

---

# CIRCUITPROBE: Tracing Visual Temporal Evidence Flow in Video Language Models

---

Yiming Zhang <sup>† \* 1 2</sup> Zhuokai Zhao <sup>\* 3</sup> Chengzhang Yu <sup>4</sup> Kun Wang <sup>5</sup> Zhendong Chu <sup>6</sup> Qiankun Li <sup>5</sup>  
Zihan Chen <sup>1 2</sup> Yang Liu <sup>5</sup> Zenghui Ding <sup>1</sup> Yining Sun <sup>1</sup> Qingsong Wen <sup>6</sup>

## Abstract

Autoregressive large vision–language models (LVLMs) interface video and language by projecting video features into the LLM’s embedding space as continuous visual token embeddings. However, it remains unclear *where* temporal evidence is represented and *how* it causally influences decoding. To address this gap, we present **CIRCUITPROBE**, a circuit-level analysis framework that dissects the end-to-end video-language pathway through two stages: (*i*) *Visual Auditing*, which localizes object semantics within the projected video-token sequence and reveals their causal necessity via targeted ablations and controlled substitutions; and (*ii*) *Semantic Tracing*, which uses logit-lens probing to track the layer-wise emergence of object and temporal concepts, augmented with temporal frame interventions to assess sensitivity to temporal structure. Based on the resulting analysis, we design a targeted surgical intervention that strictly follows our observations: identifying temporally specialized attention heads and selectively amplifying them within the critical layer interval revealed by Semantic Tracing. This analysis-driven intervention yields consistent improvements (up to 2.4% absolute) on the temporal-heavy TempCompass benchmark, validating the correctness, effectiveness, and practical value of the proposed circuit-level analysis for temporal understanding in LVLMs.

## 1 Introduction

Large vision–language models (LVLMs) have become a dominant paradigm for multimodal understanding by integrating visual inputs (images or videos) with natural language to produce free-form textual outputs (Liu et al.,

<sup>†</sup>Work done at Squirrel AI <sup>1</sup>HFIPS, Chinese Academy of Sciences <sup>2</sup>University of Science and Technology of China <sup>3</sup>University of Chicago <sup>4</sup>South China University of Technology <sup>5</sup>Nanyang Technological University <sup>6</sup>Squirrel AI. Correspondence to: Zenghui Ding <[dingzenghui@iim.ac.cn](mailto:dingzenghui@iim.ac.cn)>, Kun Wang <[wk520529@mail.ustc.edu.cn](mailto:wk520529@mail.ustc.edu.cn)>.

Preprint. February 28, 2026.

2023b;a; Maaz et al., 2023; Zhang et al., 2024b). Most LVLMs follow a modular design that couples a pre-trained visual encoder with a pre-trained large language model (LLM) through a trainable adapter, or projector (Li et al., 2022; Moon et al., 2023; Chen et al., 2023). This projector converts visual features into continuous token embeddings (often referred to as soft prompts) that are compatible with the LLM’s input space, enabling cross-modal alignment and downstream reasoning (Merullo et al., 2023; Liu et al., 2024a; Yu et al., 2025; Pan et al., 2025).

Understanding *how* video evidence influences language decoding is increasingly important for two reasons. First, it clarifies what information is actually transferred through the visual–language interface, which directly affects vision–language alignment quality (Radford et al., 2021; Zhang et al., 2024a). Second, it provides actionable guidance for designing LVLMs with robust reasoning behaviors beyond benchmark-driven improvements (Pang et al., 2024; Woo et al., 2024; Park et al., 2024; Wang et al., 2025a).

While recent interpretability work has substantially advanced our understanding of image-based LVLM representations and visual-text integration (Neo et al., 2024), it remains less clear how these tools extend to *video* inputs. Unlike images, video understanding requires the model to aggregate evidence across frames and integrate cues that are distributed over time. The challenge is that video-derived soft prompts do not correspond directly to discrete language tokens, making it difficult to attribute model outputs to specific temporal evidence or to isolate where temporal reasoning succeeds or fails (Merullo et al., 2023; Yin et al., 2023).

To address this gap, in this paper, we introduce **CIRCUITPROBE**, a circuit-based tracing framework that operationalizes two complementary analytic probes. First, *visual auditing* localizes task-critical semantics within the projected video-token sequence and tests their causal necessity through targeted token ablations and controlled substitutions. Second, *semantic tracing* tracks the layer-wise emergence of language-aligned object and action concepts by logit-lens-style (NostalgiaBraist, 2020) probe and shows how the temporal structure affects semantic consolidation. Motivated by these observations, we design targeted interventions that follow our findings to test whether the identified structures are causally responsible for temporal reasoning.

Our contributions can be summarized as follows:

- We show that video semantics remain strongly localized to object-aligned token positions after projection: ablating object tokens causes substantially larger performance degradation than removing matched control tokens.
- We find that object and action concepts become reliably language-aligned primarily in mid-to-late layers of the LLM, defining a critical interval where video-conditioned semantics are consolidated. We further observe behaviors consistent with frame-wise state aggregation rather than explicit temporal order modeling, motivating targeted temporal interventions.
- Using attention-based temporal metrics, we isolate temporally specialized attention heads and show that selectively amplifying them within the identified critical interval yields consistent improvements on temporal-heavy benchmark such as TempCompass (Liu et al., 2024b). These results validate the correctness, effectiveness, and practical value of our circuit-level analysis.

## 2 Related work

**Interpretability of LLM.** Recent research on LLM interpretability has largely focused on text-only models (Singh et al., 2024; Wang et al., 2025b). Mainstream approaches include circuit analysis, which simplifies models into sparse circuits to understand mechanisms (Hanna et al., 2023; Yao et al., 2024), and causal tracing, which tracks information flow to analyze module contributions and edit factual knowledge (Yu & Ananiadou, 2024; Sharma et al., 2024), as pioneered by ROME (Meng et al., 2022). Another prominent technique is unembedding space projection, which probes internal states by projecting them into the token space (Geva et al., 2021; Dar et al., 2023). While these methods have enhanced our understanding of language models, they focus predominantly on textual processing. Multimodal interpretability, particularly the analysis of visual-textual integration, remains an under-explored frontier.

**Interpretability of LVLMs.** Existing work on visual model interpretability typically focuses on two areas. The first examines visual encoder embedding generation, including how frozen transformers process visual tokens (Neo et al., 2024; Kaduri et al., 2024), how misalignment causes hallucinations (Jiang et al., 2025), and how modality spaces are bridged (Pang et al., 2024; Woo et al., 2024; Park et al., 2024). The second investigates visual-textual interactions in multimodal systems, often by identifying shortcomings on static images (Tong et al., 2024; Verma et al., 2024), extracting visual concept subgraphs (Rajaram et al., 2024), or reverse-engineering ViTs (Vilas et al., 2023). Recent work has begun to address video data, identifying temporal reasoning bottlenecks (Li et al., 2024c) and developing new interpretability tools (Joseph et al., 2025). However, these studies do not examine how visual temporal evidence is

consolidated over time within the LLM, particularly in the context of video input. This leaves a gap in understanding how temporal relationships are processed in LVLMs, which our work aims to fill by investigating the role of temporal attention heads and their impact on model performance.

## 3 CIRCUITPROBE

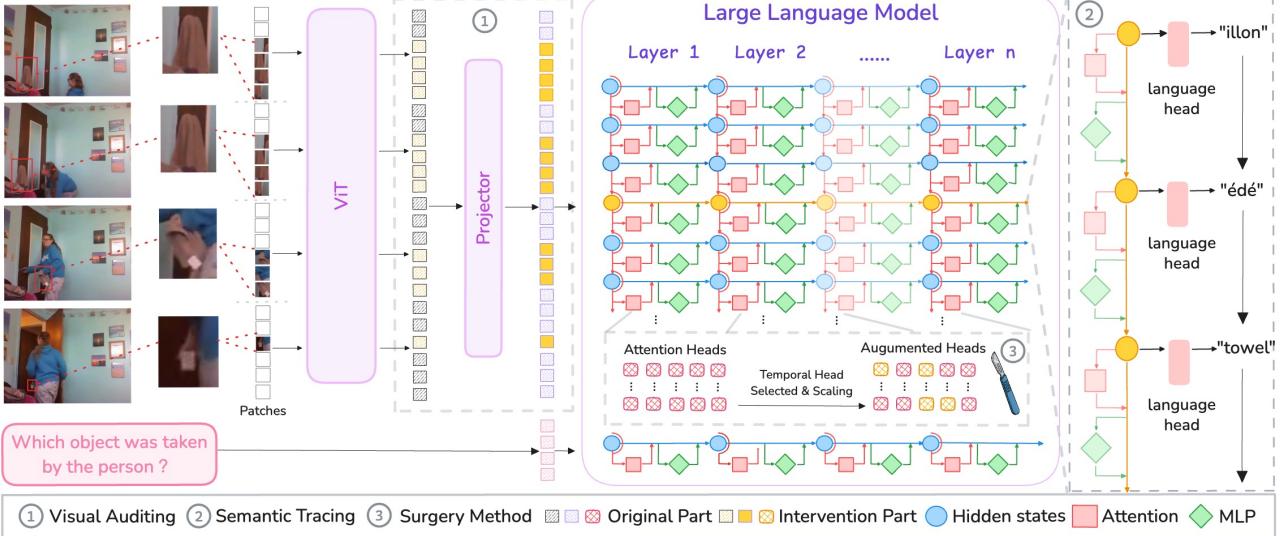
### 3.1 Preliminary Setup

Before introducing the two analytic circuits of CIRCUITPROBE, we first establish a unified notation, dataset construction protocol, and attention-based primitives that will be shared across all subsequent analyses. We evaluate on three representative LVLM backbones covering both image- and video-trained 7B variants: the LLaVA-NeXT family (Li et al., 2024b), the LLaVA-OneVision family (Li et al., 2024a), and Qwen2.5-VL (Bai et al., 2025). For brevity, we denote *image*-trained and *video*-trained model variants using the suffixes  $-I$  and  $-V$ , respectively.

