MEMORY INTERNALIZATION VIA TEST-TIME TRAINING FOR LONG-VIDEO UNDERSTANDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-video understanding often requires answering many questions about a single video whose evidence is distributed across extended temporal contexts. Conventional video-language models repeatedly encode frames or retrieve visual evidence for every question. Parameterized video memory offers a write-once alternative, but existing methods rely primarily on synthesized text and fixed adaptation configurations, leaving open where video knowledge should be written and how visual evidence should shape the memory. We introduce ***M**emory **I**nternalization via **T**est-**T**ime **T**raining* (**MIT**^{3}), a per-video adaptation framework that writes each long video into compact, video-specific adaptive parameters for video-free, text-only question answering. Under compact multi-scale supervision, causal tracing measures the causal memory utility of each model layer for video-grounded answering and selects video-specific adaptation locations. Visual memory distillation then trains the text-only memory to reproduce the representation change induced by the relevant video clip, preserving the internal effect of visual evidence beyond textual supervision. Across LVBench, LongVideoBench, and Video-MME, MIT^3 achieves data-efficient video-free question answering with substantially fewer synthesized samples than prior parameterized video-memory methods, demonstrating that effective video internalization depends on both where knowledge is written and how visual evidence is distilled into the model.

1 INTRODUCTION

Long videos increasingly serve as high-value, persistent information sources, from lecture archives and meeting recordings to egocentric streams and personal media collections. In these settings, a central task is long-video question answering (QA): a single video must support many questions whose evidence may be brief, recur with subtle changes, or depend on distant events. Modern video-language models (VLM) typically answer each question by placing sampled frames or compressed visual tokens into the language-model context (Maaz et al., 2024; Zhang et al., 2023). Sparse context can omit critical evidence, whereas dense context repeatedly incurs visual-token and memory costs, making efficient knowledge reuse for subsequent questions a central challenge for long-video QA.

Existing approaches improve the video-conditioned question-answering path through frame selection, token compression, sparse memories, redundancy reduction, and longer multimodal contexts (Song et al., 2024; Chen et al., 2025; Wang et al., 2024). Retrieval-based methods complement these architectures by preprocessing videos into searchable memories and locating relevant clips for each question (Ren et al., 2026). These strategies make long inputs more tractable, but their inference path remains video-dependent: each question still requires visual encoding, cached representations, retrieval, or inspection of relevant clips. When one video must support many questions, as in lecture archives, meetings, surveillance review, or personalized video assistants, repeatedly reconstructing much of the same visual context becomes an avoidable systems bottleneck.

Video-free inference offers a complementary scaling axis for this setting: it writes a video's knowledge into compact parameters once and answers subsequent questions from text alone. Neural Knowledge Representation (NKR) (Guan et al., 2026) instantiates this idea by generating descriptions and question-answer pairs from a video and fitting a video-specific adapter (Hu et al., 2021). Once fitted, the adapter acts as compact parametric memory for text-only question answering, a formulation contrasted with conventional video-dependent inference in Fig. 1. Effective internalization

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

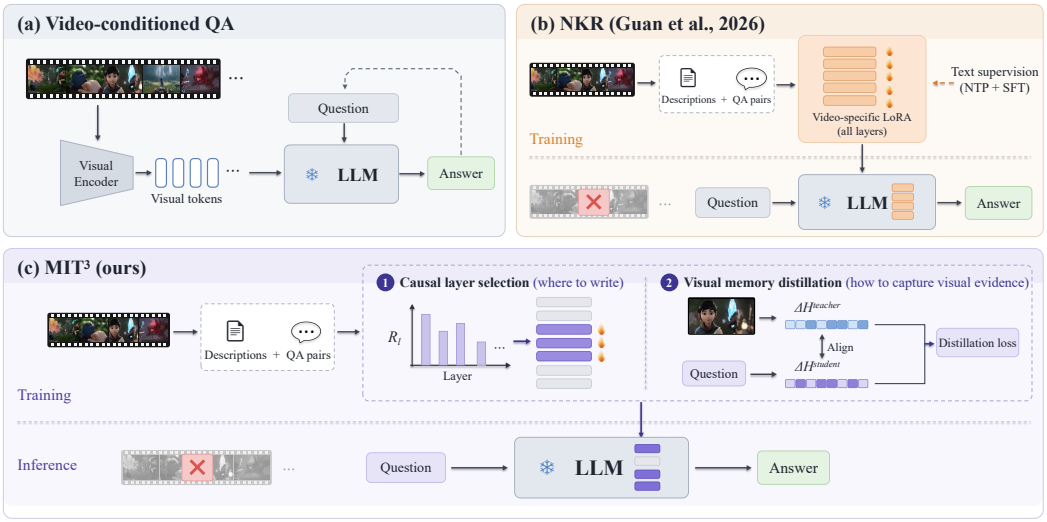

Figure 1: Comparison of video-conditioned question answering, NKR, and MIT^3. Conventional video-language models encode or retrieve visual evidence per question. NKR trains a video-specific LoRA adapter on synthesized text using a fixed all-layer configuration. MIT^3 instead selects video-specific model layers through causal tracing and distills the visual-evidence-induced representation change into these memory-writing locations, enabling video-free question answering.

therefore poses two knowledge-writing questions. First, *where should a particular video's knowledge be written?* Transformer layers play distinct computational roles, whereas existing adaptation configurations are typically fixed across videos. Second, *how should memory capture the effect of visual evidence?* Synthetic descriptions provide factual supervision, while video-conditioned representations directly signal how seeing the video changes internal model computation.

To address these two questions, we propose ***M**emory **I**nternalization via **T**est-**T**ime **T**raining* (**MIT**^{3}), a test-time training framework for video-specific adaptive memory. To determine *where* to write, MIT^3 uses causal tracing to extract the hidden-state change induced by video input, inject this change at each model layer, and score the resulting improvement in answer likelihood. Layers with the highest causal memory utility are selected as video-specific adaptive locations, where the frozen backbone most effectively converts video evidence into answer-relevant computation. To determine *how* to write, visual memory distillation uses a frozen video-conditioned teacher to supply the representation change induced by the relevant clip, and trains the text-only adapter to reproduce that change from the question. With compact multi-scale supervision, these mechanisms store video knowledge at effective computational sites while preserving the internal effect of visual evidence.

To evaluate the effectiveness of MIT^3, we conduct experiments on LVBench (Wang et al., 2025a), LongVideoBench (Wu et al., 2024), and Video-MME (Fu et al., 2025), spanning long-video and general video question answering across diverse durations. On LVBench, MIT^3 improves the matched NKR-style baseline by 6.8 points with Qwen3-VL-8B and by 5.7 points with Qwen3-VL-32B, demonstrating the value of causal localization and visual representation matching under identical textual supervision. The gains extend to LongVideoBench-Long and Video-MME, where MIT^3 achieves 63.3 and 66.7 accuracy, respectively, confirming the value of the proposed memory-writing mechanism across extreme-duration and general video understanding settings. Ablation studies isolate the contributions of memory localization and visual distillation. Our main contributions are:

- We study the layer-wise video-memory distribution in the language model and find that layers with high causal memory utility exhibit a concentration pattern across videos.
- We introduce MIT^3, a test-time memory-writing framework that combines causal localization with visual memory distillation for video-free question answering.
- We demonstrate data-efficient video internalization across LVBench, LongVideoBench, and Video-MME with substantially less synthesized supervision.

