



Temporal Self-Distillation: Learning Visual State Tracking in Videos Without Supervision

Anonymous WACV Algorithms Track submission

Paper ID ****

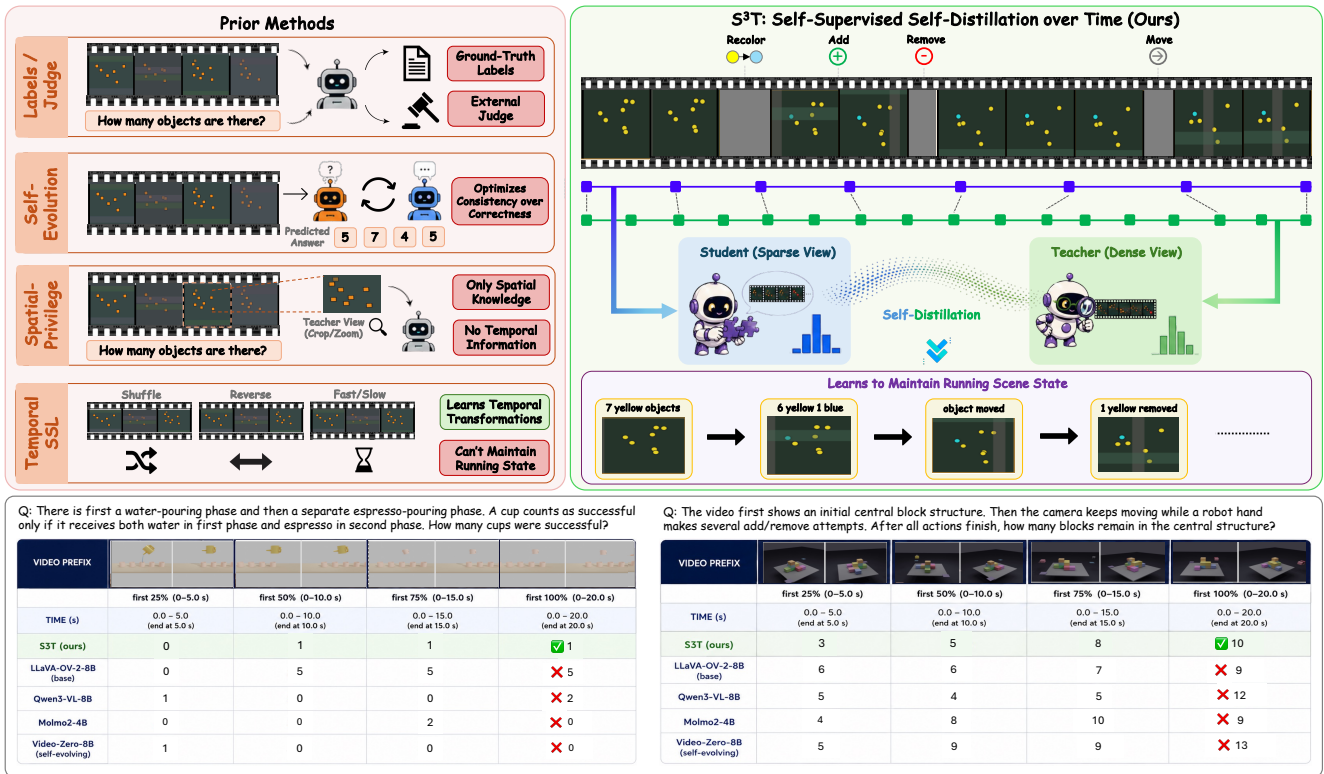


Figure 1. S^3T uses temporal sampling density as self-supervision for visual state tracking. *Prior methods* (top left) either rely on external supervision or do not directly learn this capability. Reinforcement learning needs labels or an external judge, self-evolving methods reward consistency rather than correctness, frame-based privileged-view distillation uses spatial crops without temporal information, and temporal self-supervision focuses on shuffle, reversal, or speed rather than running state. S^3T (top right) instead distills a denser-view teacher into a sparse-view student with the same frozen weights, using only a self-generated target and no labels, judge, or separate model, so it learns to maintain counts and scene states as objects are added, removed, or recoloured. *Qualitative examples* (bottom) test this on two VSTAT [57] (CC BY 4.0) clips by asking each item’s question independently on the first 25, 50, 75, and 100% of the video: S^3T converges on the correct running state early and holds it, while baselines keep revising their answer as more of the clip is revealed.

Abstract

001 Video Large Language Models (Video-LLMs) answer com-
002 plex questions about dynamic scenes but still struggle to
003 maintain visual state over time. Accurate counting and state

tracking require accumulating evidence as objects appear,
disappear, move, or become occluded. Existing methods
either do not explicitly optimize this capability or depend
on external supervision like ground-truth answers or better-
capable bigger models. We introduce S^3T (Self-Supervised

004
005
006
007
008

Self-Distillation over Time), which, to the best of our knowledge, is the first fully self-contained framework for continuous state tracking in video. Our method treats temporal sampling density as privileged information, based on the hypothesis that a denser view of the same clip provides a more reliable estimate of the running state. This view serves as the teacher, while a sparse-view student with the same frozen weights learns to match its next-token distribution. The target is generated by the model itself, so training requires no labels, separate teacher, or reward signal, and adds no inference cost. On LLaVA-OneVision-2-8B, S³T improves VSTAT by +2.70, while prior self-evolving methods leave state tracking largely unchanged. The capability learned from unlabeled synthetic clips transfers to real videos, improving performance by +7.95 on web-video counting questions and +4.50 on MVBench Action Count. Our code and models will be released.

1. Introduction

Recent progress in Video Large Language Models (Video-LLMs) models has improved multimodal reasoning and enabled complex question answering over dynamic visual content [9, 48, 61, 62]. However, visual state tracking remains a major limitation, as models must maintain accurate running counts and scene states while objects are added, removed, moved, or occluded over time [7, 54]. VSTAT [57] is a benchmark designed to test this cumulative state reasoning, and it shows that human performance reaches 90.5%, while the strongest open-source models score only 34–35%. A likely reason is that existing training objectives, whether supervised fine-tuning for single-answer correctness [29, 34, 35, 46] or contrastive alignment of video–text pairs [15, 18, 53, 55], do not require models to accumulate evidence across the full clip. VSTAT also notes that current benchmarks focus mainly on action recognition and moment localization, leaving cumulative state tracking weakly represented in the training signal [3, 47, 56].

Reinforcement learning with verifiable rewards improves video reasoning but depends on ground-truth labels or expensive judge models [31, 32, 60, 65]. Self-distillation aims to reduce this dependence by using model’s own outputs as supervision [13, 43]. However, the only prior self-distillation method for video reasoning, VISD, still requires ground-truth labels and a judge model [22]. Recent self-evolving frameworks [12, 14, 59, 63] use a model’s own outputs through questioner–solver co-evolution or proposer–solver agent coupling, but target general reasoning or temporal grounding, which focuses on event localization rather than maintaining a running visual state.

We present S³T (Self-Supervised Self-Distillation over Time), the first fully self-contained self-distillation framework for improving continuous state tracking in videos

without external supervision. The central idea is to use temporal sampling density as privileged information. We hypothesize that a denser sampling of the same sequence provides more evidence about the running scene state, such as object counts through occlusion, than a sparse sampling, while requiring no labels or annotations. Framed under a learning-under-privileged-information (LUPI) [25, 42] perspective, S³T uses this denser view as a teacher and distills its next-token distribution into the sparse-view student, using only the model’s self-generated outputs. Our method requires no labels, separate teacher network, reward model, or tracker (see Fig. 1). S³T improves VSTAT accuracy from 34.74 to 37.44 (+2.70), with gains concentrated on cumulative-state axes, including Count (+4.9), Atomic (+5.1), and Sequence (+5.8). These gains transfer to real videos, improving VSTAT-YouTube cumulative-state questions by +7.95 and MVBench Action Count by +4.50, while maintaining general video-understanding performance.

To summarize, our contributions are:

- We introduce temporal sampling density as privileged information for self-distillation in Video-LLMs, enabling supervision from unlabeled videos without external annotations or reward models.
- We propose S³T, the first fully self-contained self-distillation framework for Video-LLMs that improves cumulative-state reasoning while preserving general video-understanding performance.
- We show the capability learned from unlabelled synthetic clips transfers to real video, improving cumulative-state questions on web video by +7.95 and MVBench Action Count by +4.50, without retraining on real data.

2. Related Work

Video Understanding and State Tracking. Video-LLMs have progressed from early cross-modal alignment methods [1, 5, 27, 38] to recent systems [23, 28, 37, 64] that extend large language models to video through frame sampling and temporal aggregation. These models support open-ended dialogue [24, 36] and temporal reasoning [20, 21], but are primarily trained for action recognition, moment localization, and video question answering [51, 52]. These tasks generally require identifying the frames that contain the answer rather than maintaining a running scene state throughout a video. Recent benchmarks such as VSTAT [57] make this limitation explicit, with current open-source VidLLMs achieving only 34–35% on state tracking compared to about 90.5% for humans. Temporal self-supervised objectives such as frame-order prediction [16, 30], arrow-of-time classification [33, 49], pace estimation [6, 45], and masked video modeling [41] learn temporal structure from chronology, motion, or missing frames, but they do not require maintaining running counts or evol-

ing scene states over long sequences and occlusions. **Self-Distillation and Self-Evolution.** Visual representations can be learned without human annotations using self-generated supervisory signals or privileged information, where additional information is available during training but not inference [39, 40, 44]. In image understanding, CVPD [43] and Vision-OPD [58] exploit privileged spatial views, such as high-resolution crops or zooms, to improve fine-grained recognition. These methods leverage spatial redundancy but do not address temporal accumulation across frames. Recent self-evolving methods improve Video-LLMs using model-generated supervision. Video-Zero [63], EvoGround [14], and CurEvo [59] optimize objectives for question difficulty, answer consistency, evidence discovery, or temporal grounding rather than cumulative-state tracking. Their self-generated signals reward identifying relevant moments or producing consistent answers, but do not require integrating evidence across an entire video to maintain a running scene state. VISD [22] is closer to video self-distillation, but relies on ground-truth answers, spatio-temporal grounding annotations, and an external video-aware judge. Thus, to our knowledge, no prior method directly targets cumulative-state tracking while remaining fully self-contained.

3. Method

Here we present the proposed framework, starting with the data, two temporal views, and the distillation objective. S³T samples each unlabeled clip at two frame densities and lets the denser view teach the sparser one. Both views pass through the same frozen model and learn from self-generated targets, requiring no labels, separate teacher, reward model, or tracker, with no inference-time cost.

