COMPRESSED SIGHT, SELECTIVE INSIGHT: LEARNING WHERE TO SPEND VISUAL TOKENS IN LONG VIDEOS

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-video understanding is constrained by a visual-budget problem: a video may span thousands of frames, but only a small fraction can be given to a language model at high spatial resolution. Existing frame-selection methods retain a few frames and discard the rest, while uniform compression preserves the timeline but allocates the same low resolution to every frame. We introduce *Compressed Sight, Selective Insight* (CSSI), an efficient video language model that keeps every sampled frame in the input: a small set of selected frames receives a grid of spatial detail tokens, and every other frame is represented by one average-pooled token. Which frames receive detail is decided by a query-conditioned selection policy that reads all frames in one linear-time pass and then selects frames one at a time, each choice conditioned on the frames already chosen, so the budget is spent on complementary evidence rather than repeated views of the same scene. The policy is trained with reinforcement learning from a reward computed by the model itself, the margin by which the model separates the correct answer from the strongest incorrect option when reading the resulting input, so training needs no external reward model and no frame-level labels. Across Video-MME, LongVideoBench, MLVU, EgoSchema, and LVBench, CSSI outperforms recent efficiency-oriented video LLMs, with its largest gains on long videos and on object-reasoning questions. CSSI is also the most memory and latency-efficient method at large frame counts, scaling to 16K-frame videos where competing methods run out of memory.

1 INTRODUCTION

Long videos typically contain more visual information than a language model can process at high spatial resolution. Sampled at multiple frames per second, a video spans hundreds to thousands of frames, and each frame yields hundreds of patch embeddings. Passing all of them to the model makes the input length grow rapidly with duration. This inflates prefill cost, key-value-cache memory, and per-step attention cost, so even strong video-language models struggle to hold a full movie, a long egocentric recording, or a multi-hour compilation in one context (Lin et al., 2024; Shu et al., 2024; Bai et al., 2023; Wang et al., 2024a; Bai et al., 2025b;a).

A common approach is to compress the visual representation before it is passed to the language model. Temporal subsampling keeps a sparse set of frames but can miss a brief event. Frame selection keeps a few relevant frames but discards their surrounding temporal context (Tang et al., 2025; Zhu et al., 2025; Yao et al., 2025). A mixed-resolution representation can be more robust: the model sees the whole timeline in coarse form and spends fine spatial detail on a small subset of highly-informative frames, but in turn requires a mechanism for identifying that subset.

Our method *Compressed Sight, Selective Insight* (CSSI) casts this as a learned allocation problem with two kinds of visual tokens: a compressed view of the whole timeline and selective spatial detail on a few frames. An unselected frame is represented by a single *timeline token*, one pooled embedding that preserves the frame's place in the global sequence but not its internal layout. A selected frame is represented by a grid of *detail tokens*, where each token covers one cell of a coarse spatial grid over the frame, so the grid as a whole retains the frame's spatial layout. Every frame receives one representation or the other, so no frame is removed: selection allocates more tokens to a frame's representation rather than deciding whether the frame is retained.

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.

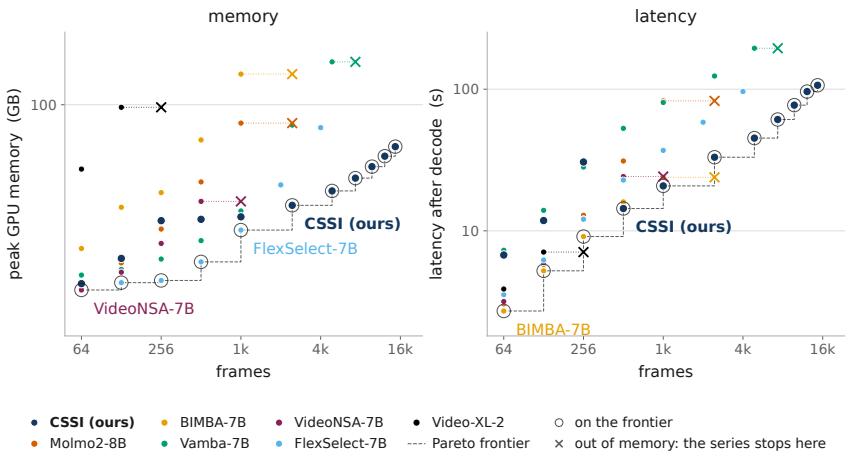


Figure 1: **Efficiency at scale.** Peak GPU memory (left) and end-to-end latency after frame decoding (right) as a function of the number of input frames. Both axes are logarithmic. Circles indicate configurations on the Pareto frontier, and a cross marks the final configuration before out-of-memory failure. CSSI remains on both frontiers beyond roughly one thousand frames and is the only compared configuration that can accommodate 16K frames long videos.

Matching frames to the query text alone is an unreliable guide to which frames will most help the model: a similarity score can retrieve a visually related frame yet miss the evidence that separates the correct answer from a plausible distractor. We therefore score a *set* of selected frames by its effect on the model's answer. The selection policy reads all frame features and the query in one linear-time pass and promotes frames one at a time. For a multiple-choice question, we run the model on each sampled set and use the margin between the correct option and the strongest incorrect option as the reward, comparing rewards only within the same video-question pair.

The system is trained in two stages. Stage I trains the projector and model to read mixed-resolution inputs produced by a fixed, parameter-free selection rule. Stage II then freezes the model and trains only the selection policy, using the answer-margin reward of complete sampled trajectories. The visual encoder remains frozen throughout and is run once per video, and the policy adds a single linear-time scan over the encoder's frame features.

We evaluate the main claim with a controlled comparison: for each question we fix the model, the sampled frame list, the number of selected frames, and the token budget, and vary only the selection rule. Under this protocol, CSSI improves over its heuristic-selection counterpart across long-video benchmarks, with the largest gains on the Video-MME long split and on object-reasoning questions.

Our contributions are threefold. First, we frame long-video visual-budget allocation as *selection* within a mixed-resolution representation that retains every sampled frame. Second, we introduce a question-conditioned selection policy that promotes frames sequentially, trained by an answer-margin reward the model computes for multiple-choice questions, without an external reward model or frame-level labels. Third, we evaluate CSSI on long-video benchmarks and analyze its behavior through paired comparisons at a fixed visual-token budget, isolating the effect of the selection rule, together with measurements of memory and latency at scale (Figure 1). (Figure 1).

2 Related Work

Efficient long-video language models. Long-video systems reduce decoder cost through temporal sampling, visual-token compression, cache compression, or specialized temporal modules (Song et al., 2026; Yan et al., 2025; Jiang et al., 2025; Islam et al., 2025; Ren et al., 2025b). Some systems use a small number of high-resolution frames together with pooled representations of the remaining video, keeping one coarse token for each of the remaining frames. CSSI adopts this representation and focuses on the allocation rule: it learns which frames receive the scarce high-resolution budget, while every unselected frame remains visible as one timeline token.

Frame selection and visual-budget allocation. Recent work selects frames using temporal coverage, visual diversity, motion, or question-frame similarity (Tang et al., 2025; Zhu et al., 2025; Yao et al., 2025). These approaches show that uniform sampling is not always the best allocation rule. Such proxy scores, however, need not directly reflect how useful a frame is to the downstream model, and pure selection systems typically remove frames that fall outside the selected set. CSSI uses question information to make a sequential selection decision, evaluates a selection set by the model's own answer margin, and retains all unselected frames at coarse resolution.

Sequential selection and answer-margin rewards. Selective state-space models provide linear-time sequence mixing and have been used for long visual and multimodal sequences (Gu & Dao, 2024; Dao & Gu, 2024; Ren et al., 2025a; Jiang et al., 2025). Their recurrent state can carry information about earlier choices when a policy selects items sequentially, and CSSI uses this property to choose a fixed number of high-resolution frames without replacement. Unlike methods that rank frames with an external scoring network or per-frame labels, our policy is supervised only by the reward the model assigns to a complete selected set. Our setting combines a fixed visual budget, sequential selection, and a model that retains a pooled token for every unselected frame.

3 METHOD

We describe the method in execution order for a single video-question pair, summarized in Figure 2. We first define the shared visual representation and the two token types, then the model trained in Stage I to read the mixed-resolution input, the selection policy introduced in Stage II, and the answer-margin reward that trains the policy.

3.1 VIDEO REPRESENTATION AND NOTATION

Let $V = (f_1, \dots, f_T)$ be a video of T sampled frames and let Q denote the question text. A frozen visual encoder $\mathcal{E}_\xi$ maps each frame to a grid of N_p visual patch embeddings,

$$\mathbf{V}_t = \mathcal{E}_\xi(f_t) \in \mathbb{R}^{N_p \times d_v}, \qquad t \in \{1, \dots, T\}. \tag{1}$$

A trainable multimodal projector $\mathcal{P}_\phi$ maps these embeddings into the language-model width d,

$$\mathbf{H}_t = \mathcal{P}_\phi(\mathbf{V}_t) \in \mathbb{R}^{N_p \times d}. \tag{2}$$

The question tokenizer maps Q to prompt embeddings $\mathbf{E}_Q \in \mathbb{R}^{L_Q \times d}$.

We use four terms consistently. A *patch embedding* is a row of $\mathbf{V}_t$ or $\mathbf{H}_t$ before spatial pooling. A *selected frame* is a frame assigned the high-resolution budget. A selected frame contributes m *detail tokens*, which retain a spatial grid. An *unselected frame* contributes one *timeline token*, which preserves its place in the video but not its internal spatial layout.

Let $\mathcal{P} \subseteq \{1, \dots, T\}$ be the set of selected-frame indices, with $|\mathcal{P}| = K$.

3.2 MIXED-RESOLUTION REPRESENTATION

Constructing the mixed-resolution timeline. For a selected frame $t \in \mathcal{P}$, we spatially pool the projected patch grid,

$$\mathbf{D}_t = \mathrm{Pool}(\mathbf{H}_t) \in \mathbb{R}^{m \times d}. \tag{3}$$

The m rows of $\mathbf{D}_t$ are the detail tokens. They preserve the frame's coarse spatial arrangement and are intended to supply finer spatial detail. For an unselected frame $t \notin \mathcal{P}$, we form one timeline token by average pooling,

$$\mathbf{c}_t = \frac{1}{N_p} \sum_{i=1}^{N_p} \mathbf{H}_{t,i} \in \mathbb{R}^d. \tag{4}$$

This operation adds no trainable parameters.

