Memory Consolidation Enables Long-Context Video Understanding

Ivana Balažević [*1] **Yuge Shi** [*1] **Pinelopi Papalampidi** [*1]
Rahma Chaabouni [1] **Skanda Koppula** [1] **Olivier J. Hénaff** [1]

Abstract

Most transformer-based video encoders are limited to short temporal contexts due to their quadratic complexity. While various attempts have been made to extend this context, this has often come at the cost of both conceptual and computational complexity. Instead, we propose to re-purpose existing pretrained video transformers by simply fine-tuning them to attend to memories derived non-parametrically from past activations. By leveraging redundancy reduction, our *memory-consolidated vision transformer* (MC-ViT) effortlessly extends its context far into the past and exhibits excellent scaling behavior when learning from longer videos. In doing so, MC-ViT sets a new state-of-the-art in long-context video understanding on EgoSchema, Perception Test, and Diving48, outperforming methods that benefit from orders of magnitude more parameters.

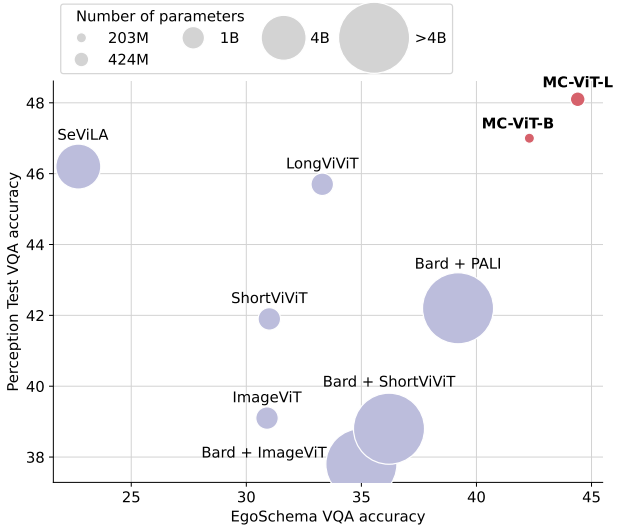

Figure 1. **Long-context video understanding on EgoSchema and Perception Test.** The proposed Memory-Consolidated Vision Transformer (MC-ViT-{B,L}, shown in bold) surpasses both public and large-scale proprietary models, despite using orders of magnitude fewer parameters and requiring only short fine-tuning schedules on top of standard pretrained models.

1. Introduction

Humans and animals reason about events extending over days, weeks, and years (Tulving, 1985), yet current artificial vision systems live largely in the present. While architectures that model the dynamics of natural videos have grown ever more sophisticated (Carreira & Zisserman, 2017; Feichtenhofer et al., 2019; Arnab et al., 2021), the temporal extent over which they reason has typically been limited to a small number of frames. In particular, transformer architectures (Vaswani et al., 2017) which power most applications in vision and language do not scale to the vast number of tokens present in natural videos due to their quadratic complexity. For example, 30 minutes of video sampled at standard rates may contain half a million tokens— more than what current state-of-the-art architectures using optimized attention algorithms (*e.g.* Dao et al., 2022) can process. Several attempts have been made to extend the temporal context of video transformers, including masking, attention approximations, and parametric memory modules (*e.g.* Wu et al., 2022; Piergiovanni et al., 2024). However, these approaches often introduce additional complexity, requiring specialized architectures and training paradigms.

In this work, we question whether such modifications are indeed necessary to enable long-context modeling. Starting from standard pretrained video transformers (Arnab et al., 2021), we process videos in a streaming setting in order to bound their complexity by the length of short segments (Dai et al., 2019). Crucially, we process individual segments in relation to a memory bank which is populated non-parametrically with the consolidated activations from past segments. This allows us to re-purpose pretrained video transformers for long-context understanding without any architectural modification, by simply fine-tuning them to attend to this memory with short training schedules.

[*]Equal contribution [1]Google DeepMind. Correspondence to: Ivana Balažević <balazevic@google.com>, Olivier J. Hénaff <henaff@google.com>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

A central question we are faced with is therefore how to choose which of the quasi-infinite tokens from past frames to store in memory. Inspired by evidence from psychology and neuroscience which formulates memory as a reconstructive process (Bartlett, 1932; Marr, 1971; Spens & Burgess, 2024), we adopt simple nonparametric schemes that form memories that are maximally representative of the full set of past activations. We find these mechanisms to effectively compress memories by an order of magnitude, and allow our *memory-consolidated vision transformer* (MC-ViT) to extend its context to significantly longer videos while maintaining a bounded complexity. In particular,

1. MC-ViT strikes a favorable trade-off between computational complexity and expressivity, outperforming standard video transformers and efficient approximations thereof with $10\times$ less memory and computation.

2. The non-parametric nature of MC-ViT allows us to straightforwardly re-purpose off-the-shelf pretrained video transformers by fine-tuning them to use their consolidated memory, yielding large efficiency gains by decreasing overall training time on long videos.

3. MC-ViT sets a new state-of-the-art on long-context video understanding tasks such as fine-grained action recognition (Diving48) and video question answering (EgoSchema and Perception Test), outperforming methods which benefit from orders of magnitude more parameters.

4. MC-ViT is competitive with large-scale proprietary systems such as GPT-4V and Bard, despite using a small, standard, and open architecture and training paradigm.

2. Related Work

Long-context architectures. Prior work has thoroughly explored approaches for handling long textual or visual inputs, by sparsifying either the input tokens or the attention applied over these tokens. In natural language processing, notable examples include Big Bird (Zaheer et al., 2020) and LongFormer (Beltagy et al., 2020) that employ local self-attention over restricted windows combined with global tokens that attend over the entire sequence. Alternative attention mechanisms in vision have utilized pooling (Wang et al., 2021; Li et al., 2022b), linear (Bolya et al., 2022) and windowed formulations (Dong et al., 2022; Li et al., 2022a; Ryali et al., 2023). Several works reduce the number of tokens via multi-resolution patchification, thus processing the input video at different granularities (Feichtenhofer et al., 2019; Yan et al., 2022a; Piergiovanni et al., 2023). Similarly, Papalampidi et al. (2024) showcase the benefits of this approach by training video encoders on long contexts with high ratios of input masking. Current state-of-the-art approaches for processing long videos consist of modular systems for captioning and extracting frame-level information, followed by a billion-scale LLM for aggregating this information (Zeng et al., 2022; Wang et al., 2022c; Li et al., 2023; Lin et al., 2023; Wang et al., 2023a; Zhang et al., 2023). The approach proposed in this work is orthogonal to these, by re-purposing standard transformer architectures for long-context modeling, whose representations can be incorporated into LLMs.

Memory-augmented transformers. Since the introduction of transformers (Vaswani et al., 2017), several works have sought to give them additional context via auxiliary memory banks. In NLP, TransformerXL does so by simply attending to recent activations in a streaming setting (Dai et al., 2019), whereas Retro (Borgeaud et al., 2022) does so by retrieving semantically related content. In vision, memory-augmented architectures have also been shown to enable video object segmentation (Oh et al., 2019), tracking (Lai et al., 2020), and action recognition (Wu et al., 2019).

Memory-compressing transformers. Several transformer-based architectures explored compressing past activations into a finite-length memory. In NLP, Neural Turing Machines (Graves et al., 2014) and Token Turning Machines (Ryoo et al., 2023) learn to read and write from a memory bank in an end-to-end manner. Similarly, Compressive Transformers (Rae et al., 2020), ∞-former (Martins et al., 2022)—and in vision, MemDPC (Han et al., 2020), LSTR (Xu et al., 2021b), MeMViT (Wu et al., 2022) and LongMem (Wang et al., 2023b)—extend the effective context length by compressing prior activations with additional parametric modules. Concurrent work Mirasol3B (Piergiovanni et al., 2024) showcases the power of this approach by combining these memory modules with large language models and a bespoke pretraining protocol.

Our work differs from these in that we find that a simple, non-parametric mechanism followed by light-weight fine-tuning is sufficient to re-purpose standard pretrained video transformer architectures (*e.g.* ViViT, Arnab et al., 2021) to achieve strong long-context modeling. Closest to our approach is concurrent work MovieChat (Song et al., 2024) that also uses a non-parametric memory, but focuses on extending the visual context of an LLM for video-to-text generation while using fixed frame-level representations via a pre-trained image encoder. In contrast, we directly add memory consolidation to the visual backbone.