3.1. Data

We train on a fixed set of 300 unlabeled clips from *StateGen*, our procedural video generator. Each clip is a 512×512 video at 12 FPS and lasts about 16 seconds, giving $T = 192$ frames. Scenes contain 3–11 near-identical entities on a textured background and evolve through 3–11 non-overlapping events sampled from a fixed set of add, remove, move, swap, and recolour operations. Events are at least 18 frames apart, while occlusion and global camera motion continue throughout the clip (Fig. 2). Because entities can appear and disappear over time, recovering the correct scene state requires information from the full sequence. The clips contain no state annotations, question-answer pairs, or captions. The model only receives RGB frames, and all supervision comes from its own generated outputs (§3.2). We use procedurally generated videos because temporal-density distillation needs many clips where the observed scene state changes with sampling density, which would be expensive to collect and annotate at scale.

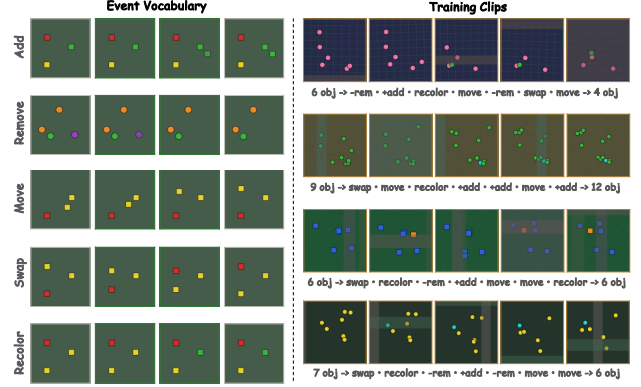


Figure 2. **Training data generated by *StateGen*.** *Left*: The five event types: add, remove, move, swap, and recolour. *Right*: Four example training clips shown at five uniformly spaced time points. Scenes contain near-identical objects undergoing these events under occlusion and global camera motion. Object counts below each strip are for visualization only and are not used in training.

StateGen randomly samples entity counts, event counts, and event timings to create clips with different levels of difficulty (sampling details given in §4). Although training uses only synthetic videos, S³T transfers to real videos and improves cumulative-state tracking on the real-world evaluation split (§4.4).

3.2. Temporal Self-Distillation

Let f_θ be the frozen base model with parameters θ , and $f_{\theta+\phi}$ the same model with a trainable low-rank adapter ϕ , so that $f_{\theta+\phi}(\cdot | x, k)$ denotes its next-token distributions from k frames of clip x .

Two temporal views. For a clip x decoded to T native frames, a view is a uniform set of k frame indices spanning the whole clip. The student view samples $k_s = 12$ frames and the teacher view samples $k_t = 24$ frames over the same span, so the only difference is temporal density: both cover $[0, T - 1]$ with no crop, window, or sub-span. With more frames, the teacher has a more complete view of how the scene changes over time and can better recover its cumulative state. We treat these additional frames as privileged information: they are available only to the teacher during training, while the student sees the sparse view and carries the gradient (LUP; [25, 42]). The student thus learns to recover this readout from fewer frames, and §4 shows that it transfers to the frame budget used at evaluation. Teacher and student use the same weights $f_{\theta+\phi}$ and differ only in the number of input frames; they are not separate networks. The student indices are also not a subset of the teacher’s. The two uniform grids over $[0, T - 1]$ share only their endpoints, so the teacher sees additional temporal evidence rather than a deterministic superset of the student’s frames. The goal is to make the teacher more informative

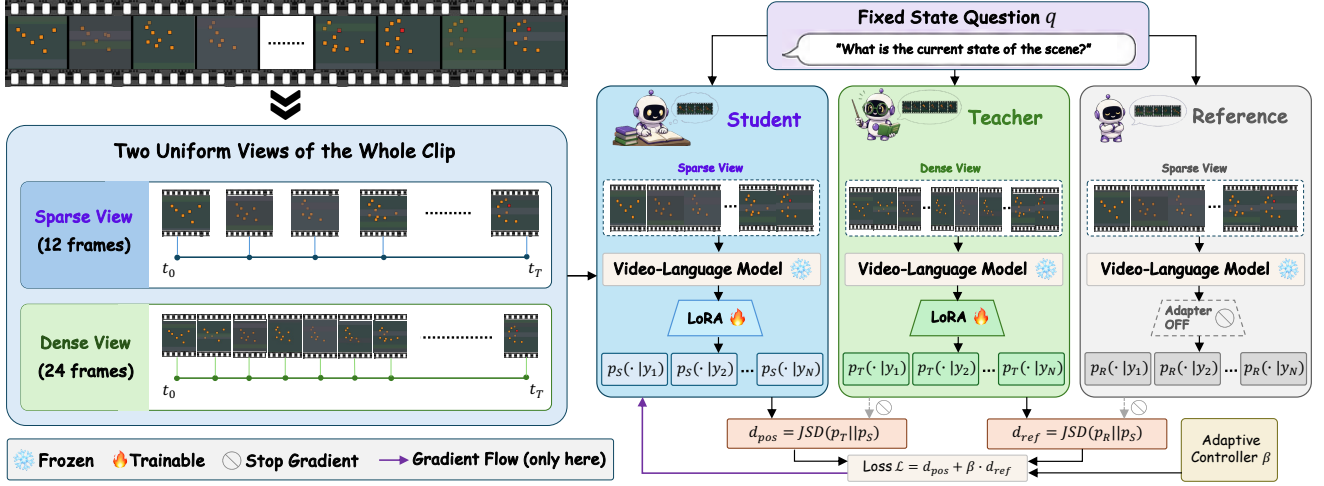


Figure 3. **Overview of S^3T .** (Left) We uniformly sample the clip at two densities: 12 frames for the sparse view and 24 for the dense view. (Right) A frozen video-language model with a trainable LoRA adapter processes the same fixed state question q in three roles: the *Student* on the sparse view, the *Teacher* on the dense view, and the *Reference* on the sparse view with the adapter disabled. All three score the same self-generated answer, yielding next-token distributions p_S, p_T, p_R . The loss $\mathcal{L} = d_{\text{pos}} + \beta d_{\text{ref}}$ pulls the Student toward the Teacher via $d_{\text{pos}} = \text{JSD}(p_T || p_S)$, while $d_{\text{ref}} = \text{JSD}(p_R || p_S)$ anchors it to the base model, with β set by an adaptive controller. Gradients flow only through the Student and only the adapter is trained, so the extra teacher frames are used during training but not at inference.

while keeping its readout recoverable from the sparse student view. We study this trade-off in §4.5.

On-policy, self-generated target. We use a single fixed probe question q for every clip and view; the model does not generate it, and it is not derived from the clip. It names the domain objects and asks for a short description of the current state, including which balls are visible, their colors, and their count. The model generates only the answer. Because q is shared across views, the frame budget is the only difference between the teacher and student passes, making their answer distributions comparable. Our strongest model soups [50] two runs trained with different probes, one asking for the full visible state and the other asking a single question about what changed. Together, they produce complementary models that improve performance across question types. Given q , the student greedily decodes its own answer on its own sparse view,

$$a = \arg \text{decode } f_{\theta+\phi}(\cdot | x, k_s, q), \quad (1)$$

and this single string a becomes the shared teacher-forcing target for every view. Let $T_a = |a|$ denote its length. For a view at k frames we read the next-token distributions at exactly the T_a positions that predict the tokens of a ,

$$\begin{aligned} P^{(k)} &= (p_1^{(k)}, \dots, p_{T_a}^{(k)}), \\ p_j^{(k)} &= \text{softmax } f_{\theta+\phi}(\cdot | x, k, q, a_{<j}). \end{aligned} \quad (2)$$

All views are therefore scored on the same answer region, position by position. Targeting the student’s own greedy

rollout keeps the objective on-policy because the aligned tokens are exactly those the student emits at inference, making the distilled readout reachable from the sparse policy. **Objective.** The model is trained to match the teacher’s next-token distribution while remaining anchored to the base, and both terms use the same token-averaged JSD over the answer region,

$$\begin{aligned} D_{\text{JSD}}(P || Q) &= \frac{1}{T_a} \sum_{j=1}^{T_a} \left[\frac{1}{2} D_{\text{KL}}(p_j || m_j) + \frac{1}{2} D_{\text{KL}}(q_j || m_j) \right], \\ m_j &= \frac{1}{2} (p_j + q_j), \end{aligned} \quad (3)$$

computed in bits ($\log_2$), so each position contributes at most one bit. We use JSD rather than KL because it is symmetric and bounded on $[0, 1]$, so the objective and its gradient stay controlled even when the moving teacher and student momentarily diverge on some tokens. Holding the teacher-student gap fixed, forward and reverse KL yield a roughly $3.8\times$ larger and markedly spikier objective, and a $\sim 3.8\times$ larger gradient norm (see Fig. 4). The distillation term moves the student toward the teacher’s reading,

$$d_{\text{pos}} = D_{\text{JSD}}(\text{sg}[P^{(k_t)}] || P^{(k_s)}), \quad (4)$$

where $\text{sg}[\cdot]$ is the stop-gradient, so with the teacher view at $k_t = 24$ frames detached the gradient trains only the student view at $k_s = 12$ frames toward it. The anchor keeps the adapted student within a bounded distance of the base on identical input,

$$\begin{aligned} d_{\text{ref}} &= D_{\text{JSD}}(\text{sg}[P^{\text{ref}}] || P^{(k_s)}), \\ P^{\text{ref}} &= f_{\theta}(\cdot | x, k_s, q, a), \end{aligned} \quad (5)$$

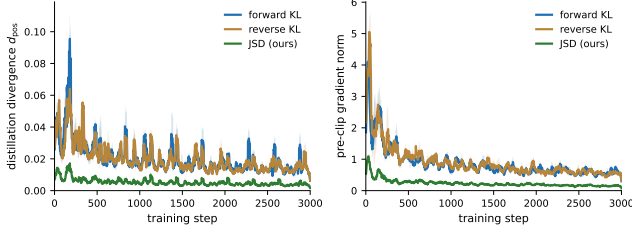


Figure 4. **JSD gives a bounded and smoother objective than KL for the same teacher–student gap.** We vary only the distillation divergence and plot d_{pos} (left) and pre-clip gradient norm (right) over full training for JSD, forward KL, and reverse KL, averaged over three seeds. The teacher–student gap is comparable across runs, with the JSD monitor staying near 0.005, so the differences come from the divergence choice. Forward and reverse KL turn the same gap into an objective that is about $3.9\times$ larger and much spikier, with an average pre-clip gradient norm about $3.9\times$ larger as well. Both curves use seed-averaged rolling means. Since gradient clipping keeps all runs stable, the difference lies in objective boundedness and smoothness rather than overall training stability.

where P^{ref} is the base distribution on the student’s own sparse view, obtained by disabling ϕ to recover f_{θ} . Because d_{ref} is a JSD to the base rather than a KL, it inherits the same boundedness. The full objective is

$$\mathcal{L} = d_{\text{pos}} + \beta d_{\text{ref}} \quad (6)$$

with d_{pos} at unit weight and β weighting the anchor.

