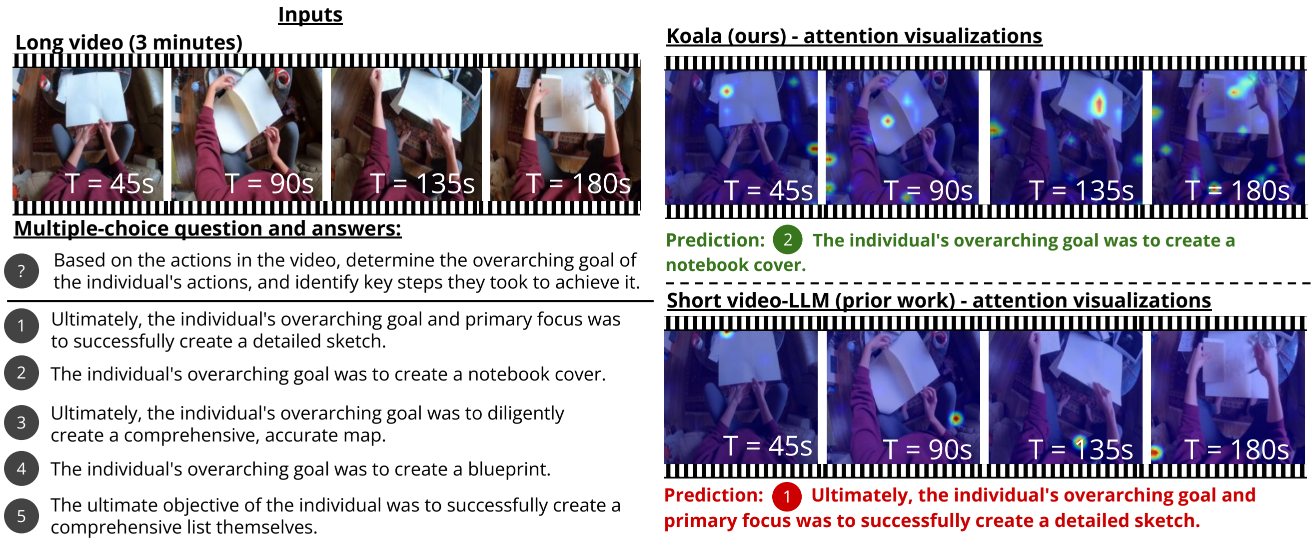
Koala: Key frame-conditioned long video-LLM

Reuben Tan[1] Ximeng Sun[1] Ping Hu[2] Jui-hsien Wang[3] Hanieh Deilamsalehy[3]
Bryan A. Plummer[1] Bryan Russell[3] Kate Saenko[1]
[1]Boston University, [2]University of Electronic Science and Technology of China, [3]Adobe Research
{rxtan, sunxm, pinghu, bplum, saenko}@bu.edu, {juiwang, deilamsa, brussell}@adobe.com
`https://cs-people.bu.edu/rxtan/projects/Koala`



Multiple-choice question and answers:

? Based on the actions in the video, determine the overarching goal of the individual's actions, and identify key steps they took to achieve it.

1. Ultimately, the individual's overarching goal and primary focus was to successfully create a detailed sketch.
2. The individual's overarching goal was to create a notebook cover.
3. Ultimately, the individual's overarching goal was to diligently create a comprehensive, accurate map.
4. The individual's overarching goal was to create a blueprint.
5. The ultimate objective of the individual was to successfully create a comprehensive list themselves.

Prediction: ② The individual's overarching goal was to create a notebook cover.

Prediction: ① Ultimately, the individual's overarching goal and primary focus was to successfully create a detailed sketch.

Figure 1. Given a video-Large Language Model that was pretrained on millions of short *seconds*-long video clips, we propose a lightweight approach (Koala) to extend its short-term video tokenizer function for understanding and answering questions about *minutes*-long videos. We are the first to use sparsely sampled key frames to condition the LLM. As shown, our Koala approach is more effective at focusing on relevant regions in the input frames than the short vLLMs, allowing it to make more informed predictions based on a more holistic understanding of the video. These regions help facilitate our model in predicting the correct answer to the question (highlighted in green).

Abstract

Long video question answering is a challenging task that involves recognizing short-term activities and reasoning about their fine-grained relationships. State-of-the-art video Large Language Models (vLLMs) hold promise as a viable solution due to their demonstrated emergent capabilities on new tasks. However, despite being trained on millions of short seconds*-long videos, vLLMs are unable to understand* minutes*-long videos and accurately answer questions about them. To address this limitation, we propose a lightweight and self-supervised approach, Key frame-conditioned long video-LLM (Koala), that introduces learnable spatiotemporal queries to adapt pretrained vLLMs for generalizing to longer videos. Our approach introduces two new tokenizers that condition on visual tokens computed from sparse video key frames for understanding short and long video moments. We train our proposed approach on HowTo100M and demonstrate its effectiveness on zero-shot long video understanding benchmarks, where it outperforms state-of-the-art large models by 3 - 6% in absolute accuracy across all tasks. Surprisingly, we also empirically show that our approach not only helps a pretrained vLLM to understand long videos but also improves its accuracy on short-term action recognition.*

1. Introduction

Answering questions about minutes-long videos is an inherently challenging task that involves recognizing multiple actions and how they fit together to form the overall activity. To recognize that the person is making a notebook cover instead of a sketch in Figure 1, a model must spot key actions (taping, measuring) and objects (paper), and understand how they are related to each other. Instruction-tuned multimodal-Large Language Models (mLLMs) [12, 34, 38, 69, 72] and their video variants (vLLMs) [35, 40, 41, 70] offer a promising avenue for understanding long videos, as demonstrated by their emergent capabilities in downstream multimodal tasks including perception [56] and commonsense reasoning [12, 62]. By learning to tokenize a small number of key frames from *seconds*-long videos into visual tokens that are mapped to the same latent space as language word tokens, vLLMs are able to leverage the knowledge encapsulated in their LLM to describe visual concepts, such as actions, in short videos.

However, existing vLLMs trained on millions of short videos still struggle with *minutes*-long videos that contain significantly more frames [32]. A naive solution is to extract the same number of key frames at a coarse rate, but this leads to a significant loss of fine-grained spatiotemporal information. Thus, this approach results in poor performance on complex and long-term temporal understanding tasks in benchmarks including EgoSchema [42] and Seed-Bench [32]. Another possibility for extending these pretrained vLLMs to long videos is to pass multiple segments of key frames into their learned tokenizer function. However, this extension may negatively affect the ability of the vLLMs to understand long videos holistically since their tokenizer function only aggregates spatiotemporal context *within* segments rather than *between* them.

