection (including ChartQA (Masry et al., 2022), AI2D (Hiippala et al., 2021), DocVQA (Mathew et al., 2021), DVQA (Kafle et al., 2018), COCO (Lin et al., 2014)), ShareGPT4V (Chen et al., 2023a), synthdog-en (Kim et al., 2021)), ShareGPT-4o (Chen et al., 2023b), ALLaVA instruct (Chen

Table 3: Performance comparison and language bias analysis of different models on the EgoMemoria benchmark. Our MM-Ego model demonstrates the best performance both before and after excluding the language bias of different models.

Method	LLaVA-OV (Li et al., 2024a)				Ego SFT				MM-Ego			
	Short	Medium	Long	Avg	Short	Medium	Long	Avg	Short	Medium	Long	Avg
Original	70.24	64.94	61.19	65.45	79.06	76.34	73.51	76.30	79.96	79.64	79.09	**79.56**
Exclude *LLaVA-OV* Bias	56.44	49.64	44.83	50.30	66.37	64.15	60.03	63.52	71.97	70.68	68.15	**70.26**
Exclude *Ego SFT* Bias	55.75	49.27	45.21	50.08	61.73	59.59	54.50	58.61	67.70	66.33	63.89	**65.97**
Exclude *MM-Ego* Bias	47.41	42.11	35.22	41.58	50.60	46.39	40.38	45.79	49.80	49.11	43.81	**47.58**
Mean Debiased Accuracy (MDA)	53.20	47.01	41.76	47.32	59.56	56.71	51.64	55.97	63.16	62.04	58.62	**61.27**

Table 4: Comparison with state-of-the-art video MLLMs. MM-Ego shows strong performance on egocentric understanding and competitive performance on Internet video understanding.

Method	EgoMemoria (MDA)				EgoSchema	Video-MME (w/o subs)			
	Short	Medium	Long	Avg	Full	Short	Medium	Long	Entire
GPT-4o	**64.31**	59.47	57.65	60.48	**72.2**	**80.00**	**70.30**	**65.30**	**71.90**
LLaVA-NeXT-Video-7B-DPO (Zhang et al., 2024b)	30.38	25.95	21.49	25.94	-	-	-	-	-
LLaVA-NeXT-Video-32B-Qwen (Zhang et al., 2024b)	43.78	33.76	31.04	36.19	60.85	-	-	-	60.20
LLaVA-OV 7B (Li et al., 2024a)	53.20	47.01	41.76	47.32	60.10	69.30	56.00	49.40	58.30
MM-Ego (ours)	63.16	**62.04**	**58.62**	**61.27**	69.03	67.60	55.70	47.80	57.00

et al., 2024a), ShareGPT4Video (Chen et al., 2024b), sherlock (Hessel et al., 2022), ScienceQA (Lu et al., 2022), NExT-QA (Xiao et al., 2021), and ActivityNet-QA (Yu et al., 2019).

Implementation Details. The model is trained for 1 epoch with a base learning rate of 1×10^{-5}, using a cosine scheduler. The batch size is set to 128. We sample a maximum of 300 frames ($N = 300$) and select 32 visual embeddings in the proposed memory pointer prompting mechanism. By default, we set the explore-exploit balancing parameter α to 0.1. Greedy decoding is used in generation.

Pretrained Models. Our MM-Ego model is initialized from LLaVA-OV 7B (Li et al., 2024a), a state-of-the-art MLLM known for its good performance on general multimodal understanding tasks. Following the same architecture, we use the SigLip-so400M ViT (Zhai et al., 2023) as the visual encoder for embedding video frames and Qwen2-7B (Yang et al., 2024) as the LLM architecture.

3.3 Main Results

We first conduct experiments on our EgoMemoria benchmark, primarily comparing three models: LLaVA-OV (Li et al., 2024a), its fine-tuned version using our MM-Ego SFT data mixture (referred to as "Ego SFT"), and our MM-Ego model, which incorporates the proposed Memory Pointer Prompting mentioned in Section 2.2.2. We show the EgoMemoria accuracy in the first row of Table 3. We observe a significant improvement in the model's performance on egocentric QAs after training on our MM-Ego data mixture, attributed to the rich egocentric knowledge provided by our curated egocentric QA training data. Moreover, leveraging the MM-Ego model architecture further enhances performance, thanks to the effective Memory Pointer Prompting mechanism.