3. Method

3.1. Overview of Video Vision Transformers (ViViT)

Video Vision Transformers (ViViT; Arnab et al. 2021) adapt Vision Transformers (Dosovitskiy et al., 2021) to straightforwardly process videos. Specifically, ViViT divides a video $V \in \mathbb{R}^{T\times H\times W}$ into N_T non-overlapping spatio-temporal patches $x_i \in \mathbb{R}^{t\times h\times w}$ such that $N_T = \frac{T}{t}\cdot\frac{H}{h}\cdot\frac{W}{w}$, and linearly

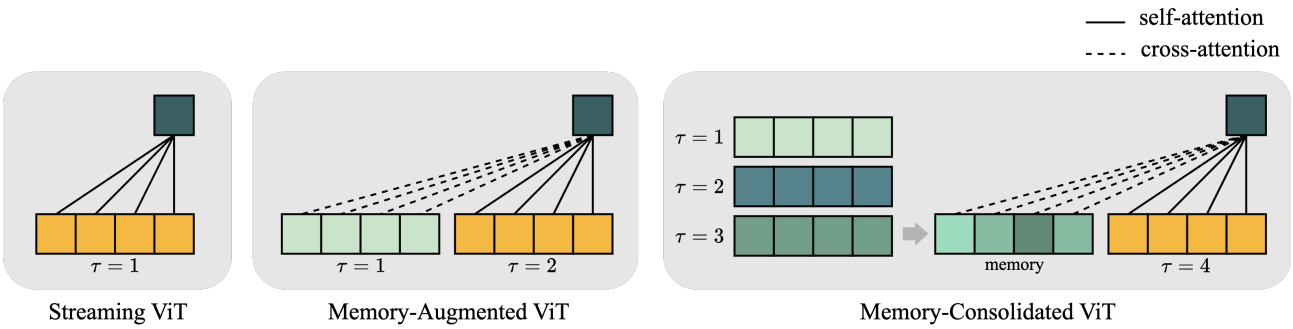

Figure 2. **Visualization of the proposed method. Left:** Streaming ViT processes each segment of the sequence independently by attending over activations within a segment. **Middle:** Memory-Augmented ViT, similar to Transformer XL (Dai et al., 2019), attends to current activations (yellow blocks) and those in recent history (green blocks). **Right:** In Memory-consolidated ViT, we consolidate the extended context into shorter memory and cross-attend over them, which enables us to effectively attend over longer sequences.

projects these patches into 1D embedding space:

$$z_i = \boldsymbol{E} x_i + p_i, \tag{1}$$

where $\boldsymbol{E}$ denotes a learnable projection layer and $p_i \in \mathbb{R}^d$ additional positional embeddings. The resulting token sequence $\boldsymbol{z}^0 = [z_i, i \in [1, N_T]] \in \mathbb{R}^{N_T \times d}$ is then passed through a series of L transformer layers, which alternate Multi-head Self-Attention (MSA; Vaswani et al. 2017), layer normalization (LN; Ba et al. 2016) and MLP blocks:

$$\boldsymbol{y}^l = \text{MSA}(\text{LN}(\boldsymbol{z}^l)) + \boldsymbol{z}^l \tag{2}$$

$$\boldsymbol{z}^{l+1} = \text{MLP}(\text{LN}(\boldsymbol{y}^l)) + \boldsymbol{y}^l. \tag{3}$$

While various schemes for factorizing the attention have been proposed for ViViT, we build our model upon the simplest joint space-time attention which models dependencies between all tokens in the sequence. We leave exploring other factorization methods for future research. In contrast to ViViT's self-attention which spans the entire video, our MC-ViT model uses self-attention within much shorter segments, and cross-attention across segments via a set of consolidated memories, which we detail below.

3.2. Memory-Consolidated Vision Transformers

In this section, we explore three successive modifications to the original ViViT architecture that enable efficient and expressive scaling to longer videos (see visualization in Figure 2). The culmination of these modifications represents our proposed method: Memory-Consolidated ViT (MC-ViT). We apply consistent pre- and post-processing steps across all three approaches: we divide the video V into s temporal segments $v_\tau \in \mathbb{R}^{S \times H \times W}$, where $S = \frac{T}{s}$ is the number of frames per segment and $S = 16$ in our experiments. We then process each segment (either individually or jointly, see below), yielding a list of s representations $\{\boldsymbol{z}_1, \cdots, \boldsymbol{z}_s\}$, one for each segment. All of these are then concatenated as the final representation of the video.

Streaming ViT (ST-ViT). Since the computational complexity of transformers scales quadratically with the number of tokens, full joint space-time attention becomes intractable for video lengths that exceed even small numbers of frames. To counteract this, we start with a simple streaming-based extension of ViViT, which processes each segment $v_\tau, \tau \in [1, s]$ independently, as described in Section 3.1, with positional embeddings spanning the entire video. Crucially, the number of tokens processed by the ViViT encoder at a given time is instead $N = \frac{S}{t} \cdot \frac{H}{h} \cdot \frac{W}{w}$, bounding the quadratic complexity by the segment length S rather than the total video length T. We include the pseudocode for the streaming ViT implementation in Appendix A, Algorithm 2.

Memory-Augmented ViT (MA-ViT). While more scalable, the streaming setting limits the encoder's ability to reason over events which span multiple segments. Hence, as in Dai et al. (2017), we augment the self-attention module with an additional set of memories $\boldsymbol{m}_\tau^l = [\boldsymbol{z}_0^l; \boldsymbol{z}_1^l; ...; \boldsymbol{z}_{\tau-1}^l] \in \mathbb{R}^{M \times d}$ consisting of concatenated activations of previous segments at each layer l:

$$\boldsymbol{y}_\tau^l = \text{MCA}(\underbrace{\text{LN}(\boldsymbol{z}_\tau^l)}_{\text{query}}, \underbrace{[\text{LN}(\boldsymbol{z}_\tau^l); \text{LN}(\boldsymbol{m}_\tau^l)]}_{\text{key-value}}) + \boldsymbol{z}_\tau^l, \tag{4}$$

where $[\cdot;\cdot]$ denotes the concatenation operation and Multi-head Cross-Attention (MCA; Dai et al. 2019) generalizes MSA by decoupling the inputs to the query and key/value heads. Specifically, the MCA operation allows activations from the current segment $\boldsymbol{z}_\tau^l$ to attend both to themselves (as in MSA) and to memories of all past activations $\boldsymbol{m}_\tau^l$, while keeping the quadratic complexity limited to $N + M$. We include the pseudocode for Memory-Augmented ViT in Appendix A, Algorithm 3.

Memory-Consolidated ViT (MC-ViT). Given the memory-augmented vision transformer architecture, a central question is how to consolidate the (potentially infinite) activations of previous segments into a finite (and ideally small) set of memories. We consider three simple instances of memory consolidation that model memory through a non-parametric reconstructive process.

Algorithm 1 Memory-consolidated ViT.

```
def mc_vit(
      video, n_chunks, n_layers,
      pos_emb, mc_method, num_mem
):
  emb = linear_proj(video) + pos_emb # [B, N, D]
  chunked_video = np.split(emb, n_chunks, axis=1)
  memory = {layer: None for layer in range(n_layers)}
  zs = []
  for z in chunked_video:
    z_norm = layer_norm(z)
    for layer in range(n_layers):
       if memory[layer] is None:
          y = self_attention(z_norm) + z
       else:
          kv = np.concatenate(z_norm, memory[layer]))
          y = cross_attention(q=z_norm, kv=kv) + z
       y_norm = layer_norm(y)
       z = mlp(y_norm) + y
       memory[layer] = memory_consolidation(
          memory[layer], z, num_mem, mc_method)
       memory[layer] = layer_norm(memory[layer])
       zs.append(z)
  return np.concatenate(zs, axis=1)
```

To produce a new consolidated memory $\boldsymbol{m}_\tau$ for the current segment (dropping the layer index l for concision), we consolidate the set of activations from the preceding segment $\boldsymbol{z}_{\tau-1} \in \mathbb{R}^{N\times d}$ into $\hat{\boldsymbol{z}}_{\tau-1} \in \mathbb{R}^{K\times d}$ ($K \leq N$) and concatenate them to the memories consolidated from all prior segments $\boldsymbol{m}_\tau = [\boldsymbol{m}_{\tau-1}, \hat{\boldsymbol{z}}_{\tau-1}] \in \mathbb{R}^{(M+K)\times d}$. The proposed instances of non-parametric memory consolidation differ in their way of computing $\hat{\boldsymbol{z}}_{\tau-1}$, which we detail below.

MC-ViT-R (random) is the simplest non-parametric baseline which randomly selects a set of K activations from $\boldsymbol{z}_{\tau-1}$ and uses them as the consolidated memory for the preceding segment:

$$\hat{\boldsymbol{z}}^{\mathrm{R}}_{\tau-1} = \{\boldsymbol{z}_{\tau-1,k} \mid k \in \mathcal{I}\} \in \mathbb{R}^{K\times d}, \tag{5}$$

where $\mathcal{I} \in [1, N]^K$ is a set of K randomly selected indices.