In light of these limitations, we propose our Key frame-conditioned long video-LLM (Koala), a novel and self-supervised approach that introduces spatiotemporal queries to adapt the *frozen* video tokenizer in pretrained vLLMs to aggregate spatiotemporal context over longer temporal horizons. Our main hypothesis is that the video tokenizer function in vLLMs, having learned to aggregate spatiotemporal context for a fixed number of frames, can generalize to understanding longer videos using the same number of input frames. More specifically, we first encode the global context of a long video by extracting the same number of input frames at a very coarse sampling rate, referred to as *key frames*. To mitigate the loss of fine-grained spatiotemporal information, we then extract a sequence of video segments at a higher sampling rate to complement the global context with local spatiotemporal information.

The key insight underlying Koala is that the global video context can be utilized to model individual video segments and the contextual relations *between* multiple video segments, which plays a crucial role in understanding long videos. To this end, we further introduce our Conditioned Segment (CS) and Conditioned Video (CV) tokenizer functions. Intuitively, the former function leverages learnable segment queries that use the global context of the video to identify and aggregate frame-level concepts within each segment; such concepts are important to both short-term context of the segment and the global context of the entire video. The latter function further introduces temporal concept queries to reason about the contextual relationships between segments to generate an enriched sequence of visual tokens as inputs into the subsequent LLM.

While the idea of using frames extracted at different sampling rates bears similarities to existing approaches [28, 37, 52] including slowfast network [16], these aforementioned approaches focus on modeling static and motion contexts in *short* videos, especially in a closed-world setting. In contrast, we focus on a task-agnostic approach for computing enriched visual tokens that are well-aligned with the base LLMs. More significantly, reasoning about global and short-term semantics of videos in vLLMs makes our setting different and challenging. By facilitating long video understanding with LLMs, our Koala approach helps to address the inherent problem of summarizing and understanding high-level temporal context which is prevalent in downstream open-world applications including video recommendations [22, 25, 39], embodied AI [13, 19, 29, 33] and robotics [31, 47].

We demonstrate the effectiveness of our proposed Koala approach through extensive evaluations on multiple zero-shot long and short-term temporal understanding tasks on the EgoSchema [42] and the Seed-Bench [32] benchmarks. We show that our proposed light-weight finetuning approach is able to incorporate long-term temporal understanding capabilities into pretrained vLLMs despite training on noisy and uncurated video and text data from the Howto100M dataset [43], and outperforms state-of-the-art mLLMs by a significant margin of 3 - 6% across all tasks. Furthermore, we show that our CS and CV tokenizer functions also help the base vLLM to improve its performance on short-term action recognition. We provide a comprehensive ablation of our approach to analyze the effectiveness of the spatiotemporal queries introduced in the proposed tokenizer functions in Koala. We are the first work to explore extending the video tokenizer function of pretrained short-term vLLMs to long-term video understanding.

2. Related work

Video understanding. The field of video understanding encompasses core research problems including action recognition [17, 64], action prediction [21] and temporal action localization [37]. Older prior work addressing these problems are often task-specific and either rely on hand-crafted features [30, 57, 68] or video encoders that are carefully designed to exploit temporal information from RGB frames and optical flow information [9, 15, 16, 55]. Moreover, understanding action sequences has often been constrained to short video clips. vLLMs are also similar to more recent fully attentional video encoders [5, 7, 14, 44] that leverage self-attention between spatiotemporal regions to compute more effective video representations.

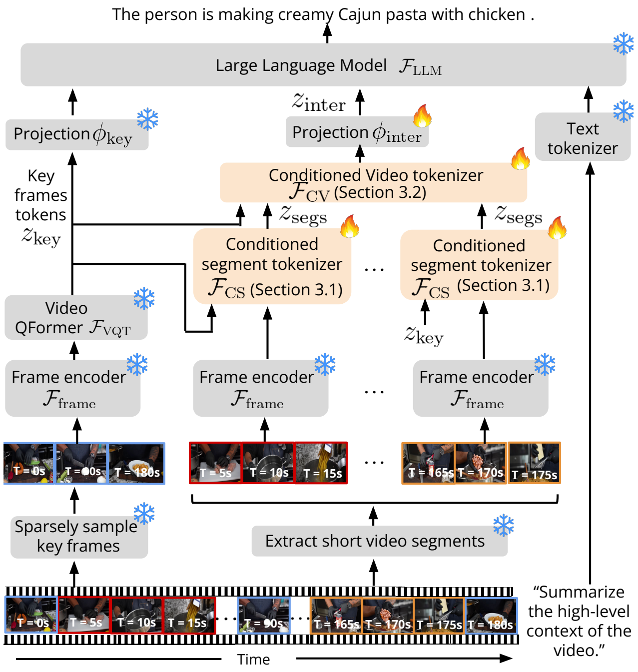


Figure 2. **Overview of our full Koala approach.** For a given video, we extract a set of coarsely-sampled key frames and non-overlapping frame segments with a much higher sampling rate. We use the key frames to provide high-level global context of the video to compute a final sequence of soft visual tokens that encode both global context as well as fine-grained spatiotemporal information via the Conditioned Segment (CS) and Conditioned Video (CV) tokenizer functions.

Additionally, there are also existing works which aim to address the task of understanding long videos [58, 63]. While these aforementioned approaches are similar in spirit to our proposed approach, they are focused on recognizing actions instead of generating language as in our case.

Instruction-tuning and multimodal foundation models. Recently, instruction-tuned multimodal-LLMs [12, 35, 40, 69, 70] have demonstrated surprising emergent capabilities on unseen tasks. We make the distinction between two main types of multimodal LLMs - image-based [4, 38, 51, 59, 69, 72] and video-based [35, 40, 41, 67, 70]. In general, mLLMs learn an adaptor between the frozen visual encoders and the LLM that generates a sequence of soft visual tokens. The base LLMs are often kept frozen or lightly finetuned with the LORA framework [24] to leverage their vast amount of knowledge gleaned from large-scale pretraining [10, 11, 54, 71]. While our proposed Koala model is also built upon a base vLLM, a key difference between prior mLLMs and ours lies in the way temporal information is aggregated in the video domain. Prior vLLMs [40, 41, 70] are often pretrained on large-scale and publicly available video and text datasets, as well as a highly curated instructional video dataset that has been annotated with temporal and spatial relations by Chat-GPT [2]. However, despite tuning on this dataset, state-of-the-art video-LLMs are still limited at understanding temporal relationships. Furthermore, while there are existing multimodal approaches [20, 53] that have also been introduced to address the task of long video question answering, they differ from ours in different aspects. [20] conditions the computation of visual attention on the question but ours uses global visual context. [53] relies on fine-grained paragraph annotations while ours only relies on coarse and noisy goal labels.