A fixed β is sensitive to the chosen value. If it is too small, the adapter drifts from the base model and loses general calibration; if it is too large, the adapter fails to absorb the teacher’s readout. Hence, we adapt β during training so that d_{ref} stays near a target κ ,

$$\begin{aligned} \beta &\leftarrow \text{clip}\left(\beta \cdot \exp(\eta \cdot \text{clip}(\delta, -3, 3)), \beta_{\min}, \beta_{\max}\right), \\ \delta &= \frac{d_{\text{ref}} - \kappa}{\kappa}, \end{aligned} \quad (7)$$

at adaptation rate η . When the student drifts past the set point, so that $d_{\text{ref}} > \kappa$, the weight grows and pulls it back, and when the student is well anchored it relaxes; the inner clip caps the per-step move so a single d_{ref} spike cannot drive β to its bound, and the outer clip keeps it within $[\beta_{\min}, \beta_{\max}]$. Thus, the anchoring budget is a controlled quantity rather than a hand-tuned coefficient. Algorithm 1 summarizes one complete S³T training step.

4. Experiments and Analyses

Implementation Details. We finetune LLaVA-OV-2-8B [2] using LoRA [11]. The base model weights and vision encoder are kept frozen, and only the adapter is updated. Our strongest configuration also adapts the vision encoder,

Algorithm 1 One S³T training step

Require: frozen base f_{θ} , adapter ϕ , probe q , frame counts $k_s=12$, $k_t=24$, anchor set point κ , rate η , bounds $[\beta_{\min}, \beta_{\max}]$, anchor weight β

- 1: sample clip x ; decode to T frames
- 2: $V_s \leftarrow k_s$ uniform frames of x ▷ student view
- 3: $V_t \leftarrow k_t$ uniform frames of x ▷ same span, denser
- 4: $a \leftarrow$ greedy decode $f_{\theta+\phi}(\cdot | V_s, q)$ ▷ self-generated target
- 5: $P^{(k_t)} \leftarrow f_{\theta+\phi}(\cdot | V_t, q, a)$, no grad ▷ teacher
- 6: $P^{\text{ref}} \leftarrow f_{\theta}(\cdot | V_s, q, a)$, no grad ▷ adapter off
- 7: $P^{(k_s)} \leftarrow f_{\theta+\phi}(\cdot | V_s, q, a)$ ▷ student
- 8: $d_{\text{pos}} \leftarrow D_{\text{JSD}}(\text{sg}[P^{(k_t)}] \| P^{(k_s)})$
- 9: $d_{\text{ref}} \leftarrow D_{\text{JSD}}(\text{sg}[P^{\text{ref}}] \| P^{(k_s)})$
- 10: $\mathcal{L} \leftarrow d_{\text{pos}} + \beta d_{\text{ref}}$
- 11: back-propagate $\mathcal{L}$ through ϕ only; clip; step
- 12: $\delta \leftarrow (d_{\text{ref}} - \kappa)/\kappa$
- 13: $\beta \leftarrow \text{clip}(\beta e^{\eta \cdot \text{clip}(\delta, -3, 3)}, \beta_{\min}, \beta_{\max})$ ▷ anchor

which we report as a separate row throughout. The adapter is applied to the language model’s attention and feed-forward projections with rank $r=32$, $\alpha=64$, and dropout 0.05; the vision-to-language projector is never adapted, so a frame becomes the same tokens in every view and the views differ only in frame count. We explain why the decoder is the right place for this capacity at the end of §4.5. We train for 3000 steps with one clip per step using AdamW [26], weight decay 0.01, gradient clipping at 1.0, and a learning rate of 1.5×10^{-5} , on 4×A100 GPUs in bfloat16. The student view samples 12 frames, while the teacher view samples 24 frames from the same clip. The anchor set point is $\kappa=0.06$ with adaptation rate $\eta=0.05$, and its weight β starts at 0.05 and is clipped to $[10^{-3}, 5]$.

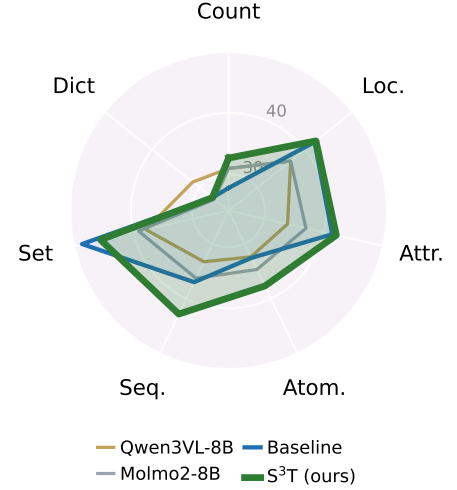
Baselines and Evaluation. We evaluate on VSTAT [57], a visual state-tracking benchmark with 1,500 questions: 536 multiple-choice questions and 964 numerical questions. It includes both procedurally generated and real videos, and its questions cannot be answered from a single keyframe or from the last frame alone. We report the benchmark metric, defined as the average of multiple-choice accuracy and mean relative accuracy on numerical questions. Each question also includes a state element label (Count, Location, Attribute) and a state structure label (Atomic, Sequence, Set, Dictionary), which indicate the aspects of state tracking being evaluated. We compare against a broad set of open-source video models, self-evolving methods, and established temporal self-supervised objectives. All models are evaluated under the same protocol for fair comparison.

4.1. Main Results

Tab. 1a compares S³T against open-source video models, prior self-evolving methods, and our own reproduction of the baseline(s). Visual state tracking improves with no supervision beyond the model itself; our baseline improves

Method	State Element				State Structure			
	Avg	Count	Loc.	Attr.	Atom.	Seq.	Set	Dict
Human Performance [57]	90.5	92.8	89.9	86.4	93.7	77.5	90.0	92.4
Open-source Models								
LLaVA-OV-2-8B [2]	35.1	28.3	43.0	40.5	33.5	38.7	46.9	27.3
LLaVA-OV-2-8B (codec) [2]	35.0	28.6	42.0	40.6	33.9	37.0	46.3	27.6
Molmo2-4B [8]	34.4	31.6	39.7	34.5	37.1	33.6	36.7	27.1
Cambrian-S-7B [56]	34.2	33.2	33.6	36.9	34.0	30.6	40.2	32.5
Molmo2-8B [8]	34.0	30.9	37.0	37.0	34.7	36.3	39.1	27.0
Qwen3VL-8B [3]	33.2	30.9	37.0	33.9	32.4	33.3	37.9	31.5
InternVL3.5-2B [47]	31.8	29.6	33.9	34.1	31.7	29.9	36.3	29.9
Cambrian-S-3B [56]	31.8	29.7	32.7	35.0	32.7	31.9	35.1	27.2
VITA-1.5-7B [10]	31.5	25.5	36.3	38.6	29.4	33.0	43.1	26.3
Qwen3VL-4B [3]	31.3	27.0	33.3	37.9	30.4	32.8	39.8	25.8
InternVL3.5-8B [47]	30.6	25.1	33.2	39.2	26.9	33.8	41.8	28.3
Qwen3VL-2B [3]	29.4	29.4	28.2	30.5	32.5	24.9	32.1	23.5
Cambrian-S-1.5B [56]	29.3	26.0	34.1	31.0	28.0	31.0	31.8	29.3
LLaVA-OV-7B [17]	28.6	20.1	34.8	39.4	24.5	30.0	43.8	25.0
LLaVA-OV-0.5B [17]	21.3	14.6	33.9	21.7	19.7	25.8	22.2	20.9
Self-Evolving Methods								
Qwen3-VL-4B [†] [3]	31.43	27.7	33.7	36.6	31.9	32.5	38.5	24.4
Video-Zero-4B [†] [63]	31.63	27.7	34.5	36.5	32.1	32.3	37.8	25.5
Qwen3-VL-8B [†] [3]	34.11	32.6	37.8	33.4	35.0	33.1	37.1	30.4
Video-Zero-8B [†] [63]	33.89	32.0	37.2	34.3	34.1	35.5	38.3	29.1
Qwen2.5-VL-7B [†] [4]	31.84	25.2	37.4	39.5	28.8	39.4	41.6	26.2
EvoGround [†] [14]	32.66	26.7	36.6	40.6	29.7	39.5	42.9	27.0
Ours								
LLaVA-OV-2-8B [†]	34.74	27.7	42.0	41.4	32.6	37.0	48.6	27.4
S ³ T (SFT teacher)	34.45	26.0	42.2	43.6	30.9	38.0	50.9	27.5
S ³ T	36.48	30.4	41.7	43.3	35.2	39.4	47.8	28.8
S ³ T (soup)	37.12	32.4	40.7	43.0	37.2	41.4	44.9	28.4
S ³ T (soup, + vision enc.)	37.44	32.6	42.3	42.1	37.7	42.8	45.5	27.4

(a) VSTAT leaderboard.



(b) Per-axis profile against the size-matched 8B open models.

Training objective	Avg	Count	Dict
Base (no training)	34.74	27.7	27.4
Frame-order [16, 30]	34.98	27.7	30.5
Arrow-of-time [33, 49]	34.75	27.4	26.5
Pace [6, 45]	35.45	29.0	27.1
Mask-infill [41]	34.25	25.2	26.3
S ³ T (ours)	36.48	30.4	28.8

(c) Established temporal pretexts.

Table 1. **Performance comparison of S³T variants vs prior methods on VSTAT.** (a) Scores are VSTAT accuracy (%) following the protocol given by [57]. [†] marks scores we reproduced. Our reproduction of the LLaVA-OV-2-8B base gives 34.74, compared with the benchmark’s reported 35.1 for the same model, and all deltas are computed from our reproduced score. Each self-evolving method is listed directly under the base model it evolves from, and both are measured under our protocol so the two are directly comparable. Per column, best and second are shaded. (b) The improvement is strongest on the cumulative-state axes, such as Count, Atomic, and Sequence, while performance on the binding-oriented axes is largely preserved. (c) Comparison between four established temporal self-supervised objectives, trained with the same recipe while changing only the target. The S³T row is the single model from (a) (36.48) and not either soup, since the pretext baselines are unsouped and a single model is the like-for-like comparison. Only S³T visibly improves overall.

from 34.74 to 37.44, a gain of +2.70. This gain does not depend on souping; a single S³T run already scores above every open-model result in the table. Runs trained with different state-probe questions come out strong on different axes, so we soup two such runs, and adapting the vision encoder adds a little more improvement. The closest comparisons are the prior self-evolving methods, since they also train on model-generated outputs rather than external labels. Under our protocol, however, they produce at most marginal gains over their respective base models, and sometimes regress. S³T is the only method that produces a clear improvement.