MC-ViT-CS (coreset) constructs a maximally representative set of memories by applying the greedy coreset selection algorithm (Agarwal et al., 2005) to the activations of the preceding segment $\boldsymbol{z}_{\tau-1}$ by iteratively adding the most distant activations to the ones already included in the consolidated memory for that segment. One iteration of the algorithm is defined as:

$$k^* = \underset{k\in[1,N]}{\arg\max} \min_{j\in\mathcal{M}^*} ||\boldsymbol{z}_{\tau-1,k} - \boldsymbol{z}_{\tau-1,j}||_2^2 \tag{6}$$

$$\mathcal{M}^* \leftarrow \mathcal{M}^* \cup \{k^*\}, \tag{7}$$

where $\mathcal{M}^*$ is the set of activation indices chosen to be added to the consolidated memory $\hat{\boldsymbol{z}}^{\mathrm{CS}}_{\tau-1}$. The greedy coreset selection algorithm is run for K iterations to produce the consolidated memory $\hat{\boldsymbol{z}}^{\mathrm{CS}}_{\tau-1} \in \mathbb{R}^{K\times d}$ for the segment $v_{\tau-1}$. Due to its iterative nature, the coreset selection algorithm becomes increasingly computationally expensive as the size of the segment memory $K = |\mathcal{M}^*|$ increases.

MC-ViT-KM (k-means) randomly initializes K cluster centroids as $\hat{\boldsymbol{z}}^{\mathrm{R}}_{\tau-1}$ (see Equation 5) and then performs 5 iterations of k-means clustering on all activations of the previous segment $\boldsymbol{z}_{\tau-1}$ to compute the updated cluster centroids, which we use as the consolidated memory $\hat{\boldsymbol{z}}^{\mathrm{KM}}_{\tau-1} \in \mathbb{R}^{K\times d}$ for the segment $v_{\tau-1}$.

We include the pseudocode for MC-ViT in Algorithm 1. The newly consolidated memory $\boldsymbol{m}_\tau$ is then jointly processed with the current segment activations $\boldsymbol{z}_\tau$ via MCA, analogously to MA-ViT (see Equation 4).

We compare these different consolidation methods in Section 4.4 and find that MC-ViT-KM performs better than the others. Therefore, unless specified otherwise, MC-ViT refers to MC-ViT-KM in the following sections.

3.3. Training and Evaluation

Initialization. Since the parameters of MC-ViT are almost identical to those of ViViT (Arnab et al., 2021), we initialize most parameters from a ViViT encoder pretrained on short (16-frame) video clips using multimodal contrastive learning (Xu et al., 2021a; Papalampidi et al., 2024), see Appendix B.1. The only parameters which differ are positional embeddings, as we fine-tune MC-ViT on significantly longer videos (*e.g.* up to 128 frames) than the short clips used for pretraining. We therefore initialize the positional embeddings by linearly upsampling ViViT's positional embeddings through interpolation along the time dimension (note that we experimented with both interpolation and extrapolation through repetition and did not find it to have a big impact on performance). Similarly, we re-use and fine-tune a BERT-style language encoder pretrained in the same setup.

Fine-tuning. For each evaluation, we fine-tune on a dataset mixture that enables a like-for-like comparison with the previous state-of-the-art. All datasets are composed of video-text pairs, and we therefore simply fine-tune the model with noise contrastive estimation. Given the video and text embeddings $\boldsymbol{z}^v_i$ and $\boldsymbol{z}^t_i$ of an example i, we minimize

$$\ell_i = -\log \frac{\exp(\boldsymbol{z}^v_i \cdot \boldsymbol{z}^t_i)}{\sum_j \exp(\boldsymbol{z}^v_i \cdot \boldsymbol{z}^t_j)} - \log \frac{\exp(\boldsymbol{z}^t_i \cdot \boldsymbol{z}^v_i)}{\sum_j \exp(\boldsymbol{z}^t_i \cdot \boldsymbol{z}^v_j)} \tag{8}$$

where the "negative" embeddings $\boldsymbol{z}^v_j$ and $\boldsymbol{z}^t_j$ are the in-batch examples unless otherwise specified. We provide further training details in Appendix B.2.

Evaluation. We employ the standard zero-shot transfer paradigm from CLIP (Radford et al., 2021) to perform all downstream tasks. In all cases, a test video is equipped with multiple possible "captions", only one of which is correct. For action recognition, these captions are simply the class names. For video question answering, captions are question-answer pairs constructed from the set of multiple-choice answers. We utilize the language model to compute caption embeddings $\boldsymbol{z}^t_i$, and compare them to the video em-

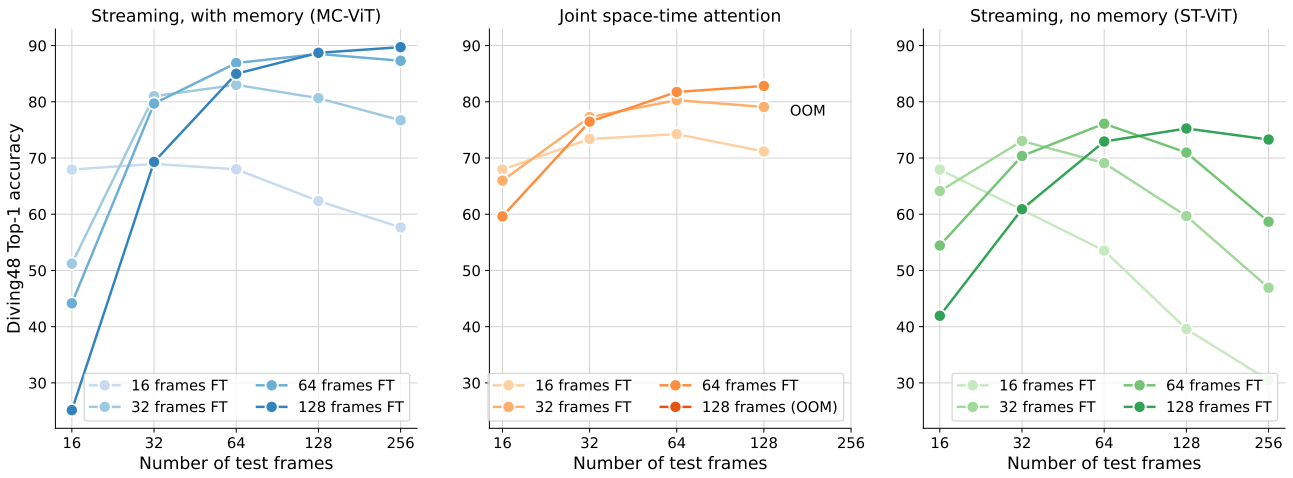

Figure 3. **MC-ViT effectively learns from long videos. Left:** MC-ViT scales to long Diving48 videos at both training and inference time, and benefits from fine-tuning on longer videos. **Middle:** Joint space-time attention benefits from fine-tuning on longer videos, but cannot learn from long (128 frame) videos due to its large complexity and memory footprint. **Right:** ST-ViT scales to longer videos but does not benefit from training on them.

bedding $\boldsymbol{z}_i^v$. The model's prediction $i^* = \arg\max_i \boldsymbol{z}_i^v \cdot \boldsymbol{z}_i^t$ is simply the caption with the highest similarity to the video embedding.

4. Experiments

4.1. Datasets

We evaluate our method on four challenging datasets for long-context video understanding, namely Diving48, EgoSchema, Next-QA, and Perception Test.

Diving48 (Li et al., 2018) was specifically designed to assess the importance of dynamic and long-term temporal reasoning in action recognition. Video lengths vary between 24 and 822 frames, with 158 frames on average. Each video is categorized into 48 fine-grained classes based on the specific dive type it depicts. Consequently, correct classification requires dense video sampling and fine-grained understanding in addition to retaining information over a long temporal extent, which necessitates reasoning over a large number of frames. To align with prior methods, we fine-tune on the Diving48 training set and re-initialize the language encoder randomly with a linear embedding function.

EgoSchema (Mangalam et al., 2023) is a long-form multiple-choice video question answering dataset derived from Ego4D (Grauman et al., 2022). The task involves selecting the correct answer out of five options based on a three-minute-long video clip. This task is particularly interesting for evaluating long-context understanding, as it benefits from long "temporal certificate" lengths, *i.e.* the minimum video duration a human needs to answer the question accurately. The model is fine-tuned on a mixture of HowTo100M and Ego4D, and we ensure that there is no overlap between Ego4D training and EgoSchema examples.

Next-QA (Xiao et al., 2021) emphasizes testing causal and temporal reasoning with open- and close-ended (multiple-choice) QA tasks. Videos in this dataset have an average duration of 44 seconds but can be as long as 2 minutes. We use the close-ended version for both fine-tuning and inference. Since the training set is fairly small and in order to avoid over-fitting on this domain, we add and only tune low-rank adapters (LoRA; Hu et al. 2021) at the self-attention and feed-forward blocks of every layer, which account for $\sim$12% of model parameters. For fine-tuning on this multiple-choice QA dataset, we use the four incorrect answers to the given question as hard negatives in Equation (8).