**Model input representation.** Given a video sampled into  $T$  frames and a textual query  $Q$ , we construct the combined input embedding sequence  $\mathbf{X}_0 = [\mathbf{X}_{\text{sys}}; \mathbf{X}_V; \mathbf{X}_Q] \in \mathbb{R}^{N \times d}$ , where  $N$  is the total number of tokens and  $d$  is the hidden dimension of the LLM. Here,  $\mathbf{X}_{\text{sys}}$  denotes system tokens,  $\mathbf{X}_Q$  corresponds to the query tokens, and  $\mathbf{X}_V$  represents the video-conditioned tokens injected into the LLM. Specifically, video tokens  $\mathbf{X}_V \in \mathbb{R}^{N_V \times d}$  are obtained by encoding each frame into  $P$  visual patches using the vision encoder  $f_C$  followed by the trainable adapter (projector). This results in  $N_V = T \times P$  video tokens, which we index by  $\mathcal{I}_V = \bigcup_{t=1}^T \mathcal{I}_{V,t}$ , where  $\mathcal{I}_{V,t}$  denotes the subset of tokens originating from frame  $t$ . We denote all non-video token indices (system and query tokens) by  $\mathcal{I}_Q$ .

**Circuit-based analysis paradigm.** With this setup in place, we adopt a circuit-based analysis paradigm to dissect the internal computation of LVLMs. Our approach explicitly manipulates subsets of video tokens and traces their transformation across layers, enabling us to test concrete predictions about information localization and semantic emergence. Importantly, both visual auditing (Circuit ①, §3.2) and semantic tracing (Circuit ②, §3.3), operate on this shared foundation, ensuring that conclusions drawn from different probes are directly comparable. Unless otherwise specified, our analyses focus on the prefill stage of autoregressive decoding and on video-token positions, where visual information is first injected into the LLM and can directly influence downstream reasoning.

**Dataset curation and probing setup.** To probe genuine temporal understanding, we curate a subset of the STAR benchmark (Wu et al.) focusing on two tasks: *Interaction* (action-object binding) and *Sequence* (chronological dependencies). To strictly isolate visual reasoning from



**Figure 1.** CIRCUITPROBE consists of two circuits: ① *Visual Auditing*, which localizes task-critical object semantics within the projected video-token sequence, and ② *Semantic Tracing*, which tracks the layer-wise emergence of language-aligned concepts. Based on these observations, *Surgical Recovery* amplifies *temporal attention heads* within the identified layers, yielding gains on temporal benchmarks.

language priors, we enforce three-step filtering process: (1) **Indispensability**: We retain only samples where the answer requires integrating video information, filtering out those solvable by single frames or common sense. (2) **Visual Clarity**: We select key frames that clearly depict the target actions or objects. (3) **Counterfactual Validity**: We verify that the model answers correctly on the original input but fails when the critical object is masked. This process yields a controlled testbed for analyzing causal visual integration over time. We employ two complementary probing formats:

1. *Open-ended questions*. (e.g., "Which object...") with prefix-forcing to trace specific semantic concepts;
2. *Close-ended questions*. ("Yes/No") to measure reasoning robustness via prediction flips under intervention.

Unless otherwise specified, Circuit ① reports results under both formats, while Circuit ② focuses on the open-ended format only for concept-level tracing. Representative example frames and more dataset details are shown in Appendix A.

### 3.2 Circuit ①: Visual Auditing

As illustrated in Fig. 1, Circuit ① seeks to identify where task-critical visual semantics reside after projection into the LLM input space, and what form those semantics take for downstream reasoning. We address these questions by using controlled manipulations of projected video tokens to identify where task-critical visual semantics reside in the input sequence and how these semantics are represented for downstream language decoding.

#### 3.2.1 WHERE DOES TASK-CRITICAL VISUAL INFORMATION RESIDE?

We begin by asking whether object-level visual semantics are diffusely encoded across the projected video-token se-

quence  $X_V$ . Inspired by (Neo et al., 2024), Following the notation introduced in §3.1, let  $\mathcal{I}_V$  denote the index set of all projected video tokens in  $X_V$ . We define an object-centric index set  $\mathcal{I}_O \subset \mathcal{I}_V$  consisting of tokens whose corresponding image patches are fully contained within the ground-truth bounding box of the target object. The remaining video tokens  $\mathcal{I}_V \setminus \mathcal{I}_O$  serve as a pool from which control subsets are constructed.

**Intervention.** To causally test which subsets of video tokens carry task-critical semantics, we remove information from selected token subsets by *replacement rather than deletion*. Specifically, for any subset  $\mathcal{S} \subset \mathcal{I}_V$ , we replace the corresponding token embeddings in  $X_V$  with an *uninformative visual embedding*  $x_{\text{null}}$ , defined as the mean visual embedding computed over all visual tokens extracted from 10,000 images randomly sampled from the ImageNet (Deng et al., 2009) validation set. This replacement preserves embedding scale and distribution, while selectively removing semantic content, thereby avoiding artifacts introduced by out-of-distribution perturbations. All non-video tokens  $X_{\text{sys}}$ ,  $X_Q$ , and all remaining video tokens, are left unchanged. We consider four ways of selecting the ablated token subset  $\mathcal{S}$ :

1. *Register Tokens*. Following Dariset et al. (2024), within each key frame, we select tokens whose  $L_2$  norms are more than two standard deviations above the mean token norm, and ablate these tokens as a control subset.
2. *Random Tokens*. As a baseline, we randomly sample a subset  $\mathcal{I}_R \subset \mathcal{I}_V$  such that  $|\mathcal{I}_R| = |\mathcal{I}_O|$ . This controls for the effect of removing an equal number of video tokens without targeting any specific semantic content.
3. *Object Tokens*. We ablate the object-centric token set  $\mathcal{I}_O$ , corresponding to image patches fully contained within

*Table 1.* Accuracy drop ( $\downarrow$ ) from visual token ablation. All performance drops are *relative* to the *No Ablation Baseline* (which has a drop of 0). For object-based ablation, the under-script  $bf$  represents the number of spatially adjacent tokens, as described in object tokens with buffer. We see that ablating object tokens leads to significantly higher drops compared to random tokens, verifying semantic localization.

Type	Avg Tokens	LLaVA-NeXT-I		LLaVA-NeXT-V		LLaVA-OV-I		LLaVA-OV-V	
		Open	Close	Open	Close	Open	Close	Open	Close
No Ablation	0	100.0 $\downarrow_0$							
Register	13	96.6 $\downarrow_{3.4}$	98.3 $\downarrow_{1.7}$	98.0 $\downarrow_{2.0}$	98.2 $\downarrow_{1.8}$	84.7 $\downarrow_{15.3}$	98.2 $\downarrow_{1.8}$	90.3 $\downarrow_{9.7}$	96.9 $\downarrow_{3.1}$
Random	100	97.7 $\downarrow_{2.3}$	99.1 $\downarrow_{0.9}$	97.7 $\downarrow_{2.3}$	98.4 $\downarrow_{1.6}$	85.2 $\downarrow_{14.8}$	98.0 $\downarrow_{2.0}$	90.5 $\downarrow_{9.5}$	97.1 $\downarrow_{2.9}$
	500	94.0 $\downarrow_{6.0}$	98.6 $\downarrow_{1.4}$	93.3 $\downarrow_{6.7}$	98.7 $\downarrow_{1.3}$	84.9 $\downarrow_{15.1}$	97.7 $\downarrow_{2.3}$	89.4 $\downarrow_{10.6}$	96.5 $\downarrow_{3.5}$
	900	89.4 $\downarrow_{10.6}$	98.0 $\downarrow_{2.0}$	89.3 $\downarrow_{10.7}$	98.4 $\downarrow_{1.6}$	85.3 $\downarrow_{14.7}$	97.2 $\downarrow_{2.8}$	90.4 $\downarrow_{9.6}$	97.1 $\downarrow_{2.9}$
Object	354	18.1 $\downarrow_{81.9}$	45.9 $\downarrow_{54.1}$	17.3 $\downarrow_{82.7}$	62.3 $\downarrow_{37.7}$	40.5 $\downarrow_{59.5}$	67.9 $\downarrow_{32.1}$	60.8 $\downarrow_{39.2}$	78.8 $\downarrow_{21.2}$
Object( $bf=1$ )	476	10.0 $\downarrow_{90.0}$	34.5 $\downarrow_{65.5}$	12.5 $\downarrow_{87.5}$	49.3 $\downarrow_{50.7}$	33.6 $\downarrow_{66.4}$	53.3 $\downarrow_{46.7}$	52.6 $\downarrow_{47.4}$	67.1 $\downarrow_{32.9}$
Object( $bf=2$ )	634	9.3 $\downarrow_{90.7}$	27.8 $\downarrow_{72.2}$	7.4 $\downarrow_{92.6}$	44.9 $\downarrow_{55.1}$	30.9 $\downarrow_{69.1}$	45.9 $\downarrow_{54.1}$	51.7 $\downarrow_{48.3}$	63.0 $\downarrow_{37.0}$

the object’s bounding box.