The per-axis breakdown shows that the gains are concentrated on the cumulative-state axes where answering requires integrating evidence across the entire clip. Accord-

ingly, S³T variants achieve the top two overall scores as well as the top two scores on Atomic and Sequence. Performance on the binding-oriented axes changes little overall, suggesting that S³T improves cumulative-state reasoning rather than general visual reasoning. The radar plot (Fig. 1b) shows the same trend against models of matching 8B size. Supervised finetuning (SFT) helps isolate the effect of the distillation objective. It uses the same dense teacher view but fine-tunes on the teacher’s generated text instead of its predictive distribution. Although it achieves the best Attribute and Set scores, it remains below the base model overall. It shows that improving individual axes alone is not sufficient, and that the gains of S³T arise from better cumulative-state tracking (we return to this in §4.3).

Table 2. **Component ablations of S³T.** Each block changes one component of the method. Scores are VSTAT accuracy (%) overall and on the Count and Dictionary axes. Rows marked (ours) are the setting we use. Within each block, **best** and **second** are shaded per column; blocks are separate comparisons and are not ranked against each other.

Setting	Overall	Count	Dict
<i>Teacher view</i>			
Temporal stretch 12 → 24 (ours)	36.48	30.4	28.8
Temporal zoom (windowed)	33.91	26.1	25.2
Event-aware	35.11	28.5	27.5
Multi-scale {16, 32, 48}	36.23	29.9	28.1
Multi-scale {12, 24, 48}	36.04	29.3	28.0
Adaptive (router)	36.26	30.9	27.9
Adaptive (budget-controlled)	36.49	30.4	27.7
Identical 12 → 12 (no privileged view)	34.83	27.7	27.6
<i>Objective</i>			
JSD distillation (ours)	36.48	30.4	28.8
SFT (cross-entropy on teacher text)	34.45	26.0	27.5
JSD to generator’s true state	33.90	27.5	29.2
The distillation divergence (JSD vs. forward/reverse KL) is ablated in Fig. 4.			
<i>Anchor</i>			
Reference-JSD anchor (ours)	36.48	30.4	28.8
No anchor ($\beta=0$)	33.68	27.3	27.7
<i>Frame ratio</i>			
12 → 24 (our ratio)	36.48	30.4	28.8
12 → 36	35.56	27.6	27.8
12 → 48	34.67	27.1	27.5
8 → 32	33.81	28.6	26.7
24 → 48	32.63	26.5	27.5

4.2. Comparison with Temporal Pretexts

Tab. 1c compares S³T against four established temporal self-supervised objectives. *Frame-order prediction* [16, 30] asks the model to recover the correct order of shuffled clip segments, *arrow-of-time* [33, 49] predicts whether a clip is played forward or backward, *pace prediction* [6, 45] identifies the playback speed, and *masked temporal infill* [41] predicts the content of a masked temporal span. All are trained using the same S³T recipe on the same base model and configurations while changing only the learning objective. Appendix A2 provides the complete definition of each pretext. None of the four matches the gains of S³T. *Pace prediction* performs best among them, consistent with its similarity to temporal-density supervision, but recovers less than half of the improvement, while *masked temporal infill* falls below the baseline. The clearest difference appears on Count. S³T is the only objective that improves it, while the remaining pretexts leave it unchanged or reduce it. *Frame-order prediction* is the only exception on a single axis, achieving the best Dictionary score without improving Count or the overall score. Thus, we can conclude that the gains arise from temporal-density distillation rather than

Table 3. **S³T on real video, where a running tally is needed.** Change in accuracy (%) over our base. The first row is the 185 web-video (VSTAT YouTube [57]) questions that require a running tally; the second is the one MVBench [19] question type that does. *soup* is the frozen-encoder model and *+vis.* also adapts the vision encoder.

Real-video benchmark	S ³ T (soup)	S ³ T (soup, +vis.)
VSTAT-YouTube, cumulative-state	+7.35	+7.95
MVBench, Action Count	+3.00	+4.50

temporal self-supervision alone.

4.3. Ablations

Teacher View. The first block of Tab. 2 fixes the 12-frame student and changes only the teacher view. We compare a windowed zoom, an event-aware teacher, two multi-scale teachers, two adaptive teachers, and an identical 12 → 12 control. None outperforms the fixed 12 → 24 teacher. The windowed zoom performs worst, particularly on Count, showing that full temporal coverage is more important than concentrating frames within a short window. Event-aware, multi-scale, and adaptive teachers provide no consistent advantage over the fixed view. The identical 12 → 12 control removes only the privileged frames while keeping the objective unchanged, causing performance to fall back to roughly the base model. This shows that the gain comes from the additional teacher frames rather than self-distillation alone.

Objective and Anchor. Both supervised alternatives perform below the base model. Fine-tuning on the teacher’s generated text reduces both the overall score and Count, while distilling toward the generator’s true state performs worse despite using the same divergence and soft targets. The latter achieves the best Dictionary score, but this does not translate into better cumulative-state tracking. The choice of divergence is evaluated separately in Fig. 4. The reference anchor is also essential; removing it ($\beta=0$) drops performance below the base, showing that the teacher provides the supervision while the anchor stabilizes adaptation.

Frame Ratio. The final block varies the student–teacher frame ratio. Increasing the gap from 12 → 24 to 12 → 36 and 12 → 48 steadily reduces performance, while changing the student’s sampling budget is more detrimental still. Among the tested settings, the fixed 2× ratio performs best. §4.5 explains why moving in either direction hurts performance.

4.4. Generalization to Real Video

S³T is trained only on synthetic clips, so we evaluate whether the learned capability transfers to real videos. Tab. 3 reports the improvement over our base. VSTAT’s web-video benchmark [57] contains multiple source tasks. Before evaluation, we assigned each task to the cumulative-state group based only on its description. This group

Table 4. **Five competing explanations for the gain and the measurement that rules out each one.** Scores are VSTAT accuracy (%) over all 1,500 questions unless stated otherwise. The base model scores 34.74. *Rate of change* groups clips by state changes per second. “Slow” clips contain few changes, while “fast” clips contain frequent object motion, appearances, or disappearances. *Count error* is the model’s reported count minus the true count, averaged over counting questions, so a value nearer 0 means the count itself is more accurate rather than the answer merely being phrased differently.

Could the gain instead be...	Measurement that rules it out
... sensitivity to fast motion?	It is largest where the scene changes <i>least</i> : +6.84 on the slowest-changing clips against +0.79 on the fastest
... simply reading more frames at test time?	S ³ T on 32 frames (34.93) already beats the base on 64 (34.74)
... extra spatial detail rather than time?	A teacher magnifying a short window scores 33.91, against 35.95 for one that samples the whole clip more densely
... self-distillation on its own?	A teacher given the student’s own 12 frames returns 34.83, i.e. near the base model’s performance
... a different answer style, not better counting?	The count error shrinks toward zero, $-1.59 \rightarrow -1.31$

contains 185 questions from assembly, pouring, nesting, and counting. On these questions, the vision-adapted model improves by +7.95 and the frozen-encoder model by +7.35, the largest gains we observe. The same pattern appears on external benchmarks, where Action Count on MVBench [19] improves by +4.50. These results show that the capability learned from synthetic clips transfers to real videos when cumulative-state reasoning is required, while leaving general video understanding largely unchanged (additional results are provided in Appendix A6). We also study generalization across baselines in Appendix A7.

4.5. Discussion

A benchmark gain of this size could have several explanations unrelated to cumulative-state tracking. So we test five of them before offering any account of the method, as listed in Tab. 4. The clearest is motion. If S³T simply made the model better at fast-moving video, the gain should be largest where the scene changes most; it is the reverse, +6.84 where the scene changes least against +0.79 where it changes most. The gain also does not come from reading more frames at test time, since S³T on 32 frames already beats the base on 64. It is not extra spatial detail either: a teacher that magnifies a short window, instead of covering the whole clip more densely, scores 2.04 points lower than one that does. Nor is it the act of self-distillation on its own, because a teacher restricted to the student’s own 12 frames hardly improves the model from its baseline (initial) performance. The model is indeed counting better and not just answering differently, since its average count moves closer to the true count.

S³T is an instance of learning under privileged information [25, 42]: the denser view exists during training but not at inference. The teacher has to show the student something it cannot already see, while staying close enough to be imitated, and Fig. 5 shows both limits on the two axes that set

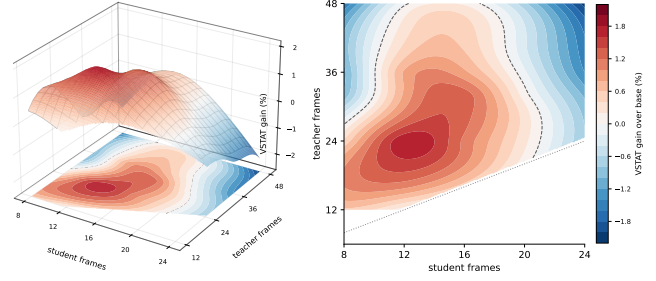


Figure 5. **The gain depends on both frame budgets.** Each surface is fitted to training runs that share our base, our data and our schedule, and differ only in how many frames the student and the teacher receive; the dotted line marks teacher = student, below which the teacher holds no extra frames. *Left*: the same quantity as a surface, with its contours projected onto the floor, so the height of the ridge and its fall-off are visible together. *Right*: the surface viewed from above. Warm colours are gains over the base and cool colours are losses; the dashed line is the zero contour, which separates settings that help from settings that hurt.

them. Along the diagonal the teacher adds no frames and there is nothing to distil, and the identical 12 → 12 control returns +0.09. Moving away from the diagonal helps until the teacher’s reading is too far from the student’s to be followed, after which the gain falls back through zero; the windowed-zoom teacher (Tab. 2) which keeps 24 frames but confines them to a short window, returns −0.83 and shows that coverage rather than frame count is what the student is following. The optimum forms a closed region centred near a 12-frame student and a roughly 24-frame teacher, rather than an open ridge. This means that neither budget can be inferred from the other. The student budget is also more sensitive. Increasing the student above roughly 20 frames crosses the zero contour for *every* teacher we tested. In contrast, with a 12-frame student, the gain remains positive across a wide range of teacher budgets.

5. Conclusion

We introduced S³T, the first fully self-contained framework for improving continuous visual state tracking in video-language models without any supervision. S³T treats a denser temporal sampling of the same clip as privileged information and distills its predictions into a sparse view of the same model, requiring no labels, external teacher, reward model, or tracker. It improves cumulative-state reasoning on VSTAT while preserving general video understanding and transfers to real videos on running-tally tasks. These gains persist under sparse inference and concentrate on cumulative-state tasks, indicating better state maintenance over time. More broadly, our results identify temporal sampling density as an effective source of label-free supervision for learning long-horizon visual state tracking.