Perception Test (Pătrăucean et al., 2023) is inspired by assessment in developmental psychology and features a collection of games or daily activities that evaluate a model's grasp of physics, reasoning, memory, and semantic extraction. Although videos in this dataset are short with an average duration of 30 seconds, accurate localization and recognition of actions and objects require a higher FPS rate (we use an FPS of 4), resulting in sequences of hundreds of frames. We evaluate on the multiple-choice video question answering task by selecting one out of three possible answers, while training on Next-QA for zero-shot evaluation on this benchmark.

4.2. MC-ViT Effectively Learns from Long Videos

We start by assessing the ability of MC-ViT to model videos of increasing lengths. For this we fine-tune MC-ViT on videos with different number of frames (16, 32, 64, or 128) by varying the FPS rate. At inference time, we also apply the model to videos with 16 to 256 frames. Figure 3 (left) shows

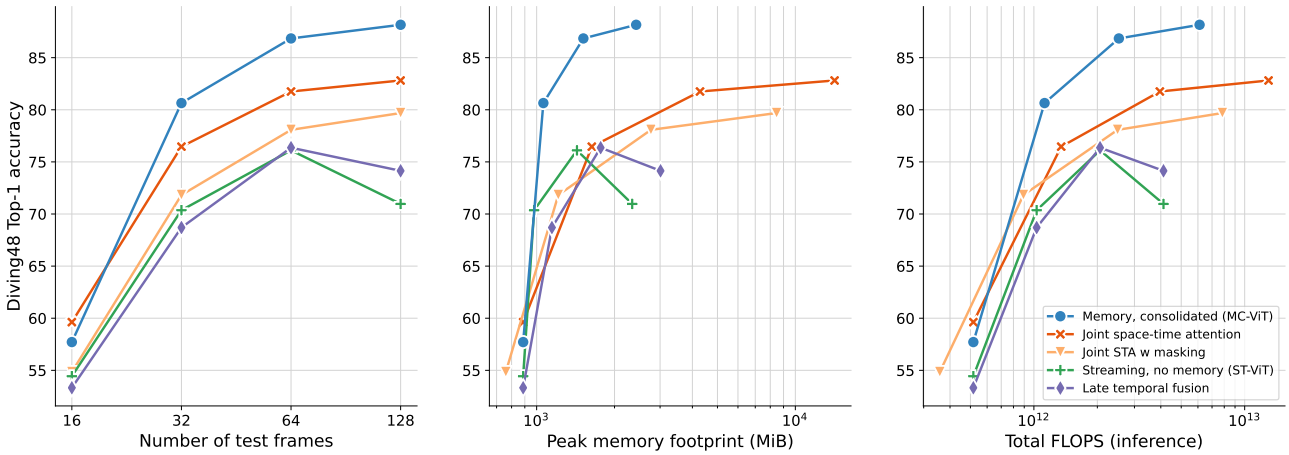

Figure 4. **MC-ViT efficiently models long videos.** Fine-grained video understanding on Diving48 as a function of number of test frames (**left**), memory consumption (**middle**), and computational complexity (FLOPS, **right**), for joint space-time attention w/ and w/o masking (yellow and red respectively), memory-less streaming setting (green), the late temporal fusion baseline (purple) and our proposed method MC-ViT (blue). MC-ViT reaches the highest accuracy with 10× less memory and FLOPS than the joint space-time attention method.

that MC-ViT's performance improves with more, densely sampled frames at both training and inference time on Diving48 fine-grained action recognition. In particular, training with longer contexts allows MC-ViT to benefit from more frames at inference time, with the optimal inference-time video length being twice that of the train-time video length, demonstrating reasonable generalization of the consolidated cross-attention mechanism.

In contrast, neither joint space-time attention (Figure 3, middle) nor a memory-less streaming ST-ViT architecture (Figure 3, right) effectively learn from long videos. While joint-space time attention benefits from training on more frames in terms of performance, its memory footprint prevents it from training or evaluating on the longest videos. ST-ViT on the other hand scales to more frames, but does not benefit from them, since it lacks the ability to reason over events that span multiple segments.

4.3. MC-ViT Efficiently Models Long Videos

Inference-time efficiency. We evaluate the performance of joint space-time attention, ST-ViT, and MC-ViT in relation to their memory and computational complexity, by varying the number of frames at inference time (all models are trained with 64 frames). MC-ViT's memory consumption is bounded by the number of tokens within a segment, similar to memory-less ST-ViT, whereas that of joint space-time attention increases with video length (Figure 4, middle). Similarly, while the computational complexity of joint space-time attention is quadratic in the video length, it is linear for both ST-ViT and MC-ViT (Figure 4, right).

In terms of performance, Figure 4 demonstrates that MC-ViT remarkably outperforms joint space-time attention with a 10× smaller memory footprint (middle) and FLOPS (right). We additionally test other scalable baselines, such as applying 25% input token masking to joint space-time attention (Papalampidi et al., 2024), and late temporal fusion (Alayrac et al., 2022; Yan et al., 2022b), where we add a learnable module on top of ST-ViT for contextualizing information across segments (see Appendix C). Not only does MC-ViT display a better scaling behavior than these baselines (Figure 4, left), but it does so with robust improvements in memory footprint and computational complexity.

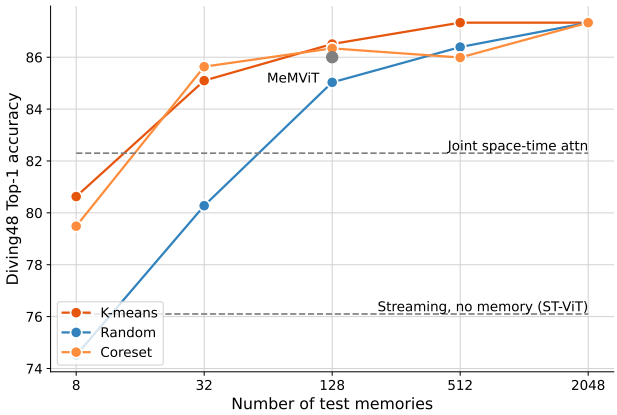

Figure 5. **MC-ViT makes efficient use of finite-length context.** We show three MC-ViT instances and compare them to relevant baselines (dashed horizontal lines). K-means (red) and coreset (orange) surpass all methods at 16× compression rate with 128 memories per segment, demonstrating the efficiency of our approach. Surprisingly, even random memory selection (blue) achieves impressive performance on this task, outperforming all baselines at 4× compression rate with 512 memories, which further showcases efficiency and robustness of the MC-ViT framework.

Training-time efficiency. We additionally assess whether these inference-time gains in efficiency translate in similar gains when fine-tuning. Specifically, we vary the number

of frames used for fine-tuning, and measure the resulting performance when evaluating the model with twice the number of frames (as suggested in Figure 3), as well as the training-time memory footprint and computational complexity. Figure A.1, left, shows that MC-ViT quickly surpasses joint space-time attention and the streaming memory architecture when provided with longer videos for training. In particular, MC-ViT surpasses the performance of joint space-time attention with a 3.5× smaller memory footprint and 3.2× fewer FLOPS (Figure A.1, middle and right).

4.4. Memory Consolidation Makes Efficient Use of a Finite Context Window

Consolidation mechanisms. We now analyze the computational efficiency and expressiveness of MC-ViT's consolidation methods. We compare our methods to three baselines: (1) joint space-time attention, (2) ST-ViT, and (3) MeMViT (Wu et al., 2022). Notably, MeMViT employs a parametric approach to memory compression, requiring a convolutional module to be trained alongside the network, where we experimented with different convolutional kernel sizes and found the kernel size $4\times2\times2$ (equivalent to MC-ViT's $K=128$) to perform the best (see Appendix C for details). Figure 5 illustrates the performance of these methods on Diving48 as a function of the number of memories K per segment. Given $K=128$ memories obtained through k-means consolidation (i.e. a 16× compression compared to MA-ViT; red curve), MC-ViT-KM outperforms all baselines. Remarkably, even random selection of $K=128$ memories (with MC-ViT-R) is sufficient to surpass ViViT and ST-ViT. Finally, consolidating past-activations with MC-ViT-CS (coreset, orange curve) performs similarly to MC-ViT-KM, highlighting the robustness of MC-ViT to the particular choice of memory consolidation algorithm. K-means consolidation is used as the default method given its greater computational efficiency and slightly higher performance for larger sets of memories.