Comparisons to existing prompting approaches. Koala is similar in spirit to existing approaches that use learnable queries for foundational image-text models [48] for short-term action recognition [27, 45, 61]. However, their purpose is introducing temporal prompts to transform the learned spatial aggregation function to reason about the temporal relations between a small number of frames. In contrast, we use spatiotemporal prompts to extend the learned short-term temporal aggregation function for long-term understanding of videos at least 10 times longer. Furthermore, our proposed approach provides an efficient mechanism for aggregating long-term temporal context over multiple segments.

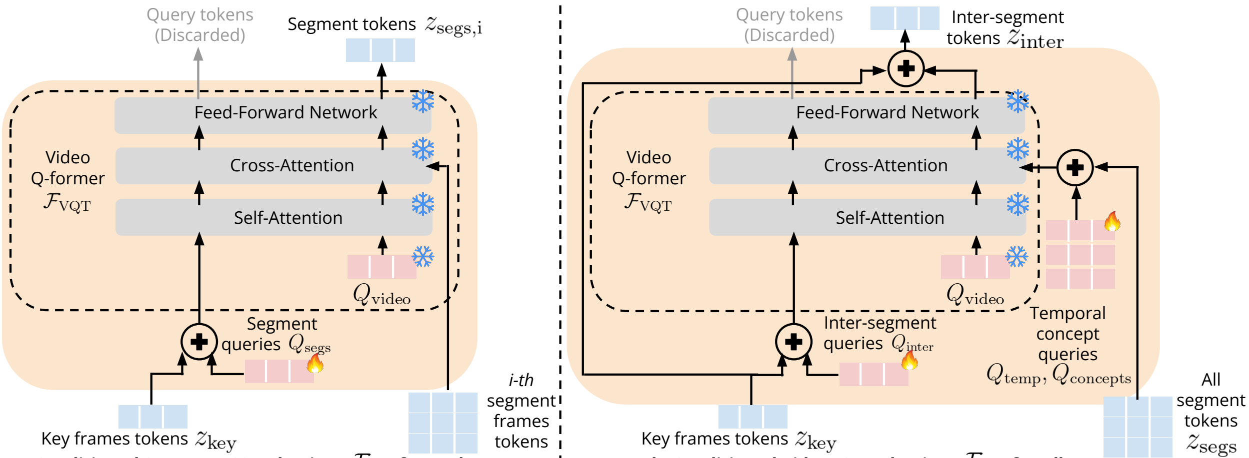
3. Koala

We propose Koala, a lightweight finetuning approach that takes a frozen vLLM, which is pretrained on short video clips, and adapts it to longer temporal settings. The key components of Koala are visual tokenizers that condition on representations of a sparse set of video key frames to adaptively select and aggregate information at the *segment* and *video* levels. We assume the vLLMs [35, 70] are trained to generate a textual response that is conditioned on an input text query and a short (seconds-long) video. The input text query is encoded to a set of text tokens z_{text}. To encode the video V, the pretrained vLLM samples a fixed number of key frames $V_{\text{key}} \subset V$, and then applies a key frames tokenizer function $\mathcal{F}_{\text{key}}$. $\mathcal{F}_{\text{key}}$ aggregates the spatiotemporal context over the visual features within the set of key frames and returns a set of key frames tokens z_{key}.

Let $z_{\text{key}} = \mathcal{F}_{\text{key}}(V_{\text{key}}) = \mathcal{F}_{\text{VQT}}(\mathcal{F}_{\text{frame}}(V_{\text{key}}); Q_{\text{video}})$, where $\mathcal{F}_{\text{VQT}}$ and $\mathcal{F}_{\text{frame}}$ denote pretrained video QFormer [34, 70] and frame encoding functions, respectively. Similar to the Perceiver model [26], $\mathcal{F}_{\text{VQT}}$ is partly parameterized by a set of frozen video queries Q_{video} (*cf*., Figure 3) for aggregating the spatiotemporal information within V_{key}. In this work, we term the information encoded by z_{key} as the global context of the video. Given the sets of text and key frames tokens z_{text} and z_{key}, respectively, the LLM function $\mathcal{F}_{\text{LLM}}$ computes the output textual response r as:

$$r = \mathcal{F}_{\text{LLM}}(\text{concat}\{z_{\text{text}}, \phi_{\text{key}}(z_{\text{key}})\}), \quad (1)$$

where concat$\{\}$ is the concatenation operation and ϕ_{key} is an affine transformation that projects the visual tokens to the LLM token space.



(a) Conditioned Segment (CS) tokenizer $\mathcal{F}_{\text{CS}}$ for *each* segment

(b) Conditioned Video (CV) tokenizer $\mathcal{F}_{\text{CV}}$ for *all* segments

Figure 3. **CS and CV tokenizer functions.** (a) Our CS tokenizer introduces learnable segment queries and fuses the global semantics of a video with fine-grained frame concept representations within each segment to compute segment tokens. (b) In the CV module, we introduce learnable inter-segment queries as well as temporal concept queries to model the contextual relations between segments.

While the key frames tokenizer $\mathcal{F}_{\text{key}}$ encodes the global context of a long video by reasoning about the high-level relationships between key frames, the coarse sampling rate results in a loss of fine-grained spatiotemporal information that is crucial for understanding long videos effectively. To address this limitation, we propose to enrich the key frames tokens with the spatiotemporal information of *local* video segments, illustrated in Figure 2. Specifically, we compute a set of contextualized inter-segment tokens z_{inter} from N non-overlapping video segments $S = \{S_1, \cdots, S_N\}$, where each segment $S_i \subset V$ is sampled at a higher frame rate. We modify Eq (1) to include the inter-segment tokens z_{inter} and the learnable affine transformation ϕ_{inter}:

$$r = \mathcal{F}_{\text{LLM}}(\text{concat}\{z_{\text{text}}, \phi_{\text{key}}(z_{\text{key}}), \phi_{\text{inter}}(z_{\text{inter}})\}). \quad (2)$$

To compute z_{inter}, we introduce our Conditioned Segment (CS) and Conditioned Video (CV) tokenizer functions, which repurpose the *frozen* $\mathcal{F}_{\text{VQT}}$ function to select local spatiotemporal information that are most relevant to the conditioned global context at the *segment* and *video* levels.