The model receives the temporally interleaved sequence

$$\mathbf{Z}_t = \begin{cases} \mathbf{D}_t \in \mathbb{R}^{m \times d}, & t \in \mathcal{P}, \\ \mathbf{c}_t \in \mathbb{R}^{1 \times d}, & t \notin \mathcal{P}. \end{cases}, \qquad \widetilde{\mathbf{S}}(\mathcal{P}) = [\mathbf{Z}_1; \dots; \mathbf{Z}_T; \mathbf{E}_Q] \tag{5}$$

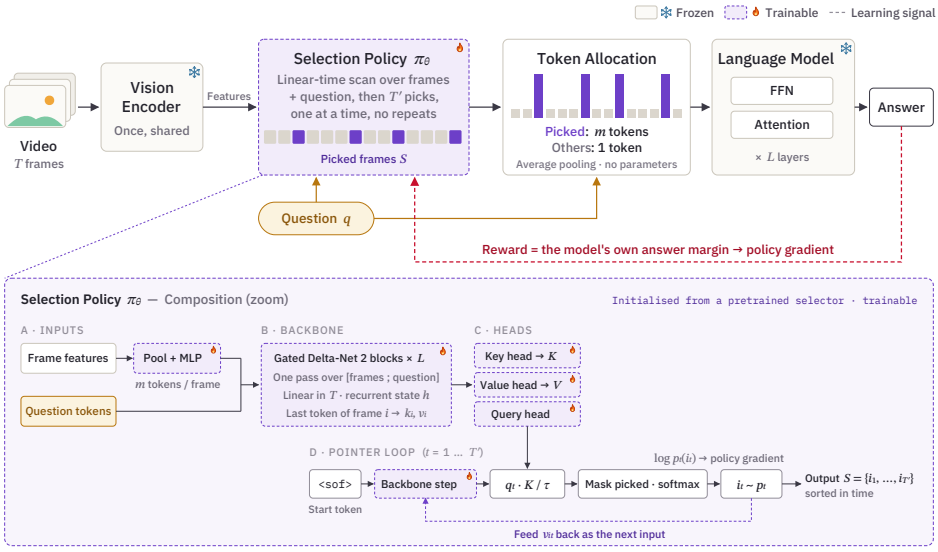


Figure 2: **CSSI overview.** A frozen vision encoder produces frame features once and shares them between the selection policy and the model. The policy reads all frame features together with the question in one linear-time scan, then samples K frame indices sequentially without replacement. A selected frame contributes a grid of spatial detail tokens. Every other frame contributes one average-pooled timeline token. The model receives the temporally interleaved mixed-resolution sequence, and its correct-option margin provides the reward that trains the selection policy. The language model itself is frozen in Stage II.

Every sampled moment therefore has a representation in the input, and every selected moment receives additional spatial detail. The visual length is $Km + (T - K)$ rather than TN_p, a large reduction for long videos.

Stage I: learning to read the representation. Before learning where to promote frames, we train the projector and model to interpret the mixed-resolution sequence, with the visual encoder frozen. Because a learned selection policy is not yet available, we initialize $\mathcal{P}$ with a parameter-free CLIP similarity rule. A frozen CLIP vision encoder produces a global frame embedding $\mathbf{x}_t$, and its text encoder produces an embedding $\mathbf{q}$ for the question. The initialization score is

$$a_t = \frac{\mathbf{x}_t^\top \mathbf{q}}{\|\mathbf{x}_t\| \, \|\mathbf{q}\|}. \tag{6}$$

We admit frames in decreasing score order subject to a diversity threshold on pairwise CLIP cosine similarity, and space any remaining slots evenly. This bootstrap rule exposes the model to a variety of mixed-resolution inputs during Stage I and is not used in Stage II.

Let $p_\omega(y_j \mid y_{<j}, \widetilde{\mathbf{S}}(\mathcal{P}))$ denote the next-token distribution of the language model with parameters ω (which include the multimodal projector), conditioned on the mixed-resolution sequence and the answer tokens generated so far. For an answer token sequence $Y = (y_1, \ldots, y_M)$, Stage I minimizes the representation-learning loss

$$\mathcal{L}_{\text{rep}} = -\sum_{j=1}^{M} \log p_\omega\big(y_j \mid y_{<j}, \widetilde{\mathbf{S}}(\mathcal{P})\big). \tag{7}$$

3.3 Question-conditioned sequential selection

The Stage II selection policy replaces the CLIP initializer with a policy tailored to the model. We design it to have three properties: (i) inspecting every frame, so a brief event can still be eligible for selection, (ii) be conditioned on the question, and (iii) accounting for frames already chosen, so the budget is not spent repeatedly on the same scene.

Shared features and a linear-time scan. The selection policy reuses the frozen visual features $\mathbf{V}_t$ computed for the model. A small policy projector maps the patch embeddings of each frame to a fixed number of policy tokens. We feed the question embeddings first, followed by the per-frame tokens in temporal order, into a Gated DeltaNet-2 backbone (Hatamizadeh et al., 2026). Because the scan is causal, placing the question first lets every frame state depend on it. Let $\mathbf{g}_t$ denote the final state associated with frame t. One pass produces all T frame states in time linear time complexity. Two heads form a frame key and a frame value,

$$\mathbf{k}_t = W_K \mathbf{g}_t, \qquad \mathbf{v}_t = W_V \mathbf{g}_t. \tag{8}$$

The key enters the dot-product selection score below, and the value represents the selected frame and is fed back into the selector state after each selection.

selection without replacement. The policy chooses frames sequentially, maintaining a fast-weight memory $\mathbf{S}_\ell \in \mathbb{R}^{d\times d}$ of the frames already selected, initialized to $\mathbf{S}_1 = \mathbf{0}$. At selection step ℓ, a learned probe $\mathbf{q}$ reads the memory into a selection query vector $\mathbf{u}_\ell = \mathbf{S}_\ell \mathbf{q}$, and the distribution over unselected frame indices is

$$p_\theta(i_\ell = t \mid \mathcal{P}_{<\ell}, V, Q) = \mathrm{softmax}_{t\notin\mathcal{P}_{<\ell}}\left(\frac{\mathbf{u}_\ell^\top \mathbf{k}_t}{\tau_\pi}\right), \qquad i_\ell \sim p_\theta. \tag{9}$$

Once frame i_ℓ is drawn, its key-value pair is written into the memory with the Gated DeltaNet-2 gated delta rule (Hatamizadeh et al., 2026),

$$\mathbf{S}_{\ell+1} = \left(\mathbf{I} - \mathbf{k}_{i_\ell}(\mathbf{b}_\ell \odot \mathbf{k}_{i_\ell})^\top\right) \mathrm{Diag}(\boldsymbol{\alpha}_\ell)\, \mathbf{S}_\ell + \mathbf{k}_{i_\ell}(\mathbf{w}_\ell \odot \mathbf{v}_{i_\ell})^\top, \tag{10}$$

where $\mathbf{b}_\ell, \mathbf{w}_\ell \in [0,1]^d$ are channel-wise erase and write gates and $\mathrm{Diag}(\boldsymbol{\alpha}_\ell)$ is a channel-wise decay, all predicted from $\mathbf{g}_{i_\ell}$. Masking the already-selected indices enforces selection without replacement, and writing each selected frame into $\mathbf{S}_\ell$ makes each decision conditional on the earlier selections. The policy therefore produces an ordered, budget-aware selection set, rather than ranking frames independently by question-frame similarity. After K steps, the selected indices are sorted in temporal order and substituted for $\mathcal{P}$ in Equation (5). At inference, the policy requires one linear-time scan and no CLIP scorer or second vision encoder.

3.4 Answer-margin reward and policy optimization

The policy does not score a frame in isolation. Its action is a trajectory $a = (i_1, \ldots, i_K)$, where each i_ℓ is a newly selected-frame index. We evaluate the reward only after all K decisions are made, by constructing the mixed-resolution input and measuring how the selected detail affects the answer.

Sampling a selection trajectory. For a training example (V, Q, y^*), we draw N trajectories from the policy in Equation (9). The jth trajectory is $a_j = (i_{j,1}, \ldots, i_{j,K})$, with selected-frame set $\mathcal{P}_j = \{i_{j,1}, \ldots, i_{j,K}\}$. Its probability is the product of the conditional decisions:

$$\log p_\theta(a_j \mid V, Q) = \sum_{\ell=1}^{K} \log p_\theta(i_{j,\ell} \mid i_{j,<\ell}, V, Q). \tag{11}$$

The indices are sorted only when assembling $\widetilde{\mathbf{S}}(\mathcal{P}_j)$ for the model. Equation (11) retains the trajectory order because it records the decisions that produced the set.

Answer-margin reward. The model evaluates each sampled input $\widetilde{\mathbf{S}}(\mathcal{P}_j)$ and produces multiple-choice logits $\mathbf{z}_j$. We use the bounded margin between the correct option and its strongest competitor as the reward,

$$r_j = \tanh\left[\frac{z_j(y^*) - \max_{y\neq y^*} z_j(y)}{2}\right] \in [-1, 1]. \tag{12}$$

The margin provides a graded signal rather than a binary correct/incorrect one: two selection sets may both yield the correct answer while one separates it more clearly from the nearest distractor. Because the reward is evaluated after the model sees both the selected detail tokens and the pooled timeline tokens for all other frames, it scores the *allocation of detail within the complete video input*, not the usefulness of a frame once the rest of the video has been removed.

Questions differ in intrinsic difficulty, so absolute rewards are not comparable across examples. For the N trajectories from one video-question pair, we form a centered, normalized advantage,

$$\hat{r}_j = \frac{r_j - \bar{r}}{\sqrt{N^{-1}\sum_{a=1}^{N}(r_a - \bar{r})^2 + 10^{-6}}}, \qquad \bar{r} = \frac{1}{N}\sum_{a=1}^{N} r_a. \tag{13}$$

This makes the update invariant to each example's baseline reward. A positive $\hat{r}_j$ indicates only that a_j received more reward than the other trajectories for the same example. If the sampled sets receive identical rewards, their advantage terms are zero.

Policy optimization. Stage I has already trained the model. In Stage II the model is frozen and only the selection policy is trained. We update the policy with the reward-weighted trajectory loss

$$\mathcal{L}_\pi = -\frac{1}{N}\sum_{j=1}^{N} \mathrm{sg}(\hat{r}_j)\log p_\theta(a_j \mid V, Q) - \beta\,\overline{\mathcal{H}}, \qquad \overline{\mathcal{H}} = \frac{1}{NK}\sum_{j=1}^{N}\sum_{\ell=1}^{K}\mathcal{H}(p_\theta(\cdot \mid i_{j,<\ell}, V, Q)), \tag{14}$$

where sg stops gradients through the reward and $\overline{\mathcal{H}}$ is the mean policy entropy over sampled decisions. The first term raises the probability of trajectories that earn larger answer-margin rewards for the same example, and the entropy term is intended to discourage the policy from collapsing onto a small group of temporal locations before reward has been sampled from other regions of the video. Because the frozen model both reads the mixed-resolution input and provides the reward, Stage II trains the selection policy alone: no additional supervised answer-prediction loss on the model and no external reward model are needed.

4 Experiments

4.1 Protocol and implementation

All experiments use a frozen SigLIP-So400M vision encoder (Zhai et al., 2023) at 384×384 resolution, a two-layer multimodal projector, and a Qwen2-7B model (Yang et al., 2024a). Videos are sampled at one frame per second and capped at $T = 1{,}024$ frames unless the scaling experiment states otherwise. The default configuration promotes $K = 64$ frames and uses $m = 196$ detail tokens per selected frame. The selection policy is a 24-layer Gated DeltaNet-2 (Hatamizadeh et al., 2026) with hidden dimension $2{,}048$ and key, value, and selection-query heads. The vision encoder is run once per video, and its features are shared by the policy and the model.

Training proceeds in two stages. In Stage I, we train the projector and model with the parameter-free bootstrap selection rule on image-text pairs, video instruction-following data, and long-form video question answering. In Stage II, we freeze the model and train only the selection policy with $\mathcal{L}_\pi$, sampling selection trajectories on multiple-choice examples from LLaVA-Video-178K, the multiple-choice subsets of VISTA-400K (Ren et al., 2024), and CinePile. Both stages optimize with AdamW under a cosine schedule in bfloat16. Full architecture, data composition, and optimizer settings are given in Sections B to D.

We evaluate multiple-choice accuracy on Video-MME (Fu et al., 2025), LongVideoBench (Wu et al., 2024b), MLVU (Zhou et al., 2024), EgoSchema (Mangalam et al., 2023), LVBench (Wang et al., 2024c), and VideoEval-Pro (Ma et al., 2025). Unless a result specifies another protocol, comparisons within a table are paired: every condition uses the same questions, sampled frame list, model, and visual-token budget. The only difference is the selection rule specified in each comparison. Full benchmark and scoring details are in Section E.