2 Related Work

2.1 Video Understanding

Video question answering requires models to ground language in visual entities, actions, temporal order, and causal relations. Early benchmarks progressively expanded this reasoning demand from short animated clips to more complex web videos and causal-temporal questions(Tapaswi et al., 2016; Lei et al., 2018; Xiao et al., 2021). Modern video-language models connect a visual encoder to an LLM through learned projection or cross-modal modules and use instruction tuning to support question answering (Maaz et al., 2024; Zhang et al., 2023; Lin et al., 2024; Jin et al., 2024). They answer by integrating sampled frames or visual tokens into the language-model context, making the quality and cost of reasoning depend on the visual information available for each question.

2.2 Long-Video Context Modeling

Long-video question answering further requires locating and combining evidence scattered over minutes or hours. Recent benchmarks evaluate long-range retrieval, temporal reasoning, and event comprehension in this setting (Mangalam et al., 2023; Fu et al., 2025; Wang et al., 2025a; Wu et al., 2024). Existing methods extend multimodal context capacity (Chen et al., 2025; Zhang et al., 2024), compress observations into sparse memories (Song et al., 2024; He et al., 2024), reduce temporal and knowledge redundancy (Wang et al., 2024; Ren et al., 2024), or retrieve relevant evidence from external multimodal indices (Ren et al., 2026). Although these approaches improve access to long-range evidence, answering each question still requires video tokens, cached representations, or retrieval. They therefore optimize access to video evidence rather than internalizing a video's knowledge for subsequent text-only question answering.

A complementary direction internalizes a video's knowledge into video-specific parameters, enabling video-free, text-only question answering after one-time per-video adaptation. NKR (Guan et al., 2026) synthesizes multi-scale descriptions and question-answer pairs and trains a video-specific adapter (Hu et al., 2021), demonstrating the feasibility of parameterized video memory. However, its internalization process primarily relies on synthesized text and adopts a fixed adaptation configuration, without explicitly determining where video knowledge should be written within the backbone or how visual evidence should shape the resulting memory.

2.3 Test-Time Training

Test-time training adapts models on test inputs with auxiliary self-supervised objectives to improve robustness under distribution shifts (Sun et al., 2020; Liu et al., 2021; Gandelsman et al., 2022; Zhang et al., 2022). Related fully test-time adaptation methods also update models directly from unlabeled test data, for example through entropy minimization (Wang et al., 2020; 2022; Niu et al., 2022; 2023). Unlike these methods, MIT^3 treats each long video as a test-time memory-writing instance, using the video once to construct adaptive parameters that answer subsequent text-only questions. This formulation introduces two video-memory-specific requirements beyond standard test-time training: determining where video knowledge should be written and how visual evidence should shape the resulting memory. MIT^3 addresses them through causal tracing for video-specific adaptation locations and visual memory distillation for video-induced representation changes.

3 Method

3.1 Overview

Let V be a long video, Q a question about its content, and A the corresponding answer. A conventional video-language model $\mathcal{M}$ performs video-conditioned question answering as

$$A \sim p_{\mathcal{M}}(A \mid V, Q), \tag{1}$$

requiring V, or representations derived from it, whenever a question is asked. We instead write the knowledge of V into compact adaptive parameters θ_V and answer subsequent questions from text:

$$A \sim p_{\mathcal{M}_{\theta_V}}(A \mid Q). \tag{2}$$

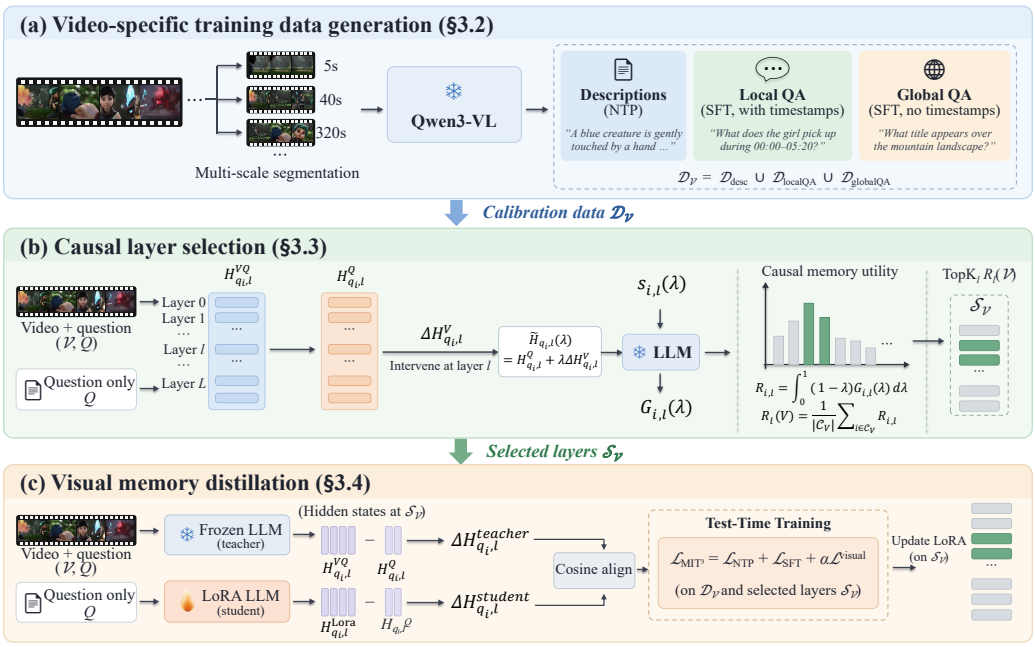

Figure 2: Overview of MIT^3. (a) A VLM constructs multi-scale descriptions, temporally grounded local QA, and global QA to cover video knowledge. (b) Delta-injection causal tracing estimates each model layer's causal memory utility and selects top-K layers under a fixed adaptation budget. (c) Visual memory distillation optimizes LoRA modules in selected layers to reproduce normalized video-induced representation changes, transferring visual evidence into text-only memory.

We instantiate θ_V as LoRA parameters (Hu et al., 2021) inserted into the attention sublayers of selected model layers, where each model layer corresponds to one Transformer block. The backbone remains frozen, and each video is represented by its own lightweight adapter.

MIT^3 internalizes each video through three stages, shown in Fig. 2. First, a VLM constructs compact multi-scale descriptions and local and global question-answer pairs. Second, delta-injection causal tracing estimates each model layer's causal memory utility from a small set of temporally grounded local QA pairs and selects top-K layers under a fixed adaptation budget. Third, LoRA modules in selected layers are optimized with textual supervision and visual memory distillation, which trains the adapter to reproduce normalized video-induced representation changes. These stages determine what knowledge to write, where to write it, and how visual evidence shapes the memory.

3.2 Video-Specific Training Data Generation

The first stage constructs compact, video-specific supervision that preserves both fine-grained evidence and longer temporal structure. Following NKR (Guan et al., 2026), we adopt a multi-scale temporal partitioning paradigm. Specifically, our implementation segments V into a temporal pyramid whose window durations double from 5 seconds (i.e., 5, 10, 20, 40, ... seconds) up to the full video length. Short-scale windows are non-overlapping, while longer-scale windows overlap to retain events crossing segment boundaries. For each window, a VLM generates complementary low-, mid-, and high-level descriptions covering visible details, actions and transitions, and the overall content. For windows up to 40 seconds, overlapping information across the three levels is fused into one coherent, redundancy-reduced description; for windows of 80 seconds or longer, the three descriptions are retained separately. Further details on overlap and related configurations are provided in Appendix A.