Finite memory size. Further, to adapt MC-ViT to very long videos and decouple memory growth from video length, we compare two simple modifications to MC-ViT in Table 1 on the Diving-48 dataset: (1) keep all tokens from the last N segments of length K (MC-ViT-last) or (2) randomly select a fixed number of $N\times K$ tokens across all past segments (MC-ViT-global). The results show that 5 segments (or 2560 tokens) should be saved in memory with no loss in performance. Further, we see that selecting $N\times K$ random tokens across segments (with $K=512$ for all experiments) is more beneficial than keeping the last $N\times K$ tokens, which shows that relevant information for the current segment is contained throughout the whole video and not just in the last several segments. This also emphasizes the importance of redundancy reduction in the consolidation scheme, as uniformly sampled memories from the entire video are less redundant than all tokens from the most recent segments.

4.5. MC-ViT Achieves State-of-the-Art Long-Context Video Understanding

Fine-grained action recognition. In Table 3, we compare MC-ViT to prior methods on Diving48, and find that it delivers state-of-the-art results. Unlike previous methods that require object tracking models (Herzig et al., 2022) or additional modeling components, MC-ViT achieves strong performance by simply re-purposing a general transformer architecture for long-context modeling: while previous methods are limited to 32 frames of video, the efficient scaling properties of MC-ViT allow it to process 128 frames. Further, MC-ViT does not require multiple spatial crops at inference time to achieve state-of-the-art results.

Long video question answering. We compare MC-ViT to prior methods on long video question answering in Table 2. We find that our approach outperforms prior works that use up to 10× more parameters. Most notably, even our smaller model version (MC-ViT-B, with 200M parameters in total) is able to achieve a 10% improvement on EgoSchema in comparison to much larger models (up to 5B parameters). This demonstrates the importance of processing more frames, which our straightforward memory consolidation method enables, as well as the effectiveness of fine-tuning MC-ViT from standard pretrained video encoders.

It is particularly notable that MC-ViT is competitive with models such as Flamingo (Alayrac et al., 2022) and SeViLA (Yu et al., 2023), which boast billion-scale LLM decoders. Such methods benefit from the language bias in VQA—which allows for some questions to be trivially answered without any visual input—and extensive textual training data. While MC-ViT surpasses these models on EgoSchema and Perception Test, SeViLa maintains stronger performance on Next-QA. We hypothesize that this benchmark is not challenging enough for long video understanding and relies heavily on language-only reasoning, since Yu et al. (2023) achieve their results while using a single input frame. Thus, frame-level models with strong decoders, such as SeViLA, may be sufficient for benchmarks requiring language-only reasoning and localization (Next-QA, Perception Test), but fail to capture a summary representation of the entire video (EgoSchema). In contrast, our method, despite lacking large language decoders, performs competitively across the board, demonstrating strong localization and long-context modeling capabilities. Finally, MC-ViT requires minimal architectural changes and training overhead for adapting to long-context understanding, in contrast to modular methods (e.g., Yu et al., 2023) which involve multiple modules and complex training regimes.

MC-ViT vs. large-scale proprietary models. Additionally, in Table 4 we compare our method to large-scale proprietary models such as GPT-4V (Achiam et al., 2023), Gemini (Anil

Table 1. **MC-ViT with memory size decoupled from video length.** For MC-ViT-last, we store the last N segments (of length $K = 512$) into memory. The equivalent for MC-ViT-global is to store randomly selected $N \times K$ tokens across segments.

	N / $N \times K$					
Method	2 / 1024	3 / 1536	4 / 2048	5 / 2560	6 / 3072	7 / 3584
MC-ViT-B-last	81.5	85.0	87.0	**87.8**	**87.8**	**87.8**
MC-ViT-B-global	**84.8**	**86.8**	**87.5**	**87.8**	**87.8**	**87.8**

Table 2. **Long video question answering, compared to public models.** Performance is calculated as percentage correct on multiple-choice video question answering on EgoSchema, Perception Test and Next-QA. By scaling to significantly longer videos, MC-ViT outperforms models that benefit from an order of magnitude more parameters. We highlight the **best** and second-best methods per dataset. 128+ stands for 128 or more frames, where we use 128 frames for EgoSchema and Next-QA and 256 frames for Perception Test.

Method	Params	Frames	EgoSchema		Perception Test	Next-QA
			Subset	Full		
CoVGT (Xiao et al., 2023)	149M	32	–	–	–	60.0
SeViT$_{\text{FiD}}$ (Kim et al., 2023)	215M	10	–	–	–	60.6
HiTeA (Ye et al., 2023)	297M	16	–	–	–	63.1
InternVideo (Wang et al., 2022b)	478M	90	–	32.1	–	63.2
ImageViT (Papalampidi et al., 2024)	1B	16	40.8	30.9	39.1	–
ShortViViT (Papalampidi et al., 2024)	1B	16	47.9	31.0	41.9	–
Flamingo (Alayrac et al., 2022)	3B	32	–	–	43.6	–
SeViLA Localizer + ShortViViT (Papalampidi et al., 2024)	5B	32	49.6	31.3	–	–
LongViViT (Papalampidi et al., 2024)	1B	256	56.8	33.3	45.7	–
SeViLA (Yu et al., 2023)	4B	32	25.7	22.7	46.2	**73.8**
MC-ViT-B	203M	128+	61.2	42.3	47.0	60.6
MC-ViT-L	424M	128+	**62.6**	**44.4**	**48.1**	65.0

Table 3. **Fine-grained action classification on Diving48.** Prior methods use 3× more spatial crops at inference time (SC) and/or bounding box information (BB), which MC-ViT does not require.

Method	Params	Extra	Top-1
TimeS-L (Bertasius et al., 2021)	121M	SC	81.0
VideoSwin-B (Liu et al., 2022)	88M	SC	81.9
BEVT (Wang et al., 2022a)	88M	SC	86.7
SIFAR-B-14 (Fan et al., 2021)	87M	SC	87.3
ORViT (Herzig et al., 2022)	160M	SC+BB	88.0
AIM ViT-B (Yang et al., 2023)	97M	SC	88.9
AIM ViT-L (Yang et al., 2023)	341M	SC	90.6
MC-ViT-B	99M	✗	**89.7**
MC-ViT-L	313M	✗	**91.0**

et al., 2023) and Bard[1] + PALI (Google AI, 2023; Chen et al., 2023). While their exact implementation details are not publicly available, these models are thought to contain hundreds of billions to trillions of parameters, *i.e.* 1000× more than MC-ViT. It is also important to note that these proprietary models are trained on massive amounts of data from the internet, resulting in potential data contamination, which we proactively avoid in our training pipeline.

We showcase the performance of all models in the "raw" columns. Additionally, we also present (in grey font) the blind model performance where the model is trained only on question-answer pairs without the visual modality. In order to isolate the visual perception capabilities from the natural language reasoning of the model, we add the "visual" column in Table 4 which is computed as the difference between "raw" and "blind". Examining the visual-only capabilities, we conclude that our small-scale model is competitive against the large proprietary ones and even surpasses GPT-4V performance on the subset of EgoSchema.

Despite using a fraction of the parameters and training data, our method remains competitive and, in some cases, outperforms these models. In particular, MC-ViT achieves 5% improvements on EgoSchema and Perception Test against the sophisticated Bard + PALI modular system used for information aggregation and frame captioning, respectively.

5. Discussion

In this work, we introduced the Memory-Consolidated Vision Transformer (MC-ViT), which efficiently models long-range dependencies in videos by consolidating past activations into a compact memory bank. MC-ViT achieves state-of-the-art performance on multiple long video benchmarks by repurposing existing video architectures without the need for specialized architectures and training regimes. Our small-scale model outperforms approaches that bene-

[1]Release of September 2023.

Table 4. **Long video question answering on EgoSchema and Perception Test, compared to large-scale proprietary models.** Performance is evaluated on the original ("raw") dataset, as well as on the "visual" subset of questions that cannot be answered by a blind language model and on Perception Test for the validation set. For each model, we compute the performance of a "blind" variant on EgoSchema that only has access to question-answer pairs. The performance of the blind model is subtracted from that of the full model to compute "visual" performance. We underline the top 2 performing models for each benchmark and subset.

	EgoSchema Raw		EgoSchema Visual		Perception Test Raw	Perception Test Visual
Method	Subset	Full	Subset	Full		
Random chance	20.0	20.0	–	–	33.3	–
Bard only (blind)	27.0	33.2	0.0	0.0	36.8	0.0
Bard + ImageViT (Papalampidi et al., 2024)	35.0	35.0	8.0	1.8	37.8	1.0
Bard + ShortViViT (Papalampidi et al., 2024)	42.0	36.2	15.0	3.0	38.8	2.0
Bard + PALI (Papalampidi et al., 2024)	44.8	39.2	17.8	6.0	42.4	5.6
GPT-4 Turbo (blind)	31.0	30.8	0.0	0.0	–	–
GPT-4V	63.5	55.6	32.5	24.8	–	–
Gemini Ultra (Anil et al., 2023)	–	–	–	–	54.7	–
MC-ViT-B (blind)	18.2	23.4	0.0	0.0	37.6	0.0
MC-ViT-B	61.2	42.3	43.0	18.9	47.1	9.5
MC-ViT-L (blind)	15.0	22.7	0.0	0.0	35.1	0.0
MC-ViT-L	62.6	44.0	47.6	21.3	47.6	12.5

fit from orders of magnitude more parameters, and is even competitive with large-scale proprietary systems such as GPT-4V and Bard, demonstrating the importance of strong compressed video representations. As an extension, these representations could be fed into large language models to augment their long-range temporal reasoning capabilities.