At the segment level, our CS tokenizer $\mathcal{F}_{\text{CS}}$ (Section 3.1) uses learnable queries that are conditioned on the encoded global context of z_{key} to identify visual concepts in frames. We seek visual concepts that are not only relevant to the local context within each segment, but also to the global context of the entire video. This context is needed because $\mathcal{F}_{\text{VQT}}$ only aggregates the contextual relationships *within* segments of frames and not *between* them. **At the video level**, our CV tokenizer function $\mathcal{F}_{\text{CV}}$ (Section 3.2) leverages $\mathcal{F}_{\text{VQT}}$ to reason about the contextual relationships of spatiotemporal concepts across different segments conditioned on the global context of z_{key}. Taken together, the final sequence of contextual inter-segment tokens z_{inter} is the output of the composition of these tokenizers:

$$z_{\text{inter}} = \mathcal{F}_{\text{CV}}(\{\mathcal{F}_{\text{CS}}(S_i \mid z_{\text{key}})\}_{i=1}^{N} \mid z_{\text{key}}) \quad (3)$$

Note that the attention mechanism encapsulated by the CS and CV tokenizers facilitates the dissemination of global video context to more fine-grained visual concepts. Finally, we describe our learning objective in Section 3.3.

3.1. Conditioned Segment Tokenizer

We illustrate our Conditioned Segment (CS) tokenizer in Figure 3a. This tokenizer repurposes the key frames tokenizer $\mathcal{F}_{\text{key}}$ to select important frame-level information that is pertinent to both the local context of each segment and the global context of the key frames tokens z_{key}. As we will demonstrate empirically in Section 4.2, naively increasing the number of key frames as input into $\mathcal{F}_{key}$ during finetuning does not help the vLLM to generalize to longer videos, even when accounting for the quadratic complexity of the attention operation.

For video segment $S_i \in S$, we repurpose the key frames tokenizer $\mathcal{F}_{\text{key}}$ via two simple modifications to the video QFormer $\mathcal{F}_{\text{VQT}}$. First, we concatenate the key frame tokens z_{key} with the video QFormer's pretrained video queries Q_{video}. This modification allows the video QFormer to condition on the key frame tokens when aggregating the input video segment features $\mathcal{F}_{\text{frame}}(S_i)$ via cross-attention with Q_{video} and z_{key}. Second, to ensure that the key frame tokens z_{key} are compatible with the video QFormer, we adapt them via addition with a set of learnable queries Q_{segs}. For video segment S_i, we define the CS tokenizer $\mathcal{F}_{\text{CS}}$ as:

$$\mathcal{F}_{\text{CS}}(S_i \mid z_{\text{key}}) = \mathcal{F}_{\text{VQT}}(\mathcal{F}_{\text{frame}}(S_i); \text{concat}\{Q_{\text{video}}, z_{\text{key}} + Q_{\text{segs}}\}). \quad (4)$$

Note that this CS tokenizer outputs tokens for Q_{video} and z_{key}. We empirically find that it is beneficial to discard the output tokens for z_{key}.

3.2. Conditioned Video Tokenizer

While our CS tokenizer helps to augment the local context of segment tokens with the global context of the entire video, the resulting tokens for each segment still lack contextual information from other segments. As such, we further propose our Conditioned Video (CV) tokenizer $\mathcal{F}_{\text{CV}}$ to reason about important spatiotemporal relationships *across* segments (Figure 3b).

Modeling spatiotemporal context across segments. We model how the local segments are related to each other conditioned on the global context of the entire video. This objective involves a granular understanding of how specific concepts such as entities and action sequences are interconnected throughout the entire video. Let $z_{\text{segs},i} = \mathcal{F}_{\text{CS}}(S_i \mid z_{\text{key}})$ be the set of conditioned tokens for segment S_i. To ensure that these tokens are compatible with the video QFormer $\mathcal{F}_{\text{VQT}}$, we introduce a set of learnable temporal queries Q_{temp}, where the i-th query $Q_{\text{temp},i}$ is added to all tokens in $z_{\text{segs},i}$. Furthermore, we introduce learnable concept queries Q_{concepts}, where the t-th query $Q_{\text{concepts},t}$ is added to the t-th token across all segment tokens $z_{\text{segs}} = \{z_{\text{segs},i}\}_{i=1}^{N}$. Taken together, we compute the adapted segment tokens for the t-th token of segment S_i as:

$$Q_{\text{final},i,t} = z_{\text{segs},i,t} + Q_{\text{temp},i} + Q_{\text{concepts},t}. \tag{5}$$

We denote the full adapted segment token set as $Q_{\text{final}} = \{Q_{\text{final},i,t}\}_{i,t}$. Similar to our $\mathcal{F}_{\text{CS}}$ function, we introduce learnable inter-segment queries Q_{inter} to adapt the key frames tokens z_{key} to be compatible with the video QFormer $\mathcal{F}_{\text{VQT}}$. We define our CV tokenizer as a weighted sum of the key frames tokens (to retain the global video context) and the repurposed video QFormer $\mathcal{F}_{\text{VQT}}$:

$$\mathcal{F}_{\text{CV}}(z_{\text{segs}} \mid z_{\text{key}}) = z_{\text{key}} + w\mathcal{F}_{\text{VQT}}(Q_{\text{final}}; \text{concat}\{Q_{\text{video}}, z_{\text{key}} + Q_{\text{inter}}\}), \tag{6}$$

where w is a learnable scalar.

3.3. Learning objective

We define the learning objective for optimizing the parameters of the introduced tokenizer functions $\mathcal{F}_{\text{CS}}$ and $\mathcal{F}_{\text{CV}}$ and the global affine transformation ϕ_{inter} as predicting the high-level task labels of instructional videos spanning at least a few minutes. This objective is akin to summarizing the long videos concisely. Given the instruction-tuned nature of the pretrained vLLM, we convert the high-level task labels such as "fix a car engine" into the instruction format by manually crafting a set of query and response templates for training (see supplemental). Let P be a question prompt for a given input video V and R its corresponding response comprising a sequence of M words $R = \{\hat{l}_1, \cdots, \hat{l}_M\}$ (each word is represented as a one-hot vector). We minimize the cross-entropy loss:

$$\mathcal{L}(V, P, R) = -\sum_{j=1}^{M} \hat{l}_j \log p(l_j | \hat{l}_{<j}, V, P), \tag{7}$$

where $p(l_j | \hat{l}_{<j}, V, P)$ denotes the probabilities for the j-th word given the preceding ground truth words $\hat{l}_{<j}$.

4. Experiments

Datasets. We train our Koala approach on a filtered subset of 250K videos from the HowTo100M instructional video dataset [43]. The filtered subset contains longer videos that span from four to over thirty minutes. Please see the supplemental for details on how we filter the training data. We evaluate our approach on two zero-shot long video question answering tasks – the multiple choice format in EgoSchema [42] and procedure-understanding in Seed-Bench [32]. Additionally, we evaluate on the task of short-term action recognition [32] to analyze if the introduced CS and CV functions are detrimental to understanding short videos. Note that we report the best results across different numbers of frames.

Implementation details. We build our approach off the publicly available Video-LLama model [70] and train for 2 epochs on the final filtered subset of Howto100M. During evaluation, we compute the log-likelihood for each candidate answer and select the highest-scoring option for fair comparison [8, 32]. We provide further details about our training setup in the supplemental.