Using the generated descriptions, we construct local and global QA supervision without another video pass. Local QA pairs are generated from individual window descriptions and cover entities, actions, attributes, temporal order, and clip-level reasoning. Each local question includes the times-

tamp or temporal range of its source window, allowing the adapter to encode temporal position and order while identifying the supporting video segment for visual memory distillation in Section 3.4. At inference, questions retain the native format of each evaluation benchmark. To capture video-wide knowledge, we merge recurring entities and selected long-window descriptions into a global memory that summarizes the event timeline, state changes, and major stages of the video. This global memory conditions global-QA generation and is also included as one next-token prediction example. Each generation round selects a different group of focus clips and produces N timestamp-free global QA pairs, cycling through the groups to distribute supervision across the video. These global questions target cross-clip retrieval and reasoning. The resulting descriptions and QA pairs form the video-specific training set:

$$\mathcal{D}_V = \mathcal{D}_{\text{desc}} \cup \mathcal{D}_{\text{localQA}} \cup \mathcal{D}_{\text{globalQA}}. \tag{3}$$

During adapter optimization, $\mathcal{D}_{\text{desc}}$ provides next-token prediction supervision, while $\mathcal{D}_{\text{localQA}}$ and $\mathcal{D}_{\text{globalQA}}$ provide fine-tuning examples. Temporally grounded local QA pairs associate each example with its supporting video segment for visual memory distillation, while description and global-QA examples remain text-only. Our description and entity construction follow the multi-level generation design of NKR (Guan et al., 2026), while local-QA generation adapts its evidence format to open-ended answering and the global-memory and full-video QA components provide video-wide supervision introduced by MIT^3. Appendix B provides the corresponding construction details.

3.3 Causal Localization of Video Memory

The second stage localizes where to write knowledge from V in the frozen backbone. Adapting every model layer distributes memory parameters throughout the backbone, while a fixed subset assumes shared storage locations across videos. MIT^3 instead uses delta-injection causal tracing to measure each model layer's video-specific causal memory utility. For a candidate layer l, we ask: *if the hidden-state change induced by the video is injected during question-only inference, how strongly can the remaining computation convert it into a higher likelihood for the paired answer?* The resulting answer gain ranks model layers by utility as video-specific memory-writing locations.

Video-induced representations. For each calibration triple (V_i, Q_i, A_i), we run the frozen model with the relevant video segment and question, then with the question alone. Let q_i denote the aligned positions of identical question tokens in both forward passes. Let $H^{VQ}_{q_i,l}$ and $H^{Q}_{q_i,l}$ denote their hidden states at model layer l under video-conditioned and question-only inference, respectively. We define the video-induced representation change on these aligned question tokens as

$$\Delta H^{V}_{q_i,l} = H^{VQ}_{q_i,l} - H^{Q}_{q_i,l}. \tag{4}$$

This delta captures how observing the relevant video segment changes the same question representations and provides the visual signal shared by causal localization and visual memory distillation.

Hidden-state intervention. We next measure how effectively the computation after model layer l can use the video-induced delta. For an intervention strength $\lambda \in [0, 1]$, we construct

$$\widetilde{H}_{q_i,l}(\lambda) = H^{Q}_{q_i,l} + \lambda \Delta H^{V}_{q_i,l}, \qquad \lambda \in [0, 1], \tag{5}$$

where small λ values test whether the remaining computation can exploit a limited video-induced change, while $\lambda = 1$ injects the complete question-token delta at layer l. We replace the question-token states at layer l with $\widetilde{H}_{q_i,l}(\lambda)$ and continue question-only inference through subsequent model layers. Let s^{Q}_i and $s_{i,l}(\lambda)$ denote the answer log-likelihood before and after intervention:

$$s^{Q}_i = \log p_{\mathcal{M}}(A_i \mid Q_i), \qquad s_{i,l}(\lambda) = \log p_{\mathcal{M}}(A_i \mid \widetilde{H}_{q_i,l}(\lambda)). \tag{6}$$

The intervention gain is $G_{i,l}(\lambda) = s_{i,l}(\lambda) - s^{Q}_i$. A larger gain indicates that computation after layer l more effectively converts the injected video-induced change into evidence for the paired answer.

Budget-averaged causal memory utility. The intervention curve $G_{i,l}(\lambda)$ characterizes the causal memory utility of model layer l as increasing amounts of the video-induced delta are injected. A compact adapter benefits from layers where even a limited representation change produces answer-relevant gains, rather than only becoming useful near full intervention. We therefore introduce

$b \in [0, 1]$ as the maximum intervention strength under consideration. For each b, the prefix integral $\int_0^b G_{i,l}(\lambda)\, d\lambda$ measures the cumulative answer gain available within that intervention budget. Averaging these prefix gains over all budgets gives

$$R_{i,l} = \int_0^1 \int_0^b G_{i,l}(\lambda)\, d\lambda\, db = \int_0^1 (1-\lambda) G_{i,l}(\lambda)\, d\lambda. \tag{7}$$

The equivalent weighted form emphasizes gains obtained from smaller interventions while retaining the utility accumulated across the full intervention trajectory. A high $R_{i,l}$ therefore identifies a layer that responds to limited video-induced changes and continues to provide answer-relevant utility as more visual information is injected.

For each video V, let $\mathcal{C}_V$ denote a calibration set of temporally grounded triples (V_i, Q_i, A_i) sampled across temporal scales. Averaging the example-level scores gives the video-level causal memory utility of model layer l:

$$R_l(V) = \frac{1}{|\mathcal{C}_V|} \sum_{i \in \mathcal{C}_V} R_{i,l}. \tag{8}$$

Under the fixed adaptation budget K, we select the layers with the highest video-level utilities:

$$\mathcal{S}_V = \underset{l}{\mathrm{TopK}}\ R_l(V). \tag{9}$$

The utility scores are computed once with the frozen model and without gradients before adapter optimization. LoRA modules are inserted only into the attention sublayers of model layers in $\mathcal{S}_V$, while the remaining layers and all backbone parameters stay frozen. This converts adapter placement from a fixed architectural configuration into a video-specific memory-writing decision.

3.4 Visual Memory Distillation

The third stage writes visual evidence into the selected memory locations by distilling how video conditioning redirects question representations. Synthesized descriptions provide explicit semantic supervision, while video-conditioned hidden states supply complementary signals about appearance, spatial configuration, and transient events. For each temporally grounded local QA example, the frozen model processes the relevant video segment and question as teacher, while the adapter-enabled student receives the same question without video input. Visual memory distillation trains the normalized adapter-induced changes to reproduce the normalized video-induced representation changes at the selected model layers $\mathcal{S}_V$. This transfers the internal computational effect of the relevant video segment into text-only memory without retaining frames or visual tokens.