However, we notice that the original overall performance metrics are higher than anticipated, raising curiosity about the extent to which language bias contributes to the models' accuracy. To answer this question, we conduct additional experiments aimed at eliminating these language biases. Specifically, we test the three model variants on the EgoMemoria benchmark without any visual inputs, identifying questions that could be correctly answered without videos as "language-biased questions". Then, we evaluate the models' performance on the subset of the benchmark without language-biased questions. For fairness, we apply this debiasing process across all three models so that they are evaluated on the same sets of data. We calculate the mean accuracy of the debiased variants, referred to as the "Mean Debiased Accuracy (MDA)". The results are presented in Table 3.

As expected, after removing the language-biased questions, the accuracy of all three models drops significantly to a more reasonable level. The performance decline is notably more pronounced in the "Medium" and "Long" classes compared to the "Short" class. For example, the average accuracy of LLaVA-OV across the three classes (short, medium, and long) drops from 65.45 to 47.32. The decrease in the "Short" class is 17.04, in the "Medium" class is 17.93, and in the "Long" class is 19.43. Despite this, we still observe improvements in MDA after training with SFT data generated by our MM-Ego data engine (**+8.65**) and applying our Memory Pointer Prompting method (**+13.95**). These results demonstrate the effectiveness of our approach even after considering language bias.

Table 5: MDA on EgoMemoria when inferring with different numbers of frames. Our MM-Ego model shows a smaller relative drop on average when decreasing the number of sampled frames.

Frames	Short			Medium			Long			Avg		
	LLaVA-OV	Ego SFT	MM-Ego	LLaVA-OV	Ego SFT	MM-Ego	LLaVA-OV	Ego SFT	MM-Ego	LLaVA-OV	Ego SFT	MM-Ego
32	53.20	59.56	63.16	47.01	56.71	62.04	41.76	51.64	58.62	47.32	55.97	61.27
16	52.68	60.45	63.82	46.37	55.99	60.81	40.12	51.15	58.16	46.39	55.86	60.93
8	50.76	59.59	62.22	44.82	54.55	58.23	39.41	49.11	55.19	44.99	54.42	58.55
4	50.43	55.36	62.30	42.54	52.08	58.44	38.88	48.40	54.65	43.95	51.95	58.46
Rel. Diff	5.20%	7.07%	**1.36%**	9.49%	8.16%	**5.81%**	6.89%	**6.26%**	6.77%	7.12%	7.19%	**4.59%**

To better understand the capability of MM-Ego, we compare its performance with state-of-the-art video MLLMs on EgoMemoria and prevalent large-scale video QA benchmarks, including the long egocentric video understanding benchmark EgoSchema (Mangalam et al., 2023) and the Internet-video-based long-video understanding benchmark Video-MME (Fu et al., 2024). The results are shown in Table 4. On EgoMemoria, GPT-4o is evaluated using 32 uniformly sampled frames from the videos, while other models follow their respective official inference settings. The MDA on EgoMemoria is computed using the debiased subsets used in Table 3. Notably, MM-Ego exhibits the highest performance on EgoMemoria, particularly in the 'Medium' and 'Long' classes. On the EgoSchema benchmark, our model achieves a substantial performance gain of **+8.18** over the previous state-of-the-art open-source model, underscoring the effectiveness of both our data and model design for egocentric understanding. Additionally, on the challenging Internet video understanding Video-MME benchmark, our model is on par with the leading model of similar parameter size although our data mixture is less diverse compared with (Li et al., 2024a). These results showcase MM-Ego's capability in egocentric video understanding while preserving its general video comprehension abilities.

3.4 Model Analysis

Quantitative Analysis of Different Numbers of Frames. To evaluate the influence of sampling different numbers of frames for different models, we calculate the mean debiased accuracy (MDA) in Table 5. The relative performance drop from sampling 32 frames to sampling 4 frames is also calculated. As expected, all models exhibit a decrease in performance with fewer sampled frames. Notably, MM-Ego exhibits a smaller average performance drop when the number of frames is reduced due to its ability to identify key frames given lower computational budget. The relative performance drop in the short category is considerably smaller compared to the medium and long categories, likely because shorter videos require fewer frames to comprehend.