4.2 Results

4.2.1 Long-video accuracy at a fixed visual budget

Table 1 reports long-video accuracy across seven benchmarks. CSSI improves over its heuristic-selection counterpart under a controlled comparison: the two systems use the same 7B model, the same one-frame-per-second video list, the same cap of 1,024 frames, and the same budget of 64 selected frames, and differ only in the rule that allocates detail. With the learned policy, Video-MME

Table 1: **Long-video question answering.** Shading marks access class: API-only, open-weight, and efficient long-video methods (top to bottom). The first two blocks reproduce published numbers; the efficient-methods block is run by us, with per-baseline configurations and scoring notes in Section E. **Avg** is the mean over all seven benchmark columns; for rows with a missing VideoEval-Pro cell it is the mean over the six reported columns. **Bold** marks the best result among efficient methods. ViLAMP and CSSI are strictly paired (same 7B model, same 1 fps frame list capped at 1,024 frames, same budget K=64), so their difference isolates the selection rule. Video-MME is without subtitles unless noted; "-" marks cells we do not report (Section E).

Model	LLM	Video-MME	Video-MME$_{sub}$	EgoSchema	LongVideoBench	LVBench	MLVU	VideoEval-Pro	Avg
API call only									
GPT-5 (OpenAI, 2025)	-	83.3	86.9	75.6	72.6	65.2	77.7	68.8	75.7
GPT-5 mini	-	77.3	82.3	70.9	69.7	54.7	69.1	60.1	69.2
Gemini 3 Pro (Google DeepMind, 2025)	-	88.6	87.5	68.9	75.9	77.0	75.7	78.0	78.8
Gemini 2.5 Pro (Comanici et al., 2025)	-	87.8	87.8	72.2	76.8	75.7	81.5	78.4	80.0
Gemini 2.5 Flash	-	84.2	84.2	70.2	73.1	64.9	75.1	69.6	74.5
Claude Sonnet 4.5 (Anthropic, 2025)	-	74.2	80.5	73.1	65.1	50.5	64.0	50.5	65.4
Open weights only									
InternVL3.5-4B (Wang et al., 2025a)	Qwen3-4B	65.4	68.6	58.9	60.8	43.2	52.0	46.5	56.5
InternVL3.5-8B	Qwen3-8B	66.0	68.6	58.6	62.1	43.4	53.2	48.1	57.1
Qwen3-VL-4B (Bai et al., 2025a)	Qwen3-4B	69.3	74.0	68.4	62.8	56.2	58.4	49.8	62.7
Qwen3-VL-8B	Qwen3-8B	71.4	75.2	69.8	62.4	58.0	57.6	50.3	63.5
Keye-VL-1.5-8B (Yang et al., 2025a)	Qwen3-8B	73.0	76.2	56.3	66.0	42.8	53.8	54.9	60.4
GLM-4.1V-9B (Team et al., 2025b)	GLM-4-9B	68.2	75.6	62.6	65.7	44.0	56.6	51.1	60.5
MiniCPM-V-4.5-8B (Yu et al., 2025)	Qwen3-8B	67.9	73.5	49.6	63.9	50.4	60.6	54.9	60.1
Eagle2.5-8B (Chen et al., 2025a)	Qwen2.5-7B	72.4	75.7	72.2	66.4	50.9	60.4	58.6	65.2
PLM-3B (Cho et al., 2025)	Llama-3.2-3B	54.9	59.4	66.9	57.9	40.4	48.4	46.2	53.4
PLM-8B	Llama-3.1-8B	58.3	65.4	68.8	56.9	44.5	52.6	47.2	56.2
LLaVA-Video-7B (Zhang et al., 2024b)	Qwen2-7B	63.3	69.7	57.3	58.2	44.2	52.8	47.8	56.2
VideoChat-Flash-7B (Li et al., 2026)	Qwen2-7B	65.3	69.7	51.3	64.7	48.2	56.0	51.2	58.1
Molmo2-4B (Clark et al., 2026)	Qwen3-4B	69.6	75.7	61.2	68.0	53.9	63.0	59.9	64.5
Molmo2-8B	Qwen3-8B	69.9	75.8	62.0	67.5	52.8	60.2	60.4	64.1
Molmo2-O-7B	OLMo-3-7B	64.9	69.2	56.8	63.7	49.6	55.2	55.1	59.2
Efficient methods, 7B (our runs)									
BIMBA (Islam et al., 2025)	Qwen2-7B	63.3	66.4	73.6	57.7	43.3	56.8	-	60.2
Vamba (Ren et al., 2025b)	Qwen2-VL-7B	58.1	61.1	55.6	52.8	41.2	47.4	-	52.7
VideoNSA (Song et al., 2026)	Qwen2.5-VL-7B	59.4	61.9	39.8	52.2	41.8	44.4	-	49.9
FlexSelect (Yunzhuzhang et al., 2025)	Qwen2-7B	67.4	71.4	60.2	59.8	49.2	**59.6**	54.4	60.3
Video-XL-2 (Qin et al., 2025)	Qwen2.5-7B	65.8	67.9	48.2	59.9	44.6	53.8	49.6	55.7
AdaReTaKe (Wang et al., 2025b)	Qwen2.5-VL-7B	64.9	68.4	62.6	61.1	50.9	57.0	55.2	60.0
TimeViper (Xu et al., 2025a)	Nemotron-Nano-9B	57.5	61.1	57.0	53.9	35.8	46.6	42.8	50.7
ViLAMP	Qwen2-7B	67.7	71.6	74.2	60.2	46.3	52.4	55.3	61.1
CSSI (ours)	Qwen2-7B	**71.0**	**72.9**	**74.2**	**61.7**	**52.1**	57.2	**59.6**	**64.1**

improves from 67.7 to 71.0, LongVideoBench from 60.2 to 61.7, MLVU from 52.4 to 57.2, LVBench from 46.3 to 52.1, and VideoEval-Pro from 55.3 to 59.6. Among the efficient methods we evaluate, CSSI reaches the highest or tied-highest accuracy on every benchmark except MLVU, tying ViLAMP on EgoSchema.

In our comparison the gain is largest on the longest videos. On the Video-MME long split the learned policy improves by 3.2 points (59.7 $\rightarrow$ 62.9, 900 questions), whereas on the short EgoSchema clips, where 64 selected frames already cover most of a video sampled at one frame per second, CSSI matches its heuristic-selection counterpart. This larger gain on longer videos is consistent with the hypothesis that allocation matters more when a video offers far more candidate moments than the budget can represent.

4.2.2 Comparing selection rules at a fixed budget

The central analysis holds the model fixed. Every row in Table 2 receives the same long Video-MME videos at one frame per second, promotes the same number of frames ($K = 64$), and presents the same number of visual tokens to the model. Only the selection rule changes. The comparison therefore asks a narrow question: given the same visual budget, does the learned policy choose more useful frames than fixed or similarity-based rules?

The effect is task selective. On object-reasoning questions, every rule is at least 4.2 points below CSSI. On action reasoning, CSSI gains 5.0 points over the query-agnostic random and uniform rules, and similarity-based selectors approach but do not exceed it. The OCR subset shows large differences but contains only 14 questions, so we treat it only as a rough, low-sample diagnostic. Because the model, frame list, and token budget are held fixed and only the selection rule changes, the accuracy differences reflect which frames receive detail. Further ablations of the design choices and of what the selector reads are in Sections G and G.1, and per-example selector comparisons in Section H.

4.2.3 Qualitative allocation behavior

Figure 3 shows the learned allocation. The selected frames tend to cluster around a few short intervals that appear to carry question-relevant evidence, and long intervals (five to fourteen minutes) can

Table 2: **Comparing selection methods.** Six selection rules and the learned policy, broken out by Video-MME question type on the long split. Every rule promotes $K=64$ frames for the same model from the same 1 fps frame list capped at 1,024 frames, and every rule answers every question, so the columns are strictly paired. Δ values are w.r.t. CSSI.

	OCR		Object Reasoning.		Action Reas.	
Selection rule	**Acc.**	Δ	**Acc.**	Δ	**Acc.**	Δ
Uniform sampling	50.0	-28.6	56.7	-5.4	60.0	-5.0
CLIP similarity (Equation (6))	57.1	-21.4	57.5	-4.6	63.9	-1.1
AKS (Tang et al., 2025)	50.0	-28.6	57.9	-4.2	64.4	-0.6
BOLT (Liu et al., 2025a)	57.1	-21.4	56.7	-5.4	63.3	-1.7
FOCUS (Zhu et al., 2025)	50.0	-28.6	56.2	-5.8	62.2	-2.8
CSSI (ours)	**78.6**	-	**62.1**	-	**65.0**	-

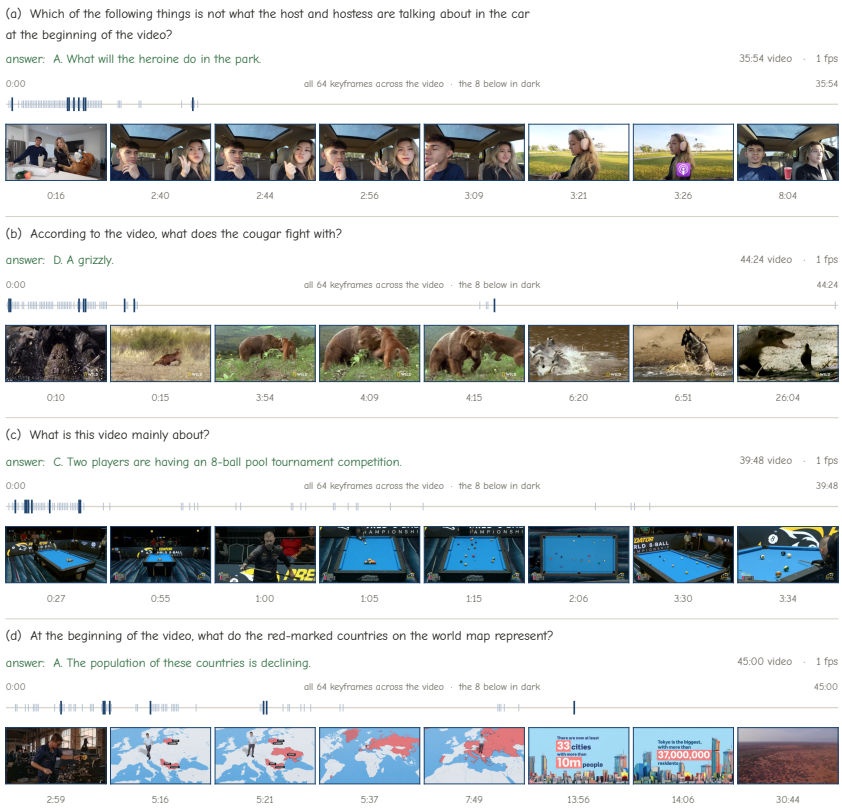


Figure 3: **Frames selected by the learned policy.** Four Video-MME long-video questions. Each timeline strip (band) marks all 64 selected frames across the full video. The eight thumbnails below it are the top-ranked selected frames, shown in temporal order. The selected frames tend to cluster around a small number of short intervals that appear to carry the question-relevant evidence, while long stretches contain no selected frames. For example, it promotes the grizzly encounter at 3:54 to 4:15 in a 44-minute wildlife video while the rest of the timeline stays coarse.

contain no selected frames. Because those intervals are kept as timeline tokens rather than removed, the surrounding temporal context stays available to the model alongside the selected detail.