4.1. Quantitative comparison to baselines

Besides the tasks of long video question answering and procedure understanding on the EgoSchema and Seed-Bench benchmarks, we also evaluate our Koala model on short-term action recognition.

EgoSchema evaluation. We report the results of our zero-shot evaluation on the EgoSchema benchmark in Table 1. In addition to state-of-the-art vLLMs, we also compare our proposed Koala approach to language prior baselines that are not included in Mangalam *et al.* [42]. The language prior baselines only use the questions and candidate answers for predictions. Please refer to the supplemental for more details on how we prompt these LLMs given a question and each candidate answer. Note that we also modify the questions and answers to replace "C" with "the camera wearer" so that the words used are more similar to the data used to pretrain these language models. To begin, we observe that the Flan-T5 [11] language prior serves as a very strong baseline on this benchmark. Despite not relying on the input videos at all, this language prior baseline outperforms most of the state-of-the-art video-language models by a significant margin. In some cases, Frozen-BiLM and InternVideo have also been finetuned on QA datasets including How2QA [36] and MSRVTT [65]. This finding suggests that existing vLLMs are not able to perform long-term temporal reasoning well although they have been trained on large-scale curated video and text data.

To better understand this finding, we also conduct an analysis of different state-of-the-art LLMs to determine their impact. In Table 2, we see that the language prior accu-

Approach	Training	LLM	LLM architecture	# input frames	Top 1 Acc (%)
Human accuracy (upper bound)	-	-	-	-	76.20
Language prior	-	Flan-T5-xl	Encoder-decoder	-	35.92
Random	-	-	-	-	20.00
VIOLET [18]	-	Bert-Base [50]	Encoder	5	19.90
Frozen-BiLM [66]	MLM	DeBERTa-V2-XLarge [23]	Encoder	90	26.90
Video-Llama (finetuned)	Captioning	Llama-2	Decoder	32	28.36
mPLUG-Owl [67]	Captioning	Llama	Decoder	5	31.10
InternVideo [60]	Contrastive	CLIP	Encoder	90	32.10
Video-Llama [70]	Captioning	Llama-2	Decoder	128	33.25
MovieChat [49]	Captioning	Llama-2	Decoder	128	33.49
Koala (ours)	Captioning	Llama-2	Decoder	64	**40.42**

Table 1. **Zero-shot long video question answering on EgoSchema benchmark.** For all models, we report the best results obtained across varying number of input frames. Our Koala approach outperforms the base Video-Llama model despite using much fewer frames. We also include the results for a strong language prior baseline as well as human performance (highlighted in gray).

LLM	Architecture	Top 1 Acc (%)
Random	-	20.00
GPT-J	decoder-only	9.94
GPT-Neo	decoder-only	17.21
Vicuna	decoder-only	21.45
Llama-2	decoder-only	26.03
Flan-T5-xl	encoder-decoder	**35.92**

Table 2. **Zero-shot long video question answering on EgoSchema with language priors.** We observe that the language priors with different LLMs serve as strong baselines.

racy varies greatly across the different LLMs. For example, the Flan-T5 model performs better than the LLama-2 model by approximately 9%. On the other end of the spectrum, a similarly-sized autoregressive GPT-J LLM with 6B parameters performs significantly worse than random. Given that the question and answer options in EgoSchema were generated using powerful LLMs (*e.g.*, GPT4 [46], Bard [1], and Claude [3]), we hypothesize that Flan-T5's accuracy on this task is due to having learned a representation that is similar to the LLMs used to generate the evaluation data.

While both Video-Llama and our approach use Llama-2 as the base LLM, we observe that Video-Llama still underperforms the Flan-T5 language prior baseline despite improving upon the Llama-2 language prior variant. In contrast, our Koala approach not only outperforms the Flan-T5 model, but also improves upon the base Video-Llama model by ∼7%. **This finding demonstrates the effectiveness of our introduced tokenizer functions at reasoning about temporal relations over longer spans.** One question that arises from these results is whether the accuracy gains by Koala can be attributed to further training on video data that may be semantically similar to the target domain. To address this question, we also finetune the Video-Llama captioning model without our CS and CV functions. Finetuning Video-LLama yields a drop of ∼5% in top-1 accuracy from the base Video-Llama model, and suggests that the improvements are not solely due to further finetuning. We include details about finetuning Video-Llama on HowTo100M in the supplemental.

Seed-Bench Procedure Understanding. We report the results of our evaluations on the procedure understanding task of the Seed-Bench benchmark in Table 3. The goal of procedure understanding is to detect all actions performed in a given video and arrange them in the correct temporal order, which requires fine-grained temporal understanding over a long span. As shown in Li *et al.* [32], state-of-the-art vLLMs (*e.g.*, mPLUG-Owl, VideoChat, and Video-Llama) often perform worse than their image-based variants such as InstructBLIP and VPGTrans. In certain cases, some vLLMs actually perform worse than their base LLM language prior baselines. For instance, using videos causes the accuracy to drop by 2-3% in the case of Valley [40] and Video-ChatGPT [41] when compared to their base Vicuna LLM [10] language prior.

It is also notable that large-scale pretraining on millions of short video and caption pairs only helps Video-Llama to improve by ∼4% over its base Llama-2 language prior. This finding suggests that learning to aggregate temporal context over a larger number of key frames without knowledge of the global context does not result in learning an effective key frames tokenizer function. In contrast, we observe that our proposed Koala model gains an improvement of ∼9% over Video-Llama in spite of the lightweight finetuning stage that uses many fewer training videos as compared to the initial pretraining on WebVid10M [6] and curated instructional video data [41]. This finding suggests that our introduced CS and CV tokenizer functions are beneficial towards reasoning about long-term temporal relations between different action sequences in videos.

Seed-Bench Action Recognition. Finally, we evaluate our Koala approach on the task of action recognition (Table 3) to study the effect of our introduced tokenizer functions for short-term temporal understanding tasks. In contrast to the longer setting in the procedure understanding task, the videos in this task generally have duration of around 10 seconds. Similar to our observations on the procedure understanding task, the mPLUG-Owl, Video-ChatGPT, and Valley vLLMs perform worse on this task than the image-based InstructBLIP and VPGTrans models.