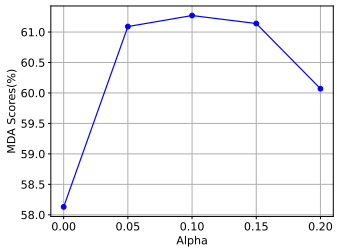


Figure 7: MDA scores with different α values for explore-and-exploit balancing.

Qualitative Analysis of Memory Pointer Prompting. In Figure 6, we present a qualitative analysis of the accuracy of Memory Pointer Prompting on EgoMemoria. We randomly select samples and visualize the key frames identified by the global glimpse step in Memory Pointer Prompting. The results show a strong alignment between the questions and the selected frames. In failure cases, we observe that the issues are often due to the ambiguity of the questions, causing the model to struggle with accurately localizing the key visual embeddings. Furthermore, the visualized correlation scores during the global glimpse step show distinct patterns across various videos and questions, confirming its effectiveness in selecting key visual embeddings tailored to the specific questions.

Quantitative Analysis of Explore-Exploit Balancing Parameter α. As discussed in Section 2.2.2, we design an explore-exploit balancing parameter α to fuse the uniform distribution and the sampling probability computed by Memory Pointer Prompting. We illustrate MM-Ego's performance with varying values of α in Figure 7. The results show that $\alpha = 0.1$ achieves the best performance, while larger or smaller values of α tend to either over-explore or over-exploit.

Conversation Examples by MM-Ego. In Figure 8, we show a real-world demo of MM-Ego, where the input video is a 2-minute long egocentric video captured by a camera on an off-the-shelf wearable device (this video is not used in our dataset). MM-Ego is able to correctly answer the episodic memory-related questions given the egocentric video, despite the difference in data domain.

2 minutes video captured by a wearable device

Question 1: Where did I leave my wallet?
MM-Ego: I left the wallet on the table. (Correct!)
Question 2: Did I see anything I can play with?
MM-Ego: Yes I have a guitar at the corner. (Correct!)
Question 3: How many times did I interact with the remoter?
MM-Ego: I interacted with the remoter three times. (Correct!)

Figure 8: Real-world conversation examples generated by MM-Ego. The input is a 2-minute long egocentric video recorded using a camera on an off-the-shelf wearable device. MM-Ego can accurately identify key visual details and provide correct answers to the user's memory-related questions.

4 Related Work

Multimodal Large Language Models. Recent advancements in Large Language Models (OpenAI, 2023; Touvron et al., 2023) have sparked significant interest in developing Multimodal Large Language Models (MLLMs) that combine the language understanding capabilities of LLMs with multi-modal perception abilities (Alayrac et al., 2022; Dai et al., 2023; Zhu et al., 2023; McKinzie et al., 2024). For video-based MLLMs, most works follow a structure akin to image-based MLLMs. To handle the large volume of video frames, some methods reduce the number of frames (Zhang et al., 2023; Wang et al., 2024; Maaz et al., 2024; Xu et al., 2024), which results in the loss of many visual details. Others extend the LLMs' context length by employing parallel techniques (Xue et al., 2024), but this often leads to low training efficiency. Unlike these approaches, our method preserves global awareness of the entire video, allows for attention to visual details, and is efficiently trainable.

Egocentric Video Understanding. While the growing field of egocentric video understanding is still in its infancy, there have been many influential works. For a comprehensive overview of egocentric vision please refer to Plizzari et al. (2024). On the data/benchmark side, representative works include Ego4D (Grauman et al., 2022), Ego-Exo4D (Grauman et al., 2024), and EPIC-KITCHENS-100 (Damen et al., 2018). When also considering language, prior work on egocentric video-language benchmarks include QaEgo4D (Bärmann & Waibel, 2022) and EgoSchema (Mangalam et al., 2023). For understanding long egocentric videos, prior modeling efforts include GroundVQA (Di & Xie, 2024), Encode-Store-Retrieve (Shen et al., 2023), and R-VLM (Xu et al., 2023). However, most previous works focus on classic video understanding tasks such as activity recognition and temporal grounding, and hence they do not involve a large language model for complex understanding. In contrast, we propose to develop an MLLM to tackle comprehensive egocentric video understanding.