4.2.4 Isolating selection: video needle in a haystack

To measure selection on its own, we adapt the V-NIAH protocol (Zhang et al., 2024a): a synthetic needle, a card with a code to read or a coloured shape to name, is hidden at a controlled depth in a haystack of concatenated long videos (5 to 120 minutes), and only the needle answers the question. Every selector promotes $K=64$ keyframes for the same model, and the needle spans several samples of the frame grid, so accuracy depends only on which frames are selected. Over 600 items (Figure 4),

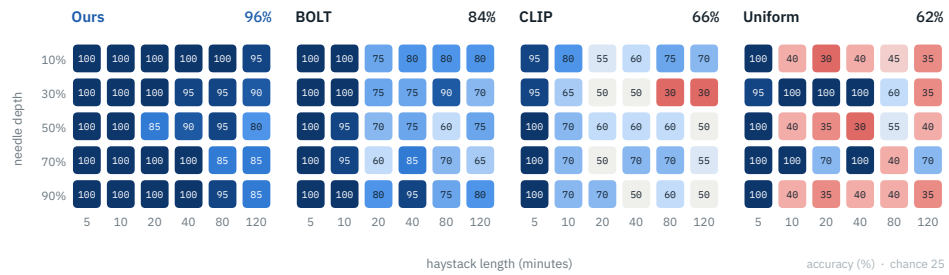


Figure 4: **Video needle in a haystack.** Accuracy (%) by haystack length and needle depth for four keyframe selectors inside the same model at K=64. Each tile is 20 needles, chance is 25%, and the number beside each name is the mean over all 600 items.

CSSI averages 95.8%, against 83.5% for BOLT, 65.7% for CLIP similarity, and 61.7% for uniform sampling. The gap widens with length, and on haystacks of 40 minutes or more CSSI keeps 92.7% while the baselines fall to 77.0%, 56.0%, and 51.0%. selection is decisive: across all selectors the model answers 99.6% of items whose needle is selected to a keyframe but only 32.6% of those left as a single timeline token, and forcing the needle into the keyframe set gives 100% at every length.

4.2.5 How much context and how much detail?

We jointly vary the number of context frames T and selected keyframes K with the model held fixed (Figure 5; full protocol in Section F). Across a sixteen-fold range of T and a fourfold range of K, accuracy spans only 4.3 points and peaks at the deployed point T=1,024, K=64, so the main results are not reported at a favourable corner of an otherwise weaker surface. Neither budget pays to increase: at every K, T=2,048 and the uncapped 1 fps stream fall below T=1,024, and raising K from 64 to 128 at T=1,024 nearly doubles the visual tokens yet lowers accuracy by 1.7 points.

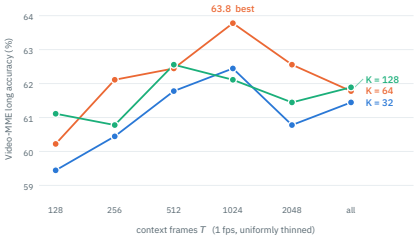


Figure 5: **Accuracy over T and K** on the Video-MME long split (18 cells, interpolated; range 59.4 to 63.8), peaking at the deployed point T=1,024, K=64.

5 Conclusion

CSSI treats long-video visual-budget allocation as selection within a complete mixed-resolution timeline rather than all-or-nothing frame selection: selected frames keep spatial detail, every other frame stays visible as one average-pooled token, and a selection policy trained from the model's own answer-margin reward on multiple-choice questions chooses what to promote from features computed once and shared with the model. Across seven long-video benchmarks it outperforms recent efficient video LLMs, and at a fixed visual-token budget its learned policy improves over heuristic selection, especially on long-video and object-reasoning sets.

Limitations The policy is trained and evaluated with multiple-choice supervision. The answer-margin reward of Equation (12) requires enumerable answer options, and all of our benchmarks score option accuracy. Although the selector itself needs only the question and can therefore run on open-ended queries (Section G.1), we have not measured whether the learned selection improves free-form answer generation at inference time, and training the policy directly for that setting would require a margin surrogate over open-ended outputs. Second, the policy's selection behavior is learned from the Stage II corpora, which are dominated by web, instructional, and movie footage; it may not transfer to domains where evidence has a different temporal structure, such as sign-language understanding, where the signal is continuous and fine-grained, or sports video, where decisive moments are brief and visually similar to their surroundings. Finally, the selection budget K is fixed per configuration rather than adapted per question, which can waste detail tokens on questions needing few frames and starve those needing many.

REPRODUCIBILITY STATEMENT

All architectural choices, token budgets, and objectives are specified in Sections B and 3, and the full training recipe (optimizer settings, learning-rate schedules, and per-stage data composition) is given in Sections C and D, with settings tabulated in Tables 3 and 4. The default configuration (patch grid, detail-token budget m, selection budget K, and frame cap) is stated in Sections 3.1 and 4.1. All evaluation benchmarks and the paired comparison protocol are described in Sections 4.1 and E. Every training and evaluation dataset is publicly available. The visual encoder, model backbone, and selection policy are standard released components, and the policy adds a single linear-time scan over precomputed frame features.

ETHICS STATEMENT

This work studies visual-budget allocation for long-video question answering using publicly available academic benchmarks and instruction-tuning corpora. It involves no human subjects, no private or personally identifying data collection, and no deployment. As with any video-language model, downstream systems built on these components can inherit biases present in the pretraining and instruction data. We do not introduce new data sources beyond those already released by the community.

USE OF AI ASSISTANTS

AI based tools were used only to polish grammar, wording, and phrasing of the manuscript. They were not used to generate research ideas, design or run experiments, produce results, or draw conclusions, all of which are the authors' own work.

REFERENCES

Anthropic. Claude sonnet 4.5 system card. Technical report, Anthropic, 2025.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond, 2023. URL https://arxiv.org/abs/2308.12966.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report, 2025a. URL https://arxiv.org/abs/2511.21631.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025b. URL https://arxiv.org/abs/2502.13923.

Mu Cai, Reuben Tan, Jianrui Zhang, Bocheng Zou, Kai Zhang, Feng Yao, Fangrui Zhu, Jing Gu, Yiwu Zhong, Yuzhang Shang, Yao Dou, Jaden Park, Jianfeng Gao, Yong Jae Lee, and Jianwei Yang. Temporalbench: Towards fine-grained temporal understanding for multimodal video models. *arXiv preprint arXiv:2410.10818*, 2024.

Guo Chen, Zhiqi Li, Shihao Wang, Jindong Jiang, Yicheng Liu, Lidong Lu, De-An Huang, Wonmin Byeon, Matthieu Le, Tuomas Rintamaki, Tyler Poon, Max Ehrlich, Tuomas Rintamaki, Tyler Poon,

Tong Lu, Limin Wang, Bryan Catanzaro, Jan Kautz, Andrew Tao, Zhiding Yu, and Guilin Liu. Eagle 2.5: Boosting long-context post-training for frontier vision-language models, 2025a. URL https://arxiv.org/abs/2504.15271.

Yukang Chen, Fuzhao Xue, Dacheng Li, Qinghao Hu, Ligeng Zhu, Xiuyu Li, Yunhao Fang, Haotian Tang, Shang Yang, Zhijian Liu, Yihui He, Hongxu Yin, Pavlo Molchanov, Jan Kautz, Linxi Fan, Yuke Zhu, Yao Lu, and Song Han. LongVILA: Scaling long-context visual language models for long videos. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=wCXAlfvCy6.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24185–24198, 2024.

Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, and Lidong Bing. Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. *arXiv preprint arXiv:2406.07476*, 2024. URL https://arxiv.org/abs/2406.07476.

Jang Hyun Cho, Andrea Madotto, Effrosyni Mavroudi, Triantafyllos Afouras, Tushar Nagarajan, Muhammad Maaz, Yale Song, Tengyu Ma, Shuming Hu, Suyog Jain, Miguel Martin, Huiyu Wang, Hanoona Rasheed, Peize Sun, Po-Yao Huang, Daniel Bolya, Nikhila Ravi, Shashank Jain, Tammy Stark, Shane Moon, Babak Damavandi, Vivian Lee, Andrew Westbury, Salman Khan, Philipp Krähenbühl, Piotr Dollár, Lorenzo Torresani, Kristen Grauman, and Christoph Feichtenhofer. Perceptionlm: Open-access data and models for detailed visual understanding, 2025. URL https://arxiv.org/abs/2504.13180.

Christopher Clark, Jieyu Zhang, Zixian Ma, Jae Sung Park, Mohammadreza Salehi, Rohun Tripathi, Sangho Lee, Zhongzheng Ren, Chris Dongjoo Kim, Yinuo Yang, Vincent Shao, Yue Yang, Weikai Huang, Ziqi Gao, Taira Anderson, Jianrui Zhang, Jitesh Jain, George Stoica, Winson Han, Ali Farhadi, and Ranjay Krishna. Molmo2: Open weights and data for vision-language models with video understanding and grounding, 2026. URL https://arxiv.org/abs/2601.10611.

Gheorghe Comanici et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Tri Dao and Albert Gu. Transformers are ssms: Generalized models and efficient algorithms through structured state space duality, 2024. URL https://arxiv.org/abs/2405.21060.

Miquel Farré, Andi Marafioti, Lewis Tunstall, Leandro Von Werra, and Thomas Wolf. Finevideo. https://huggingface.co/datasets/HuggingFaceFV/finevideo, 2024.

Jiajun Fei, Dian Li, Zhidong Deng, Zekun Wang, Gang Liu, and Hui Wang. Video-ccam: Enhancing video-language understanding with causal cross-attention masks for short and long videos, 2024. URL https://arxiv.org/abs/2408.14023.

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 *CVPR*, 2025.

Google DeepMind. Gemini 3 Pro model card. https://deepmind.google/models/model-cards/gemini-3-pro/, 2025.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *First Conference on Language Modeling*, 2024.

Ali Hatamizadeh, Yejin Choi, and Jan Kautz. Gated deltanet-2: Decoupling erase and write in linear attention. *arXiv preprint arXiv:2605.22791*, 2026.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Md Mohaiminul Islam, Tushar Nagarajan, Huiyu Wang, Gedas Bertasius, and Lorenzo Torresani. Bimba: Selective-scan compression for long-range video question answering. *arXiv preprint arXiv:2503.09590*, 2025.

Baoxiong Jia, Ting Lei, Song-Chun Zhu, and Siyuan Huang. EgotaskQA: Understanding human tasks in egocentric videos. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022.

Jindong Jiang, Xiuyu Li, Zhijian Liu, Muyang Li, Guo Chen, Zhiqi Li, De-An Huang, Guilin Liu, Zhiding Yu, Kurt Keutzer, Sungjin Ahn, Jan Kautz, Hongxu Yin, Yao Lu, Song Han, and Wonmin Byeon. Storm: Token-efficient long video understanding for multimodal llms, 2025. URL https://arxiv.org/abs/2503.04130.

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, 2024. URL https://arxiv.org/abs/2311.08046.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-onevision: Easy visual task transfer. *Transactions on Machine Learning Research*, 2025a. ISSN 2835-8856. URL https://openreview.net/forum?id=zKv8qULV6n.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-onevision: Easy visual task transfer. *Transactions on Machine Learning Research*, 2025b. ISSN 2835-8856. URL https://openreview.net/forum?id=zKv8qULV6n.

KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding, 2024a. URL https://arxiv.org/abs/2305.06355.

Xinhao Li, Yi Wang, Jiashuo Yu, Xiangyu Zeng, Yuhan Zhu, Haian Huang, Jianfei Gao, Kunchang Li, Yinan He, Chenting Wang, Yu Qiao, Yali Wang, and Limin Wang. Videochat-flash: Hierarchical compression for long-context video modeling. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=MUjdNcfNPv.

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

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 Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.342. URL https://aclanthology.org/2024.emnlp-main.342/.