Teacher and student deltas. At each selected model layer $l \in \mathcal{S}_V$, the frozen teacher provides the video-induced target defined in Equation 4:

$$\Delta H^{\text{teacher}}_{q_i,l} = H^{VQ}_{q_i,l} - H^{Q}_{q_i,l} = \Delta H^{V}_{q_i,l}. \tag{10}$$

The text-only student processes Q_i with the video-specific adapter enabled. Let $H^{\text{LoRA}}_{q_i,l}$ denote its hidden states at the same aligned question-token positions. The adapter-induced change is

$$\Delta H^{\text{student}}_{q_i,l} = H^{\text{LoRA}}_{q_i,l} - H^{Q}_{q_i,l}. \tag{11}$$

Both deltas are measured relative to the same question-only state $H^{Q}_{q_i,l}$. This shared reference isolates the transformations contributed by the video and the adapter, making their representation changes directly comparable.

Representation alignment. We align the teacher and student deltas using cosine distance. For model layer $l \in \mathcal{S}_V$, the token-averaged loss is

$$\mathcal{L}^{\text{visual}}_{i,l} = \frac{1}{|q_i|} \sum_{t \in q_i} \left(1 - \frac{\left\langle \Delta h^{\text{student}}_{i,l,t}, \Delta h^{\text{teacher}}_{i,l,t} \right\rangle}{\left\| \Delta h^{\text{student}}_{i,l,t} \right\|_2 \left\| \Delta h^{\text{teacher}}_{i,l,t} \right\|_2 + \epsilon} \right), \tag{12}$$

where ϵ provides numerical stability. The per-example visual loss averages this reproduction objective over the selected model layers:

$$\mathcal{L}^{\text{visual}}_{i} = \frac{1}{|\mathcal{S}_V|} \sum_{l \in \mathcal{S}_V} \mathcal{L}^{\text{visual}}_{i,l}. \tag{13}$$

Table 1: Accuracy and inference efficiency on LVBench and LongVideoBench. Time/query denotes inference time per question. † denotes controlled NKR-style baseline using the same data as MIT^3.

Method	Input	Time/query (s)	Mem. (GB)	LVBench		LongVideoBench	
				All	$>$ 1h	All	Long
VideoLLaMA3-7B	$<$ 768 fr.	64.5	1.2	45.3	–	–	–
AdaReTaKe-7B	$<$ 2048 fr.	45	0.9	51.2	48.4	–	55.5
VideoRAG	N/A	90	0.5	49.2	47.8	–	–
VCA	N/A	–	–	41.3	–	–	–
DVD-o3	N/A	180	0.3	74.2	–	–	–
NKR-8B	N/A	0.25	0	39.5	–	–	–
NKR-14B	N/A	0.33	0	48.8	48.6	–	52.9
Qwen3-VL-8B backbone, with 2 fps							
Direct	$<$ 768 fr.	31.1	15.1	45.6	42.1	60.2	48.0
NKR-style†	N/A	0.25	0	36.7	36.5	54.5	43.3
MIT^3 (Ours)	N/A	0.25	0	43.5	43.2	58.5	47.1
Qwen3-VL-32B backbone, with 2 fps							
Direct	$<$ 768 fr.	103.4	26.9	52.5	48.1	65.7	55.1
NKR-style†	N/A	0.46	0	44.9	45.3	57.4	47.3
MIT^3 (Ours)	N/A	0.46	0	50.6	**50.1**	63.3	**55.9**

This loss is evaluated for the temporally grounded local QA examples used for visual distillation, while description and global-QA examples provide text-only NTP and SFT supervision.

3.5 Test-Time Memory Optimization

For each video, we freeze the base VLM and optimize only LoRA parameters in attention sublayers of the causally selected model layers $\mathcal{S}_V$. Let $\mathcal{D}_{\text{visual}} \subseteq \mathcal{D}_{\text{localQA}}$ denote the temporally grounded local QA examples used for visual memory distillation. Their sample-averaged visual loss is

$$\mathcal{L}_{\text{visual}} = \frac{1}{|\mathcal{D}_{\text{visual}}|} \sum_{i \in \mathcal{D}_{\text{visual}}} \mathcal{L}_i^{\text{visual}}. \tag{14}$$

The complete test-time memory optimization objective is

$$\mathcal{L}_{\text{MIT}^3} = \mathcal{L}_{\text{NTP}} + \mathcal{L}_{\text{SFT}} + \alpha \mathcal{L}_{\text{visual}}, \tag{15}$$

where each term is averaged over its corresponding training subset and α controls the visual-distillation weight. After this one-time optimization, the resulting adapter θ_V serves as the video-specific memory. At inference, the frozen backbone receives only the question Q with θ_V enabled, requiring no video input, visual tokens, or retrieval. The same adapter can answer subsequent questions about V without reprocessing the video.

4 Experiments

4.1 Experimental Setup

Datasets and Baselines We evaluate MIT^3 on LVBench (Wang et al., 2025a), LongVideoBench (Wu et al., 2024), and Video-MME (Fu et al., 2025). For LVBench and LongVideoBench, we report results on both the full benchmarks and their long-video subsets, excluding subtitle and audio inputs. Baselines include published long-video methods (Zhang et al., 2025; Wang et al., 2025b; Ren et al., 2026; Yang et al., 2025; Zhang et al., 2026; Guan et al., 2026), direct Qwen3-VL-8B/32B inference (Bai et al., 2025), and a controlled NKR-style baseline. Since NKR's per-video training data are unavailable, we reproduce its data-generation procedure and train both NKR-style and MIT^3 on the same generated supervision. The NKR-style baseline optimizes only $\mathcal{L}_{\text{NTP}} + \mathcal{L}_{\text{SFT}}$, while MIT^3 additionally uses video-specific layer selection and visual memory distillation.

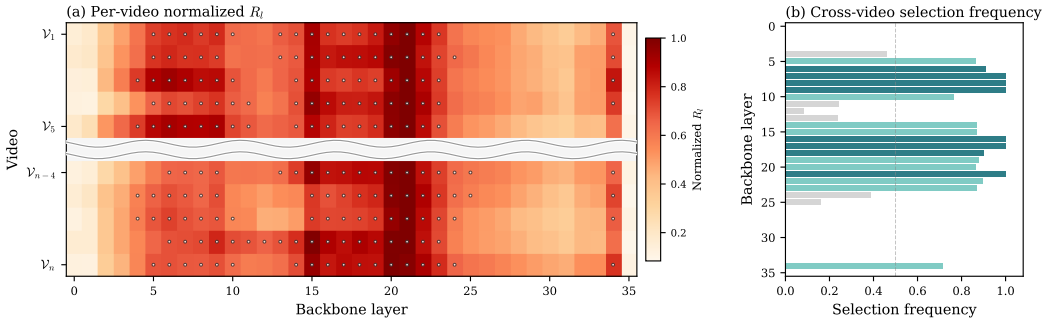

Figure 3: Layer-wise causal utility and top-18 selection frequency for Qwen3-VL-8B on LVBench.

Implementation Details Following video-NKR (Guan et al., 2026), we use LoRA with rank 256 and scaling factor 512, selecting the top 18 and 32 layers for Qwen3-VL-8B and Qwen3-VL-32B, respectively. For a one-hour video, our pipeline generates approximately 5.5k training examples, roughly one-eighth of those used by NKR. We also verify that no evaluation questions appear in adapter training or causal-localization calibration. Full dataset, data-generation, and optimization details are provided in Appendix C.