5 Discussion

Limitation and Future Work. While MM-Ego demonstrates a strong ability in egocentric understanding, there is still room for further improvement. On the data and benchmark side, we can introduce more diverse egocentric understanding corpus (Grauman et al., 2024; Huang et al., 2024). For the model itself, we plan to enhance its capacity to process a larger number of frames, such as at the order of thousands, to better handle longer or even always-on egocentric videos.

Conclusion. In this paper, we make three key contributions towards the development of egocentric foundation model: the creation of the first large-scale egocentric QA training dataset, the introduction of a novel model designed for effective long egocentric video comprehension, and the establishment of the EgoMemoria benchmark for assessing models' ability to capture visual details from egocentric videos. We hope that these efforts will benefit further research on egocentric MLLMs.

ETHICS STATEMENT

Our proposed method does not involve the creation or introduction of any new video content. All generated data is derived from publicly available, privacy-protected datasets (Grauman et al., 2022). The data is intended exclusively for academic research purposes and will not be used for any commercial applications. We have adhered to ethical standards by ensuring that no private or sensitive data has been used or compromised.

REPRODUCIBILITY STATEMENT

We provide a detailed explanation of the data synthesis process in our data engine in Section 2.1. We also elaborate on our model design in Section 2.2.2. Additionally, we outline the implementation details, including the training hyperparameters in Section 3.2.

REFERENCES

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022.

Leonard Bärmann and Alex Waibel. Where did i leave my keys?-episodic-memory-based question answering on egocentric videos. In *CVPR-W*, 2022.

Minjie Cai, Kris M Kitani, and Yoichi Sato. Understanding hand-object manipulation with grasp types and object attributes. In *Robotics: Science and Systems*, volume 3, 2016.

Alejandro Cartas, Juan Marín, Petia Radeva, and Mariella Dimiccoli. Recognizing activities of daily living from egocentric images. In *Pattern Recognition and Image Analysis*, 2017.

Guiming Hardy Chen, Shunian Chen, Ruifei Zhang, Junying Chen, Xiangbo Wu, Zhiyi Zhang, Zhihong Chen, Jianquan Li, Xiang Wan, and Benyou Wang. Allava: Harnessing gpt4v-synthesized data for a lite vision-language model. *arXiv*, 2024a.

Lin Chen, Jisong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. *arXiv*, 2023a.

Lin Chen, Xilin Wei, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Bin Lin, Zhenyu Tang, Li Yuan, Yu Qiao, Dahua Lin, Feng Zhao, and Jiaqi Wang. Sharegpt4video: Improving video understanding and generation with better captions. *arXiv*, 2024b.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *arXiv*, 2023b.

Sijie Cheng, Zhicheng Guo, Jingwen Wu, Kechen Fang, Peng Li, Huaping Liu, and Yang Liu. Egothink: Evaluating first-person perspective thinking capability of vision-language models. In *CVPR*, 2024.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *arXiv*, 2023.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Scaling egocentric vision: The epic-kitchens dataset. In *ECCV*, 2018.

Shangzhe Di and Weidi Xie. Grounded question-answering in long egocentric videos. In *CVPR*, 2024.

Peter I Frazier. A tutorial on bayesian optimization. *arXiv*, 2018.