Jiajun Liu, Yibing Wang, Hanghang Ma, Xiaoping Wu, Xiaoqi Ma, xiaoming Wei, Jianbin Jiao, Enhua Wu, and Jie Hu. Kangaroo: A powerful video-language model supporting long-context video input. *arXiv preprint arXiv:2408.15542*, 2024a.

Shuming Liu, Chen Zhao, Tianqi Xu, and Bernard Ghanem. Bolt: Boost large vision-language model without training for long-form video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2025a.

Zhijian Liu, Ligeng Zhu, Baifeng Shi, Zhuoyang Zhang, Yuming Lou, Shang Yang, Haocheng Xi, Shiyi Cao, Yuxian Gu, Dacheng Li, Xiuyu Li, Yunhao Fang, Yukang Chen, Cheng-Yu Hsieh, De-An Huang, An-Chieh Cheng, Vishwesh Nath, Jinyi Hu, Sifei Liu, Ranjay Krishna, Daguang Xu, Xiaolong Wang, Pavlo Molchanov, Jan Kautz, Hongxu Yin, Song Han, and Yao Lu. Nvila: Efficient frontier visual language models, 2024b. URL https://arxiv.org/abs/2412.04468.

Zuyan Liu, Yuhao Dong, Ziwei Liu, Winston Hu, Jiwen Lu, and Yongming Rao. Oryx MLLM: On-demand spatial-temporal understanding at arbitrary resolution. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=ODiY6pbHZQ.

Wentao Ma, Weiming Ren, Yiming Jia, Zhuofeng Li, Ping Nie, Ge Zhang, and Wenhu Chen. Videoeval-pro: Robust and realistic long video understanding evaluation, 2025. URL https://arxiv.org/abs/2505.14640.

Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.

OpenAI. GPT-4V(ision) system card. https://openai.com/index/gptv-system-card/, 2023.

OpenAI. GPT-5 system card. Technical report, OpenAI, 2025.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian

Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

Viorica Pătrăucean, Lucas Smaira, Ankush Gupta, Adrià Recasens Continente, Larisa Markeeva, Dylan Banarse, Skanda Koppula, Joseph Heyward, Mateusz Malinowski, Yi Yang, Carl Doersch, Tatiana Matejovicova, Yury Sulsky, Antoine Miech, Alex Frechette, Hanna Klimczak, Raphael Koster, Junlin Zhang, Stephanie Winkler, Yusuf Aytar, Simon Osindero, Dima Damen, Andrew Zisserman, and João Carreira. Perception test: A diagnostic benchmark for multimodal video models, 2023. URL https://arxiv.org/abs/2305.13786.

Minghao Qin, Xiangrui Liu, Zhengyang Liang, Yan Shu, Huaying Yuan, Juenjie Zhou, Shitao Xiao, Bo Zhao, and Zheng Liu. Video-xl-2: Towards very long-video understanding through task-aware kv sparsification. *arXiv preprint arXiv:2506.19225*, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 18–24 Jul 2021.

Ruchit Rawal, Khalid Saifullah, Miquel Farré, Ronen Basri, David Jacobs, Gowthami Somepalli, and Tom Goldstein. Cinepile: A long video question answering dataset and benchmark, 2024. URL https://arxiv.org/abs/2405.08813.

Weiming Ren, Huan Yang, Jie Min, Cong Wei, and Wenhu Chen. Vista: Enhancing long-duration and high-resolution video understanding by video spatiotemporal augmentation, 2024. URL https://arxiv.org/abs/2412.00927.

Weiming Ren, Wentao Ma, Huan Yang, Cong Wei, Ge Zhang, and Wenhu Chen. Vamba: Understanding hour-long videos with hybrid mamba-transformers, 2025a. URL https://arxiv.org/abs/2503.11579.

Weiming Ren, Wentao Ma, Huan Yang, Cong Wei, Ge Zhang, and Wenhu Chen. Vamba: Understanding hour-long videos with hybrid mamba-transformers, 2025b. URL https://arxiv.org/abs/2503.11579.

Saul Santos, António Farinhas, Daniel C Mcnamee, and Andre Martins. ∞-video: A training-free approach to long video understanding via continuous-time memory consolidation. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 52877–52893. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/santos25a.html.

Yan Shu, Peitian Zhang, Zheng Liu, Minghao Qin, Junjie Zhou, Tiejun Huang, and Bo Zhao. Video-xl: Extra-long vision language model for hour-scale video understanding. *arXiv preprint arXiv:2409.14485*, 2024.

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

Enxin Song, Wenhao Chai, Shusheng Yang, Ethan J. Armand, Xiaojun Shan, Haiyang Xu, Jianwen Xie, and Zhuowen Tu. VideoNSA: Native sparse attention scales video understanding. In *The Fourteenth International Conference on Learning Representations*, 2026.

Xi Tang, Jihao Qiu, Lingxi Xie, Yunjie Tian, Jianbin Jiao, and Qixiang Ye. Adaptive keyframe sampling for long video understanding. *arXiv preprint arXiv:2502.21271*, 2025.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. 2023.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. 2024.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. 2025a.

V Team, Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, Lihang Pan, Shuaiqi Duan, Weihan Wang, Yan Wang, Yean Cheng, Zehai He, Zhe Su, Zhen Yang, Ziyang Pan, Aohan Zeng, Baoxu Wang, Bin Chen, Boyan Shi, Changyu Pang, Chenhui Zhang, Da Yin, Fan Yang, Guoqing Chen, Jiazheng Xu, Jiale Zhu, Jiali Chen, Jing Chen, Jinhao Chen, Jinghao Lin, Jinjiang Wang, Junjie Chen, Leqi Lei, Letian Gong, Leyi Pan, Mingdao Liu, Mingde Xu, Mingzhi Zhang, Qinkai Zheng, Sheng Yang, Shi Zhong, Shiyu Huang, Shuyuan Zhao, Siyan Xue, Shangqin Tu, Shengbiao Meng, Tianshu Zhang, Tianwei Luo, Tianxiang Hao, Tianyu Tong, Wenkai Li, Wei Jia, Xiao Liu, Xiaohan Zhang, Xin Lyu, Xinyue Fan, Xuancheng Huang, Yanling Wang, Yadong Xue, Yanfeng Wang, Yanzi Wang, Yifan An, Yifan Du, Yiming Shi, Yiheng Huang, Yilin Niu, Yuan Wang, Yuanchang Yue, Yuchen Li, Yutao Zhang, Yuting Wang, Yu Wang, Yuxuan Zhang, Zhao Xue, Zhenyu Hou, Zhengxiao Du, Zihan Wang, Peng Zhang, Debing Liu, Bin Xu, Juanzi Li, Minlie Huang, Yuxiao Dong, and Jie Tang. Glm-4.5v and glm-4.1v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning, 2025b. URL https://arxiv.org/abs/2507.01006.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution, 2024a. URL https://arxiv.org/abs/2409.12191.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution, 2024b. URL https://arxiv.org/abs/2409.12191.

Weihan Wang, Zehai He, Wenyi Hong, Yean Cheng, Xiaohan Zhang, Ji Qi, Shiyu Huang, Bin Xu, Yuxiao Dong, Ming Ding, and Jie Tang. Lvbench: An extreme long video understanding benchmark, 2024c.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025a.

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, Vienna, Austria, July 2025b. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.283. URL https://aclanthology.org/2025.findings-acl.283/.

Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, Ping Luo, Ziwei Liu, Yali Wang, Limin Wang, and Yu Qiao. Internvid: A large-scale video-text dataset for multimodal understanding and generation. In *The Twelfth International Conference on Learning Representations*, 2024d.

Bo Wu, Shoubin Yu, Zhenfang Chen, Joshua B Tenenbaum, and Chuang Gan. Star: A benchmark for situated reasoning in real-world videos, 2024a. URL https://arxiv.org/abs/2405.09711.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. Longvideobench: A benchmark for long-context interleaved video-language understanding. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024b.

Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9777–9786, June 2021.

Boshen Xu, Zihan Xiao, Jiaze Li, Jianzhong Ju, Zhenbo Luo, Jian Luan, and Qin Jin. Timeviper: A hybrid mamba-transformer model for efficient long video understanding. *arXiv preprint arXiv:2511.16595*, 2025a.

Lin Xu, Yilin Zhao, Daquan Zhou, Zhijie Lin, See Kiong Ng, and Jiashi Feng. Pllava : Parameter-free llava extension from images to videos for video dense captioning, 2024. URL https://arxiv.org/abs/2404.16994.

Mingze Xu, Mingfei Gao, Zhe Gan, Hong-You Chen, Zhengfeng Lai, Haiming Gang, Kai Kang, and Afshin Dehghan. Slowfast-LLaVA: A strong training-free baseline for video large language models, 2025b. URL https://openreview.net/forum?id=WkpqUVcSTy.

Weili Xu, Enxin Song, Wenhao Chai, Xuexiang Wen, Tian Ye, and Gaoang Wang. Auroralong: Bringing rnns back to efficient open-ended video understanding, 2025c. URL https://arxiv.org/abs/2507.02591.

Shilin Yan, Jiaming Han, Joey Tsai, Hongwei Xue, Rongyao Fang, Lingyi Hong, Ziyu Guo, and Ray Zhang. Crosslmm: Decoupling long video sequences from lmms via dual cross-attention mechanisms. *arXiv preprint arXiv:2505.17020*, 2025.

An Yang et al. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024a.

Biao Yang, Bin Wen, Boyang Ding, Changyi Liu, Chenglong Chu, Chengru Song, Chongling Rao, Chuan Yi, Da Li, Dunju Zang, Fan Yang, Guorui Zhou, Guowang Zhang, Han Shen, Hao Peng, Haojie Ding, Hao Wang, Haonan Fan, Hengrui Ju, Jiaming Huang, Jiangxia Cao, Jiankang Chen, Jingyun Hua, Kaibing Chen, Kaiyu Jiang, Kaiyu Tang, Kun Gai, Muhao Wei, Qiang Wang, Ruitao Wang, Sen Na, Shengnan Zhang, Siyang Mao, Sui Huang, Tianke Zhang, Tingting Gao, Wei Chen, Wei Yuan, Xiangyu Wu, Xiao Hu, Xingyu Lu, Yi-Fan Zhang, Yiping Yang, Yulong Chen, Zeyi Lu, Zhenhua Wu, Zhixin Ling, Zhuoran Yang, Ziming Li, Di Xu, Haixuan Gao, Hang Li, Jing Wang, Lejian Ren, Qigen Hu, Qianqian Wang, Shiyao Wang, Xinchen Luo, Yan Li, Yuhang Hu, and Zixing Zhang. Kwai keye-vl 1.5 technical report, 2025a. URL https://arxiv.org/abs/2509.01563.

Dongjie Yang, Suyuan Huang, Chengqiang Lu, Xiaodong Han, Haoxin Zhang, Yan Gao, Yao Hu, and hai zhao. Vript: A video is worth thousands of words. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024b.

Songlin Yang, Jan Kautz, and Ali Hatamizadeh. Gated delta networks: Improving mamba2 with delta rule. In *International Conference on Learning Representations (ICLR)*, 2025b.

Yifeng Yao, Yike Yun, Jing Wang, Huishuai Zhang, Dongyan Zhao, Ke Tian, Zhihao Wang, Minghui Qiu, and Tao Wang. K-frames: Scene-driven any-k keyframe selection for long video understanding, 2025. URL https://arxiv.org/abs/2510.13891.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, et al. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv preprint arXiv:2408.01800*, 2024.

Jiabo Ye, Haiyang Xu, Haowei Liu, Anwen Hu, Ming Yan, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. mPLUG-owl3: Towards long image-sequence understanding in multi-modal large language models. In *The Thirteenth International Conference on Learning Representations*, 2025.