6. Acknowledgement

References

- [1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikołaj Bińkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Kar'en Simonyan. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35: 23716–23736, 2022. 2
- [2] Xiang An, Yin Xie, Feilong Tang, Yunyao Yan, Huajie Tan, Didi Zhu, Changrui Chen, Xiuwei Zhao, Bin Qin, Kaicheng Yang, Yifei Shen, Yuanhan Zhang, Kaichen Zhang, Wenkang Zhang, Zheng Cheng, Nansen Zhang, Chunsheng Wu, Chunjiang Ge, Zimin Ran, Dehua Song, Chunyuan Li, Shikun Feng, Ming Hu, Zhangquan Chen, Junbo Niu, Bo Li, Ziyong Feng, Ziwei Liu, Zongyuan Ge, and Jiankang Deng. Llava-onevision-2: Towards next-generation perceptual intelligence. *arXiv preprint arXiv:2605.25979*, 2026. 5, 6
- [3] 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, Sibao 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, 2025. 2, 6
- [4] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao 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, 2025. 6
- [5] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1728–1738, 2021. 2
- [6] Sagie Benaim, Ariel Ephrat, Oran Lang, Inbar Mosseri, William T Freeman, Michael Rubinstein, Michal Irani, and Tali Dekel. Speednet: Learning the speediness in videos. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9922–9931, 2020. 2, 6, 7
- [7] Xi Chen, Zuoxin Li, Ye Yuan, Gang Yu, Jianxin Shen, and Donglian Qi. State-aware tracker for real-time video object segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9384–9393, 2020. 2
- [8] Christopher Clark, Jieyu Zhang, Zixian Ma, Jae Sung Park, Rohun Tripathi, Sangho Lee, Mohammadreza Salehi, Jason Ren, Chris Dongjoo Kim, Yinuo Yang, Vincent Shao, Yue Yang, Weikai Huang, Ziqi Gao, Taira Anderson, Jianrui Zhang, Jitesh Jain, George Stoica, Ali Farhadi, and Ranjay Krishna. Molmo2: Open weights and data for vision-language models with video understanding and grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28652–28668, 2026. 6
- [9] Hao Fei, Shengqiong Wu, Wei Ji, Hanwang Zhang, Meishan Zhang, Mong-Li Lee, and Wynne Hsu. Video-of-thought: Step-by-step video reasoning from perception to cognition. *arXiv preprint arXiv:2501.03230*, 2024. 2
- [10] Chaoyou Fu, Haojia Lin, Xiong Wang, Yifan Zhang, Yunhang Shen, Xiaoyu Liu, Haoyu Cao, Zuwei Long, Heting Gao, Ke Li, Long Ma, Xiaowu Zheng, Rongrong Ji, Xing Sun, Caifeng Shan, and Ran He. Vita-1.5: Towards gpt-4o level real-time vision and speech interaction. *Advances in Neural Information Processing Systems*, 38:75300–75320, 2026. 6
- [11] 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. In *International Conference on Learning Representations*, 2022. 5
- [12] Shiqi Huang, Ziyue Wang, Zhongrong Zuo, Han Qiu, Qi She, and Bihan Wen. Evovid: Temporal-centric self-evolution for video large language models. *arXiv preprint arXiv:2605.21931*, 2026. 2
- [13] Jonas Hübner, Frederike L'ubeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, and Andreas Krause. Reinforcement learning via self-distillation. *arXiv preprint arXiv:2601.20802*, 2026. 2
- [14] Minjoon Jung, Byoung-Tak Zhang, and Lorenzo Torresani. Evoground: Self-evolving video agents for video temporal grounding. *arXiv preprint arXiv:2605.13803*, 2026. 2, 3, 6
- [15] Dohwan Ko, Joonmyung Choi, Juyeon Ko, Shinyeong Noh, Kyoung-Woon On, Eun-Sol Kim, and Hyunwoo J Kim. Video-text representation learning via differentiable weak temporal alignment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5016–5025, 2022. 2
- [16] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *Proceedings of the IEEE international conference on computer vision*, pages 667–676, 2017. 2, 6, 7
- [17] 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. *arXiv preprint arXiv:2408.03326*, 2024. 6
- [18] Dongxu Li, Junnan Li, Hongdong Li, Juan Carlos Niebles, and Steven CH Hoi. Align and prompt: Video-and-

- language pre-training with entity prompts. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4953–4963, 2022. 2
- [19] Kunchang Li, Yali Wang, Yanan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Lou, Limin Wang, and Yu Qiao. Mvbench: A comprehensive multi-modal video understanding benchmark. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22195–22206, 2024. 7, 8
- [20] Lei Li, Yuanxin Liu, Linli Yao, Peiyuan Zhang, Chenxin An, Lean Wang, Xu Sun, Lingpeng Kong, and Qi Liu. Temporal reasoning transfer from text to video. In *International Conference on Learning Representations*, pages 74070–74102, 2025. 2
- [21] Ruotong Liao, Max Erler, Huiyu Wang, Guangyao Zhai, Gengyuan Zhang, Yunpu Ma, and Volker Tresp. Videoinsta: Zero-shot long video understanding via informative spatial-temporal reasoning with llms. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 6577–6602, 2024. 2
- [22] Hao Lin, Kunyang Lv, Xu Jiang, Jingqi Tian, Zhongjing Du, Jiayu Ding, Qiaoman Zhang, and Hongbo Jin. Visd: Enhancing video reasoning via structured self-distillation. *arXiv preprint arXiv:2605.06094*, 2026. 2, 3
- [23] Ji Lin, Hongxu Yin, Wei Ping, Pavlo Molchanov, Mohammad Shueybi, and Song Han. Vila: On pre-training for visual language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26689–26699, 2024. 2
- [24] Ye Liu, Zongyang Ma, Zhongang Qi, Yang Wu, Ying Shan, and Chang W Chen. Et bench: Towards open-ended event-level video-language understanding. *Advances in Neural Information Processing Systems*, 37:32076–32110, 2024. 2
- [25] David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information. *arXiv preprint arXiv:1511.03643*, 2015. 2, 3, 8
- [26] Ilya Loshchilov, Frank Hutter, et al. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 5 (5):5, 2017. 5
- [27] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval and captioning. *Neurocomputing*, 508:293–304, 2022. 2
- [28] 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)*, pages 12585–12602, 2024. 2
- [29] Ahmad Mahmood, Ashmal Vayani, Muzammal Naseer, Salman Khan, and Fahad Shahbaz Khan. Vurf: A general-purpose reasoning and self-refinement framework for video understanding. *arXiv preprint arXiv:2403.14743*, 2024. 2
- [30] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *European conference on computer vision*, pages 527–544. Springer, 2016. 2, 6, 7
- [31] Kun Ouyang, Yuanxin Liu, Haoning Wu, Yi Liu, Hao Zhou, Jie Zhou, Fandong Meng, and Xu Sun. Spacer: Reinforcing mllms in video spatial reasoning. *arXiv preprint arXiv:2504.01805*, 2025. 2
- [32] Kun Ouyang, Yuanxin Liu, Linli Yao, Yishuo Cai, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. Conan: Progressive learning to reason like a detective over multi-scale visual evidence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 41089–41099, 2026. 2
- [33] Lyndsey C. Pickup, Zheng Pan, Donglai Wei, YiChang Shih, Changshui Zhang, Andrew Zisserman, Bernhard Scholkopf, and William T. Freeman. Seeing the arrow of time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 2, 6, 7
- [34] Rui Qian, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Shuangrui Ding, Dahua Lin, and Jiaqi Wang. Streaming long video understanding with large language models. *Advances in Neural Information Processing Systems*, 37:119336–119360, 2024. 2
- [35] Hanoona Rasheed, Muhammad Uzair Khattak, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Fine-tuned clip models are efficient video learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6545–6554, 2023. 2
- [36] Shaden Shaar, Bradon Thymes, Sirawut Chaixanien, Claire Cardie, and Bharath Hariharan. Movierecapsqa: A multimodal open-ended video question-answering benchmark. *arXiv preprint arXiv:2601.02536*, 2026. 2
- [37] Enxin Song, Wenhao Chai, Guan hong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, Yan Lu, Jenq-Neng Hwang, and Gaoang Wang. Moviechat: From dense token to sparse memory for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18221–18232, 2024. 2
- [38] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7464–7473, 2019. 2
- [39] Omkar Thawakar, Shravan Venkatraman, Ritesh Thawkar, Abdelrahman Shaker, Hisham Cholakkal, Rao Muhammad Anwer, Salman Khan, and Fahad Khan. Evolmm: Self-evolving large multimodal models with continuous rewards. *arXiv preprint arXiv:2511.16672*, 2025. 3
- [40] Ritesh Thawkar, Shravan Venkatraman, Omkar Thawakar, Abdelrahman Shaker, Fahad Khan, Hisham Cholakkal, Salman Khan, and Rao Muhammad Anwer. Ask, solve, generate: Self-evolving unified multimodal understanding and generation via self-consistency rewards. *arXiv preprint arXiv:2606.27376*, 2026. 3
- [41] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *Advances in neural information processing systems*, 35:10078–10093, 2022. 2, 6, 7