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

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh Kumar Ramakrishnan, Fiona Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Zhongcong Xu, Chen Zhao, Siddhant Bansal, Dhruv Batra, Vincent Cartillier, Sean Crane, Tien Do, Morrie Doulaty, Akshay Erapalli, Christoph Feichtenhofer, Adriano Fragomeni, Qichen Fu, Abrham Gebreselasie, Cristina Gonzalez, James Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Khoo, Jachym Kolar, Satwik Kottur, Anurag Kumar, Federico Landini, Chao Li, Yanghao Li, Zhenqiang Li, Karttikeya Mangalam, Raghava Modhugu, Jonathan Munro, Tullie Murrell, Takumi Nishiyasu, Will Price, Paola Ruiz Puentes, Merey Ramazanova, Leda Sari, Kiran Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Ziwei Zhao, Yunyi Zhu, Pablo Arbelaez, David Crandall, Dima Damen, Giovanni Maria Farinella, Christian Fuegen, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard Newcombe, Aude Oliva, Hyun Soo Park, James M. Rehg, Yoichi Sato, Jianbo Shi, Mike Zheng Shou, Antonio Torralba, Lorenzo Torresani, Mingfei Yan, and Jitendra Malik. Ego4d: Around the world in 3,000 hours of egocentric video. In *CVPR*, 2022.

Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, et al. Ego-exo4d: Understanding skilled human activity from first-and third-person perspectives. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19383–19400, 2024.

Jack Hessel, Jena D Hwang, Jae Sung Park, Rowan Zellers, Chandra Bhagavatula, Anna Rohrbach, Kate Saenko, and Yejin Choi. The Abduction of Sherlock Holmes: A Dataset for Visual Abductive Reasoning. In *ECCV*, 2022.

Tuomo Hiippala, Malihe Alikhani, Jonas Haverinen, Timo Kalliokoski, Evanfiya Logacheva, Serafina Orekhova, Aino Tuomainen, Matthew Stone, and John A Bateman. Ai2d-rst: A multimodal corpus of 1000 primary school science diagrams. *Language Resources and Evaluation*, 55:661–688, 2021.

Shaul Hochstein and Merav Ahissar. View from the top: Hierarchies and reverse hierarchies in the visual system. *Neuron*, 36(5):791–804, 2002.

Yifei Huang, Guo Chen, Jilan Xu, Mingfang Zhang, Lijin Yang, Baoqi Pei, Hongjie Zhang, Lu Dong, Yali Wang, Limin Wang, et al. Egoexolearn: A dataset for bridging asynchronous ego-and exo-centric view of procedural activities in real world. In *CVPR*, 2024.

Kushal Kafle, Brian Price, Scott Cohen, and Christopher Kanan. Dvqa: Understanding data visualizations via question answering. In *CVPR*, 2018.

Geewook Kim, Teakgyu Hong, Moonbin Yim, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. Donut: Document understanding transformer without ocr. *arXiv*, 2021.

Yong Jae Lee, Joydeep Ghosh, and Kristen Grauman. Discovering important people and objects for egocentric video summarization. In *CVPR*, 2012.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv*, 2024a.

Yanghao Li, Tushar Nagarajan, Bo Xiong, and Kristen Grauman. Ego-exo: Transferring visual representations from third-person to first-person videos. In *CVPR*, 2021.

Yanwei Li, Chengyao Wang, and Jiaya Jia. Llama-vid: An image is worth 2 tokens in large language models. In *ECCV*, 2024b.

Ji Lin, Hongxu Yin, Wei Ping, Yao Lu, Pavlo Molchanov, Andrew Tao, Huizi Mao, Jan Kautz, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models. In *CVPR*, 2024.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023.

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

Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. In *ACL*, 2024.

Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. In *NeurIPS Datasets and Benchmarks Track*, 2023.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. *arXiv*, 2022.

Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *Proc. WACV*, 2021.

Brandon McKinzie, Zhe Gan, Jean-Philippe Fauconnier, Sam Dodge, Bowen Zhang, Philipp Dufter, Dhruti Shah, Xianzhi Du, Futang Peng, Floris Weers, et al. Mm1: Methods, analysis & insights from multimodal llm pre-training. *arXiv*, 2024.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. *arXiv*, 2016.

OpenAI. ChatGPT: Optimizing language models for dialogue. `https://openai.com/blog/chatgpt`, 2023. Accessed: 2023.

Chiara Plizzari, Gabriele Goletto, Antonino Furnari, Siddhant Bansal, Francesco Ragusa, Giovanni Maria Farinella, Dima Damen, and Tatiana Tommasi. An outlook into the future of egocentric vision. *IJCV*, pp. 1–57, 2024.

Junxiao Shen, John Dudley, and Per Ola Kristensson. Encode-store-retrieve: Enhancing memory augmentation through language-encoded egocentric perception. *arXiv*, 2023.