4. *Object Tokens with Buffer.* To account for spatial imprecision and contextual spillover, we expand  $\mathcal{I}_O$  to include spatially adjacent tokens. We evaluate two buffer sizes: 1-Buffer, which includes immediately neighboring tokens, and 2-Buffer, which includes all tokens within a two-position radius of any object token.

**Observation.** As shown in Table 1, ablating object-aligned tokens induces a dramatically larger degradation in performance than ablating any of the control token subsets. This causal asymmetry persists even when the control set contains an equal or substantially larger number of tokens. For example, on LLaVA-NeXT-V (Li et al., 2024b), replacing 573 object tokens in  $\mathcal{I}_O$  leads to a 92.6% drop in open-ended accuracy, whereas replacing 900 randomly selected tokens in  $\mathcal{I}_R$  reduces performance by only 10.7%. Similar trends are observed for register tokens and buffered object tokens, and the effect is consistent across both open-ended and close-ended evaluation protocols, as well as across LLaVA-NeXT (Li et al., 2024b) and LLaVA-OneVision (Li et al., 2024a) variants.

### 3.2.2 WHAT REPRESENTATION DOES THE LLM USE FOR OBJECT SEMANTICS?

As illustrated in Fig. 1, Circuit ① operates on the projected video-token sequence  $X_V$  before any cross-layer transformation occurs. While §3.2.1 establishes where task-critical visual semantics are localized in the input sequence, we now ask a complementary question: in what representational form does the LLM most effectively use this information for downstream reasoning? Specifically, we test whether object-level semantics must be conveyed through native visual token embeddings, or whether equivalent information can be supplied directly in the language embedding space.

**Intervention.** Let  $\mathcal{I}_O \subset \mathcal{I}_V$  denote the object-aligned token indices identified in §3.2.1. Instead of masking, we perform a controlled substitution by replacing the visual em-

beddings at indices  $\mathcal{I}_O$  with the embedding of the object’s textual label (e.g., replacing visual tokens corresponding to a towel with the embedding of the word “towel”). All other inputs, including token positions, ordering, system tokens  $X_{sys}$ , and query tokens  $X_Q$  are left unchanged. We compare this condition against both the original visual-only input and the object-ablation baseline introduced in §3.2.1.

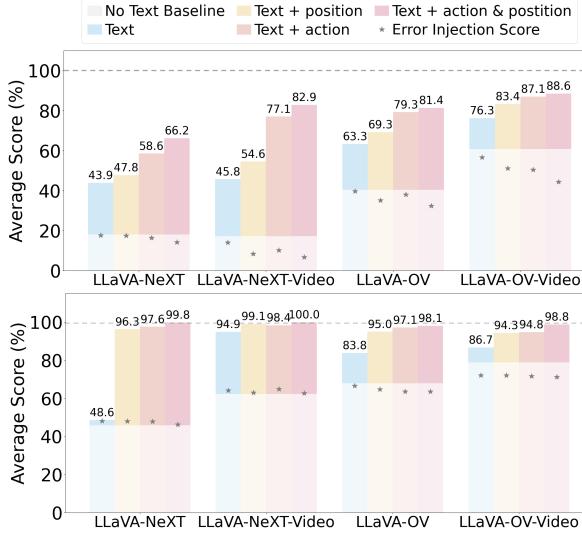
**Observation.** As shown in Fig. 1, replacing object-aligned visual tokens with their corresponding textual embeddings partially recovers the performance loss caused by object ablation. Moreover, substituting visual object tokens with their corresponding text embeddings only partially recovers performance, indicating that native multimodal evidence carries complementary cues beyond discrete label. In the open-ended question setting, examples that collapse under object ablation (e.g., 17.3% accuracy) recover to 82.9% after text injection. Conversely, Fig. 2 shows that injecting an incorrect textual label consistently degrades performance, confirming that the gains arise from supplying the correct semantic content rather than from generic perturbation effects. This behavior is consistent across LVLM variants and across both open-ended and close-ended evaluation formats.

### Circuit ① Takeaway

Object semantics are localized in a small set of object-aligned video tokens, but effective reasoning relies on their language-aligned conceptual representations.

### 3.3 Circuit ②: Semantic Tracing

Circuit ① showed that task-critical visual information enters LLM through the compact set of object-aligned video tokens, and that effective reasoning ultimately relies on language-aligned conceptual representations. We builds on these constraints and asks the process-level question: *How do object and temporal semantics carried by video tokens evolve across LLM layers before influencing decoding?*



**Figure 2. Text injection experiment.** The result shows the performance change when visual object tokens are replaced by their corresponding embedded textual labels. Error injection indicates wrong concept injection.

Answering this question is essential for identifying where temporal reasoning can be causally intervened upon. If visual semantics become usable for language decoding only after a particular stage of processing, then only mechanisms operating at that stage should meaningfully affect temporal understanding—the prediction we later test via targeted surgery in §4.

### 3.3.1 LOGIT-LENS PROBING: MEASURING CONCEPT EVOLVING ACROSS LAYERS

To trace when video-conditioned semantic concepts become expressed in the language space, we employ the logit-lens method (Nostalgebraist, 2020). Specifically, we project the intermediate hidden states  $h_i^{(\ell)}$  of visual tokens into the vocabulary space using the frozen output head (see Appendix B.1 for formal details). We quantify this emergence on the object-centric visual tokens  $\mathcal{I}_O$  (identified in Circuit ①) using two metrics relative to the target concept  $s_c$ :

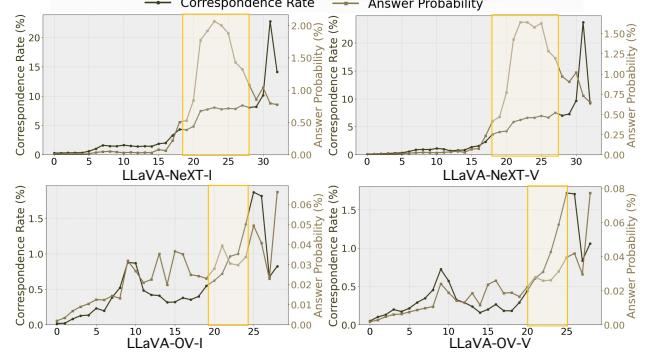
1. **Correspondence Rate ( $C_R^{(\ell)}$ )**. A hard metric measuring the fraction of tokens whose top-ranked vocabulary projection matches the target concept:

$$C_R^{(\ell)} \triangleq \frac{1}{|\mathcal{I}_O|} \sum_{i \in \mathcal{I}_O} \mathbb{I}\left(\arg \max_{w \in \mathcal{V}} P(w | h_i^{(\ell)}) \in \mathcal{T}(s_c)\right). \quad (1)$$

2. **Answer Probability ( $A_P^{(\ell)}$ )**. A soft metric capturing the average confidence (probability mass) assigned to the target concept:

$$A_P^{(\ell)} \triangleq \frac{1}{|\mathcal{I}_O|} \sum_{i \in \mathcal{I}_O} P(s_c | h_i^{(\ell)}). \quad (2)$$

Together,  $C_R^{(\ell)}$  and  $A_P^{(\ell)}$  characterize both the reliability



**Figure 3. Quantitative analysis of semantic tracing.** Both metrics show a sharp increase in the mid-to-late layers, indicating that abstract semantic concepts are consolidated deep within the network.

and the strength with which object or action semantics are expressed in the language space at each layer.