Note that the base Video-Llama model performs worse

Approach	Training	LLM	LLM architecture	# input frames	Procedure Understanding	Action Recognition
Language prior	-	Vicuna	Decoder-only	-	23.83	27.30
Language prior	-	Flan-T5	Encoder-decoder	-	25.42	23.16
Language prior	-	Llama	Decoder-only	-	26.17	32.99
Language prior	-	Llama-2	Decoder-only	-	22.65	27.07
Random	-	-	-	-	25.00	25.00
mPLUG-Owl [67]	Captioning	Llama	Decoder-only	32	26.51	26.72
VideoChat [35]	Captioning	Vicuna	Decoder-only	32	27.27	34.89
Video-ChatGPT [41]	Captioning	Vicuna	Decoder-only	32	21.14	27.59
Valley [40]	Captioning	Vicuna	Decoder-only	32	20.72	31.26
Video-Llama-2 [70]	Captioning	Llama-2	Decoder-only	32	25.42	35.52
InstructBLIP [12]	Captioning	Flan-T5	Encoder-decoder	8	27.10	33.10
MovieChat [49]	Captioning	Llama-2	Decoder-only	32	26.76	34.37
InstructBLIP Vicuna [12]	Captioning	Vicuna	Decoder-only	8	23.07	34.48
VPGTrans [69]	Captioning	Flan-T5	Encoder-decoder	8	31.88	39.54
Koala (ours)	Captioning	Llama-2	Decoder-only	64	**35.91**	**41.26**

Table 3. **Zero-shot video question answering on Seed-Bench.** Compared to state-of-the-art mLLMs, our Koala approach improves the capability of the vLLM to not only understand long temporal context in procedure understanding but also to recognize short actions. We also compare to language prior baselines with different LLMs (highlighted in gray).

Approach	EgoSchema Benchmark	Procedure Understanding	Action Recognition
Base	33.25	26.68	35.52
Base + CS	36.93	30.20	38.74
Base + CS + CV	**40.42**	**35.91**	**41.26**

Table 4. **Model ablations on the zero-shot evaluation benchmarks.** We ablate the effectiveness of different queries introduced in our Koala approach on all three evaluation tasks.

Keep z_{key} output	Condition on z_{key} in CS tokenizer	Temporal concept queries $Q_{\text{temp}}, Q_{\text{concepts}}$	EgoSchema Benchmark
✓	✓	✓	33.61
✗	✗	✓	39.12
✗	✓	✗	39.20
✗	✓	✓	**40.42**

Table 5. **Additional model ablations on the EgoSchema benchmark.** We include additional ablation experiments over adding temporal queries in our CS tokenizer function as well as retaining the learnable inter-segment queries as input into the LLM. We observe that global context conditioning and introducing learnable parameters are beneficial towards adapting pretrained vLLMs.

Approach	Aggregate pre-LLM	EgoSchema
Base	✗	33.25
Average	✓	33.39
Memory module (Moviechat) [49]	✓	34.62
Concatenation	✗	35.72
Koala (ours)	✓	**40.33**

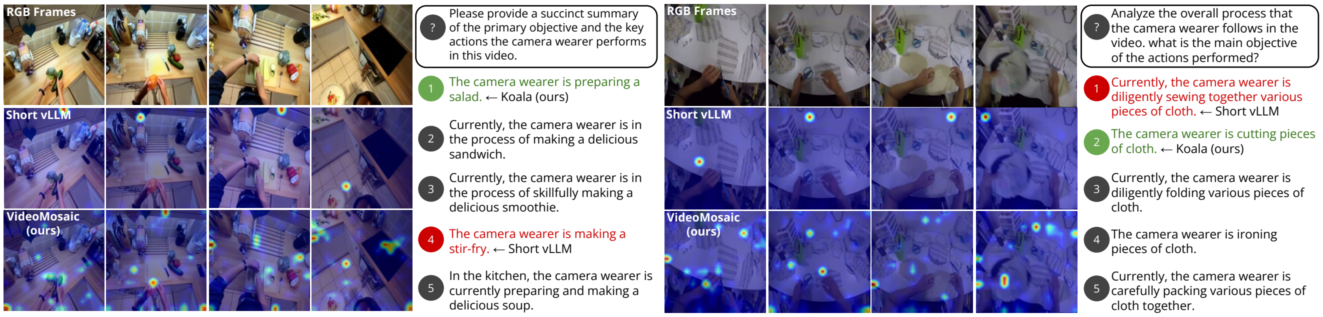
Table 6. **Comparisons between pre- and post-LLM temporal context aggregation.** We observe that naively encoding each video segment separately and concatenating the entire sequence of video tokens into the LLM performs worse than aggregating the video tokens *before* passing them into the LLM.

than the image-LLM VPGTrans by ∼4% despite its large-scale pretraining on seconds-long videos. This finding suggests that its key frames tokenizer function may be limited at reasoning about fine-grained actions and interactions between objects. While we are primarily focused on understanding long videos, we observe that our CS and CV tokenizer functions are also beneficial to understanding short actions, improving upon Video-Llama by ∼6% and outperforming VPGTrans by ∼2%. These results suggest that using key frames to provide global context for reasoning about spatiotemporal relationships between video segments may be crucial for fine-grained action understanding.

4.2. Ablation study

Overall Koala architecture. In Table 4, we ablate our CS and CV functions across all three evaluation tasks to determine their individual contributions. Consistent across all three tasks, we observe that conditioning on the key frames for global context to aggregate spatiotemporal context within each video segment in our CS function is especially crucial, as evidenced by a ∼3% improvement in top-1 accuracy on average. We also note the importance of reasoning about spatiotemporal contextual information between segments in our CIS function where our concept queries help improve accuracy on both long and short-term temporal understanding.

Tokenizer design. We ablate the design choices of the CS and CV tokenizers on the EgoSchema benchmark in Table 5. We observe that passing the output tokens corresponding to z_{key} into the LLM ("Keep z_{key} output") instead of discarding them leads to a sharp drop in accuracy of ∼7%, which may be due to the base vLLM being pretrained to accept a fixed number of video tokens as input. Additionally, we note the benefit of conditioning the CS tokenizer on the key frame tokens z_{key}, where the lack of conditioning leads to a drop of ∼1.3%. Finally, we observe the importance of introducing additional parameters in the form of the temporal concept queries Q_{temp} and Q_{concepts} in the CV tokenizer. As evidenced by the accuracy gain, it is



Please provide a succinct summary of the primary objective and the key actions the camera wearer performs in this video.

1 The camera wearer is preparing a salad. ← Koala (ours)

2 Currently, the camera wearer is in the process of making a delicious sandwich.

3 Currently, the camera wearer is in the process of skillfully making a delicious smoothie.

4 The camera wearer is making a stir-fry. ← Short vLLM

5 In the kitchen, the camera wearer is currently preparing and making a delicious soup.

(a)

Analyze the overall process that the camera wearer follows in the video. what is the main objective of the actions performed?