Gunnar A. Sigurdsson, Abhinav Gupta, Cordelia Schmid, Ali Farhadi, and Karteek Alahari. Charades-ego: A large-scale dataset of paired third and first person videos. In *arXiv*, 2018.

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

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. Pointer networks. *NeurIPS*, 28, 2015.

Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Jilan Xu, Zun Wang, et al. Internvideo2: Scaling video foundation models for multimodal video understanding. *arXiv*, 2024.

Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. In *CVPR*, 2021.

Jiaqi Xu, Cuiling Lan, Wenxuan Xie, Xuejin Chen, and Yan Lu. Retrieval-based video language model for efficient long video question answering. *arXiv*, 2023.

Mingze Xu, Mingfei Gao, Zhe Gan, Hong-You Chen, Zhengfeng Lai, Haiming Gang, Kai Kang, and Afshin Dehghan. Slowfast-llava: A strong training-free baseline for video large language models. *arXiv*, 2024.

Fuzhao Xue, Yukang Chen, Dacheng Li, Qinghao Hu, Ligeng Zhu, Xiuyu Li, Yunhao Fang, Haotian Tang, Shang Yang, Zhijian Liu, et al. Longvila: Scaling long-context visual language models for long videos. *arXiv*, 2024.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2 technical report. *arXiv*, 2024.

Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. Activitynet-qa: A dataset for understanding complex web videos via question answering. In *AAAI*, 2019.

Semir Zeki. Area v5—a microcosm of the visual brain. *Frontiers in integrative neuroscience*, 9:21, 2015.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *ICCV*, 2023.

Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv*, 2023. URL `https://arxiv.org/abs/2306.02858`.

Peiyuan Zhang, Kaichen Zhang, Bo Li, Guangtao Zeng, Jingkang Yang, Yuanhan Zhang, Ziyue Wang, Haoran Tan, Chunyuan Li, and Ziwei Liu. Long context transfer from language to vision. *arXiv*, 2024a.

Yuanhan Zhang, Bo Li, haotian Liu, Yong jae Lee, Liangke Gui, Di Fu, Jiashi Feng, Ziwei Liu, and Chunyuan Li. Llava-next: A strong zero-shot video understanding model, April 2024b. URL `https://llava-vl.github.io/blog/2024-04-30-llava-next-video/`.

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

A MORE ANALYSIS OF MEMORY POINTER PROMPTING

To further assess the effectiveness of MM-Ego and the proposed Memory Pointer Prompting mechanism, we present additional visual results of key frame identification during the global glimpse step in Figure 9. MM-Ego demonstrates the capability to extract relevant visual information from a large set of frames based on the given questions.

Question 1: Where did I walk towards with the hose in my hands?
Choices: **A**: I walked backward towards a stone wall. **B**: I walked sideways towards a wooden shed. **C**: I walked around towards a brick pathway. **D**: I walked forward towards an iron fence.

Prediction: D
Label: D

Selected Frames

Question 2: What was the color of the tape I tried to remove from the wood?
Choices: **A**: The color of the tape was blue. **B**: The color of the tape was red. **C**: The color of the my was green. **D**: The color of the tape was yellow.

Prediction: D
Label: D

Selected Frames

Question 3: Which hand did I use to pick the frying pan from the boot of the pickup truck?
Choices: **A**: I picked a frying pan from the boot of the pickup truck with my right hand. **B**: 1. I picked a frying pan from the boot of the pickup truck with my left hand. **C**: 2. I picked a fryingpan from the boot of the pickup truck with both hands. **D**: 3. I picked a frying pan from the boot of the pickup truck using a cloth in my left hand.

Prediction: A
Label: A

Selected Frames

Question 4: What did I pass to my left hand?
Choices: **A**: I passed the cup to my left hand. **B:** I passed the plate to my left hand. **C:** I passed the book to my left hand. **D:** I passed the remote to my left hand.

Prediction: B
Label: B

Selected Frames

Question 5: What action did I take with the frying pan at the end?
Choices: **A**: A: I moved the frying pan on the cooker with my left hand and then stirred the content with the chopsticks. **B**: I placed the frying pan in the sink and washed it using a sponge and dish soap. **C**: I transferred the frying pan to the dining table and served the food onto plates. **D**: I hung the frying pan on the wall rack and wiped the stove clean.