- [42] Vladimir Vapnik and Rauf Izmailov. Learning using privileged information: similarity control and knowledge transfer. *The Journal of Machine Learning Research*, 16(1):2023–2049, 2015. 2, 3, 8
- [43] Shravan Venkatraman, Omkar Thawakar, Ritesh Thawkar, Abdelrahman Shaker, and Rao Muhammad. Perception before supervision: Self-contained visual distillation from counterfactual blind spots. In *37th British Machine Vision Conference 2026, BMVC 2026, Lancaster, UK, November 23-26, 2026*. BMVA, 2026. 2, 3
- [44] Shravan Venkatraman, Ritesh Thawkar, Omkar Thawakar, Rao Muhammad Anwer, Hisham Cholakkal, Salman Khan, and Fahad Khan. Paying more attention to visual tokens in self-evolving large multimodal models. 2026. 3
- [45] Jiangliu Wang, Jianbo Jiao, and Yun-Hui Liu. Self-supervised video representation learning by pace prediction. In *European conference on computer vision*, pages 504–521. Springer, 2020. 2, 6, 7
- [46] Qi Wang, Yanrui Yu, Ye Yuan, Rui Mao, and Tianfei Zhou. Videorf: Incentivizing video reasoning capability in mllms via reinforced fine-tuning. *Advances in neural information processing systems*, 38:4350–4376, 2026. 2
- [47] Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, Zhaokai Wang, Zhe Chen, Hongjie Zhang, Ganlin Yang, Haomin Wang, Qi Wei, Jinhui Yin, Wenhao Li, Erfei Cui, Guanzhou Chen, Zichen Ding, Changyao Tian, Zhenyu Wu, Jingjing Xie, Zehao Li, Bowen Yang, Yuchen Duan, Xuehui Wang, Zhi Hou, Haoran Hao, Tianyi Zhang, Songze Li, Xiangyu Zhao, Haodong Duan, Nianchen Deng, Bin Fu, Yinan He, Yi Wang, Conghui He, Botian Shi, Junjun He, Yingdong Xiong, Han Lv, Lijun Wu, Wenqi Shao, Kaipeng Zhang, Huipeng Deng, Biquing Qi, Jiaye Ge, Qipeng Guo, Wenwei Zhang, Songyang Zhang, Maosong Cao, Junyao Lin, Kexian Tang, Jianfei Gao, Haian Huang, Yuzhe Gu, Chengqi Lyu, Huanze Tang, Rui Wang, Haijun Lv, Wanli Ouyang, Limin Wang, Min Dou, Xizhou Zhu, Tong Lu, Dahua Lin, Jifeng Dai, Weijie Su, Bowen Zhou, Kai Chen, Yu Qiao, Wenhao Wang, and Gen Luo. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025. 2, 6
- [48] Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Zun Wang, Yansong Shi, Tianxiang Jiang, Songze Li, Jilan Xu, Hongjie Zhang, Yifei Huang, Yu Qiao, Yali Wang, and Limin Wang. Internvideo2: Scaling foundation models for multimodal video understanding. In *European conference on computer vision*, pages 396–416. Springer, 2024. 2
- [49] Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8052–8060, 2018. 2, 6, 7
- [50] Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International conference on machine learning*, pages 23965–23998. PMLR, 2022. 4, 1
- [51] Junbin Xiao, Angela Yao, Yicong Li, and Tat-Seng Chua. Can i trust your answer? visually grounded video question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13204–13214, 2024. 2
- [52] Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. Video question answering via gradually refined attention over appearance and motion. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1645–1653, 2017. 2
- [53] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pages 6787–6800, 2021. 2
- [54] Guang Yang, Manling Li, Jiajie Zhang, Xudong Lin, Heng Ji, and Shih-Fu Chang. Video event extraction via tracking visual states of arguments. In *Proceedings of the AAAI conference on artificial intelligence*, pages 3136–3144, 2023. 2
- [55] Jianwei Yang, Yonatan Bisk, and Jianfeng Gao. Taco: Token-aware cascade contrastive learning for video-text alignment. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11562–11572, 2021. 2
- [56] Shusheng Yang, Jihan Yang, Pinzhi Huang, Ellis L II Brown, Zihao Yang, Yue Yu, Shengbang Tong, Zihan Zheng, Yifan Xu, Muhao Wang, Rob Fergus, Yann LeCun, Li Fei-Fei, and Saining Xie. Cambrian-s: Towards spatial supersensing in video. In *The Fourteenth International Conference on Learning Representations*, 2025. 2, 6
- [57] Sihyun Yu, Nanye Ma, Pinzhi Huang, Hyunseok Lee, Shusheng Yang, June Suk Choi, Ellis Brown, Oscar Michel, Boyang Zheng, Jinwoo Shin, and Saining Xie. Benchmarking visual state tracking in multimodal video understanding. *arXiv preprint arXiv:2606.03920*, 2026. 1, 2, 5, 6, 7
- [58] Qianhao Yuan, Jie Lou, Xing Yu, Hongyu Lin, Le Sun, Xianpei Han, and Yaojie Lu. Vision-opd: Learning to see fine details for multimodal llms via on-policy self-distillation. *arXiv preprint arXiv:2605.18740*, 2026. 3
- [59] Guiyi Zeng, Junqing Yu, Yi-Ping Phoebe Chen, Xu Chen, Wei Yang, and Zikai Song. Curevo: Curriculum-guided self-evolution for video understanding. *arXiv preprint arXiv:2604.26707*, 2026. 2, 3
- [60] Congzhi Zhang, Zhibin Wang, Yinchao Ma, Jiawei Peng, Yihan Wang, Qiang Zhou, Jun Song, and Bo Zheng. Rewatch-rl: Boosting complex video reasoning in large vision-language models through agentic data synthesis. *arXiv preprint arXiv:2509.23652*, 2025. 2
- [61] 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*, pages 543–553, 2023. 2

- 801 [62] Haoji Zhang, Xin Gu, Jiawen Li, Chixiang Ma, Sule Bai,
802 Chubin Zhang, Bowen Zhang, Zhichao Zhou, Dongliang He,
803 and Yansong Tang. Thinking with videos: Multimodal tool-
804 augmented reinforcement learning for long video reasoning.
805 In *Proceedings of the IEEE/CVF Conference on Computer*
806 *Vision and Pattern Recognition*, pages 32903–32914, 2026.
807 [2](#)
- 808 [63] Ruixu Zhang, Deyi Ji, Lanyun Zhu, Xuanyi Liu, Yuxin
809 Meng, Ruihang Chu, and Yujiu Yang. Video-zero:
810 Self-evolution video understanding. *arXiv preprint*
811 *arXiv:2605.14733*, 2026. [2](#), [3](#), [6](#)
- 812 [64] Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Zi-
813 wei Liu, and Chunyuan Li. Llava-video: Video instruction
814 tuning with synthetic data. *arXiv preprint arXiv:2410.02713*,
815 2024. [2](#)
- 816 [65] Tinghui Zhu, Sheng Zhang, James Y Huang, Selena Song,
817 Xiaofei Wen, Yuankai Li, Hoifung Poon, and Muhao Chen.
818 Video models can reason with verifiable rewards. *arXiv*
819 *preprint arXiv:2605.15458*, 2026. [2](#)



Temporal Self-Distillation: Learning Visual State Tracking in Videos Without Supervision

Supplementary Material

This supplementary material provides additional extended results, implementation details, and further observations and inferences.

A1. Model Souping

A single S^3T model reaches 36.48 on VSTAT. The 37.12 and 37.44 models reported in the main paper are obtained by model souping [50], which combines two independently trained runs by averaging their weight updates. The resulting model has the same size and inference cost as a single checkpoint. This section describes the two runs, the averaging procedure, and how the averaging strength is selected.

Complementary probes. The two runs differ only in the fixed probe q (§3.2), which changes the aspect of the running state used for self-supervision. The first uses the full-state probe “*This is a short video of colored balls moving on a background. Some balls may be briefly hidden behind moving bars, and the camera may pan or zoom. List the colors of the balls you can see and how many there are. Be concise (one short line).*” The second keeps the same first two sentences but asks one question from a small fixed set about changes across the clip, such as “*How many different ball colors appear in total during the clip? Answer with one number only.*” or “*Does any ball change color during the clip? Answer only yes or no.*” The full-state run performs best across the state axes at 36.88, while the single-question run performs best on Count at 36.11. Their complementary strengths make the resulting soup stronger than either run alone.

Weight averaging and strength selection. We soup the weight updates rather than the model outputs. A LoRA adapter adds a weight update $\Delta W = s(BA)$ to each adapted module, where A and B are its low-rank factors and s is the fixed scale. For each module, we average the two runs’ ΔW , multiply the result by a global strength α , and add it to the frozen base weights. The vision-adapted model applies the same procedure to the vision-encoder adapters, producing a single checkpoint used like any other model.

The only additional hyperparameter is α , which is selected without using VSTAT. We evaluate each merged model on 100 held-out *StateGen* synthetic clips from a separate seed range. Their program states are never used during training, and their scores do not affect benchmark evaluation. For each α , the model describes every clip using 12

Table A1. **Choosing the averaging strength on held-out data.** Count error on 100 fresh synthetic clips, whose states are never used during training or benchmark evaluation. Lower is better, and the minimum occurs at the reported $\alpha=1.3$.

α	1.0	1.1	1.2	1.3	1.4	1.5	1.7
Count error ↓	3.02	3.16	3.20	2.96	3.12	3.26	3.53

frames, and its reported count is compared with the true count. Table A1 shows that the count error is lowest at $\alpha=1.3$, which is used for the reported 37.44 model. A second label-free criterion based on the remaining teacher–student divergence decreases smoothly with α but does not identify a unique value, so we do not use it.

Additional strengths. Table A2 reports performance across a wider range of averaging strengths. The vision-adapted model reaches 37.44 at $\alpha=1.3$, while the frozen-encoder model reaches 37.12 at $\alpha=1.5$. Both remain above the base across the tested range.

Table A2. **Extended strength range.** VSTAT overall accuracy (%) across additional averaging strengths. Every reported value remains above the base score of 34.74.

α	1.25	1.3	1.35	1.5	1.7
Vision-adapted	36.12	37.44	36.62	36.76	37.06
Frozen encoder	36.13	36.33	36.64	37.12	36.87

A2. Temporal-Pretext Baselines

Table 1c compares four established temporal self-supervised tasks. Each applies a known transformation to an unlabeled clip and trains the model to identify that transformation. All baselines use the same setup as S^3T , including the LLaVA-OV-2-8B base, the same 300 unlabeled clips, LoRA rank $r=32$, 3000 training steps, and the reference anchor. Only the training target changes, isolating the effect of the learning objective.

Objective construction. Each task applies a transformation T to clip x and derives a short text label $\ell(T)$. The model is trained with cross-entropy to predict this label from the transformed clip,

$$\mathcal{L}_{\text{pretext}} = \text{CE}(f_{\theta+\phi}(T(x)), \ell(T)). \quad (8)$$

Q: The video first shows an initial central block structure. Then the camera keeps moving while a robot hand makes several add/remove attempts. After all actions finish, how many blocks remain in the central structure?

VIDEO PREFIX	first 25% (0-5.0 s)	first 50% (0-10.0 s)	first 75% (0-15.0 s)	first 100% (0-20.0 s)
TIME (s)	0.0 - 5.0 (end at 5.0 s)	0.0 - 10.0 (end at 10.0 s)	0.0 - 15.0 (end at 15.0 s)	0.0 - 20.0 (end at 20.0 s)
S3T (ours)	✓ 10	✓ 10	✓ 10	✓ 10
LLaVA-OV-2-BB (base)	✓ 10	✗ 9	✗ 7	✗ 4
Qwen3-VL-8B	✓ 10	✗ 9	✗ 7	✗ 7
Molmo2-4B	✗ 6	✗ 3	✗ 6	✗ 5
Video-Zero-8B (self-evolving)	✗ 16	✗ 9	✗ 7	✗ 7

Q: What is the final score (red-blue)? (A) 3-1 (B) 2-2 (C) 0-4 (D) 1-3

VIDEO PREFIX	first 25% (0-11.0 s)	first 50% (0-22.0 s)	first 75% (0-32.9 s)	first 100% (0-43.9 s)
TIME (s)	0.0 - 11.0 (end at 11.0 s)	0.0 - 22.0 (end at 22.0 s)	0.0 - 32.9 (end at 32.9 s)	0.0 - 43.9 (end at 43.9 s)
S3T (ours)	✓ D	✓ D	✓ D	✓ D
LLaVA-OV-2-BB (base)	✓ D	✗ B	✗ B	✗ B
Qwen3-VL-8B	✓ D	✗ A	✗ A	✗ A
Molmo2-4B	✓ D	✓ D	✓ D	✓ D
Video-Zero-8B (self-evolving)	✓ D	✗ B	✗ B	✗ A