1 Currently, the camera wearer is diligently sewing together various pieces of cloth. ← Short vLLM

2 The camera wearer is cutting pieces of cloth. ← Koala (ours)

3 Currently, the camera wearer is diligently folding various pieces of cloth.

4 The camera wearer is ironing pieces of cloth.

5 Currently, the camera wearer is carefully packing various pieces of cloth together.

(b)

Figure 4. **Example attention heatmap visualizations on EgoSchema.** We provide some qualitative examples of predictions made by our Koala approach and the base Video-Llama model based on what they focus on. We observe that Koala is generally able to focus on relevant regions better than the base vLLM.

important to adapt to the frozen video QFormer $\mathcal{F}_{\text{VQT}}$.

Temporal aggregation. Lastly, given the recent importance of vLLMs, we study in Table 6 the key factors for integrating long-term temporal context from more input frames into the frozen vLLM and compare to our Koala. For all aggregation function variants, we use 4 segments of 8 frames each. We next describe the different aggregation variants. The first variant ("Average") obtains visual tokens by averaging $\frac{1}{N}\sum_{i=1}^{N}\mathcal{F}_{\text{key}}(S_i)$ across all video segments S_i. These averaged tokens are concatenated with the key frame tokens z_{key} before being projected by ϕ_{key} and passed to the LLM. The second variant ("Memory module") utilizes short and long-term memory mechanisms [49] to compute contextualized soft visual tokens as input into the LLM. We pass in $\mathcal{F}_{\text{key}}(S_i)$ across all video segments into the short-term memory and use the long-term memory tokens as input into the LLM. In the third variant ("Concatenation"), we concatenate tokens $\mathcal{F}_{\text{key}}(S_i)$ across all segments S_i, allowing the LLM to leverage its pretrained self-attention function for temporal reasoning. We note that this variant is similar to the SlowFast approach [16], where the "slow" frame features z_{key} are fused with the "fast" frame features $\mathcal{F}_{\text{key}}(S_i)$ by concatenation.

In general, we observe that it is more beneficial to aggregate temporal context in videos and encode it in the sequence of visual tokens before passing them into the LLM. While average-pooling video segment representations or using a long-term memory module [49] may lose some fine-grained spatiotemporal information, we observe that they are outperformed by the concatenation variant on downstream evaluations by only a small margin. This finding suggests that the self-attention layers in the LLM may not understand longer sequences of visual tokens without additional large-scale pretraining. Finally, we further ablate over the training hyperparameters including the number of segments as well as frames per segment used as input into the vLLM. Please refer to the supplemental for these results.

4.3. Qualitative results

We analyze how our introduced spatiotemporal queries in the CS and CV tokenizer functions change what the vLLM focuses on in the input videos (Figure 4). Compared to the baseline Video-Llama model, we observe that our introduced queries generally help to improve the capability of the model to focus on relevant visual concepts. The visualization in Figure 4a is particularly interesting because the introduced queries help our Koala model to predict that the person is making a salad based on its attention on the empty stove in the last frame (far right). Additionally, we also observe in Figure 4b that our model generally focuses on the pieces of cloth as opposed to the background as in the case of the base Video-Llama model.

Limitations. While our Koala approach is able to extend the video tokenizer function of a pretrained vLLM to understand minutes-long videos, it may still be limited at understanding much longer videos such as movies. Since it relies on a pretrained model, we inherit as a fundamental limitation a maximum number of input tokens, thereby limiting the number of input segments. However, extending positional embeddings to longer sequences remains an open work, especially in the setting of vLLMs.

5. Conclusion

In conclusion, we propose an approach, Koala, that introduces the Conditioned Segment and Conditioned Video tokenizer functions. Our CS and CV functions leverage learnable spatiotemporal queries to adapt the frozen video tokenizer function in pretrained vLLMs to generalize to minutes-long videos. More importantly, we empirically demonstrate the benefits of our Koala approach where it improves the base vLLMs on both short and long-term temporal understanding tasks.

Acknowledgements: This material is based upon work supported, in part, by DARPA under agreement number HR00112020054.

References

[1] Google. an important next step on our ai journey. https://blog.google/technology/ai/bard-google-ai-search-updates/, 2020. 6

[2] Introducing chatgpt. https://openai.com/blog/chatgpt/, 2023. 3

[3] Introducing claude. https://www.anthropic.com/index/claude-2/, 2023. 6

[4] 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. *Advances in neural information processing systems*, 35:23716–23736, 2022. 3

[5] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6836–6846, 2021. 2

[6] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *IEEE International Conference on Computer Vision*, 2021. 6

[7] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, volume 2, page 4, 2021. 2

[8] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 5

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

[10] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023. 3, 6

[11] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*, 2022. 3, 5

[12] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning, 2023. 2, 3, 7

[13] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Kiana Ehsani, Jordi Salvador, Winson Han, Eric Kolve, Aniruddha Kembhavi, and Roozbeh Mottaghi. procthor: Large-scale embodied ai using procedural generation. *Advances in Neural Information Processing Systems*, 35:5982–5994, 2022. 2

[14] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6824–6835, 2021. 2

[15] Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 203–213, 2020. 2

[16] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6202–6211, 2019. 2, 8

[17] Christoph Feichtenhofer, Haoqi Fan, Bo Xiong, Ross Girshick, and Kaiming He. A large-scale study on unsupervised spatiotemporal representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3299–3309, 2021. 2

[18] Tsu-Jui Fu, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. An empirical study of end-to-end video-language transformers with masked visual modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22898–22909, 2023. 6

[19] Samir Yitzhak Gadre, Kiana Ehsani, Shuran Song, and Roozbeh Mottaghi. Continuous scene representations for embodied ai. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14849–14859, 2022. 2

[20] Difei Gao, Luowei Zhou, Lei Ji, Linchao Zhu, Yi Yang, and Mike Zheng Shou. Mist: Multi-modal iterative spatial-temporal transformer for long-form video question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14773–14783, 2023. 3

[21] Rohit Girdhar and Kristen Grauman. Anticipative video transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 13505–13515, 2021. 2

[22] Satya Krishna Gorti, Noël Vouitsis, Junwei Ma, Keyvan Golestan, Maksims Volkovs, Animesh Garg, and Guangwei Yu. X-pool: Cross-modal language-video attention for text-video retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5006–5015, 2022. 2

[23] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*, 2020. 6

[24] 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. 3

[25] Fan Hu, Aozhu Chen, Ziyue Wang, Fangming Zhou, Jianfeng Dong, and Xirong Li. Lightweight attentional feature fusion: A new baseline for text-to-video retrieval. In *European Conference on Computer Vision*, pages 444–461. Springer, 2022. 2

[26] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General

perception with iterative attention. In *International conference on machine learning*, pages 4651–4664. PMLR, 2021. 3

[27] Chen Ju, Tengda Han, Kunhao Zheng, Ya Zhang, and Weidi Xie. Prompting visual-language models for efficient video understanding. In *European Conference on Computer Vision*, pages 105–124. Springer, 2022. 3