Prediction: A
Label: A

Selected Frames

Question 6: How did I add spice to the frying pan the first time?
Choices: **A**: I grabbed a handful of spice and sprinkled it over the frying pan. **B**: I shook the spice container directly above the frying pan. **C**: I measured the spice with a teaspoon and added it to the frying pan. **D**: I scooped out some spice with the spoon and poured it in the frying pan.

Prediction: D
Label: D

Selected Frames

Figure 9: More key frame identification results of the global glimpse step on EgoMemoria. We find high relevance between the identified key frames and the questions, demonstrating the effectiveness of the proposed Memory Pointer Prompting method. The ✓ indicates that the selected frames are relevant to the questions.

B More Discussion on Related Works

Egocentric QA Data Generation After finishing the project, we find that the generation process in MM-Ego data engine shares some similar processes with the recently published LLaMA-VID (Li et al., 2024b) and GroundVQA (Di & Xie, 2024). LLaMA-VID utilizes movie synopses to prompt LLMs to produce movie summaries and plot-related QA pairs. GroundVQA generates short-term (around 8 minutes) episodic memory QA from video narrations, but the goal and implementation details are different. MM-Ego collects and processes videos with significantly more diverse video lengths from 30 seconds to 1 hour. The scale of our produced egocentric QA dataset is also significantly larger (7M vs. 303K).

Egocentric QA Evaluation Benchmark There have been some benchmarks for assessing the performance of MLLMs in understanding egocentric images (Cheng et al., 2024) and videos (Mangalam et al., 2023; Di & Xie, 2024; Bärmann & Waibel, 2022). QAEgo4D (Di & Xie, 2024) benchmark assesses shorter-term video understanding (around 8 minutes) with a considerably smaller dataset (1,850 questions across 166 videos). EgoSchema (Mangalam et al., 2023) has more video clips, yet the video lengths are still relatively short. Unlike these, our EgoMemoria benchmark encompasses a significantly wider range of video lengths, spanning from 30 seconds to 1 hour. The extended video lengths make the benchmark significantly more challenging and closer to real-world egocentric video use cases. We compare these benchmarks in Table 6.

Benchmark	Videos	QAs	Lengths
QAEgo4D (Di & Xie, 2024)	166	1850	8 min
EgoSchema (Mangalam et al., 2023)	**5063**	5063	3 min
EgoMemoria (ours)	629	**7026**	**0.5–60 min**

Table 6: Comparison with exiting egocentric video understanding benchmarks.

C Fine-tuning Exclusively on Egocentric QA Data

To evaluate model performance when fine-tuning exclusively on egocentric QA data, we conduct an ablation study, with the results presented in Table 7. For both LLaVA-OV (Li et al., 2024a) and MM-Ego, we consider two variants: one fine-tuned on our comprehensive data mixture and the other trained solely on egocentric QA data. The results demonstrate further performance improvements on the EgoMemoria benchmark. However, it is important to note that such domain-specific fine-tuning largely restricts the models' capacity for general video understanding. On the other hand, we observe that MM-Ego still achieves significantly better performance in egocentric video understanding, attributed to the learning of the memory pointer prompting mechanism.

Name	Short	Medium	Long	Avg
LLaVA-OV (Data Mixture)	59.56	56.71	51.64	55.97
LLaVA-OV (Egocentric Data Only)	62.58	58.13	53.61	58.11
MM-Ego (Data Mixture)	63.16	62.04	58.62	61.27
MM-Ego (Egocentric Data Only)	65.41	64.89	61.05	63.78

Table 7: Performance comparison of different models on EgoMemoria (MDA) with fine-tuning exclusively on egocentric QA data.

D Analysis of Using Language Model in Data Engine

The motivation for using a language model to convert egocentric video captions into egocentric QA conversations is to address the "chicken-or-egg dilemma". If we rely on a vision-language model (VLM) to generate egocentric QA pairs, the quality of the data is inherently limited by the egocentric understanding capabilities of the labeling VLM. Consequently, downstream VLMs trained on this synthetic data cannot outperform the labeling VLM. This creates the chicken-or-egg problem: do we have a strong egocentric VLM first, or do we have good egocentric QA data first?