We showcased the effectiveness of non-parametric memory consolidation techniques as a simple means of extending long video contexts, and future work could straightforwardly build on MC-ViT by exploring alternative consolidation strategies. For instance, incorporating insights from cognitive models of memory, such as the role of episodic and semantic memory systems, as well as theories of efficient coding (Barlow, 1961), could inspire new consolidation techniques. Furthermore, the concept of memory consolidation could be applied to other domains involving sequential data, such as natural language and audio processing, laying the foundation for personalized assistant technologies that jointly reason over multiple modalities.

Impact Statement

By adapting standard video architectures to the long-context setup, this work could potentially equip general-purpose assistant models with the ability to efficiently process long videos. These models will likely suffer from similar biases and potential harms associated with visual language models and large language models more generally.

Further, since this work focuses on efficient processing of long sequences without sacrificing performance, the corresponding methods and findings from this work could potentially be applied to other domains, such as NLP or audio, allowing for faster processing of large amounts of data and thus making long-context model training more readily available for widespread use.

Acknowledgements

We thank Andrew Zisserman, João Carreira, Carl Allen, and Nikhil Parthasarathy for their thoughtful feedback, Relja Arandjelović for fruitful discussions at the inception of this project, and Oliver Vikbladh, Eleanor Spens, and Neil Burgess for sharing their insights into memory consolidation in the human mind.

References

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Agarwal, P. K., Har-Peled, S., Varadarajan, K. R., et al. Geometric approximation via coresets. *Combinatorial and Computational Geometry*, 52(1):1–30, 2005.

Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., et al. Flamingo: A visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 2022.

Anil, R., Borgeaud, S., Wu, Y., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., et al. Gem-

ini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lučić, M., and Schmid, C. ViViT: A video vision transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

Ba, J. L., Kiros, J. R., and Hinton, G. E. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Barlow, H. B. Possible principles underlying the transformation of sensory messages. *Sensory communication*, 1 (01):217–233, 1961.

Bartlett, F. C. *Remembering: A study in experimental and social psychology*. Cambridge University Press, 1932.

Beltagy, I., Peters, M. E., and Cohan, A. Longformer: The long-document transformer. *arXiv:2004.05150*, 2020.

Bertasius, G., Wang, H., and Torresani, L. Is space-time attention all you need for video understanding? In *International Conference on Machine Learning*, 2021.

Bolya, D., Fu, C.-Y., Dai, X., Zhang, P., and Hoffman, J. Hydra attention: Efficient attention with many heads. In *European Conference on Computer Vision*, 2022.

Borgeaud, S., Mensch, A., Hoffmann, J., Cai, T., Rutherford, E., Millican, K., Van Den Driessche, G. B., Lespiau, J.-B., Damoc, B., Clark, A., et al. Improving language models by retrieving from trillions of tokens. In *International Conference on Machine Learning*, 2022.

Carreira, J. and Zisserman, A. Quo vadis, action recognition? a new model and the kinetics dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Chen, X., Wang, X., Changpinyo, S., Piergiovanni, A., Padlewski, P., Salz, D., Goodman, S., Grycner, A., Mustafa, B., Beyer, L., et al. PaLI: A jointly-scaled multilingual language-image model. In *International Conference on Learning Representations*, 2023.

Dai, Z., Yang, Z., Yang, F., Cohen, W. W., and Salakhutdinov, R. R. Good semi-supervised learning that requires a bad gan. In *Advances in Neural Information Processing Systems*, 2017.

Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V., and Salakhutdinov, R. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the Association for Computational Linguistics*, 2019.

Dao, T., Fu, D., Ermon, S., Rudra, A., and Ré, C. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. *Advances in Neural Information Processing Systems*, 2022.

Dong, X., Bao, J., Chen, D., Zhang, W., Yu, N., Yuan, L., Chen, D., and Guo, B. CSWin transformer: A general vision transformer backbone with cross-shaped windows. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Fan, Q., Panda, R., et al. Can an image classifier suffice for action recognition? In *International Conference on Learning Representations*, 2021.

Feichtenhofer, C., Fan, H., Malik, J., and He, K. SlowFast networks for video recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

Google AI. Bard [large language model], 2023.

Grauman, K., Westbury, A., Byrne, E., Chavis, Z., Furnari, A., Girdhar, R., Hamburger, J., Jiang, H., Liu, M., Liu, X., et al. Ego4D: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Graves, A., Wayne, G., and Danihelka, I. Neural Turing machines. *arXiv preprint arXiv:1410.5401*, 2014.

Han, T., Xie, W., and Zisserman, A. Memory-augmented dense predictive coding for video representation learning. In *Proceedings of the European Conference on Computer Vision*, 2020.

Herzig, R., Ben-Avraham, E., Mangalam, K., Bar, A., Chechik, G., Rohrbach, A., Darrell, T., and Globerson, A. Object-region video transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Hu, E. J., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2021.

Jaegle, A., Borgeaud, S., Alayrac, J.-B., Doersch, C., Ionescu, C., Ding, D., Koppula, S., Zoran, D., Brock, A., Shelhamer, E., et al. PerceiverIO: A general architecture for structured inputs & outputs. In *International Conference on Learning Representations*, 2022.

Jia, M., Tang, L., Chen, B.-C., Cardie, C., Belongie, S., Hariharan, B., and Lim, S.-N. Visual prompt tuning. In *Proceedings of the European Conference on Computer Vision*, 2022.

Kim, S., Kim, J.-H., Lee, J., and Seo, M. Semi-parametric video-grounded text generation. *arXiv preprint arXiv:2301.11507*, 2023.

Lai, Z., Lu, E., and Xie, W. MAST: A memory-augmented self-supervised tracker. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

Li, K., He, Y., Wang, Y., Li, Y., Wang, W., Luo, P., Wang, Y., Wang, L., and Qiao, Y. VideoChat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*, 2023.

Li, Y., Li, Y., and Vasconcelos, N. RESOUND: Towards action recognition without representation bias. In *Proceedings of the European Conference on Computer Vision*, 2018.

Li, Y., Mao, H., Girshick, R., and He, K. Exploring plain vision transformer backbones for object detection. In *Proceedings of the European Conference on Computer Vision*, 2022a.

Li, Y., Wu, C.-Y., Fan, H., Mangalam, K., Xiong, B., Malik, J., and Feichtenhofer, C. MViTv2: Improved multiscale vision transformers for classification and detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022b.

Lin, K., Ahmed, F., Li, L., Lin, C.-C., Azarnasab, E., Yang, Z., Wang, J., Liang, L., Liu, Z., Lu, Y., et al. MM-VID: Advancing video understanding with GPT-4V(ision). *arXiv preprint arXiv:2310.19773*, 2023.

Liu, Z., Ning, J., Cao, Y., Wei, Y., Zhang, Z., Lin, S., and Hu, H. Video Swin transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Mangalam, K., Akshulakov, R., and Malik, J. EgoSchema: A diagnostic benchmark for very long-form video language understanding. In *Advances in Neural Information Processing Systems*, 2023.

Marr, D. *Simple memory: a theory for archicortex*. Philosophical Transactions of the Royal Society of London, 1971.

Martins, P. H., Marinho, Z., and Martins, A. F. ∞-former: Infinite memory transformer. In *Proceedings of the Association for Computational Linguistics*, 2022.

Miech, A., Zhukov, D., Alayrac, J.-B., Tapaswi, M., Laptev, I., and Sivic, J. Howto100M: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

Oh, S. W., Lee, J.-Y., Xu, N., and Kim, S. J. Video object segmentation using space-time memory networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

Papalampidi, P., Koppula, S., Pathak, S., Chiu, J., Heyward, J., Patraucean, V., Shen, J., Miech, A., Zisserman, A., and Nematzdeh, A. A simple recipe for contrastively pre-training video-first encoders beyond 16 frames. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Pătrăucean, V., Smaira, L., Gupta, A., Continente, A. R., Markeeva, L., Banarse, D., Koppula, S., Heyward, J., Malinowski, M., Yang, Y., et al. Perception Test: A diagnostic benchmark for multimodal video models. In *Advances in Neural Information Processing Systems*, 2023.

Piergiovanni, A., Kuo, W., and Angelova, A. Rethinking video ViTs: Sparse video tubes for joint image and video learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Piergiovanni, A., Nobel, I., Kim, D., Ryoo, M. S., Gomes, V., and Angelova, A. Mirasol3B: A multimodal autoregressive model for time-aligned and contextual modalities. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognitio*, 2024.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021.