Q: What text message is encoded by the Morse light flashes? Use this Morse mapping: A=·-·, B=·-···, C=·-··-·, D=·-··, E=·, F=··-·, G=··-··, H=····, I=···, J=··-··, K=·-·-·, L=·-··, M=··, N=··-·, O=····, P=··-··, Q=··-·-·, R=···, S=····, T=·-··, U=··-·, V=····, W=··-··, X=··-·, Y=··-··, Z=····, 0=·····, 1=·-····, 2=··-···, 3=·····, 4=·····, 5=·····, 6=·····, 7=·····, 8=·····, 9=·····, . =·····, : =·····, ! =·····, (=··-··,) =··-··, [=··-··, \ =··-··,] =··-··, ^ =··-··, _ =··-··, ` =··-··, " =··-··, \$ =··-··, % =··-··, & =··-··, ' =··-··, * =··-··, + =··-··, = =··-··, < =··-··, > =··-··, ? =··-··, @ =··-··, # =··-··, (A) ANANAS (B) BANABA (C) ANANDA (D) BANANA

VIDEO PREFIX	first 25% (0-5.1 s)	first 50% (0-10.2 s)	first 75% (0-15.3 s)	first 100% (0-20.4 s)
TIME (s)	0.0 - 5.1 (end at 5.1 s)	0.0 - 10.2 (end at 10.2 s)	0.0 - 15.3 (end at 15.3 s)	0.0 - 20.4 (end at 20.4 s)
S3T (ours)	✓ D	✓ D	✓ D	✓ D
LLaVA-OV-2-BB (base)	✓ D	✗ C	✗ C	✗ C
Qwen3-VL-8B	✓ D	✓ D	✓ D	✓ D
Molmo2-4B	✓ D	✓ D	✓ D	✓ D
Video-Zero-8B (self-evolving)	✓ D	✓ D	✓ D	✓ D

Q: Balls are indexed left-to-right in the last frame before any release from 'ball_1' through 'ball_6'. Which ball took the longest time to fall through the hole after its own release? (A) ball_3 (B) ball_1 (C) ball_2 (D) ball_6

VIDEO PREFIX	first 25% (0-5.1 s)	first 50% (0-10.2 s)	first 75% (0-15.3 s)	first 100% (0-20.4 s)
TIME (s)	0.0 - 5.1 (end at 5.1 s)	0.0 - 10.2 (end at 10.2 s)	0.0 - 15.3 (end at 15.3 s)	0.0 - 20.4 (end at 20.4 s)
S3T (ours)	✓ B	✓ B	✗ A	✓ B
LLaVA-OV-2-BB (base)	✓ B	✓ B	✗ A	✗ A
Qwen3-VL-8B	✗ A	✗ A	✗ A	✗ A
Molmo2-4B	✗ D	✗ D	✗ D	✗ D
Video-Zero-8B (self-evolving)	✗ A	✗ A	✗ A	✗ A

Q: What word was typed on the keyboard? (A) burtonize (B) anatomize (C) castorize (D) customize

VIDEO PREFIX	first 25% (0-3.2 s)	first 50% (0-6.4 s)	first 75% (0-9.7 s)	first 100% (0-12.9 s)
TIME (s)	0.0 - 3.2 (end at 3.2 s)	0.0 - 6.4 (end at 6.4 s)	0.0 - 9.7 (end at 9.7 s)	0.0 - 12.9 (end at 12.9 s)
S3T (ours)	✓ D	✓ D	✓ D	✓ D
LLaVA-OV-2-BB (base)	✓ D	✗ B	✗ B	✓ D
Qwen3-VL-8B	✓ D	✓ D	✓ D	✓ D
Molmo2-4B	✓ D	✓ D	✓ D	✓ D
Video-Zero-8B (self-evolving)	✓ D	✓ D	✓ D	✓ D

Figure A1. **Additional qualitative samples.** Each panel is one VSTAT [57] (CC BY 4.0) item. We ask that item's own question four times, on the first 25, 50, 75 and 100% of the clip, as four *independent* queries; every model then samples its usual 64-frame budget inside the prefix, with greedy decoding and an identical prompt. Correctness follows VSTAT's own matching rule. These are selected examples chosen from items our model answers correctly at the full clip; Tab. 1a reports performance over the whole benchmark.

The four tasks differ only in the transformation and label:

- **Frame order** [16, 30] splits the clip into four contiguous segments and reorders them with a random permutation π . The label is the digit sequence that restores the original order.
- **Arrow of time** [33, 49] plays the clip forward or backward. The label is *forward* or *backward*.
- **Pace** [6, 45] samples the clip at one of several playback speeds. The label is the corresponding speed level.
- **Masked infill** [41] removes a contiguous span of frames. The target is a short description of the missing event.

None of these objectives matches the state-tracking gains of S³T (Table 1c). Pace is the strongest baseline at 35.45 and is closest to our temporal-density setup, but recovers less than half of the single-model gain. Masked infill falls below the base at 34.25. These tasks can often be solved from local appearance or motion cues without maintaining the scene state across the clip. Frame order is the only exception on one axis, raising Dictionary to 30.5 while leaving the overall score unchanged. Its weights are not used in

S³T.

A3. Choice of Distillation Divergence

Table A3 gives the values behind Fig. 4. We keep the training recipe fixed and vary only the distillation divergence d_{pos} . The reference anchor remains JSD in every run. Results cover the full 3000 training steps and are averaged over three seeds. A separate JSD monitor stays near 0.005 for all three objectives, showing that they act on a comparable teacher-student gap and differ mainly in their loss scale and gradient behavior.

A4. Ablation Components

Table 2 compares several teacher-view variants. Each keeps the 12-frame student fixed and changes only how the teacher frames are selected. Our default teacher uses 24 uniformly sampled frames covering the full clip.

Event-aware teacher. This variant places more teacher frames around scene changes detected by the frozen model

Table A3. **JSD and KL as the distillation divergence.** Mean and maximum values of d_{pos} and the pre-clipping gradient norm over 3000 steps, averaged across three seeds. The reference anchor uses JSD in every run. Although the monitored teacher–student gap remains near 0.005, forward and reverse KL produce losses about $3.8\times$ larger and substantially larger gradient spikes than JSD.

Divergence	d_{pos}		Pre-clip grad. norm	
	mean	max	mean	max
JSD (ours)	0.0054	0.133	0.237	5.53
Forward KL	0.0208	1.175	0.873	15.96
Reverse KL	0.0204	0.448	0.918	29.32

without labels (§A5). It therefore allocates more frames to periods where the scene state is likely to change.

Multi-scale teacher. The teacher combines predictions from several frame counts, such as $\{16, 24, 32\}$ or $\{12, 24, 48\}$, instead of using a single 24-frame view. The student is distilled toward the combined prediction.

Adaptive teachers. These variants choose the teacher density separately for each clip. The *router* selects a frame count using a lightweight rule, while the *budget-controlled* variant adjusts the density to keep the teacher–student divergence d_{pos} near a target range. Both seek a teacher that provides additional information while remaining learnable from the sparse view. Neither outperforms the fixed $12 \rightarrow 24$ teacher.

A5. Validation of Self-Detected Event Boundaries

The event-aware teacher in our ablation study places additional frames around boundaries detected by the frozen model. Since this variant performs below the fixed temporal stretch, we test whether the result is caused by unreliable boundary detection.

The detector uses no labels or external model. At each candidate time point, it extracts two short non-overlapping windows before and after the point. The frozen model answers the same state probe on both windows, and we compute the token-averaged difference between their next-token distributions. A larger difference indicates that the model reads the scene differently across that point and therefore suggests an event boundary.

The score separates true event times from ordinary time points with an AUROC of 0.851 and a confidence interval of $[0.833, 0.869]$. Shuffling the boundary labels gives 0.501 ± 0.014 , placing the observed result about 26 standard deviations above chance.

Additional checks show that the detector does not rely on fixed temporal positions or a repeated boundary pattern. When candidates are grouped by time and ranked only within each group, AUROC remains between 0.81 and 0.85. Across 253 clips, the detector produces 211 distinct boundary sets, and 75.5% of its proposed boundaries match events scheduled by StateGen.

Table A4. **Detected boundaries are semantic rather than shot cuts.** Detection of true event boundaries on 200 synthetic clips. AUROC is the probability of ranking a true boundary above a non-boundary, where 0.5 is chance. The model’s own before/after divergence far exceeds two pixel-based shot-cut detectors, which sit near chance because these renders contain no camera cuts.

Detector	AUROC	F1
Model before/after JSD (ours)	0.851	0.776
ffmpeg scene-score	0.645	0.63–0.69
PySceneDetect	0.576	0.63–0.69
Random	0.503	–

The signal also differs from standard shot-cut detection (Table A4). Pixel-based tools remain near chance because the clips are continuous renders without camera cuts. In contrast, our detector responds to changes in the model’s predicted scene state. The event-aware teacher therefore receives a reliable boundary signal, and its lower performance is unlikely to result from poor boundary detection.

A6. Additional Real-Video Results

The additional real-video results show that S^3T mainly affects tasks requiring cumulative-state reasoning. On VS-TAT’s web-video questions that require a running tally, the model improves by +7.95. On the same source split, it drops by -9.55 on selective-attention questions that require tracking one specified object while ignoring others, revealing a trade-off between cumulative tracking and entity-specific binding.

The same pattern appears on MVBench. Action Count improves by +4.50, while the overall score changes by only -0.40 . Broader temporal-reasoning benchmarks remain largely unchanged. None of the three TempCompass splits changes significantly for either released model, and MMVU is unchanged. VET-Bench’s shell-game task also remains near chance for both the base model and S^3T because it tests tracking one hidden object through swaps rather than maintaining a count.

Training on real Kinetics videos gives no clear overall improvement at +0.66, although cumulative-state questions improve by +4.09. Together, these results suggest that the learned behavior transfers beyond synthetic clips but remains specific to cumulative-state reasoning.

Table A5. **Additional real-video results.** Values are changes in accuracy relative to our base unless an absolute score is shown. Brackets give 95% bootstrap intervals over evaluation items. We mark an effect as significant when its interval excludes zero and otherwise report it as non-significant.

Benchmark	Measure	Result
VSTAT-YouTube, cumulative-state ($n=185$)	Δ overall	+7.95 [+4.22,+12.00] [sig.]
VSTAT-YouTube, selective-attention ($n=88$)	Δ overall	−9.55 [sig.]
MVBench, Action Count	Δ acc	+4.50 [+1.00,+8.50] [sig.]; +3.00 [n.s.] frozen
MVBench, overall ($n=4000$)	Δ acc	−0.40 [−1.07,+0.28] [n.s.]
Kinetics real-video training (3 seeds)	Δ overall	+0.66 $\pm$ 0.28 [n.s.]
TempCompass, caption-matching ($n=1503$)	Δ acc	+0.20 [−0.60,+1.06] [n.s.]; −1.00 [−2.26,+0.27] [n.s.] frozen
TempCompass, multiple-choice ($n=1580$)	Δ acc	−0.06 [−0.82,+0.70] [n.s.]; −1.01 [−2.47,+0.51] [n.s.] frozen
TempCompass, yes/no ($n=2453$)	Δ acc	+0.04 [−0.73,+0.82] [n.s.]; +0.24 [−1.06,+1.63] [n.s.] frozen
MMVU	score	55.2 = base (± 0.0)
VET-Bench, shell game, 64f	3-way acc	base 0.31, S ³ T 0.33 (+0.02 [n.s.])
VET-Bench, shell game, 128f	3-way acc	base 0.28, S ³ T 0.30 (+0.02 [n.s.])