### 3.3.2 WHEN DO VIDEO-CONDITIONED SEMANTICS BECOME LANGUAGE-ALIGNED?

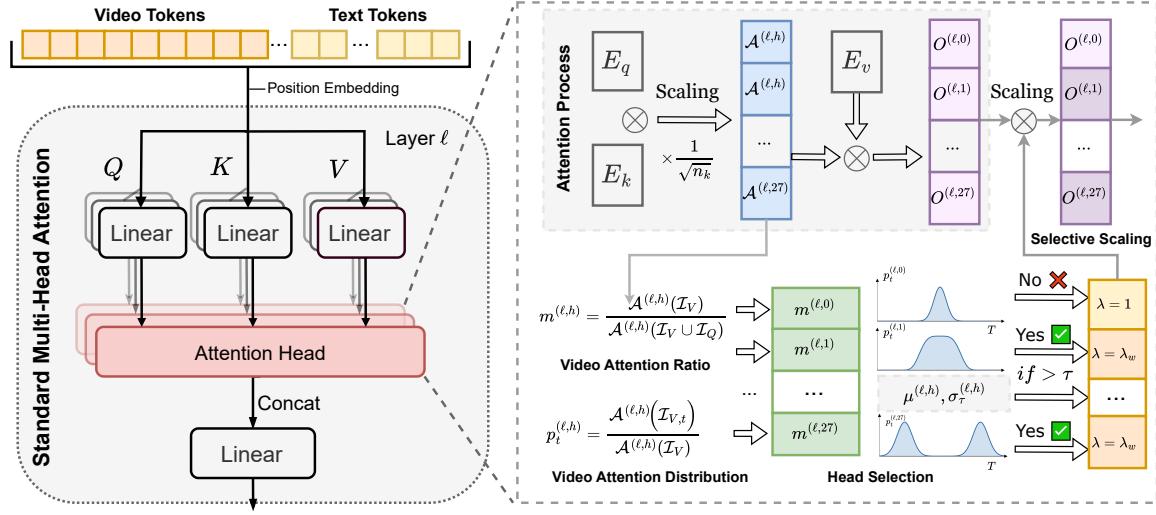
Fig. 3 plots  $C_R^{(\ell)}$  and  $A_P^{(\ell)}$  as a function of layer depth. Across models and concepts, both metrics remain low in early layers, despite the presence of object-aligned visual tokens in the input. As depth increases, we observe a sharp and consistent rise in both correspondence rate and answer probability within the mid-to-late layer interval, after which the curves plateau.

This behavior indicates that projection alone does not render visual information immediately usable for language decoding. Instead, object and action semantics undergo a depth-dependent transformation inside the LLM, becoming stably expressed in the vocabulary space only after sufficient processing. Importantly, this transition is substantially more pronounced for object-aligned tokens  $\mathcal{I}_O$  than for control token sets, indicating that the observed effect reflects video-conditioned semantic consolidation rather than a generic property of the residual stream. The attention knockout and scaling experiments presented in Appendix B further support our observations.

### 3.3.3 HOW DOES TEMPORAL STRUCTURE AFFECT SEMANTIC CONSOLIDATION?

To understand how temporal evidence contributes to this consolidation process, we repeat the same semantic tracing procedure under controlled temporal perturbations of the input video, including frame shuffling and reversal. These interventions preserve per-frame visual content while disrupting temporal structure.

As shown in Table 2, temporal perturbations degrade downstream performance, with stronger global disruptions (e.g., reversal) causing substantially larger drops than mild local reordering, indicating that temporal structure is behaviorally



**Figure 4. Surgery framework.** We first identify *temporal heads* that exhibit both high video attention and broad temporal dispersion. We then intervene by amplifying these specific heads during the decoding phase. This process validates the causal role of these computational units in handling long-range temporal dependencies.

**Table 2. Temporal Robustness Analysis.** Results are reported as the **Accuracy (%)** on the correct-only subset across three temporal perturbation types: REVERSE, LOCAL SWAP, and SHUFFLE. Higher values indicate better stability against temporal perturbations.  $\downarrow \Delta$  denotes the performance drop relative to the unperturbed baseline.

Model	REVERSE			LOCAL SWAP			SHUFFLE		
	Interaction	Sequence	Avg	Interaction	Sequence	Avg	Interaction	Sequence	Avg
LLaVA-OneVision-OV	52.9 $\downarrow 47.1$	57.0 $\downarrow 43.0$	54.4 $\downarrow 45.6$	96.4 $\downarrow 3.6$	98.2 $\downarrow 1.8$	97.0 $\downarrow 3.0$	75.5 $\downarrow 24.5$	85.5 $\downarrow 14.5$	79.3 $\downarrow 20.7$
LLaVA-OneVision-SI	54.1 $\downarrow 45.9$	52.6 $\downarrow 47.4$	53.6 $\downarrow 46.4$	96.6 $\downarrow 3.4$	98.7 $\downarrow 1.3$	97.4 $\downarrow 2.6$	79.7 $\downarrow 20.3$	87.0 $\downarrow 13.0$	82.4 $\downarrow 17.6$
Qwen 2.5 VL	80.7 $\downarrow 19.3$	77.3 $\downarrow 22.7$	79.4 $\downarrow 20.6$	88.3 $\downarrow 11.7$	92.0 $\downarrow 8.0$	89.6 $\downarrow 10.4$	87.2 $\downarrow 12.8$	92.4 $\downarrow 7.6$	89.1 $\downarrow 10.9$

relevant. This suggests that temporal information influences decoding primarily through mechanisms operating during semantic consolidation, rather than by altering early visual representations.

### 3.3.4 WHAT DOES SEMANTIC TRACING REVEAL ABOUT TEMPORAL REASONING?

Beyond aggregate layer-wise metrics, we inspect how decoded concepts evolve over time at fixed token positions. As shown in Fig. 5, the decoded concepts at a fixed token position evolve in tandem with the instantaneous visual appearance of each frame. In the standing-up example, “sitting”-related tokens persist into frames where the action has already begun, and only gradually give way to “standing”-related tokens. This lagged, per-frame semantic update is consistent with a static, frame-by-frame description strategy: the model tracks momentary visual states but does not robustly encode the transition itself as a coherent temporal relation. Taken together with the strong degradation under temporal reversal, these behaviors suggest that temporal understanding in current LVLMs relies on implicit evidence accumulation and recency-biased aggregation, rather than explicit modeling of state transitions. More interactive examples are provided in the Appendix B.4, along with an anonymous website demo for exploration.

### Circuit ② Takeaway

Video-conditioned object and action concepts become language-aligned only within the mid-to-late semantic consolidation interval, where evidence is integrated across frames.

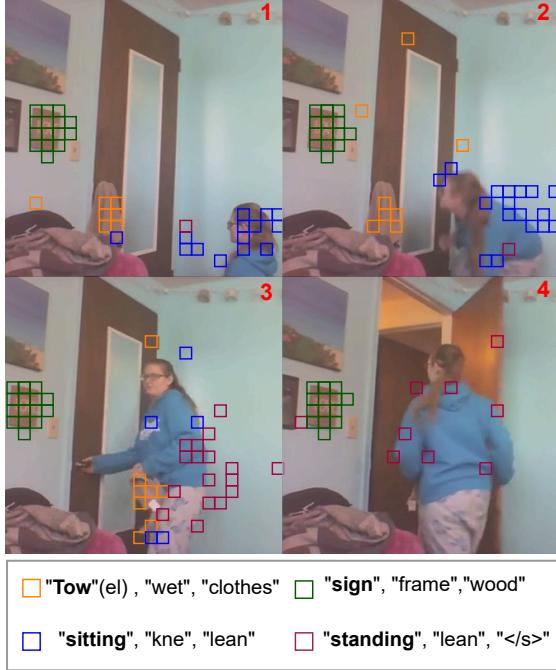
## 4 CIRCUITPROBE-derived Surgery

Circuits ① and ② identify two prerequisites on effective temporal reasoning in LVLMs: (1) task-critical semantics enter the model through object-aligned video tokens, and (2) these semantics become language-aligned only within a mid-to-late consolidation internal.

Building on these observations, we design a surgical recovery method turning these observations into a targeted intervention. Specifically, we proceed in two stages: first identifying such *temporal attention heads* using interpretable temporal routing statistics, and then selectively amplifying their contributions to test whether they causally support temporal understanding.

### 4.1 Identifying Temporal Attention Heads

The findings in Circuit ② suggest that temporal reasoning does not emerge from early visual processing, but instead de-



*Figure 5.* Qualitative example of semantic tracing illustrating how the model captures *temporal dynamics*. The sequence from (a) to (d) shows the evolution of the top-3 most frequent word groups for the single token position.

pends on how evidence from multiple frames is aggregated once concepts become language-aligned. This motivates a concrete hypothesis: only a small subset of attention heads actively integrate information across time in a way that can influence decoding. To test this hypothesis, we model each attention head as a two-stage routing mechanism. A head must first decide whether to attend to video tokens at all, and then decide how to distribute that attention across temporal slices. We utilize TempCompass (Liu et al., 2024b) as the primary benchmark to validate the effectiveness of our intervention on temporal reasoning.

**Video routing.** We first quantify whether an attention head meaningfully consumes video evidence. Heads with negligible video routing are unlikely to participate in temporal reasoning and are excluded from further consideration. Following the notation in §3, let  $q$  denote a query token position, and let  $\alpha_{q \rightarrow j}^{(\ell,h)}$  be the attention weight from  $q$  to token  $j$  at layer  $\ell$ , head  $h$ , we define the attention mass routed from  $q$  into a token subset  $S$  as  $\mathcal{A}_q^{(\ell,h)}(S) \triangleq \sum_{j \in S} \alpha_{q \rightarrow j}^{(\ell,h)}$ . We partition token indices into video tokens  $\mathcal{I}_V = \bigcup_{t=1}^T \mathcal{I}_{V,t}$ , grouped by frame (or temporal slice)  $t$ , and non-video tokens  $\mathcal{I}_Q$ . Using this partition, we measure how strongly head  $(\ell, h)$  routes attention to the video stream via the *video attention ratio*, which we define as:

$$m^{(\ell,h)} \triangleq \frac{\mathcal{A}_q^{(\ell,h)}(\mathcal{I}_V)}{\mathcal{A}_q^{(\ell,h)}(\mathcal{I}_V \cup \mathcal{I}_Q) + \varepsilon}. \quad (3)$$

Intuitively,  $m^{(\ell,h)}$  serves as a gating score. It is close to zero when the head largely ignores video tokens, and increases as the head allocates a greater ratio of its attention capacity to video-derived evidence. In the following, we retain only heads with sufficiently large video attention ratio, ensuring that subsequent temporal analysis focuses on mechanisms that actually ingest visual information.