Rae, J. W., Potapenko, A., Jayakumar, S. M., and Lillicrap, T. P. Compressive transformers for long-range sequence modelling. *International Conference on Learning Representations*, 2020.

Ryali, C., Hu, Y.-T., Bolya, D., Wei, C., Fan, H., Huang, P.-Y., Aggarwal, V., Chowdhury, A., Poursaeed, O., Hoffman, J., et al. Hiera: A hierarchical vision transformer without the bells-and-whistles. In *International Conference on Machine learning*, 2023.

Ryoo, M. S., Gopalakrishnan, K., Kahatapitiya, K., Xiao, T., Rao, K., Stone, A., Lu, Y., Ibarz, J., and Arnab, A. Token Turing machines. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Song, E., Chai, W., Wang, G., Zhang, Y., Zhou, H., Wu, F., Guo, X., Ye, T., Lu, Y., Hwang, J.-N., et al. MovieChat: From dense token to sparse memory for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Spens, E. and Burgess, N. A generative model of memory construction and consolidation. *Nature Human Behaviour*, pp. 1–18, 2024.

Sun, C., Shrivastava, A., Singh, S., and Gupta, A. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017.

Tulving, E. Memory and consciousness. *Canadian Psychology/Psychologie canadienne*, 26(1):1, 1985.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

Wang, R., Chen, D., Wu, Z., Chen, Y., Dai, X., Liu, M., Jiang, Y.-G., Zhou, L., and Yuan, L. BEVT: BERT pretraining of video transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022a.

Wang, S., Zhao, Q., Do, M. Q., Agarwal, N., Lee, K., and Sun, C. Vamos: Versatile action models for video understanding. *arXiv preprint arXiv:2311.13627*, 2023a.

Wang, W., Xie, E., Li, X., Fan, D.-P., Song, K., Liang, D., Lu, T., Luo, P., and Shao, L. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

Wang, W., Dong, L., Cheng, H., Liu, X., Yan, X., Gao, J., and Wei, F. Augmenting language models with long-term memory. *Advances in Neural Information Processing Systems*, 2023b.

Wang, Y., Li, K., Li, Y., He, Y., Huang, B., Zhao, Z., Zhang, H., Xu, J., Liu, Y., Wang, Z., et al. InternVideo: General video foundation models via generative and discriminative learning. *arXiv preprint arXiv:2212.03191*, 2022b.

Wang, Z., Li, M., Xu, R., Zhou, L., Lei, J., Lin, X., Wang, S., Yang, Z., Zhu, C., Hoiem, D., et al. Language models with image descriptors are strong few-shot video-language learners. *Advances in Neural Information Processing Systems*, 2022c.

Wu, C.-Y., Feichtenhofer, C., Fan, H., He, K., Krahenbuhl, P., and Girshick, R. Long-term feature banks for detailed video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

Wu, C.-Y., Li, Y., Mangalam, K., Fan, H., Xiong, B., Malik, J., and Feichtenhofer, C. 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*, 2022.

Xiao, J., Shang, X., Yao, A., and Chua, T.-S. Next-QA: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

Xiao, J., Zhou, P., Yao, A., Li, Y., Hong, R., Yan, S., and Chua, T.-S. Contrastive video question answering via video graph transformer. *arXiv preprint arXiv:2302.13668*, 2023.

Xu, H., Ghosh, G., Huang, P.-Y., Okhonko, D., Aghajanyan, A., Metze, F., Zettlemoyer, L., and Feichtenhofer, C. VideoCLIP: Contrastive pre-training for zero-shot video-text understanding. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2021a.

Xu, M., Xiong, Y., Chen, H., Li, X., Xia, W., Tu, Z., and Soatto, S. Long short-term transformer for online action detection. *Advances in Neural Information Processing Systems*, 2021b.

Yan, S., Xiong, X., Arnab, A., Lu, Z., Zhang, M., Sun, C., and Schmid, C. Multiview transformers for video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022a.

Yan, S., Zhu, T., Wang, Z., Cao, Y., Zhang, M., Ghosh, S., Wu, Y., and Yu, J. Video-text modeling with zero-shot transfer from contrastive captioners. *arXiv preprint arXiv:2212.04979*, 2022b.

Yang, T., Zhu, Y., Xie, Y., Zhang, A., Chen, C., and Li, M. AIM: Adapting image models for efficient video action recognition. In *International Conference on Learning Representations*, 2023.

Ye, Q., Xu, G., Yan, M., Xu, H., Qian, Q., Zhang, J., and Huang, F. HiTeA: Hierarchical temporal-aware video-language pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

Yu, S., Cho, J., Yadav, P., and Bansal, M. Self-chained image-language model for video localization and question answering. In *Advances in Neural Information Processing Systems*, 2023.

Zaheer, M., Guruganesh, G., Dubey, K. A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L., et al. Big Bird: Transformers for longer sequences. *Advances in Neural Information Processing Systems*, 2020.

Zeng, A., Attarian, M., Choromanski, K. M., Wong, A., Welker, S., Tombari, F., Purohit, A., Ryoo, M. S., Sindhwani, V., Lee, J., et al. Socratic models: Composing zero-shot multimodal reasoning with language. In *International Conference on Learning Representations*, 2022.

Zhang, C., Lu, T., Islam, M. M., Wang, Z., Yu, S., Bansal, M., and Bertasius, G. A simple LLM framework for long-range video question-answering. *arXiv preprint arXiv:2312.17235*, 2023.

A. ST-ViT and MA-ViT Pseudocode

We include pseudocode for streaming ViT and memory-augmented ViT in Algorithm 2 and Algorithm 3.

B. Training Details

B.1. Pretraining Short-Video ViViT Encoders

In this work we re-purpose a standard ViViT (Arnab et al., 2021) encoder into MC-ViT and fine-tune it for long-context understanding as described in Section 3.3. We use a pre-trained ViViT from Papalampidi et al. (2024) (referred to as ShortViViT therein), whose training details follow those of Xu et al. (2021a), and which we reproduce here for completeness. Note however that similar pretrained ViViT's are available from public repositories[2].

Phase 1: Image-text pretraining. A standard ViT image encoder and an associated BERT text encoder are pretrained with multimodal contrastive learning (Radford et al., 2021) on a mixture of paired image-text datasets (ALIGN, Jia et al. (2022); LTIP, Alayrac et al. (2022); and JFT, Sun et al. (2017)). We utilize two variants of the underlying vision and text encoder: ViT-B with BERT-M for MC-ViT-B, as well as ViT-L and BERT-B for MC-ViT-L.

Phase 2: Short-video pretraining. Initializing from the phase 1 image/text encoders, Papalampidi et al. (2024) train a ViViT video encoder and associated text encoder on short (16-frame) videos sampled from the HowTo100M (Miech et al., 2019) and VTP (Alayrac et al., 2022) datasets, together with the image datasets (treated as single-frame videos) from phase 1, again using multimodal contrastive learning (Equation 8). The parameters of the phase 1 ViT and phase 2 ViViT are identical except for the patch-embedding projection and positional embeddings, which they extend temporally at initialization time by simply replicating them as in Arnab et al. (2021).

B.2. Implementation Details

We provide all experimental settings for training our MC-ViT model variants in Table 5, categorized by dataset. We report our experimental setup for training the large model variants, but we follow the same settings when training the base versions, except for Diving48, where we use 128 instead of 64 frames.

For training on Next-QA, we additionally use low-rank adaptation (LoRA) to avoid overfitting. Given each ViViT layer, we decompose the linear QKV input projection, the output self-attention projection, and the dense feed-forward block:

$$h = W_o x + \frac{1}{\alpha} BAx, \tag{9}$$

where W_o is the frozen pretrained weight matrix, B and A are zero-initialized low-rank matrices, $B \in \mathbb{R}^{d_m \times r}$, $A \in \mathbb{R}^{r \times d_m}$, $r \ll d_m$ is the rank of the decomposition matrices, and α is a hyperparameter for easier tuning of the model, as recommended by (Hu et al., 2021). We use a r=128, resulting in 12% of model parameters for the large model version.