4.2 Main Results

Table 1 compares MIT^3 with published long-video systems, direct video-conditioned inference, and the matched NKR-style baseline. Under identical video-specific supervision, MIT^3 consistently outperforms the NKR-style baseline across both backbone sizes and all four evaluation settings, demonstrating the effectiveness of video-specific layer selection and visual memory distillation for parameterized video memory. Despite performing video-free, text-only inference after one-time internalization, MIT^3 remains close to direct video-conditioned inference on the full benchmarks, while requiring no video frames, visual tokens, retrieval, or video-dependent KV cache at query time. More importantly, with the 32B backbone, MIT^3 reaches 50.1 on LVBench $>$ 1h and 55.9 on LongVideoBench-Long, surpassing direct video-conditioned inference on both long-video subsets. Together with the small variation between the full LVBench and its $>$ 1h subset, these results highlight the effectiveness and stability of MIT^3 for long-video understanding. This makes MIT^3 particularly suitable for scenarios that repeatedly revisit long videos, such as archived surveillance review, recorded meetings, and lecture or instructional video analysis. We further analyze the resource advantages of MIT^3 over alternative inference paradigms in Appendix F.

4.3 Layer-Wise Distribution

We analyze the layer-wise causal utilities of Qwen3-VL-8B on LVBench to examine whether video-specific localization selects arbitrary layers or reveals a consistent organization of video memory. For each video, we normalize $R_l(V)$ across layers and identify the top-18 layers selected for adaptation. Figure 3(a) shows the resulting video-layer heatmap, while Figure 3(b) reports the selection frequency of each layer across videos.

The selected layers show substantial cross-video agreement. A compact group of layers is repeatedly assigned high causal memory utility, forming a shared memory-writing core that accounts for **65%** of all top-18 selections. At the same time, a smaller set of selected layers varies across videos, introducing content-dependent refinements around this common structure. These results suggest that causal localization captures both a consistent cross-video organization of memory and video-specific adaptation needs, rather than reducing to either a fixed layer subset or arbitrary per-video selection. The corresponding analysis for Qwen3-VL-32B is provided in Appendix E.

Table 2: Component and LoRA-target ablations on LVBench with Qwen3-VL-8B.

Configuration	Layer Selection	LoRA Target	Visual Loss	Accuracy (%)
Baseline	All	Attention	–	36.7
Random-K	Random K	Attention	–	34.1
All + Visual	All	Attention	✓	41.2
Causal Top-K	Causal Top-K	Attention	–	39.2
MIT^3	Causal Top-K	Attention	✓	43.5
MIT^3 (FFN)	Causal Top-K	FFN	✓	43.9

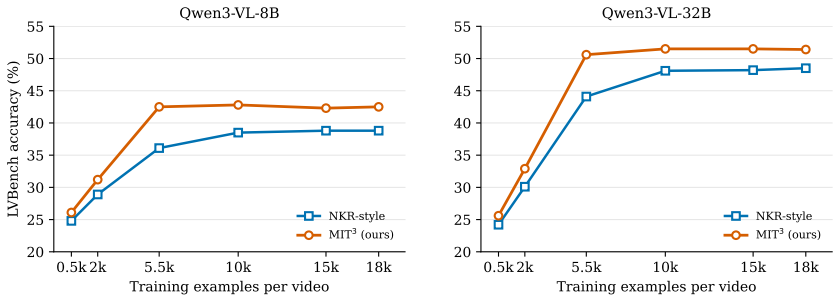

Figure 4: Data efficiency on LVBench under matched per-video supervision budgets.

4.4 Ablation Study

Component ablations. Table 2 isolates the contributions of layer selection and visual memory distillation on LVBench using Qwen3-VL-8B. All component ablations apply LoRA to the attention projections and compare all-layer, random-K, and causal top-K placement with or without the visual loss. We further examine the choice of LoRA target by applying the complete MIT^3 configuration to either the attention projections or the FFN while keeping all other settings fixed. The two choices yield comparable performance, we therefore follow the NKR configuration and use attention-targeted LoRA in all main experiments.

Data efficiency. We further compare NKR-style and MIT^3 under matched supervision budgets using Qwen3-VL-8B and Qwen3-VL-32B. The budgets in Fig. 4, 0.5k, 2k, 5.5k, 10k, 15k, and 18k examples per video, are estimated from a one-hour video, with 5.5k corresponding to our standard configuration. MIT^3 consistently outperforms the matched NKR-style baseline across all supervision budgets and both backbone sizes, indicating that the proposed memory-writing strategy improves the utilization of synthesized supervision rather than relying on a larger training set. Notably, the performance of MIT^3 approaches saturation once the supervision reaches a moderate scale, and further increasing the number of generated examples yields only marginal gains beyond the 5.5k setting. This trend suggests that the benefit of MIT^3 is driven primarily by how video knowledge is localized and distilled into the model, rather than by continued scaling of synthetic supervision. In other words, causal localization and visual memory distillation allow the model to extract more useful video-specific knowledge from a compact set of training examples.

5 Conclusion

We introduced MIT^3, a per-video test-time training framework that converts each long video into compact adaptive memory for write-once, query-many understanding. By answering subsequent questions through text-only inference, MIT^3 eliminates repeated visual encoding or retrieval while retaining video-specific knowledge in lightweight parameters. Causal localization identifies where this knowledge can be written most effectively, and visual memory distillation trains the adapter to reproduce the representation change induced by visual evidence. Across LVBench, LongVideoBench, and Video-MME, MIT^3 consistently improves controlled text-only internalization using roughly one-eighth of NKR's synthesized supervision. Its per-video state is $45.6\times$ smaller than persistent KV reuse with Qwen3-VL-32B, enabling scalable access to growing videos.

Ethics Statement

This work uses publicly available video-understanding benchmarks and does not involve collecting new personal data or conducting human-subject experiments. As video-memory systems may raise privacy concerns and inherit biases from generated supervision, we recommend appropriate consent, data governance, and careful auditing, particularly for deployment in sensitive or high-stakes settings.

Reproducibility Statement

We provide the model formulation, causal-localization procedure and training objectives in the main paper. Section 4.1 reports the principal experimental settings, including the calibration-set size, intervention grid, numerical integration method, LoRA configuration, and training hyperparameters. Appendix C gives additional implementation details, while Appendix A specifies the multi-scale data construction and provides representative training examples. The complete data-generation prompts are included in Appendix B.

AI Use Statement

Generative AI tools were used to assist with language editing and polishing of the manuscript and with the implementation of selected code modules. All AI-assisted text and code were reviewed, revised where necessary, and manually verified by the authors. The experimental design, methodological decisions, analysis, and scientific conclusions were determined by the authors, who take full responsibility for the final content of this work.

References

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Yukang Chen, Fuzhao Xue, Dacheng Li, Qinghao Hu, Ligeng Zhu, Xiuyu Li, Yunhao Fang, Haotian Tang, Shang Yang, Zhijian Liu, et al. Longvila: Scaling long-context visual language models for long videos. In *International Conference on Learning Representations*, volume 2025, pp. 18227–18246, 2025.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24108–24118. IEEE, 2025.

Yossi Gandelsman, Yu Sun, Xinlei Chen, and Alexei Efros. Test-time training with masked autoencoders. *Advances in Neural Information Processing Systems*, 35:29374–29385, 2022.