Table A6. **S³T across eleven base models: it clearly helps one, moves two others slightly in the same direction, and is flat or harmful on the rest.** Every row runs the same self-supervised objective on the same data; only the base model changes. Δ S³T is the change in VSTAT accuracy (%) after training, reported at the *best* setting we found for that base rather than at the setting tuned for LLaVA-OV-2-8B, so that the comparison is not handicapped by our own tuning. Changes smaller than the ± 0.52 seed spread of our recipe are within noise. *Headroom* is how much the frozen base gains on VSTAT when its frame budget is raised from the student’s 12 to the teacher’s 24, measured with no training and no adapter; it is the privileged information the objective has available to distill. It does not separate the one base that improves from the ten that do not: across the models we test, it is uncorrelated with Δ S³T (Spearman $\rho = +0.29$, $p = 0.39$, $n = 11$), and the *largest* headroom in the table belongs to Qwen3-VL-8B, which S³T leaves unchanged.

Base model	Backbone family	Base	Δ S ³ T	Headroom 12 $\rightarrow$ 24
LLaVA-OV-2-8B	OneVision-2 / Qwen3-8B	34.74	+1.74	+2.91
InternVL3.5-14B	InternViT / Qwen3-14B	31.97	+0.27	+1.28
Qwen3-VL-8B	Qwen3-VL	34.11	+0.13	+3.42
InternVL3.5-8B	InternViT / Qwen3-8B	30.24	≈ 0 [n.s.]	−0.45
Molmo2-4B [†]	Molmo2	35.81	−0.18	+1.64
InternVL3.5-4B	InternViT / Qwen3-4B	29.76	−0.20	+1.12
LLaVA-OV-7B	OneVision / Qwen2-7B	28.49	−0.86	−0.76
Qwen3-VL-4B	Qwen3-VL	31.43	−0.86	+0.79
Cambrian-S-7B [‡]	Cambrian-S	31.53	−1.16	−0.33
Qwen2.5-VL-7B	Qwen2.5-VL	31.84	−1.43	+1.77
InternVL3.5-2B	InternViT / Qwen3-2B	31.31	−4.84	+1.53

[†]Best of four settings (our recipe, and three re-tunings of the learning rate and the anchor target); the full checkpoint ladder of the best setting peaks at 35.82, i.e. exactly the base, so the result is a null rather than an optimisation failure.

[‡]Best-anchored of three runs; Cambrian-S is unstable under this objective and the three runs span −1.16 to −14.02. See Appendix A7.

to 14B parameters (Table A6). S³T clearly improves one model. InternVL3.5-14B and Qwen3-VL-8B also reach slightly above their bases by +0.27 and +0.13, respectively. Their per-axis results are reported in Appendix A8.

To reduce the chance that these failures come from poor tuning, we re-tune the two strongest alternatives. Molmo2-4B reaches 35.82 at its best learning rate, anchor target, and checkpoint, compared with a base score of 35.81. On Cambrian-S-7B, raising a hard-coded ceiling that had capped the anchor weight, and so left the anchor too weak, improves the result from −4.24 to −1.16, which is the value reported in Table A6. Neither model shows a positive effect after re-tuning.

We also test whether success can be predicted from properties of the base model or the two temporal views. From our experiments, we can conclude that teacher-view headroom is not predictive. Across the eleven bases, the gain from increasing the frame budget from 12 to 24 has little relation to the effect of S³T (Spearman $\rho = +0.29$, $p = 0.39$). Qwen3-VL-8B has more headroom than our base, +3.42 versus +2.91, yet S³T changes it by only +0.13. Cambrian-S is the only failure we can partly explain. Its accuracy drops when the frame budget is doubled, leaving little useful signal in the teacher view, and its training is unstable under our objective. The remaining failures are not explained by the factors we measured.

However, two possibilities remain open. The models may differ in how much temporal post-training they have already received, or in how effectively a language-model adapter can act on visual evidence. Testing these possibilities would require models matched on post-training and a broader study of adapter placement. We therefore scope our claim to the base model on which S³T shows a clear gain and leave reliable base-model selection to future work.

996

A7. Transfer Across Base Models

997

We apply the same objective, data, and training schedule to eleven base models from five families, ranging from 1.5B

998

Table A7. **Per-axis breakdown for the two bases whose overall change is positive but inside the seed spread.** Per-axis change on VSTAT for the two bases of Table A6 whose overall delta is positive but inside our recipe’s ± 0.52 seed spread, with 95% paired-bootstrap intervals over 10 000 resamples of the items. S³T on LLaVA-OV-2-8B is shown for comparison. Intervals excluding zero are bold. n is fixed by the benchmark. All three are single seed (seed 7), so these intervals capture item noise only; the per-axis *seed* spread is larger than every delta in the top two rows.

n	Count 747	Location 384	Attribute 369	Atomic 710	Sequence 211	Set 249	Dictionary 330
InternVL3.5-14B (+0.27)	−0.11	+1.28	+0.00	+0.21	+1.42	−0.08	−0.06
95% CI	[−1.30, 1.14]	[−0.26, 2.87]	[−1.03, 1.14]	[−0.90, 1.35]	[0.00, 3.32]	[−1.69, 1.61]	[−1.97, 1.94]
Qwen3-VL-8B (+0.13)	+0.95	− 2.94	+1.68	−0.39	+2.27	+1.00	−0.76
95% CI		[− 5.26 , − 0.70]					
S ³ T on LLaVA-OV-2-8B (+1.74)	+2.69	−0.29	+1.92	+2.58	+2.37	−0.84	+1.48

A8. Explanations Tested and Ruled Out

S³T clearly improves only one of the eleven base models in Table A6. We report below the explanations we tested for this outcome and the measurement that rules each one out.

Teacher-view headroom. A natural hypothesis is that a base must benefit from the denser teacher view before it can learn from that view. We test this by evaluating all eleven bases at 12 and 24 frames without training, using the same 1500 VSTAT items and scorer as the main experiments.

Teacher-view headroom does not predict the effect of S³T. Across the eleven bases, Spearman correlation is $\rho = +0.29$ ($p = 0.39$) and Pearson correlation is $r = +0.21$ ($p = 0.54$). Qwen3-VL-8B provides the clearest counterexample. Its score increases by +3.42 when moving from 12 to 24 frames, exceeding the +2.91 headroom of LLaVA-OV-2-8B, yet S³T changes it by only +0.13. Three additional bases have more than half the headroom of our successful base and still decline after training. A base can therefore expose useful information at the denser frame budget without transferring that information to the sparse view.

We run two controls on this measurement. First, every model reproduces its published performance at its native frame budget. No model returns the same score at 12 and 24 frames, and the per-item answer-change rate is 22–35% for ten models and 11.2% for Cambrian-S. The models are therefore responding to the change in frame budget.

Second, InternVL uses a fixed 64-tile budget through `dynamic_max_num = max(1, min(12, 64/T))`. At 12 frames it allocates about five tiles per frame, compared with two at 24 frames, so its default evaluation changes both temporal and spatial resolution. We repeat the experiment on all four InternVL sizes with one tile per frame. The conclusion remains unchanged, with $\rho = +0.17$ and $p = 0.62$, and Qwen3-VL-8B still has the largest headroom.

Taken together, these experiments rule out the expla-

nations we could test with the available models. Differences in prior temporal post-training and in how effectively language-model adapters act on visual evidence remain plausible, but evaluating them requires models matched on post-training and a broader comparison of adapter placement, which we leave for future work.

Hyperparameter sensitivity. The recipe was developed using LLaVA-OV-2-8B, so unsuccessful transfer could reflect poor tuning. We therefore re-tuned Molmo2-4B and Cambrian-S-7B before reporting their results.

For Molmo2-4B, we halve the learning rate, tighten the anchor target, combine both changes, and evaluate every checkpoint from the strongest configuration. Its best score is 35.82 against a base score of 35.81. Re-tuning reduces the original deficit from −0.69 to −0.18, showing that some of the loss came from our initial configuration, but no learning rate, anchor target, or checkpoint gives a clear improvement.

For Cambrian-S-7B, the original anchor weight reaches its hard-coded upper bound during 9.3% of training steps. Raising this bound reduces d_{ref} from 0.0758 to 0.0664 and lowers the proportion of saturated steps from 9.3% to 2.6%. The result improves from −4.24 to −1.16, which is the value reported in Table A6. Cambrian-S nevertheless remains unstable under the objective, which is possibly attributed to the nature of the base model itself (because we do not observe such an instability with any other base). All three runs diverge during training with gradient norms between 8.5 and 37. Halving the learning rate does not prevent this, and that run produces empty answers on 19.7% of evaluation items. Its observed range across three runs is −1.16 to −14.02.

Small positive changes on two bases. InternVL3.5-14B and Qwen3-VL-8B yield +0.27 and +0.13 above their respective bases. Both values fall within the ± 0.52 seed variation of our recipe. We nevertheless examine whether their

per-axis changes match the pattern produced by S^3T on LLaVA-OV-2-8B. Table A7 reports paired bootstrap intervals from 10 000 resamples.

For InternVL3.5-14B, no axis has an interval excluding zero, and four of the seven axes change by at most 0.21 points. For Qwen3-VL-8B, only Location excludes zero, and it moves in the negative direction by -2.94 with interval $[-5.26, -0.70]$. This result does not survive Holm correction across the seven axes. With seven tests at the 95% level, one uncorrected interval excluding zero is also consistent with chance.

Neither per-axis profile clearly resembles the profile of our successful model. Relative to the single-run S^3T vector on LLaVA-OV-2-8B, Spearman correlation is -0.04 ($p = 0.96$) for InternVL3.5-14B and $+0.21$ ($p = 0.66$) for Qwen3-VL-8B. Their signs agree on 3/7 and 4/7 axes, respectively. A more favourable comparison against our four-seed mean profile raises the correlations but leaves both insignificant, and on Qwen3-VL-8B the axes that define our result, Atomic and Set, move in the opposite direction.

These estimates capture item-level variation but not seed variation because each base was trained with one seed. On LLaVA-OV-2-8B, four seeds produce per-axis standard deviations of up to 1.80. The two valid InternVL3.5-8B seeds differ by 2.1 to 3.3 points on five axes and reverse direction on Atomic, Sequence, and Set. This variation exceeds every change measured on InternVL3.5-14B. We therefore report the two positive aggregate changes without treating them as established gains.