[28] Kumara Kahatapitiya and Michael S Ryoo. Coarse-fine networks for temporal activity detection in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8385–8394, 2021. 2

[29] Apoorv Khandelwal, Luca Weihs, Roozbeh Mottaghi, and Aniruddha Kembhavi. Simple but effective: Clip embeddings for embodied ai. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14829–14838, 2022. 2

[30] Alexander Klaser, Marcin Marszałek, and Cordelia Schmid. A spatio-temporal descriptor based on 3d-gradients. In *BMVC 2008-19th British Machine Vision Conference*, pages 275–1. British Machine Vision Association, 2008. 2

[31] Sateesh Kumar, Jonathan Zamora, Nicklas Hansen, Rishabh Jangir, and Xiaolong Wang. Graph inverse reinforcement learning from diverse videos. In *Conference on Robot Learning*, pages 55–66. PMLR, 2023. 2

[32] Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. Seed-bench: Benchmarking multimodal llms with generative comprehension. *arXiv preprint arXiv:2307.16125*, 2023. 2, 5, 6

[33] Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Michael Lingelbach, Jiankai Sun, et al. Behavior-1k: A benchmark for embodied ai with 1,000 everyday activities and realistic simulation. In *Conference on Robot Learning*, pages 80–93. PMLR, 2023. 2

[34] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 2, 3

[35] KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*, 2023. 2, 3, 7

[36] Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. Hero: Hierarchical encoder for video+ language omni-representation pre-training. *arXiv preprint arXiv:2005.00200*, 2020. 5

[37] Zhi Li, Lu He, and Huijuan Xu. Weakly-supervised temporal action detection for fine-grained videos with hierarchical atomic actions. In *European Conference on Computer Vision*, pages 567–584. Springer, 2022. 2

[38] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. 2, 3

[39] Yuqi Liu, Pengfei Xiong, Luhui Xu, Shengming Cao, and Qin Jin. Ts2-net: Token shift and selection transformer for text-video retrieval. In *European Conference on Computer Vision*, pages 319–335. Springer, 2022. 2

[40] Ruipu Luo, Ziwang Zhao, Min Yang, Junwei Dong, Minghui Qiu, Pengcheng Lu, Tao Wang, and Zhongyu Wei. Valley: Video assistant with large language model enhanced ability. *arXiv preprint arXiv:2306.07207*, 2023. 2, 3, 6, 7

[41] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. *arXiv preprint arXiv:2306.05424*, 2023. 2, 3, 6, 7

[42] Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. *arXiv preprint arXiv:2308.09126*, 2023. 2, 5

[43] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2630–2640, 2019. 2, 5

[44] Daniel Neimark, Omri Bar, Maya Zohar, and Dotan Asselmann. Video transformer network. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3163–3172, 2021. 2

[45] Bolin Ni, Houwen Peng, Minghao Chen, Songyang Zhang, Gaofeng Meng, Jianlong Fu, Shiming Xiang, and Haibin Ling. Expanding language-image pretrained models for general video recognition. In *European Conference on Computer Vision*, pages 1–18. Springer, 2022. 3

[46] R OpenAI. Gpt-4 technical report. arxiv 2303.08774. *View in Article*, 2023. 6

[47] Yuzhe Qin, Yueh-Hua Wu, Shaowei Liu, Hanwen Jiang, Ruihan Yang, Yang Fu, and Xiaolong Wang. Dexmv: Imitation learning for dexterous manipulation from human videos. In *European Conference on Computer Vision*, pages 570–587. Springer, 2022. 2

[48] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 3

[49] Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Xun Guo, Tian Ye, Yan Lu, Jenq-Neng Hwang, et al. Moviechat: From dense token to sparse memory for long video understanding. *arXiv preprint arXiv:2307.16449*, 2023. 6, 7, 8

[50] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*, 2019. 6

[51] Yixuan Su, Tian Lan, Huayang Li, Jialu Xu, Yan Wang, and Deng Cai. Pandagpt: One model to instruction-follow them all. *arXiv preprint arXiv:2305.16355*, 2023. 3

[52] Guolei Sun, Yun Liu, Henghui Ding, Thomas Probst, and Luc Van Gool. Coarse-to-fine feature mining for video semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3126–3137, 2022. 2

[53] Yuchong Sun, Hongwei Xue, Ruihua Song, Bei Liu, Huan Yang, and Jianlong Fu. Long-form video-language pre-training with multimodal temporal contrastive learning. *Advances in neural information processing systems*, 35:38032–38045, 2022. 3

[54] 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 preprint arXiv:2302.13971*, 2023. 3

[55] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018. 2

[56] Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems*, 34:200–212, 2021. 2

[57] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Dense trajectories and motion boundary descriptors for action recognition. *International journal of computer vision*, 103:60–79, 2013. 2

[58] Jue Wang, Wentao Zhu, Pichao Wang, Xiang Yu, Linda Liu, Mohamed Omar, and Raffay Hamid. Selective structured state-spaces for long-form video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6387–6397, 2023. 3

[59] Wenhai Wang, Zhe Chen, Xiaokang Chen, Jiannan Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu Qiao, et al. Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. *arXiv preprint arXiv:2305.11175*, 2023. 3

[60] Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, et al. Internvideo: General video foundation models via generative and discriminative learning. *arXiv preprint arXiv:2212.03191*, 2022. 6

[61] Syed Talal Wasim, Muzammal Naseer, Salman Khan, Fahad Shahbaz Khan, and Mubarak Shah. Vita-clip: Video and text adaptive clip via multimodal prompting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23034–23044, 2023. 3

[62] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022. 2

[63] Chao-Yuan Wu and Philipp Krahenbuhl. Towards long-form video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1884–1894, 2021. 3

[64] Chao-Yuan Wu, Yanghao Li, Karttikeya Mangalam, Haoqi Fan, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Memvit: Memory-augmented multiscale vision transformer for efficient long-term video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13587–13597, 2022. 2

[65] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5288–5296, 2016. 5

[66] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Zero-shot video question answering via frozen bidirectional language models. *Advances in Neural Information Processing Systems*, 35:124–141, 2022. 6

[67] Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*, 2023. 3, 6, 7

[68] Junsong Yuan, Zicheng Liu, and Ying Wu. Discriminative subvolume search for efficient action detection. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2442–2449. IEEE, 2009. 2

[69] Ao Zhang, Hao Fei, Yuan Yao, Wei Ji, Li Li, Zhiyuan Liu, and Tat-Seng Chua. Transfer visual prompt generator across llms. *arXiv preprint arXiv:2305.01278*, 2023. 2, 3, 7

[70] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023. 2, 3, 5, 6, 7

[71] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022. 3

[72] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 2, 3