Yuchen Guan, Xiao Li, Zongyu Guo, Xiaoyi Zhang, Xiulian Peng, Chun Yuan, and Yan Lu. From content to knowledge: Lightning fast long-video understanding with neural knowledge representations. *arXiv preprint arXiv:2606.11913*, 2026.

Bo He, Hengduo Li, Young Kyun Jang, Menglin Jia, Xuefei Cao, Ashish Shah, Abhinav Shrivastava, and Ser-Nam Lim. Ma-lmm: Memory-augmented large multimodal model for long-term video understanding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13504–13514. IEEE, 2024.

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.

Peng Jin, Ryuichi Takanobu, Wancai Zhang, Xiaochun Cao, and Li Yuan. Chat-univi: Unified visual representation empowers large language models with image and video understanding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13700–13710. IEEE, 2024.

Jie Lei, Licheng Yu, Mohit Bansal, and Tamara Berg. Tvqa: Localized, compositional video question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 1369–1379, 2018.

Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pp. 5971–5984, 2024.

Yuejiang Liu, Parth Kothari, Bastien Van Delft, Baptiste Bellot-Gurlet, Taylor Mordan, and Alexandre Alahi. Ttt++: When does self-supervised test-time training fail or thrive? *Advances in Neural Information Processing Systems*, 34:21808–21820, 2021.

Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12585–12602, 2024.

Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. *Advances in Neural Information Processing Systems*, 36:46212–46244, 2023.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *International conference on machine learning*, pp. 16888–16905. PMLR, 2022.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. *arXiv preprint arXiv:2302.12400*, 2023.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. Timechat: A time-sensitive multimodal large language model for long video understanding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14313–14323. IEEE, 2024.

Xubin Ren, Lingrui Xu, Long Xia, Shuaiqiang Wang, Dawei Yin, and Chao Huang. Videorag: Retrieval-augmented generation with extreme long-context videos. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, pp. 2390–2401, 2026.

Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, et al. Moviechat: From dense token to sparse memory for long video understanding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18221–18232. IEEE, 2024.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning*, pp. 9229–9248. PMLR, 2020.

Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. Movieqa: Understanding stories in movies through question-answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4631–4640, 2016.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020.

Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7191–7201. IEEE, 2022.

Weihan Wang, Zehai He, Wenyi Hong, Yean Cheng, Xiaohan Zhang, Ji Qi, Ming Ding, Xiaotao Gu, Shiyu Huang, Bin Xu, et al. Lvbench: An extreme long video understanding benchmark. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22958–22967. IEEE, 2025a.

Xiao Wang, Qingyi Si, Jianlong Wu, Shiyu Zhu, Li Cao, and Liqiang Nie. Retake: Reducing temporal and knowledge redundancy for long video understanding. *arXiv preprint arXiv:2412.20504*, 2024.

Xiao Wang, Qingyi Si, Shiyu Zhu, Jianlong Wu, Li Cao, and Liqiang Nie. Adaretake: Adaptive redundancy reduction to perceive longer for video-language understanding. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 5417–5432, 2025b.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. Longvideobench: A benchmark for long-context interleaved video-language understanding. *Advances in Neural Information Processing Systems*, 37:28828–28857, 2024.

Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9772–9781. IEEE, 2021.

Zeyuan Yang, Delin Chen, Xueyang Yu, Maohao Shen, and Chuang Gan. Vca: Video curious agent for long video understanding. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20168–20179. IEEE, 2025.

Boqiang Zhang, Kehan Li, Zesen Cheng, Zhiqiang Hu, Yuqian Yuan, Guanzheng Chen, Sicong Leng, Yuming Jiang, Hang Zhang, Xin Li, et al. Videollama 3: Frontier multimodal foundation models for image and video understanding. *arXiv preprint arXiv:2501.13106*, 2025.

Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. In *Proceedings of the 2023 conference on empirical methods in natural language processing: system demonstrations*, pp. 543–553, 2023.

Marvin Zhang, Sergey Levine, and Chelsea Finn. Memo: Test time robustness via adaptation and augmentation. *Advances in neural information processing systems*, 35:38629–38642, 2022.

Peiyuan Zhang, Kaichen Zhang, Bo Li, Guangtao Zeng, Jingkang Yang, Yuanhan Zhang, Ziyue Wang, Haoran Tan, Chunyuan Li, and Ziwei Liu. Long context transfer from language to vision. *arXiv preprint arXiv:2406.16852*, 2024.

Xiaoyi Zhang, Zhaoyang Jia, Zongyu Guo, Jiahao Li, Bin Li, Houqiang Li, and Yan Lu. Deep video discovery: Agentic search with tool use for long-form video understanding. *Advances in Neural Information Processing Systems*, 38:89863–89895, 2026.

A Training Data Construction and Examples

This section provides additional implementation details for the multi-scale supervision introduced in Section 3.2, together with representative training examples. To balance temporal coverage and redundancy, the 5-, 10-, and 20-second windows are non-overlapping, while windows of 40 seconds or longer use 50% overlap with a stride equal to half the window length. For windows up to 40 seconds, overlapping content from the low-, mid-, and high-level descriptions is fused into a single coherent, redundancy-reduced training example; for windows of 80 seconds or longer, the three descriptions are retained as separate examples. Table 3 summarizes the resulting configuration. Figure 5 illustrates the corresponding training formats using a 5-second fused-description example, an 80-second separate-description example, and representative local and global QA pairs. Ellipses indicate omitted content for compact presentation.

Table 3: Per-video multi-scale training-data construction. "Merged" indicates that overlapping information from the three description levels is fused into one coherent, redundancy-reduced training example.

Window	Stride	Overlap	Descriptions	Local QA
5 s	5 s	0%	Merged	1
10 s	10 s	0%	Merged	1
20 s	20 s	0%	Merged	3
40 s	20 s	50%	Merged	5
80 s	40 s	50%	Separate	5
$\geq$160 s	Half-window	50%	Separate	10

Short window (5 seconds): fused description example
Time range: 00:00:00–00:00:05
Fused example: White serif text on a dark background explains that unique wooden masks are carved to honor the dead and placed in sacred shrines, where they gradually return to dust and symbolize the spirit's peaceful journey into the next life. Subtle blue light orbs drift in the background ...

Long window (80 seconds): separate description examples
Time range: 00:00:00–00:01:20
[High] A young girl and two small green-haired children interact in a mystical forest ...
[Mid] The video opens with text about wooden masks and spirit guides; the girl then observes the children nearby ...
[Low] White text appears on screen; the girl has dark hair and wears a blue top, while the children have green hair with colorful ornaments ...

Local QA
Q: From 00:00:00 to 00:02:40, what does the girl do while interacting with the creatures?
A: She moves through the forest and interacts closely with the two creatures ...

Global QA
Q: What does the text in the video say about the purpose of the wooden masks?
A: They are carved to honor the dead and symbolize the spirit's peaceful journey ...

Figure 5: Representative training examples for the multi-scale supervision pipeline. For short windows ($\leq$ 40 s), overlapping information from the high-, mid-, and low-level descriptions is fused into one coherent, redundancy-reduced training example, whereas for long windows ($\geq$ 80 s), the descriptions are retained as three separate examples. Local QA pairs include explicit temporal grounding, while global QA pairs are timestamp-free. Ellipses denote omitted content.