**Temporal distribution across frames.** Conditioned on attending to video tokens, we next characterize how each head distributes its attention across frames. Intuitively, a head that only attends to a single frame—even if it attends strongly—cannot integrate temporal evidence. In contrast, a head that draws information from multiple frames can better support temporal aggregation.

To make this notion precise, we define a frame-wise (slice-wise) attention distribution. Let  $\mathcal{I}_{V,t}$  denote the set of video tokens corresponding to frame (or temporal slice)  $t$ . We define the normalized attention mass assigned by head  $(\ell, h)$  to slice  $t$  as  $p_t^{(\ell,h)} \triangleq \frac{\mathcal{A}_q^{(\ell,h)}(\mathcal{I}_{V,t})}{\mathcal{A}_q^{(\ell,h)}(\mathcal{I}_V) + \varepsilon}$ ,  $\sum_{t=1}^T p_t^{(\ell,h)} = 1$ . This slice distribution makes temporal specialization interpretable: a head that locks onto a single frame yields a highly peaked  $p_t$ , whereas genuine cross-frame integration yields a broader distribution.

To quantitatively characterize temporal specialization, we measure both the *breadth* of the slice distribution (how many frames receive non-trivial attention) and its *temporal span* (how far that attention extends along time). We first quantify how many temporal slices contribute non-trivially using the effective number of slices defined via the Hill number of order 2 (inverse Simpson index) (Somerville et al., 2008):

$$n_{\text{eff}}^{(\ell,h)} \triangleq \frac{1}{\sum_{t=1}^T \left(p_t^{(\ell,h)}\right)^2 + \varepsilon}. \quad (4)$$

which captures the breadth of temporal support:  $n_{\text{eff}} \approx 1$  indicates near-exclusive focus on a single slice, while larger values indicate that attention is distributed across frames.

To capture *how far* evidence spans along time, we compute the second central moment of the slice distribution. Using normalized slice indices  $\tau_t \triangleq \frac{t-1}{T-1} \in [0, 1]$  (for comparability across different  $T$ ), we define:

$$\sigma_{\tau}^{(\ell,h)} = \sqrt{\sum_{t=1}^T p_t^{(\ell,h)} (\tau_t - \mu^{(\ell,h)})^2}, \text{ where } \mu^{(\ell,h)} = \sum_{t=1}^T p_t^{(\ell,h)} \tau_t. \quad (5)$$

While  $n_{\text{eff}}$  measures the breadth of the support, standard deviation  $\sigma_{\tau}$  distinguishes local multi-slice smoothing from long-range temporal integration.

**Temporal head selection.** We designate a head  $(\ell, h)$  as *temporal* only if it consistently (i) routes non-trivial attention mass to the video stream, (ii) draws support from multiple

frames, and (iii) spans a sufficiently wide temporal range. Concretely, we require

$$\bar{m}^{(\ell,h)} \geq \tau_m, \quad \overline{n_{\text{eff}}}^{(\ell,h)} \geq \tau_{\text{eff}}, \quad \bar{\sigma}_{\tau}^{(\ell,h)} \geq \tau_{\sigma}, \quad (6)$$

All thresholds  $(\tau_m, \tau_{\text{eff}}, \tau_{\sigma})$  and metric ablations are further specified in Appendix C, and the full selection pipeline is illustrated in Fig. 4.

## 4.2 Temporal Head Amplification

Having identified a sparse set of temporally specialized heads  $\mathcal{S}_{\text{temp}}$  using the routing and dispersion criteria in Eq. (6), we now test whether strengthening these heads causally improves temporal understanding. Guided by Circuit ②, which localizes the semantic consolidation interval, we restrict amplification to the critical layer range  $\mathcal{L}_{\text{crit}}$  (typically mid-to-late layers) where video-conditioned concepts become language-aligned and most likely to influence decoding. For an attention head  $h$  at layer  $\ell$ , let  $\mathbf{O}^{(\ell,h)} \in \mathbb{R}^{d_k}$  denote its output vector after attention but before the final linear projection  $\mathbf{W}_O$ . We apply a scaling factor  $\lambda$  to amplify the contribution of identified temporal heads, i.e.

$$\tilde{\mathbf{O}}^{(\ell,h)} = \begin{cases} \lambda \cdot \mathbf{O}^{(\ell,h)}, & \text{if } (\ell, h) \in \mathcal{S}_{\text{temp}} \text{ and } \ell \in \mathcal{L}_{\text{crit}} \\ \mathbf{O}^{(\ell,h)}, & \text{otherwise} \end{cases}$$

where  $\alpha$  is a hyperparameter controlling the intervention strength. Crucially, this operation amplifies the *magnitude* of the temporal evidence routed into the residual stream without altering the distribution of the attention weights itself, thereby preserving the head’s intrinsic routing logic while enhancing its impact on the final generation.

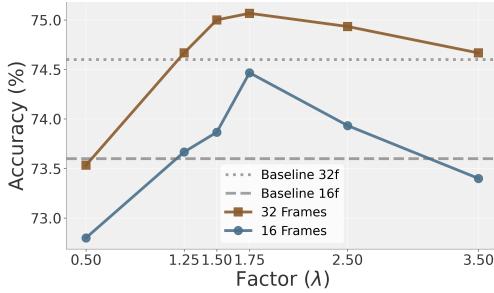


Figure 6. Sensitivity analysis of the head amplification factor  $\lambda$  across layer windows.

## 4.3 Experiments and Validation of Findings

**Experiment results.** As shown in Table 3, amplifying the identified temporal heads improves performance on TempCompass (Liu et al., 2024b) across all evaluated LVLMs, with gains reaching up to +2.4% absolute accuracy. These improvements are observed under both 16-frame and 32-frame settings, indicating that the intervention benefits genuine temporal reasoning rather than exploiting a specific

Table 3. Effectiveness of Temporal Head Amplification. We evaluate the proposed inference-time intervention on the TempCompass benchmark under 16- and 32-frame settings. Results are reported without retraining, following the protocols in §3.1.

Injection Layers	Qwen 2.5 VL		LLaVA-OV-V		LLaVA-OV-I	
	T = 16	T = 32	T = 16	T = 32	T = 16	T = 32
Baseline (None)	73.6	74.6	72.2	72.2	63.6	63.6
Layers 0–5	73.6 <small>±0.0</small>	74.2 <small>±0.4</small>	72.0 <small>±0.2</small>	71.4 <small>±0.8</small>	63.8 <small>±0.2</small>	63.6 <small>±0.0</small>
Layers 5–10	73.4 <small>±0.2</small>	74.4 <small>±0.2</small>	72.2 <small>±0.0</small>	72.4 <small>±0.2</small>	63.6 <small>±0.0</small>	63.8 <small>±0.2</small>
Layers 10–15	73.2 <small>±0.4</small>	74.2 <small>±0.4</small>	72.4 <small>±0.4</small>	72.2 <small>±0.0</small>	63.2 <small>±0.4</small>	63.8 <small>±0.2</small>
Layers 15–20	73.4 <small>±0.2</small>	74.8 <small>±0.2</small>	71.8 <small>±0.4</small>	72.2 <small>±0.0</small>	62.6 <small>±1.0</small>	63.2 <small>±0.4</small>
Layers 20–25	76.0 <small>±2.4</small>	75.4 <small>±0.8</small>	72.2 <small>±0.0</small>	72.6 <small>±0.4</small>	62.8 <small>±0.8</small>	62.6 <small>±1.0</small>
Layers 25–30	74.0 <small>±0.4</small>	75.0 <small>±0.4</small>	73.0 <small>±1.0</small>	72.2 <small>±0.0</small>	63.0 <small>±0.6</small>	63.2 <small>±0.4</small>

input length or configuration. Notably, these gains are achieved without modifying model parameters or retraining, demonstrating that temporal reasoning capacity already exists within the model and can be selectively strengthened at inference time, with the optimal scaling factor  $\alpha$  for these heads further illustrated in Fig. 6.

**Validation of CIRCUITPROBE findings.** Table 3 shows that gains concentrate in *mid-to-late* layers: injecting into earlier layers yields negligible improvements or even slight degradation, while later blocks (e.g., Layers 20–25/25–30) produce the most reliable uplift across models and frame counts. This layer locality is consistent with observation in Fig. 3: semantic concepts become strongly decodable only in mid-to-late layers, implying that temporal evidence routing is most effective once object semantics have been consolidated into language-aligned representations. In other words, early layers primarily transmit and normalize visual features, whereas later layers are where cross-slice evidence aggregation can directly influence answer tokens. More qualitative examples can be found in Appendix C.3, which further demonstrate the robustness of our findings.