[2] https://huggingface.co/docs/transformers/model_doc/vivit

Algorithm 2 Streaming ViT.

```
def streaming_vit(video, n_chunks, n_layers, pos_emb):
  emb = linear_proj(video) + pos_emb
  chunked_video = np.split(emb, n_chunks, axis=1)
  zs = []
  memory = None
  for z in chunked_video:
     z_norm = layer_norm(z)
     for _ in range(n_layers):
        y = self_attention(z_norm) + z
        y_norm = layer_norm(y)
        z = mlp(y_norm) + y
        zs.append(z)
  return np.concatenate(zs, axis=1)
```

Algorithm 3 Memory-augmented ViT.

```
def ma_vit(video, n_chunks, n_layers, pos_emb):
  emb = linear_proj(video) + pos_emb
  chunked_video = np.split(emb, n_chunks, axis=1)
  memory = {layer: None for layer in range(n_layers)}
  zs = []
  for z in chunked_video:
     z_norm = layer_norm(z)
     for _ in range(n_layers):
        if memory[layer] is None:
           y = self_attention(z_norm) + z
        else:
           kv = np.concatenate(z_norm, memory[layer]))
           y = cross_attention(q=z_norm, kv=kv) + z
        y_norm = layer_norm(y)
        z = mlp(y_norm) + y
        memory[layer] = memory_concatenation(memory[
             layer], z)
        memory[layer] = layer_norm(memory[layer])
        zs.append(z)
  return np.concatenate(zs, axis=1)
```

C. Additional Baselines

Joint space-time attention with masking. To implement joint space-time attention with masking, we follow Papalampidi et al. (2024) and randomly mask 25% of the input video tokens before feeding them into the transformer. To better align training with evaluation, we perform masking both at fine-tuning and inference time. In particular, this allows the masked model to be more memory- and compute-efficient than full space-time attention.

Late temporal fusion. A common approach for efficiently processing videos is applying a learnable late temporal fusion module for aggregating information over frame-level (Alayrac et al., 2022; Yan et al., 2022b) or short-video representations (Piergiovanni et al., 2024). We follow Alayrac et al. (2022) and Yan et al. (2022b) and use

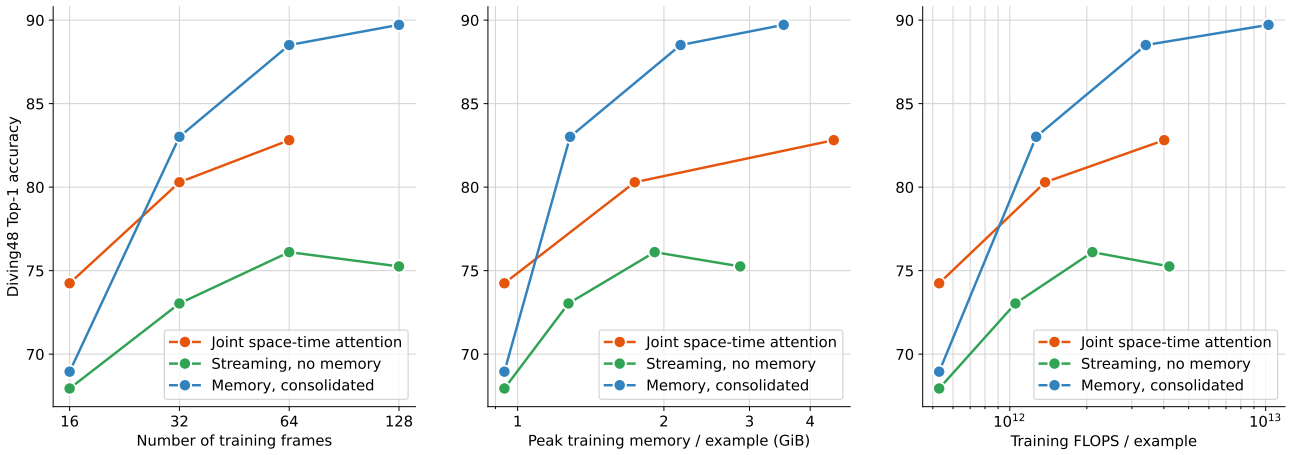

Figure A.1. **MC-ViT efficiently trains on long videos.** Fine-grained video understanding on Diving48 as a function of number of training frames (**left**), train-time memory consumption (**middle**), and training complexity (FLOPS, **right**), for joint space-time attention (red), memory-less streaming setting (green), and our proposed method MC-ViT (blue). MC-ViT reaches the highest accuracy with 3.5× less training memory and 3.2× fewer training FLOPS than joint space-time attention.

Algorithm 4 GPT-4V Zero-Shot Prompt.

```
You are a helpful assistant, an expert in answering
     questions about videos. You will be given a
     question about a video and five possible answer
     options. You will be provided frames from the
     video, sampled evenly across the video. You are
     very capable, think step-by-step when answering
     the question.

Question: <question>

Possible answer choices:
A. <Answer Choice 1>
B. <Answer Choice 2>
C. <Answer Choice 3>
D. <Answer Choice 4>
E. <Answer Choice 5>

ONLY output A, B, C, D, or E to answer. DO NOT OUTPUT
     with the full answer text or any other words,
     output only the single letter indicating the
     correct choice: one of A, B, C, D, or E.

<video frames attached to the HTTP request>
```

a light-weight Perceiver-resampler (Jaegle et al., 2022) to contextualize and compress information across segment-level information. We process the long video in 16-frame segments similarly to ST-ViT and feed all output tokens across segments into a single layer of a randomly initialized Perceiver-resampler. We use 256 latent learnable queries for learning a global compressed representation of the video.

MeMViT. We implement MeMViT's convolutional compression module (Wu et al., 2022) and apply it to every video segment as a parametric alternative to MC-ViT's consolidation mechanism. We experiment with 4× compression (i.e. kernel sizes of 1×2×2 and 4×1×1), 16× compression (kernel size 4×2×2) and 64× compression (kernel size 4×4×4). Similarly to the original paper, we find the 4×2×2 kernel to yield best results on the Diving48 dataset, corresponding to MC-ViT's memory segment length of $K = 128$. We also analyze the memory footprint and FLOPs of MeMViT during training and inference and we find the differences between MC-ViT and MeMViT to be almost indistinguishable, with MeMViT's memory footprint slightly higher (+4%) due to additional convolutional kernel parameters.

Algorithm 5 GPT-4 Turbo Zero-Shot Prompt.

```
You are a helpful assistant, an expert in answering
     questions about videos. You will be given a
     question about a video and five possible answer
     options. You will not be provided frames from the
      video, but still do your best to guess an answer.
      You are very capable, think step-by-step when
     answering the question.

Question: <question>

Possible answer choices:
A. <Answer Choice 1>
B. <Answer Choice 2>
C. <Answer Choice 3>
D. <Answer Choice 4>
E. <Answer Choice 5>

ONLY output A, B, C, D, or E to answer. DO NOT OUTPUT
     with the full answer text or any other words,
     output only the single letter indicating the
     correct choice: one of A, B, C, D, or E.
```

D. Evaluating Proprietary Models

We evaluated multiple choice video question answering using the proprietary GPT-4V ("gpt-4-vision-preview") model (version as deployed on January 6 through 9th, 2023). This model has a context window of 128K tokens, with training data up to April 2023. The model was queried in a zero-shot manner, using the prompt described in Algorithm 4. Each request to the model was attached with a fixed number of frames, varying from 16 frames to 512 frames. We use 256×256 resolution frames for all queries. Note that the token count (and corresponding cost) also increases dramat-

Table 5. Training specifications for fine-tuning MC-ViT per dataset.

	EgoSchema	Diving48	Next-QA	Perception Test
Optimizer	AdamW			
Learning rate schedule	Cosine with linear warmup			
Gradient clip	2.0			
Linear warmup steps	1k			
Frame-level resolution	256×256			
Frame-level cropping	center crop			
Convolution kernel	2×16×16			
Batch size	128	256		
Label smoothing	0	0.1		
# memories/segment (K)	128	512		
Frame sampling	Uniform			4 FPS
Weight decay rate	0	0	1e-2	
Base learning rate	2e-5	5e-5	1e-6	
Training steps	5k	30k	20k	

ically as the number of frames accompanying the question increases: 16 frames require roughly 4.3K tokens per question/answer pair, 64 frames require 16.5K tokens, and 256 frames require 65.5K tokens. On the EgoSchema subset we found similar performance from using 16, 64, and 256 frames (63.5%, 63.9%, and 63.6% accuracy respectively), hence we used 16 frames when evaluating on the full set. Frames were sampled linearly from the video, and we use the 'auto'(-select) resolution quality setting.

For obtaining blind results, we used the "gpt-4-1106-preview" (GPT-4 Turbo) model. The model was queried in a zero-shot manner, using the prompt described in Algorithm 5. Models were queried on January 17, 2023. Each request was approximately 250 tokens.

For both the blind and vision-enabled model, a small number of requests come back with a format that doesn't adhere to the A/B/C/D/E characters representing the five possible answer choices (uppercase or lowercase). For example, the model sometimes responds with "The answer is C", or the actual text of the answer choice, instead of the corresponding answer choice letter. In those cases, we apply answer post-processing to extract out the corresponding answer choice by lowercasing the response and checking for the common regex, `the answer is (a|b|c|d|e)`. If the match fails, we compute the Levenshtein edit distance with each answer choice, choosing the lowercased answer response that most closely matches the lowercased proprietary model output.