Kexin Yi*, Chuang Gan*, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B. Tenenbaum. Clevrer: Collision events for video representation and reasoning. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=HkxYzANYDB.

Tianyu Yu, Zefan Wang, Chongyi Wang, Fuwei Huang, Wenshuo Ma, Zhihui He, Tianchi Cai, Weize Chen, Yuxiang Huang, Yuanqian Zhao, Bokai Xu, Junbo Cui, Yingjing Xu, Liqing Ruan, Luoyuan Zhang, Hanyu Liu, Jingkun Tang, Hongyuan Liu, Qining Guo, Wenhao Hu, Bingxiang He, Jie Zhou, Jie Cai, Ji Qi, Zonghao Guo, Chi Chen, Guoyang Zeng, Yuxuan Li, Ganqu Cui, Ning Ding, Xu Han, Yuan Yao, Zhiyuan Liu, and Maosong Sun. Minicpm-v 4.5: Cooking efficient mllms via architecture, data, and training recipe, 2025. URL https://arxiv.org/abs/2509.18154.

Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. Activitynet-qa: A dataset for understanding complex web videos via question answering. In *AAAI*, pp. 9127–9134, 2019.

Yunzhuzhang, Yu Lu, Tianyi Wang, Fengyun Rao, Yi Yang, and Linchao Zhu. Flexselect: Flexible token selection for efficient long video understanding. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=0D3ja9s17M.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11975–11986, October 2023.

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, Singapore, December 2023a. Association for Computational Linguistics. URL https://aclanthology.org/2023.emnlp-demo.49/.

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

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

Ruohong Zhang, Liangke Gui, Zhiqing Sun, Yihao Feng, Keyang Xu, Yuanhan Zhang, Di Fu, Chunyuan Li, Alexander G Hauptmann, Yonatan Bisk, and Yiming Yang. Direct preference optimization of video large multimodal models from language model reward. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, Albuquerque, New Mexico, April 2025a. Association for Computational Linguistics.

Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun MA, Ziwei Liu, and Chunyuan Li. Video instruction tuning with synthetic data, 2024b. URL https://openreview.net/forum?id=8Livf4oZxz.

Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Llava-video: Video instruction tuning with synthetic data, 2025b. URL https://arxiv.org/abs/2410.02713.

Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Shitao Xiao, Xi Yang, Yongping Xiong, Bo Zhang, Tiejun Huang, and Zheng Liu. Mlvu: A comprehensive benchmark for multi-task long video understanding. *arXiv preprint arXiv:2406.04264*, 2024.

Bin Zhu, Bin Lin, Munan Ning, Yang Yan, Jiaxi Cui, WANG HongFa, Yatian Pang, Wenhao Jiang, Junwu Zhang, Zongwei Li, Cai Wan Zhang, Zhifeng Li, Wei Liu, and Li Yuan. Languagebind: Extending video-language pretraining to n-modality by language-based semantic alignment. In *The Twelfth International Conference on Learning Representations*, 2024.

Zirui Zhu, Hailun Xu, Yang Luo, Yong Liu, Kanchan Sarkar, Zhenheng Yang, and Yang You. Focus: Efficient keyframe selection for long video understanding, 2025.

A Extended related work

This appendix expands Section 2 with a broader survey of the work CSSI builds on and is compared against: video-language models, efficient long-video architectures and visual-token reduction, frame and keyframe selection, linear-time sequence models, and long-video benchmarks and training corpora.

Image- and video-language models. Video-language models inherit their visual grounding from image-text pretraining with contrastive encoders such as CLIP (Radford et al., 2021), SigLIP (Zhai et al., 2023), and LanguageBind (Zhu et al., 2024). A large family of instruction-tuned video LLMs couples a visual encoder to a language model through a projector: Video-LLaMA and its successors (Zhang et al., 2023a; Cheng et al., 2024; Zhang et al., 2023b), Video-LLaVA (Lin et al., 2024), VideoChat (Li et al., 2024a), Chat-UniVi (Jin et al., 2024), mPLUG-Owl3 (Ye et al., 2025), MiniCPM-V (Yao et al., 2024), PLLaVA (Xu et al., 2024), and the LLaVA-OneVision / LLaVA-Video line (Li et al., 2025a;b; Zhang et al., 2024b; 2025b). General-purpose multimodal systems, including GPT-4V and GPT-4o (OpenAI, 2023; OpenAI et al., 2024; Hurst et al., 2024), the Gemini and Gemma families (Team et al., 2023; 2024; 2025a), the Qwen-VL series (Bai et al., 2023; Wang et al., 2024b; Bai et al., 2025b;a), and InternVL (Chen et al., 2024), set the accuracy frontier but read only a modest number of frames at once, since their cost grows with the number of visual tokens. CSSI targets the regime in which the video is far longer than such a window.

Efficient long-video models and visual-token reduction. Because a decoder's cost grows with the visual-token count, long-video systems reduce that count before or during language reasoning. One line pools or merges tokens spatially and temporally: LLaMA-VID represents a frame with as few as two tokens (Li et al., 2024b), PLLaVA pools image features into video (Xu et al., 2024), Oryx adapts spatial resolution on demand (Liu et al., 2025b), and SlowFast-LLaVA mixes a dense-temporal and a dense-spatial stream (Xu et al., 2025b). A second line extends the context or compresses the key-value cache: LongVILA (Chen et al., 2025b), long-context transfer (Zhang et al., 2024a), Kangaroo (Liu et al., 2024a), NVILA (Liu et al., 2024b), Video-XL (Shu et al., 2024), CrossLMM (Yan et al., 2025), and native sparse attention (Song et al., 2026). A third line adds explicit temporal modules or memory: STORM (Jiang et al., 2025), BIMBA (Islam et al., 2025), Video-CCAM (Fei et al., 2024), and training-free memory consolidation in ∞-Video (Santos et al., 2025). CSSI instead keeps every sampled frame as a single timeline token and spends spatial detail only on selected frames, reducing tokens by *allocation* rather than by discarding temporal coverage.

Frame and keyframe selection. Closest to our selection step is work that chooses which frames a model should read. Query-conditioned selectors score frames by similarity, coverage, or motion: adaptive keyframe sampling (Tang et al., 2025), FOCUS (Zhu et al., 2025), BOLT (Liu et al., 2025a), and scene-driven any-k keyframe selection (Yao et al., 2025). These methods generally *replace* the video with the selected frames. CSSI differs in two ways: the selection is trained end-to-end from the model's own answer margin rather than from a proxy similarity, and unselected frames remain visible as timeline tokens, so selection promotes resolution instead of deciding survival.

Linear-time sequence models. Selective state-space models (Gu & Dao, 2024; Dao & Gu, 2024) provide linear-time sequence mixing and a recurrent state, and have been applied to long video by Vamba (Ren et al., 2025a;b), STORM (Jiang et al., 2025), and RNN-style efficient models such as AuroraLong (Xu et al., 2025c). CSSI uses a Gated DeltaNet-2 backbone (Hatamizadeh et al., 2026), which decouples the erase and write gates of the gated delta rule (Yang et al., 2025b), for the selection policy: one causal scan produces a question-conditioned state per frame, and the recurrent selector memory carries the frames already selected so that the budget is spent on complementary evidence.

Long-video benchmarks and training corpora. We evaluate on Video-MME (Fu et al., 2025), LongVideoBench (Wu et al., 2024b), MLVU (Zhou et al., 2024), EgoSchema (Mangalam et al., 2023), LVBench (Wang et al., 2024c), and VideoEval-Pro (Ma et al., 2025). Related benchmarks include TemporalBench (Cai et al., 2024), NExT-QA (Xiao et al., 2021), STAR (Wu et al., 2024a), EgoTaskQA (Jia et al., 2022), CLEVRER (Yi* et al., 2020), and MovieChat (Song et al., 2023). Training draws on video instruction and long-form question-answering corpora including LLaVA-Video-178K (Zhang et al., 2024b), CinePile (Rawal et al., 2024), FineVideo (Farré et al., 2024),

Table 3: **Training data by stage.** Approximate sample counts. All corpora are public. Stage II counts are after temporal-sensitivity filtering (Section C).

Stage	Substage	Datasets	#Samples	Trained params
I	Modality alignment	Image-text, InternVid, Vript, web captions	∼7.4M	projector
I	Instruction tuning	LLaVA-Video-178K, NExT-QA, VideoChat, PerceptionTest, STAR, EgoTaskQA, CLEVRER, MovieChat	∼1.3M	projector + model
I	Long-form QA	CinePile, FineVideo	∼0.5M	projector + model
II	Policy training	LLaVA-Video-178K (MC), VISTA-400K (MC), CinePile	393K / 962K	policy

InternVid (Wang et al., 2024d), and Vript (Yang et al., 2024b). Preference- and reward-based tuning of video LMMs (Zhang et al., 2025a) is complementary to our answer-margin signal, which supervises frame selection rather than answer generation.

B Architecture and inference

Encoder, projector, and pooling. The visual encoder is a frozen SigLIP-So400M model (Zhai et al., 2023) run at 384×384. Each frame produces N_p=729 patch embeddings (27×27) of width d_v. A two-layer GELU MLP projector $\mathcal{P}_\phi$ maps them to the model width d (Equation (2)). A selected frame is 2×2 average-pooled to m=196 detail tokens (14×14, Equation (3)). Every other frame is mean-pooled to a single timeline token (Equation (4)). The model is a Qwen2-7B decoder. Pooling adds no parameters, and the encoder is run once per video so its features are shared by the model and the policy.

Selection policy. The policy is a 24-layer Gated DeltaNet-2 stack (Hatamizadeh et al., 2026) of hidden width $2{,}048$. A lightweight policy projector maps each frame's patch grid to a fixed number of policy tokens. The question embeddings, followed by these per-frame tokens, are scanned once (causally) to yield a per-frame state $\mathbf{g}_t$ (Section 3.3). Two linear heads produce a frame key $\mathbf{k}_t$ and value $\mathbf{v}_t$ (Equation (8)). A learned probe reads the fast-weight selector memory into the selection query $\mathbf{u}_\ell$ (Equation (9)), and each selected frame's key-value pair is written into that memory with the gated delta rule (Equation (10)), so each selection is conditioned on earlier ones. The final blocks of the stack are re-initialized, while earlier blocks retain their video-language pretrained weights.

Inference. At test time the policy selects K indices greedily (argmax at each step) without replacement, using no CLIP scorer or second vision encoder. The frame features are computed in a single pass. The policy then performs K sequential selection steps, and the model decodes the assembled mixed-resolution sequence of Equation (5) once.

C Training data

All videos are decoded at one frame per second. Table 3 summarizes the corpora used in each stage, whose roles we describe below.

Stage I corpora. Modality alignment uses image-text pairs and short video-caption corpora, including InternVid (Wang et al., 2024d) and Vript (Yang et al., 2024b), to align the projector while both token types are present. Video instruction tuning uses LLaVA-Video-178K (Zhang et al., 2024b) with NExT-QA (Xiao et al., 2021), together with short-video question-answering and perception sources: VideoChat-style instruction data (Li et al., 2024a), PerceptionTest (Pătrăucean et al., 2023), STAR (Wu et al., 2024a), EgoTaskQA (Jia et al., 2022), CLEVRER (Yi* et al., 2020), and MovieChat (Song et al., 2023). Long-form tuning continues on CinePile (Rawal et al., 2024) and FineVideo (Farré et al., 2024) at the full $1{,}024$-frame cap.

Stage II corpora. We define the policy reward only for multiple-choice examples, so the correct-option margin of Equation (12) is well defined: LLaVA-Video-178K multiple-choice items (with its NExT-QA, ActivityNet-QA (Yu et al., 2019), and PerceptionTest subsets), the multiple-choice subsets of VISTA-400K, and CinePile. Starting from roughly 9.6×10^5 candidate QA pairs we apply the temporal-sensitivity filter below, which retains 41% ($\approx 3.9 \times 10^5$ pairs).