Multi-level clip-description prompt

Instructions. You are a professional video annotator. Based on the input video clip, generate three levels of natural, fluent, and descriptive captions in English.

Output format. Return a JSON object with the fields `caption_high_level`, `caption_mid_level`, and `caption_low_level`.

Guidelines.

- Capture temporal dynamics, including movements, actions, and transitions.
- Describe scene evolution factually.
- Translate factual entities into English and ignore irrelevant expressions.

Caption levels.

- **Level 1:** a short summary of the main subject.
- **Level 2:** a richer description of entities, actions, and context.
- **Level 3:** a comprehensive narrative detailing attributes, spatial relations, and temporal evolution.

Figure 6: Prompt for multi-level clip descriptions, adapted from NKR.

Stage 1: clip-level entity extraction

Instructions. You are a professional video entity extractor. Analyze the input video clip and detect all entities. For each entity, provide its name, appearance, identity, and first-seen timestamp within this clip. Avoid duplicates and return a JSON object.

Stage 2: video-level entity merging

Instructions. You are a professional assistant for merging entities. Given entity registries from different clips of the same video, merge duplicate entities, retain the earliest first-seen time, and create a `timestamps` field containing all occurrence times.

Figure 7: NKR-style prompts for clip-level entity extraction and video-level merging.

B Data Generation Prompts

This appendix documents the prompts used for data generation in MIT^3. We provide the newly introduced prompts in full, while components inherited from or adapted from NKR (Guan et al., 2026) are summarized at the level of their structure and modifications.

B.1 NKR-Based Components

Our data-generation pipeline retains several components from NKR (Guan et al., 2026). Specifically, we preserve its three-level captioning format within a single model call and its two-stage entity-processing procedure. For local clip-level QA generation, we follow the same caption-and-entity evidence organization, but adapt the task from multiple-choice to open-ended QA and require each question to be temporally grounded.

B.2 MIT^3-Specific Prompts

MIT^3 introduces two additional prompts for constructing global video knowledge and generating full-video open-ended QA. We provide their complete templates below.

Local open-ended QA prompt

Instructions. You are a careful dataset constructor that converts one video clip into high-quality open-ended QA pairs. Given multi-level captions and entities for a video clip, produce diverse QA pairs covering multiple scales and aspects.

Question composition.

- Cover high-level gist, mid-level attributes and relations, and low-level fine-grained details.
- Questions may concern a single moment, an action sequence, or multiple events.

Timestamp rules.

- Each timestamp must lie within the clip's time range.
- Use floating-point seconds and prefer ranges for sustained actions.
- Include the timestamp or temporal range directly in the question.

Workflow.

1. Parse the captions into atomic facts.
2. Fuse them into a contradiction-free fact set.
3. Draft questions spanning content and temporal granularities.
4. Enforce the timestamp rules.
5. Retain only unique, answerable questions.
6. Return JSON only.

Figure 8: Prompt for temporally grounded, open-ended clip-level QA.

Global memory construction prompt

Create a comprehensive global memory document for the entire video from selected clip descriptions, sorted by time, and a merged entity registry.

Required sections.

1. **Global entity registry:** disambiguated entities with unique names, appearances, roles, and first/last appearance times.
2. **Event timeline:** chronological major events with timestamps.
3. **State changes and causal relationships:** major transitions and their causes.
4. **Structural overview:** major phases or chapters of the video.

Figure 9: MIT^3 prompt for global memory construction.

C Additional Implementation Details

Datasets and Baselines LVBench (Wang et al., 2025a) contains 103 long videos and 1,549 question–answer pairs covering entity recognition, event understanding, key-information retrieval, temporal grounding, reasoning, and summarization. We report results on both the full benchmark and the subset containing videos longer than one hour. For LongVideoBench (Wu et al., 2024), we evaluate on the full benchmark and the LongVideoBench-Long subset adopted by NKR (Guan et al., 2026), which contains 564 questions over 188 videos of 900–3,600 seconds. Subtitle and audio inputs are excluded from both benchmarks. We additionally evaluate on Video-MME (Fu et al., 2025), which covers short-, medium-, and long-duration videos across diverse perception and reasoning tasks.

Published baselines include VideoLLaMA3 (Zhang et al., 2025), AdaReTaKe (Wang et al., 2025b), VideoRAG (Ren et al., 2026), VCA (Yang et al., 2025), Deep Video Discovery (DVD-o3) (Zhang et al., 2026), and NKR (Guan et al., 2026). For direct Qwen3-VL inference, frames are sampled

Full-video QA prompt

Generate exactly {N} distinct open-ended QA pairs about the entire video.

Critical disambiguation rules.

1. Each question must have a unique, unambiguous answer across the entire video.
2. A question is invalid if multiple scenes or moments could answer it.
3. Questions must not contain timestamps or time references, although answers may mention times.
4. Answers must be definitive rather than speculative.
5. Answers must contain concise factual content (30–100 tokens).
6. Do not generate multiple-choice questions, refusals, or answers requiring knowledge beyond the provided memory.

Figure 10: MIT^3 prompt for open-ended full-video QA.

at 2 fps and capped at 768 frames. Since the per-video training data of NKR are unavailable, we follow its data-generation procedure and train the controlled NKR-style baseline and MIT^3 using exactly the same generated supervision. The NKR-style baseline optimizes LoRA parameters with only $\mathcal{L}_{\mathrm{NTP}} + \mathcal{L}_{\mathrm{SFT}}$, allowing the effects of causal layer selection and visual memory distillation to be evaluated independently of differences in synthetic supervision.

Optimization Details MIT^3 is implemented in PyTorch (Paszke et al., 2019). Following the video-NKR configuration, we use LoRA with rank 256 and scaling factor 512 and optimize the video-specific parameters with AdamW. Unless otherwise specified, each adapter is trained for 5 epochs with a per-device batch size of 4, an initial learning rate of 2×10^{-4} with cosine annealing, and visual-loss weight $\alpha = 1$.

For causal localization, we sample at most 20 calibration QA pairs from each temporal scale. For each candidate layer, the intervention gain $G_{i,l}(\lambda)$ is evaluated on

$$\Lambda = \{0.25, 0.5, 0.75, 1.0\}, \tag{16}$$

and approximate the equivalent weighted integral in Equation 7 by averaging over these four sampled intervention strengths:

$$R_{i,l} \approx \frac{1}{|\Lambda|} \sum_{\lambda \in \Lambda} (1 - \lambda) G_{i,l}(\lambda). \tag{17}$$

Thus, b serves only as an auxiliary variable in the continuous formulation and is not separately discretized during implementation. For visual memory distillation, we randomly sample 50% of the temporally grounded QA pairs at each temporal scale and precompute their teacher deltas before adapter optimization. Sensitivity analyses of the top-K layer budget and visual-supervision sampling ratio are provided in Appendix D.

Training Data For a one-hour video, the data-generation pipeline produces approximately 2k multi-scale description examples, 3k temporally grounded local QA pairs, and 500 timestamp-free global QA pairs, yielding approximately 5.5k training examples in total. This corresponds to roughly one-eighth of the training samples used by NKR. To prevent evaluation leakage, we compare the generated supervision against the benchmark evaluation sets and verify that no evaluation questions are included in either adapter training or causal-localization calibration.