## 5 Conclusion and Limitation

We propose CIRCUITPROBE to mechanistically dissect video LVLMs by (i) *Visual Auditing* that localizes task-critical object-aligned video tokens and causally verifies their necessity (text-label injection only partially recovers), and (ii) *Semantic Tracing* that shows video semantics become reliably language-aligned mainly in mid-to-late layers, where models often behave closer to evidence aggregation (often bag-of-frames-like) than explicit temporal-state modeling. Guided by these findings, we identify temporally dispersed, video-routing attention heads and selectively amplify them within the consolidation interval, yielding consistent inference-time gains on temporal-heavy benchmarks across backbones and frame budgets. However, we do not yet systematically quantify potential side effects such as calibration shifts or robustness under distribution shift. Finally, our evaluation covers a limited set of model families, and extending the analysis and interventions to broader architectures and larger models remains important future work.

## Impact Statement

This work aims to enhance our understanding of temporal reasoning in LVLMs by dissecting their ability to interpret and process temporal information across frames. Through detailed analysis of perturbation behaviors, we uncover how LVLMs rely on temporal cues and the limitations they face in modeling complex temporal dynamics. By proposing targeted interventions like temporal head identification and amplification, we seek to advance the design of LVLMs that can robustly handle temporal dependencies, ensuring more accurate and context-aware multimodal understanding. We advocate for the integration of temporal reasoning as a key aspect of model alignment, urging the research community to prioritize this alongside other capabilities, fostering the development of more reliable, robust, and ethically sound AI systems.

## References

- Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.
- Chen, J., Zhu, D., Shen, X., Li, X., Liu, Z., Zhang, P., Krishnamoorthi, R., Chandra, V., Xiong, Y., and Elhoseiny, M. MiniGPT-v2: Large language model as a unified interface for vision-language multi-task learning, 2023. URL <https://arxiv.org/abs/2310.09478>.
- Dar, G., Geva, M., Gupta, A., and Berant, J. Analyzing transformers in embedding space. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 16124–16170, 2023.
- Darcet, T., Oquab, M., Mairal, J., and Bojanowski, P. Vision transformers need registers, 2024. URL <https://arxiv.org/abs/2309.16588>.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255. Ieee, 2009.
- Geva, M., Schuster, R., Berant, J., and Levy, O. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5484–5495, 2021.
- Hanna, M., Liu, O., and Variengien, A. How does gpt-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model. *Advances in Neural Information Processing Systems*, 36:76033–76060, 2023.
- Jiang, N., Kachinthaya, A., Petryk, S., and Gandelsman, Y. Interpreting and editing vision-language representations to mitigate hallucinations, 2025. URL <https://arxiv.org/abs/2410.02762>.
- Joseph, S., Suresh, P., Hufe, L., Stevenson, E., Graham, R., Vadi, Y., Bzdok, D., Lapuschkin, S., Sharkey, L., and Richards, B. A. Prisma: An open source toolkit for mechanistic interpretability in vision and video. *arXiv preprint arXiv:2504.19475*, 2025.
- Kaduri, O., Bagon, S., and Dekel, T. What's in the image? a deep-dive into the vision of vision language models, 2024. URL <https://arxiv.org/abs/2411.17491>.
- Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Zhang, P., Li, Y., Liu, Z., and Li, C. Llava-onevision: Easy visual task transfer, 2024a. URL <https://arxiv.org/abs/2408.03326>.
- Li, F., Zhang, R., Zhang, H., Zhang, Y., Li, B., Li, W., Ma, Z., and Li, C. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. *arXiv preprint arXiv:2407.07895*, 2024b.
- Li, J., Li, D., Xiong, C., and Hoi, S. C. H. BLIP: bootstrapping language-image pre-training for unified vision-language understanding and generation. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvári, C., Niu, G., and Sabato, S. (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 12888–12900. PMLR, 2022. URL <https://proceedings.mlr.press/v162/li22n.html>.
- Li, L., Liu, Y., Yao, L., Zhang, P., An, C., Wang, L., Sun, X., Kong, L., and Liu, Q. Temporal reasoning transfer from text to video. *arXiv preprint arXiv:2410.06166*, 2024c.
- Liu, H., Li, C., Li, Y., and Lee, Y. J. Improved baselines with visual instruction tuning, 2023a.
- Liu, H., Li, C., Wu, Q., and Lee, Y. J. Visual instruction tuning, 2023b.
- Liu, H., Li, C., Li, Y., Li, B., Zhang, Y., Shen, S., and Lee, Y. J. Llava-next: Improved reasoning, ocr, and world knowledge, January 2024a. URL <https://llava-vl.github.io/blog/2024-01-30-llava-next/>.
- Liu, Y., Li, S., Liu, Y., Wang, Y., Ren, S., Li, L., Chen, S., Sun, X., and Hou, L. Tempcompass: Do video llms really understand videos? *arXiv preprint arXiv: 2403.00476*, 2024b.
- Maaz, M., Rasheed, H., Khan, S., and Khan, F. S. Video-ChatGPT: Towards Detailed Video Understanding via Large Vision and Language Models, 2023. URL <https://arxiv.org/abs/2306.05424>.

- Meng, K., Bau, D., Andonian, A., and Belinkov, Y. Locating and editing factual associations in GPT. *Advances in Neural Information Processing Systems*, 35, 2022.
- Merullo, J., Castricato, L., Eickhoff, C., and Pavlick, E. Linearly mapping from image to text space, 2023. URL <https://arxiv.org/abs/2209.15162>.
- Moon, S., Madotto, A., Lin, Z., Nagarajan, T., Smith, M., Jain, S., Yeh, C.-F., Murugesan, P., Heidari, P., Liu, Y., Srinet, K., Damavandi, B., and Kumar, A. AnyMAL: An Efficient and Scalable Any-Modality Augmented Language Model, 2023. URL <https://arxiv.org/abs/2309.16058>.
- Neo, C., Ong, L., Torr, P., Geva, M., Krueger, D., and Barez, F. Towards Interpreting Visual Information Processing in Vision-Language Models, 2024. URL <https://arxiv.org/abs/2410.07149>.
- Nostalgiaist. Interpreting GPT: The logit lens - AI Alignment Forum, 2020. URL <https://www.alignmentforum.org/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens>.
- Pan, X., Shukla, S. N., Singh, A., Zhao, Z., Mishra, S. K., Wang, J., Xu, Z., Chen, J., Li, K., Juefei-Xu, F., et al. Transfer between modalities with metaqueries. *arXiv preprint arXiv:2504.06256*, 2025.
- Pang, Z., Xie, Z., Man, Y., and Wang, Y. Frozen transformers in language models are effective visual encoder layers. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL <https://openreview.net/forum?id=t0FI3Q66K5>.
- Park, J., Lee, J., and Sohn, K. Bridging vision and language spaces with assignment prediction. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL <https://openreview.net/forum?id=1K2V2E2MNv>.
- Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021. URL <https://api.semanticscholar.org/CorpusID:231591445>.
- Rajaram, A., Chowdhury, N., Torralba, A., Andreas, J., and Schwettmann, S. Automatic discovery of visual circuits, 2024. URL <https://arxiv.org/abs/2404.14349>.
- Sharma, A. S., Atkinson, D., and Bau, D. Locating and editing factual associations in mamba. *arXiv preprint arXiv:2404.03646*, 2024.
- Singh, C., Inala, J. P., Galley, M., Caruana, R., and Gao, J. Rethinking interpretability in the era of large language models. *ArXiv*, abs/2402.01761, 2024. URL <https://api.semanticscholar.org/CorpusID:267412530>.
- Somerfield, P., Clarke, K., and Warwick, R. Simpson index. In *Encyclopedia of ecology*, pp. 3252–3255. Elsevier, 2008.
- Tong, S., Liu, Z., Zhai, Y., Ma, Y., LeCun, Y., and Xie, S. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pp. 9568–9578. IEEE, 2024. doi: 10.1109/CVPR52733.2024.00914. URL <https://doi.org/10.1109/CVPR52733.2024.00914>.
- Verma, G., Choi, M., Sharma, K., Watson-Daniels, J., Oh, S., and Kumar, S. Cross-modal projection in multimodal llms doesn't really project visual attributes to textual space. In *Annual Meeting of the Association for Computational Linguistics*, 2024. URL <https://api.semanticscholar.org/CorpusID:268032058>.
- Vilas, M. G., Schaumlöffel, T., and Roig, G. Analyzing vision transformers for image classification in class embedding space, 2023. URL <https://arxiv.org/abs/2310.18969>.
- Wang, K., Zhang, G., Zhou, Z., Wu, J., Yu, M., Zhao, S., Yin, C., Fu, J., Yan, Y., Luo, H., et al. A comprehensive survey in llm (-agent) full stack safety: Data, training and deployment. *arXiv preprint arXiv:2504.15585*, 2025a.
- Wang, R., Si, S., Yu, F., Rothuizen, D. W., Hsieh, C.-J., and Dhillon, I. S. Large language models are interpretable learners. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL <https://openreview.net/forum?id=hTphfqtafO>.
- Woo, S., Kim, D., Jang, J., Choi, Y., and Kim, C. Don't Miss the Forest for the Trees: Attentional Vision Calibration for Large Vision Language Models, 2024. URL <https://arxiv.org/abs/2405.17820>.
- Wu, B., Yu, S., Chen, Z., and Tenenbaum, J. B. STAR: A Benchmark for Situated Reasoning in Real-World Videos.
- Yao, Y., Zhang, N., Xi, Z., Wang, M., Xu, Z., Deng, S., and Chen, H. Knowledge circuits in pretrained transformers. *arXiv preprint arXiv:2405.17969*, 2024.