Temporal-sensitivity filtering. An example enters Stage II only if the selected-frame set can affect the predicted answer. For each candidate we draw several random selection sets and score them with

Table 4: **Optimization settings by stage.** "n/a" denotes not applicable. Stage II trains only the selection policy, directly at the deployed budget K=64.

Setting	Stage I (model)	Stage II (policy)
Optimizer	AdamW	AdamW (β_1=0.9, β_2=0.99)
Learning rate	10^{-5} (all)	6×10^{-5} reinit / 10^{-5} pretrained
Schedule	cosine, warm-up ratio 0.03	cosine, 256-step warm-up
Effective batch	4 (bs 1×accum 4)	64 video-question pairs
Gradient clip	n/a	1.0
Precision	bfloat16	bfloat16
Duration	1 epoch / substage	1 epoch (filtered data)
Entropy weight β	n/a	10^{-4}
Group size N	n/a	16
Trainable modules	projector, model	selection policy only

the frozen Stage-I model. We discard questions that are answerable from almost any selection set or from none of the sampled sets, i.e., whose answer-margin variance falls below a threshold τ_{var}, which concentrates the policy-gradient signal on allocation-sensitive questions.

D TRAINING PROCEDURE

Both stages use AdamW in bfloat16 with a cosine schedule and keep the visual encoder frozen. Table 4 lists the settings.

Stage I: model training. Stage I trains the projector and model to read the mixed-resolution sequence, with selections supplied by the parameter-free CLIP rule of Equation (6): a frozen CLIP encoder (Radford et al., 2021) scores each frame against the question, frames are admitted in decreasing score subject to a 0.85 pairwise-similarity diversity threshold, and any remaining slots are evenly spaced. The projector is trained first in isolation (modality alignment). The projector and model are then tuned jointly on instruction and long-form data. We use a learning rate of 10^{-5}, warm-up ratio 0.03, per-device batch size 1 with gradient accumulation 4, and one epoch per substage. Only the multimodal projector and model weights are updated.

Stage II: policy training. Stage II freezes the encoder and the entire Stage-I model, and trains only the selection policy by minimizing the policy-gradient loss $\mathcal{L}_\pi$ (Equation (14)) on the sampled selection sets. We optimize with AdamW (β_1=0.9, β_2=0.99), a 256-step linear warm-up then cosine decay, gradient clipping at 1.0, and 64 video-question pairs per update under memory-efficient data parallelism with gradient checkpointing. Re-initialized policy layers use learning rate 6×10^{-5} and pretrained policy layers 10^{-5}, with a fixed entropy weight β=10^{-4} and group size N=16 (the number of selection trajectories sampled per video-question pair for the advantage in Equation (13)). The policy is trained directly at the deployed budget K=64. The reward is produced by the frozen model itself, so no external reward model and no frame-level selection labels are used.

E EVALUATION BENCHMARKS AND PROTOCOL

We report multiple-choice accuracy on seven long-video benchmarks: Video-MME (Fu et al., 2025) and its subtitled variant (Video-MME$_{\text{sub}}$), LongVideoBench (Wu et al., 2024b), the MLVU multiple-choice test split, EgoSchema (Mangalam et al., 2023), LVBench (Wang et al., 2024c), and the multiple-choice VideoEval-Pro. Video-MME is reported without subtitles unless a column states otherwise. Its long split contains 900 questions and is the setting for Table 2 and Figure 5. Unless a table states otherwise, comparisons are strictly paired: every condition sees the same questions, the same 1 fps frame list (capped at 1,024 frames), the same model, and the same visual-token budget, and only the selection rule varies. Each option is scored by the length-normalized log-likelihood the model assigns to it, and the highest-scoring option is taken as the prediction. For the baselines run through `lmms-eval` we instead read the first emitted option letter, as noted in Table 1.

F BUDGET SWEEP: PROTOCOL AND DETAILS

This appendix gives the full protocol and per-region analysis behind Section 4.2.5 and Figure 5.

Setup. We use the Video-MME long split (300 videos, 900 questions) and answer the same 900 questions in every cell, so any two settings are paired. The context is the 1 fps stream thinned uniformly to $T \in \{128, 256, 512, 1{,}024, 2{,}048\}$ frames, together with an uncapped setting ("all" in Figure 5) that keeps every 1 fps frame. Since no video in the split reaches 4,096 s, this uncapped setting holds 2,512 frames on average and 3,732 at most. Keyframes come from the selection policy of Section 3.3: we sample one 128-step trajectory per question and keep its first $K \in \{32, 64, 128\}$ indices, so the keyframe sets are nested across K. The model, decoding, and prompt are held fixed, and a cell costs $196K + (T - K)$ visual tokens. The 18 measured cells are interpolated with a shape-preserving (PCHIP) scheme that stays within the measured range (59.4 to 63.8).

More context stops helping past 1,024 frames. At every K, both T=2,048 and the uncapped stream fall below T=1,024, and the drop occurs between 1,024 and 2,048 frames. Beyond that point, reading every frame changes little: on the 714 questions whose video exceeds 2,048 frames, the uncapped stream differs from T=2,048 by $+0.84$, -0.98, and $+0.56$ points at K=32, 64, and 128, negligible in all three cases. As a consistency check, on the 62 videos with at most 2,048 frames the two settings receive identical inputs and their predictions agree exactly at every K. Extra single-token frames therefore add no evidence that the selected frames lack, and past a point they dilute the sequence the model attends over.

More keyframes do not help either. At T=1,024, raising K from 64 to 128 nearly doubles the visual tokens (13,504 to 25,984) yet lowers accuracy by 1.7 points. Much larger budgets are unreachable regardless: with 196 tokens per keyframe and a 32,768-position context without RoPE scaling, K is capped at 156 even at T=2,048, and K=256 does not fit at any T.

The small-T corner partly reflects uniform sampling. When T is small relative to K, the policy's selections map onto fewer distinct context frames than there are keyframe slots, and the remaining slots are filled with evenly spaced frames. At K=128 the mean number of distinct selected frames is 67.3 at T=128 and 94.4 at T=256, and the set is complete only from T=512 onward. The weaker cells in the lower-left of Figure 5 therefore partly measure uniform sampling rather than the policy, and should not be read as a failure of the policy at small budgets.

G ADDITIONAL ABLATIONS

Table 5 varies the main design choices of CSSI on a fixed set of Video-MME long questions with the model held constant, so every comparison is paired. Autoregressive selection slightly outperforms ranking the same policy scores without feedback. selection resolution matters most, with accuracy dropping 5.6 points from 196 to 16 tokens per keyframe, while spending the same token budget on more, lower-resolution keyframes is roughly neutral. Presenting keyframes in temporal order rather than selection order also helps.

G.1 WHAT THE SELECTOR READS

The policy is conditioned on the question and, by default, on the answer options. A selector that reads the options could in principle look for frames that support an option rather than frames that answer the question, which would make the gain a form of confirmation bias rather than evidence seeking. We test this directly by changing what the *selector* reads while the model always sees the full question with every option. Keyframes are re-selected with the question and all options (our setting), with the question alone, with the question and the correct option only, and with the question and a single fixed wrong option. All four are evaluated on the same 545 Video-MME long questions.

Accuracy is flat across the four: 65.0 with all options, 65.5 with the question alone, 65.1 when the selector is shown the correct option, and 64.6 when it is steered by a wrong one. Two conclusions follow. The policy does not need the options, so selection is driven by the question. Revealing the correct option to the selector does not help, so the frames it promotes are not simply those that match

Table 5: **Design choices of CSSI.** Every row is evaluated on the same 587 Video-MME long questions with the same model, so the comparisons are paired. Δ is against the setting we use (in bold) within each block. Unless a row says otherwise, the configuration is $K{=}64$ keyframes at 196 tokens, a 1 fps context capped at 1,024 frames, and the remaining frames at one token each.

Setting	Acc.	Δ	Note
Selection			
autoregressive selection (ours)	**64.9**		each pick conditions on the previous ones
top-K by the policy marginal	64.4	-0.5	no feedback. The same scores ranked
candidate grid 2 fps	63.9	-1.0	half the candidate frames
candidate grid 1 fps	64.1	-0.9	a quarter of them
Keyframe resolution			
196 tokens per keyframe (ours)	**64.9**		$27{\times}27$ patches pooled to $14{\times}14$
64 tokens per keyframe	63.2	-1.7	pooled to $8{\times}8$
64 tokens, $K{=}192$	64.1	-0.9	token-matched to the first row
16 tokens per keyframe	59.3	-5.6	pooled to $4{\times}4$
Keyframe order			
keyframes in temporal order (ours)	**64.9**		compressed stream removed in both rows
keyframes in selection order	62.7	-2.2	the order the policy produced them

the answer text. The adversarial condition is the sharper test: steering the selector with a single wrong option raises how often the model then chooses that option only from 12.5% to 13.4%. Mis-targeted selection therefore costs a little accuracy, but it does not drag the model toward the option the selector was given.

H WHERE EACH SELECTOR LOOKS

Figures 6 to 9 compare the keyframes chosen by CSSI with those of the CLIP similarity rule of Equation (6), AKS (Tang et al., 2025), and BOLT (Liu et al., 2025a) on four long Video-MME questions. Every method promotes $K{=}64$ keyframes from the same list of context frames (1 fps, capped at 1,024 frames), and the same model reads each selection, so the four rows of a figure differ only in which frames are selected.

To make the evidence visible, each frame is scored by how much better it matches the correct option than the strongest wrong option, using SigLIP image and text embeddings. This score is drawn as the shaded curve behind each row and is identical across the four rows of a figure. The ticks mark the 64 keyframes of each method, and below them are that method's eight highest-scoring keyframes in temporal order. The same rule chooses the thumbnails for every method, so a row with no relevant thumbnail is a selection that contains no relevant frame.

We chose questions whose answer is visible in the frames. In every example CSSI answers correctly while CLIP similarity and BOLT do not. AKS answers three of the four correctly, and we keep those cases because they show what a partial hit looks like.