We additionally report the per-video construction time measured on four NVIDIA A100 GPUs. For Qwen3-VL-8B, multi-scale data generation, causal localization, visual-supervision generation, and adapter optimization take approximately 30, 20, 10, and 10 minutes, respectively, for approximately 70 minutes in total. For Qwen3-VL-32B, the corresponding stages take approximately 60, 30, 15, and 40 minutes, for approximately 145 minutes in total. Visual-supervision generation includes computing and caching the video-induced teacher deltas used during optimization.

Table 4: Effect of the selected-layer budget K for Qwen3-VL-8B on LVBench.

K	1	3	6	12	18	24
Accuracy (%)	26.1	24.9	35.6	39.6	**43.5**	41.7

Table 5: Effect of the selected-layer budget K for Qwen3-VL-32B on LVBench.

K	12	16	24	32	48
Accuracy (%)	37.9	42.1	46.7	**50.6**	48.1

D Hyperparameter Analysis

We analyze the sensitivity of MIT^3 to two key hyperparameters: the number of selected layers K and the sampling ratio used for visual memory distillation. All experiments are evaluated on LVBench, varying one factor at a time while keeping the remaining configuration fixed.

Number of Selected Layers. Tables 4 and 5 report the effect of the layer budget K for Qwen3-VL-8B and Qwen3-VL-32B, respectively. For both backbones, performance generally improves as more causally relevant layers are included, but begins to decline once the selected set becomes overly broad. Qwen3-VL-8B achieves its best performance at $K = 18$, while Qwen3-VL-32B peaks at $K = 32$. This behavior suggests that effective memory internalization benefits from concentrating adaptation on a sufficiently expressive yet selective subset of layers: too few layers provide insufficient capacity for storing video-specific knowledge, whereas expanding the adaptation set beyond the causally salient region may dilute this specialization. We therefore use $K = 18$ and $K = 32$ for the 8B and 32B models, respectively, throughout the main experiments.

Visual-Supervision Sampling Ratio. Table 6 studies the fraction of temporally grounded QA pairs used for visual memory distillation with Qwen3-VL-8B. Removing visual supervision reduces accuracy from 43.5 to 38.2, confirming its importance for effective memory internalization. Increasing the sampling ratio from 50% to 100%, however, yields only a marginal improvement of 0.2 points. This saturation indicates that visual memory distillation does not require dense supervision over all local QA pairs; a moderate subset already provides sufficient visual guidance. We therefore use a 50% sampling ratio in the main experiments to retain nearly the full benefit while reducing the cost of teacher-delta generation.

E Layer-Wise Distribution with Qwen3-VL-32B

We repeat the layer-distribution analysis with Qwen3-VL-32B, selecting the top-32 adaptation layers for each LVBench video. As in the 8B setting, Figure 11 visualizes the normalized per-video causal utilities together with the cross-video selection frequency. A similar structure emerges at the larger model scale: high-utility layers are concentrated within a shared region across videos, while a smaller subset of selected layers varies with individual video content. The frequently selected layers account for approximately **72%** of all top-32 selections, indicating substantial cross-video consistency in the preferred adaptation locations. At the same time, the remaining selections preserve video-specific variation around this shared structure. Overall, the 32B analysis exhibits the same combination of a common memory-writing core and content-dependent refinements observed with Qwen3-VL-8B.

Table 6: Effect of the visual-distillation sampling ratio on LVBench with Qwen3-VL-8B.

Visual QA Sampling Ratio	Accuracy (%)
0%	38.2
50%	43.5
100%	**43.7**

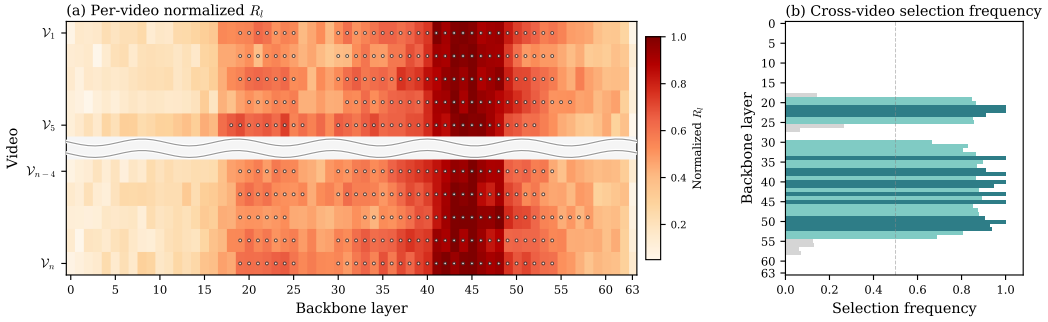

Figure 11: Layer-wise causal utility and top-32 selection frequency for Qwen3-VL-32B on LVBench.

F Deployment and Resource Analysis

MIT^3 offers practical advantages in settings where long videos serve as persistent information sources and are revisited repeatedly. We compare its serving footprint with persistent KV-cache reuse using Qwen3-VL-32B under the same one-hour, 768-frame setting as Table 1. KV reuse avoids repeated video prefill by retaining the video-conditioned KV cache, whereas MIT^3 represents each video with a compact LoRA adapter and performs subsequent inference from text alone.

Table 7: Persistent-state comparison for Qwen3-VL-32B.

Setting	KV Reuse	MIT^3
Per-video persistent state	26.9 GB	0.59 GB
Backbone + 1 video	89.0 GB	62.7 GB
Backbone + 10 videos	331.1 GB	68.0 GB
Reduction vs. KV reuse	–	45.6× smaller

The resource advantage becomes increasingly pronounced as more long videos are retained for subsequent access. Under persistent KV reuse, each additional video introduces a 26.9-GB cache, whereas MIT^3 adds only a 0.59-GB adapter. As a result, maintaining ten one-hour videos requires approximately 331 GB including the backbone under KV reuse, compared with only 68 GB for MIT^3. This compact per-video state substantially reduces the memory required to maintain and switch among multiple persistent video memories.

Such a deployment pattern is particularly relevant to repeated analysis of long-video collections, including retrospective surveillance review, recorded meetings, lecture archives, and instructional or maintenance videos. In these settings, multiple video memories can share the same backbone and be stored or switched through lightweight adapters, without keeping the original frames or a large video-dependent KV cache resident for every video.

G Additional Benchmark Results

G.1 Video-MME

Video-MME evaluates general video understanding across short-, medium-, and long-duration videos. Table 8 reports overall accuracy with the Qwen3-VL-8B backbone, providing an additional evaluation beyond the dedicated long-video benchmarks. MIT^3 consistently improves over the matched NKR-style baseline, while remaining competitive with direct video-conditioned inference despite operating without video input at query time.

Table 8: Video-MME results. † denotes methods trained with the same data as MIT^3.

Method	Input	Accuracy (%)
Qwen3-VL-8B (Bai et al., 2025)	< 768 fr.	69.1
Qwen3-VL-8B + NKR-style†	N/A	62.9
Qwen3-VL-8B + MIT^3 (Ours)	N/A	**66.7**

H Limitations

Our study focuses on test-time training, where each video is internalized through lightweight, video-specific adaptation of a pretrained model. This setting allows us to isolate and evaluate the proposed memory-writing mechanisms, but computational constraints limit our exploration of how these mechanisms scale to large-scale pretraining. Extending them to the pretraining stage may support more persistent and reusable video knowledge across videos and tasks, which we leave for future work.