Yin, S., Fu, C., Zhao, S., Li, K., Sun, X., Xu, T., and Chen, E.  
A Survey on Multimodal Large Language Models, 2023.  
URL <https://arxiv.org/abs/2306.13549>.

Yu, H., Zhao, Z., Yan, S., Korycki, L., Wang, J., He, B., Liu, J., Zhang, L., Fan, X., and Yu, H. Cafe: Unifying representation and generation with contrastive-autoregressive finetuning. *arXiv preprint arXiv:2503.19900*, 2025.

Yu, Z. and Ananiadou, S. Interpreting arithmetic mechanism in large language models through comparative neuron analysis. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 3293–3306, 2024.

Zhang, Y., Zhao, Z., Chen, Z., Chen, Z., Feng, Z., Ding, Z., and Sun, Y. Rankclip: Ranking-consistent language-image pretraining. *ArXiv*, abs/2404.09387, 2024a. URL <https://api.semanticscholar.org/CorpusID:269149157>.

Zhang, Y., Zhao, Z., Chen, Z., Ding, Z., Yang, X., and Sun, Y. Beyond training: Dynamic token merging for zero-shot video understanding. *ArXiv*, abs/2411.14401, 2024b. URL <https://api.semanticscholar.org/CorpusID:274166248>.

## Appendix

### A Analysis Setup

#### A.1 Question Probing Setup

We use two complementary question formats that probe object and action understanding to evaluate the impact of token-level interventions under controlled decoding conditions,

1. *Open-ended questions.* We formulate specific questions targeting objects and actions depicted in the video (e.g., “Which object did the person throw in the video?”). These questions are designed to be answerable solely based on the provided video key frames. To ensure controlled comparison across ablation conditions, we pre-fill the model’s response with a fixed prefix (e.g., “The object is”) and analyze the next generated token before and after intervention.
2. *Close-ended questions.* We prompt the model with binary questions (e.g., “Is there a/an [object] in this video?”), and assess the impact of intervention by comparing the model’s next-token prediction. A flip from “Yes” to “No” after the intervention serves as evidence that the manipulation was crucial for the model’s reasoning and understanding of the input video.

#### A.2 Frames Visualization

This section visualizes sample input keyframes from the videos to provide a clearer understanding of the data used in our experiments. Each row in Figure 7 represents a distinct video sequence fed into the LVLMs for analysis. These examples are representative of the scenarios in our dataset, encompassing a variety of everyday actions, objects, and environments. By visualizing the raw inputs, we aim to illustrate the visual complexities, such as changes in viewpoint, object scale, and partial occlusions, that the model must handle to perform accurate semantic reasoning.



*Figure 7.* Visualization of Input Keyframes. Each row displays a sequence of frames provided to the model as input for a specific video. The red bounding boxes highlight the ground-truth object pertinent to the task’s question (e.g., the object being picked up, kicked, or taken). It is important to note that these bounding boxes are included here for clarity and were not provided to the model during inference.

## B Supplementary Experiments and Details: Circuit ① and ②

### B.1 Logit lens Setup

Our analysis focuses exclusively on the internal representations of projected visual tokens as they propagate through the LLM backbone, before any autoregressive generations. Let  $\mathcal{I}_O$  denote the set of object-centric visual token indices identified in Circuit ①, and let  $n_O \triangleq |\mathcal{I}_O|$ . For each token position  $i \in \mathcal{I}_O$ , we denote by  $h_i^{(\ell)} \in \mathbb{R}^d$  the hidden state at layer  $\ell$ . At each layer, we project this hidden state into the vocabulary space using the frozen language-model output projection  $W_{LM}$ , defining a layer-wise token distribution

$$P(\cdot | h_i^{(\ell)}) = \text{softmax}(W_{LM} h_i^{(\ell)}). \quad (7)$$

For each sample, we define a target semantic concept  $s_c$  corresponding to the ground-truth object or action referenced by the question (e.g., “microwave”). If  $s_c$  is tokenized into multiple sub-word tokens, we define the target token set  $\mathcal{T}(s_c) \subseteq \mathcal{V}$  accordingly, and use

$$P(s_c | h_i^{(\ell)}) \triangleq \sum_{w \in \mathcal{T}(s_c)} P(w | h_i^{(\ell)}). \quad (8)$$

When  $s_c$  maps to a single token,  $\mathcal{T}(s_c) = \{w_c\}$  and the definition reduces to  $P(w_c | h_i^{(\ell)})$ .

### B.2 Scaling Experiments

We conducted supplementary experiments to verify that our conclusions generalize to larger-scale models. We replicated our core analyses on the LLaVA-NeXT-34B model variants, with results that closely mirror those presented in the main body of the paper. The visual token ablation study on the 34B models reaffirms the principle of spatial localization for semantic information. Ablating object-specific tokens incurs significantly more substantial performance degradation than removing larger quantity of random tokens (in Table 4).

Furthermore, our semantic tracing analysis on the 34B architecture, depicted in Fig. 8, reveals a conceptual emergence pattern consistent with our earlier observations. Both the Correspondence Rate and Answer Probability remain negligible through the initial layers before exhibiting a sharp, concurrent rise beginning around layer 40. This trend indicates that abstract, language-aligned concepts are consolidated in the deeper layers of the network, irrespective of model scale. These scaling experiments provide robust evidence that the mechanisms of semantic localization and late-stage conceptual formation are fundamental properties of the tested LVLM architectures.

Ablation	Tokens	LLaVA-NeXT-34B-I		LLaVA-NeXT-34B-V	
		Type	Number	Open	Close
<i>Control Groups (Low Tokens)</i>					
Baseline	0			100.0 <sub>±0</sub>	100.0 <sub>±0</sub>
Register	13			55.3 <sub>±44.68</sub>	96.9 <sub>±3.14</sub>
<i>Object-based Ablation</i>					
Object	304			9.3 <sub>±90.75</sub>	29.8 <sub>±70.16</sub>
	413			5.8 <sub>±94.24</sub>	21.1 <sub>±78.88</sub>
	573			5.8 <sub>±94.24</sub>	18.0 <sub>±82.02</sub>
<i>Control Groups (High Tokens)</i>					
Random	100			54.3 <sub>±45.72</sub>	96.9 <sub>±3.14</sub>
	350			55.5 <sub>±44.5</sub>	96.5 <sub>±3.49</sub>
	500			54.5 <sub>±45.55</sub>	96.3 <sub>±3.66</sub>
	900			59.9 <sub>±40.14</sub>	96.8 <sub>±3.17</sub>
					78.3 <sub>±21.67</sub>
					29.7 <sub>±70.33</sub>

Table 4. Accuracy (%) from visual token ablation on question-answering performance across 34B models. The  $\downarrow$  symbol indicates the magnitude of this performance drop.

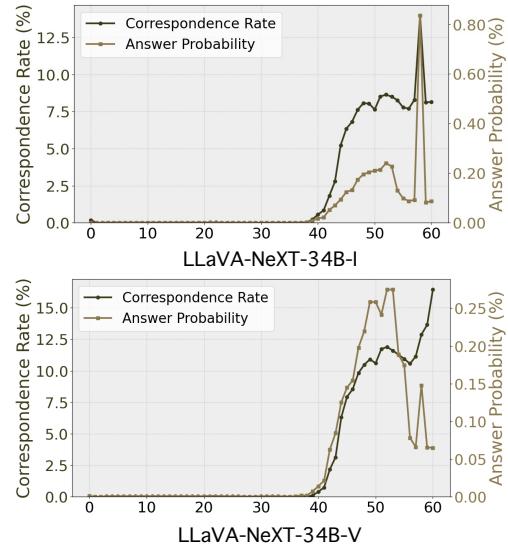


Figure 8. Quantitative analysis of semantic tracing on 34B model size.

### B.3 Attention Knockout

Our earlier token-level interventions establish that task-relevant object semantics are spatially localized in the video embedding (Circuit ①). However, localization alone does not clarify *how* such evidence is routed through the LLM backbone when producing an answer, nor whether temporal reasoning can be supported by object-centric evidence in isolation. This appendix section therefore characterizes the *attention flow* that supports the *first generated token*, with the goal of providing auxiliary evidence for the layer-localization claims used in Circuit ②.

**Setup and masking protocol.** We conduct all experiments on the LLaVA-NeXT family. To control information flow during prediction of the first output token, we apply an attention knockout mask that blocks attention edges from a specified source token set to the final answer token, restricted to a chosen layer window. We partition layers into five windows (Early, Early-to-Mid, Mid, Mid-to-Late, Late), and apply the same mask to *all heads* within the selected window.

We consider two complementary masking regimes: (i) **object-centric masking** blocks attention from object-region tokens and buffers ( $O, O+1, O+2$ ) to the final token; (ii) **contextual masking** blocks attention from all non-object tokens and their complements ( $C, C+1, C+2$ ) to the final token. For each configuration, we report (a) **accuracy drop** (Table 5) and (b) **answer probability drop** for the correct token (Fig. 9).