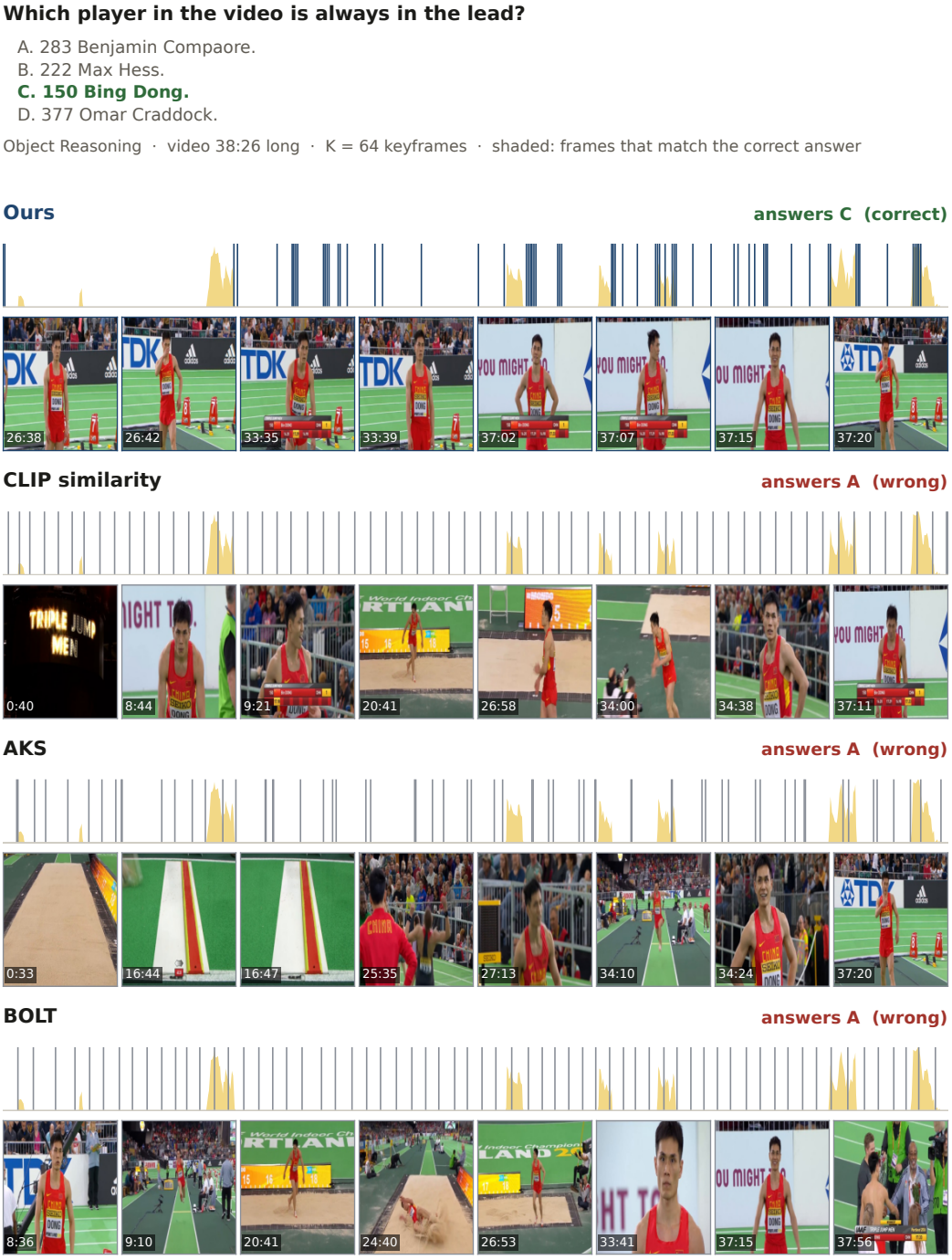


Figure 6: **Which runner is always in the lead?** A 38-minute athletics broadcast that also covers the jumping events. The best-matching keyframes of CSSI all show the leading runner, Bing Dong, on the track or in the race leaderboard, and its keyframes follow the shaded evidence from the 26th minute onward. CLIP similarity, AKS, and BOLT spread their budget over the whole broadcast, their best matches include the long-jump pit and the triple jump, and all three answer A.

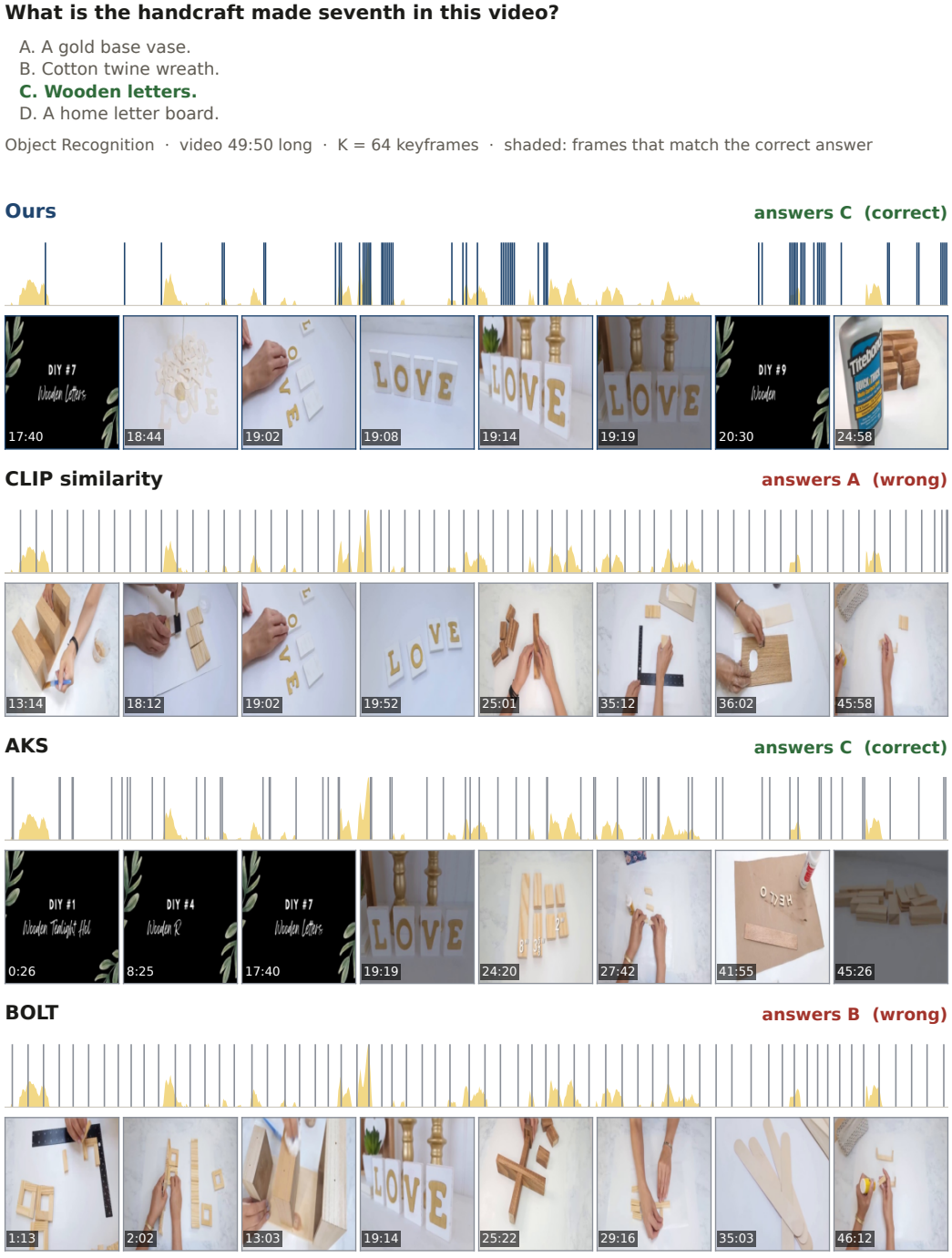


Figure 7: **Which handcraft is made seventh?** A 50-minute compilation of DIY projects. CSSI promotes the title card "DIY #7 Wooden Letters" at 17:40 together with the wooden letters being assembled between 18:44 and 19:19, which together identify the seventh project. CLIP similarity and BOLT reach the letters in single frames, but none of their best-matching keyframes shows the title card that numbers the project, and they answer A and B. AKS also promotes the title card and answers correctly.

Which of the following steps has the most scenarios during the demonstration according to the video?

A. Solving the White Corners.
B. Solving the Corners.
C. Solving the Middle Layer.
D. Solving the Yellow.

Object Reasoning · video 40:28 long · K = 64 keyframes · shaded: frames that match the correct answer



Figure 8: **Which step has the most scenarios?** A 40-minute Rubik's cube tutorial. CSSI promotes the "Step 4: Solving the Yellow" caption at 25:54 and a run of on-screen scenario overlays with their move sequences between 36:09 and 36:46. CLIP similarity and BOLT sample the tutorial almost evenly and answer B. AKS also reaches the scenario overlays and answers correctly.

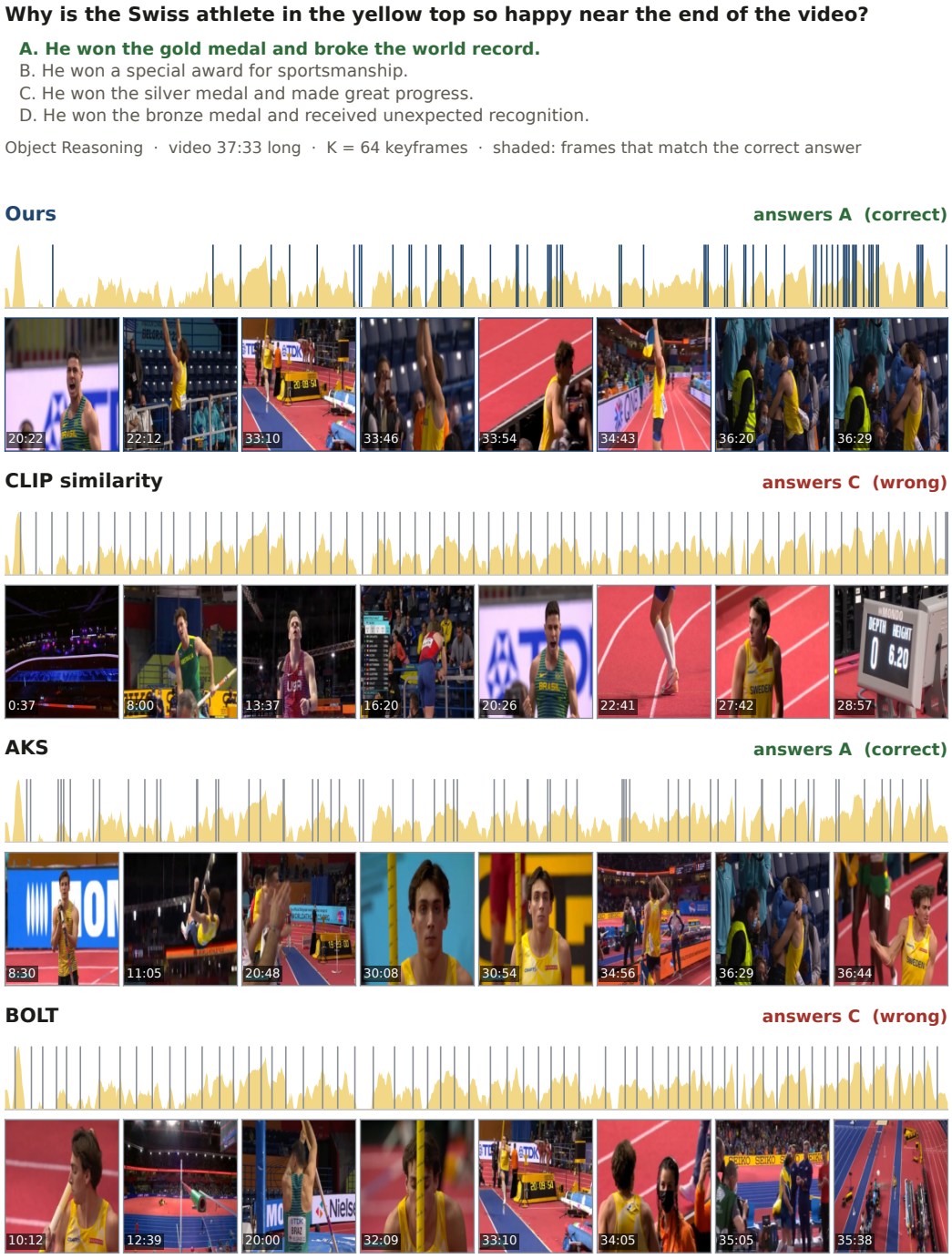


Figure 9: **Why is the athlete in the yellow top so happy near the end?** A 38-minute pole-vault broadcast. The question points to the end of the video, and CSSI places its densest cluster in the last five minutes, covering the final attempt and the celebration that follows. CLIP similarity and BOLT spread their keyframes evenly and answer C. AKS also covers the celebration and answers correctly.