To address this dilemma, our "narration-to-egocentric QA" data engine leverages a language-only model to generate QA samples. This approach circumvents the hallucination and inaccuracy issues often associated with long video understanding, which is a very challenging task by itself. Since we have access to high-quality, fine-grained, and densely annotated video narrations created by human labelers in Ego4D, the essential visual information has already been effectively translated into the narrations.

We conduct a preliminary experiment of using image frames augmented GPT-4o to produce egocentric QA. We uniformly sample 32 frames with a resolution of 336x448 from a 30 seconds egocentric video.

Input Narrations: VID_NARRATION = ["I close the tap.", "I press the buttons on the rice cooker.", "I open the rice cooker.", "I stir the rice.", "I close the rice cooker.", "I press the buttons on the rice cooker."]

Text Prompts: ``Please design at most 5 questions and answers about self-questioning my memory based on a video. For example, ask questions about what I have done, what objects did I interact, where did I go, what did other people do, the locations, colors, and shapes of the objects in the scenes if mentioned, based solely on the video narrations $VID_NARRATION. The answers should be in first-person perspective or objective description of the objects. Remember this rule: special word 'I', 'c', or 'C' represent me (first-person perspective), special word 'x', 'y', 'Y', 'Z', 'z', 'o', 'O', 'X', etc. represent somebody else, this is very important. Ask very diverse questions that you are sure about the answers, and the questions cannot be answered without watching the video, which means no common sense questions. Don't generate anything if you do not have enough information. Attach index of the sentence that answer is based upon. Directly reply by strictly following this example: 'QUESTION: Where did I leave my key? ANSWER:I put it on the table._0,2 QUESTION:Did I close the door when I left? ANSWER:Yes I did._3 QUESTION:What is the color of the cup? ANSWER:White._1''

Generated Egocentric QAs: We present the generated QAs by GPT-4o both with and without video input in Table 8 and Table 9. In this example, we observe that the generated QAs have similar quality whether videos are included or not. However, incorporating video into GPT-4o's input significantly increases the time and computational costs. Moreover, when applying VLM to a large volume of videos, it becomes challenging to verify if any visual hallucinations arise from the video understanding process. Therefore, we choose to use a mature pure language model in our data engine.

Question	**Answer**
What did I do to the tap in the video?	I closed the tap.
How did I interact with the rice cooker buttons?	I pressed the buttons on the rice cooker twice.
What did I use to stir the rice?	I used a spoon to stir the rice.
Did I leave the rice cooker open after stirring the rice?	No, I closed the rice cooker.
At what points in the video was the rice cooker interacted with?	The rice cooker was opened, stirred, and closed.

Table 8: GPT-4o generated egocentric QAs with video input.

Question	Answer
What action did I perform on the tap?	I closed it.
What did I do after pressing the buttons on the rice cooker the first time?	I opened the rice cooker.
What did I use to stir the rice?	I stirred the rice.
How many times did I press the buttons on the rice cooker?	Twice.
What sequence of actions involved the rice cooker?	I pressed the buttons, opened it, stirred the rice, closed it, and pressed the buttons again.

Table 9: Generated egocentric QAs without video input.

E MM-EGO'S CONNECTION WITH COGNITIVE NEUROSCIENCE

Our proposed "Memory Pointer Prompting" two-step mechanism is inspired by the way human naturally process videos. When answering a specific question about a long video, we typically start by quickly scanning the entire video to identify frames relevant to the question, which is similar to our "global glimpse" step. We then closely examine those frames to find the answers, which resembles our "fallback" step.

In the study of human visual perception system, researchers identify two distinct pathways for processing visual information in the brain: the magnocellular and parvocellular pathways (Zeki, 2015; Hochstein & Ahissar, 2002). Our "global glimpse" step mirrors the functionality of the magnocellular pathway which is responsible for handling information about large, fast-moving objects. On the other hand, our "fallback" step aligns with the role of the parvocellular pathway which specializes in processing details of small, slow-moving objects.