**Analysis.** This attention-flow diagnostic is designed to *triangulate* the conclusions of Circuits ①–② by asking a different question: given that object evidence is spatially localized in the video embedding (Circuit ①), *which* information source (object vs. context) is actually *consumed* by the LLM backbone, and *where* in depth this consumption is most critical when forming the first answer token. Table 5 and Fig. 9 show a consistent separation between the roles of contextual versus object-centric information for first-token prediction.

**From localization to utilization.** Circuit ① shows that ablating object-region tokens catastrophically harms object identification, establishing that object semantics are indeed present and localized. In contrast, Table 5 and Fig. 9 show that *blocking access to non-object context* produces a substantially larger degradation than blocking access to the object region when predicting the first token. Taken together, these results refine the interpretation of Circuit ①: although object semantics are necessary and localized, *they are not sufficient for the model’s initial reasoning state*; the model must first read broader contextual evidence to decide *how* to use object information.

**Depth-specific division of labor.** The layer-window results provide a depth-resolved counterpart to Circuit ②. Circuit ② (logit-lens tracing) indicates that semantically decodable representations emerge sharply in mid-to-late layers. Consistently, we find that contextual masking is most harmful in early-to-mid windows, while object-centric masking becomes relatively more harmful in mid-to-late windows. This alignment supports a two-stage computation: (i) early-to-mid layers integrate global/contextual cues to establish a coarse reasoning scaffold, and (ii) mid-to-late layers selectively incorporate fine-grained object evidence as semantic representations become language-aligned, directly shaping the answer distribution.

Overall, attention-flow analysis offers a complementary lens that links (1) *spatial localization* of object evidence (Circuit ①) and (2) *depth-wise emergence* of language-aligned semantics (Circuit ②) to a concrete, causal *routing* process about which evidence is accessed at which depth during generation.

Model	Condition (From*)	Early	Early-Mid	Mid	Mid-Late	Late	All Layers
<i>Object-centric masking</i>							
LLaVA-NeXT-I	O	0.19	0.20	0.24	<b>0.25</b>	0.23	0.29
	O+1	0.20	0.23	0.25	<b>0.28</b>	0.27	0.35
	O+2	0.23	0.23	0.26	<b>0.28</b>	0.26	0.39
LLaVA-NeXT-V	O	0.25	0.26	0.27	<b>0.30</b>	0.27	0.32
	O+1	0.25	0.26	0.29	<b>0.33</b>	0.29	0.38
	O+2	0.25	0.26	0.31	<b>0.33</b>	0.32	0.42
<i>Contextual masking</i>							
LLaVA-NeXT-I	C	0.82	<b>0.99</b>	0.89	0.66	0.44	0.99
	C+1	0.83	<b>0.97</b>	0.86	0.59	0.39	0.97
	C+2	0.83	<b>0.97</b>	0.86	0.54	0.39	0.97
LLaVA-NeXT-V	C	0.49	<b>0.91</b>	0.89	0.92	0.62	0.97
	C+1	0.47	<b>0.90</b>	0.86	0.88	0.56	0.94
	C+2	0.46	<b>0.90</b>	0.83	0.87	0.52	0.92

Table 5. Accuracy drop for LLaVA-NeXT models across various layer windows after attention knockout. Masking conditions: **O**, object tokens; **O+1/2**, object with 1/2 buffer; **C**, all tokens except the object; **C+1/2**, all tokens except the object with 1/2 buffer.

#### B.4 Interactive Examples

We provide additional qualitative examples to visually illustrate the findings from our circuit-based analysis. These examples showcase the model’s process of interpreting video frames to answer specific questions about objects and actions. Each figure includes the input video frames, the posed question, the model’s response, and a table showing the semantic evolution of key tokens across different layers, as analyzed through our semantic tracing circuit (Circuit ②). More interactive examples in the anonymous interactive [demo website](#).



**Prompt**

**USER:** The input consists of a sequence of key frames from a video. Question: Which object was taken by the person?  
**ASSISTANT:** The object is the

**Answer:** Towel

**Response:** Towel

Token/Layer	Layer 1	Layer 2	Layer 3	Layer 4	Layer 5
<s>	Institution	Arхив	sierp	sierp	sierp
US	она	Архив	pert	tap	pert
ER	Session	пуга	oded	庄	pel
:	Portail	пуга	<>	<>	Sav
		пуга	Carter	❖	❖
<IMG001>	sak	sak	sak	sak	sak
<IMG002>	gresql	gresql	gresql	gresql	gresql
<IMG003>	olas	olas	olas	olas	olas
<IMG004>	<>	<>	<>	<>	<>
<IMG005>	gresql	gresql	gresql	gresql	gresql
<IMG006>	sak	sak	sak	sak	sak
<IMG007>	yter	yter	yter	yter	yter
<IMG008>	nahm	nahm	nahm	nahm	nahm
<IMG009>	loop	loop	loop	loop	sak
<IMG010>	loop	loop	loop	loop	loop
<IMG011>	gresql	gresql	gresql	gresql	gresql
<IMG012>	yter	yter	yter	yter	yter
<IMG013>	izi	izi	Emp	Emp	conf
<IMG014>	alle	alle	alle	alle	conf
<IMG015>	yter	yter	yter	yter	yter
<IMG016>	alle	alle	alle	arab	anno
<IMG017>	yter	yter	yter	yter	yter

Figure 10. Qualitative example of the model correctly identifying an object. The user asks which object was taken by the person. The model correctly identifies the "Towel". The accompanying table shows the layer-by-layer semantic tracing for visual and text tokens.

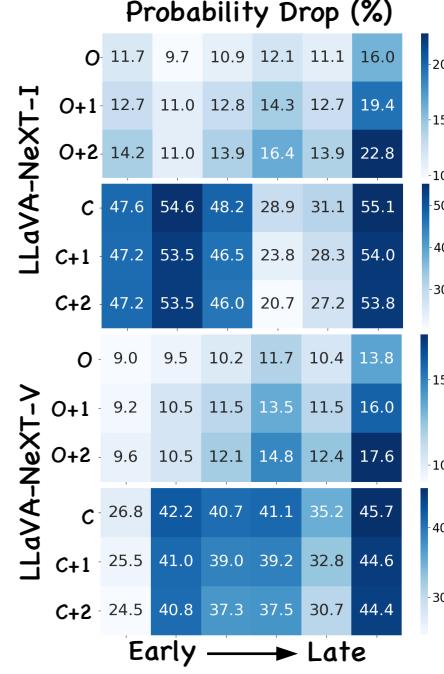


Figure 9. Answer probability drop under attention knockout across various layers.

Token/Layer	Layer 1	Layer 2	Layer 3	Layer 4	Layer 5
<>	Institution	Архив	sierp	sierp	sierp
US	ona	Архив	pert	tap	pert
ER	Session	пуга	oded	庄	pel
:	Portal	пуга	<>	<>	Sav
		пуга	Carter	❖	❖
<IMG001>	yter	yter	yter	yter	yter
<IMG002>	yter	yter	yter	yter	yter
<IMG003>	middle	middle	middle	middle	middle
<IMG004>	yter	yter	yter	yter	yter
<IMG005>	yter	yter	yter	yter	yter
<IMG006>	kwiet	kwiet	kwiet	kwiet	kwiet
<IMG007>	onymes	onymes	onymes	onymes	stycz
<IMG008>	dek	dek	dek	dek	dek
<IMG009>	yter	yter	yter	yter	yter
<IMG010>	assen	assen	assen	assen	assen
<IMG011>	wheel	wheel	wheel	wheel	wheel
<IMG012>	assen	assen	assen	emann	emann
<IMG013>	alet	alet	alet	empio	empio
<IMG014>	gresql	gresql	gresql	gresql	gresql
<IMG015>	alet	alet	alet	alet	alet
<IMG016>	azon	azon	azon	azon	azon
<IMG017>	fle	fle	fle	fle	wi

**Prompt**  
**USER:** The input consists of a sequence of key frames from a video. Question: Which object was thrown by the person?  
**ASSISTANT:** The object is the

**Answer:** Clothes

**Response:** Green shirt that the person is throwing.

Figure 11. Qualitative example where the model is prompted to identify a thrown object. The model successfully responds that a "Green shirt" was thrown, correctly identifying both the object and its color. The table illustrates the semantic trace, showing how the model processes the visual information through its layers.

## C Surgery Details and Setup

### C.1 Threshold Setting

This section details the thresholding protocol used to identify *temporal heads* (Eq. 6) and the hyperparameters used for head amplification. We report all thresholds and aggregation choices to ensure the surgery procedure is fully reproducible. All thresholds ( $\tau_m, \tau_{\text{eff}}, \tau_\sigma, \tau_{\text{occ}}$ ) are selected on a held-out development split to avoid test leakage. Concretely, we use a percentile-based calibration to balance (i) head sparsity and (ii) downstream gains under surgery. Across the explored configurations, we find that the best-performing setting is  $\tau_m = 0.1$ ,  $\tau_{\text{eff}} = 2.0$ , and  $\tau_\sigma = 0.8, \tau_{\text{occ}} = 0.6$  which achieves the strongest accuracy improvement while maintaining a compact headset.

### C.2 Metric Ablation

The head-type ablation in Table 6 shows that partial criteria (e.g., using only  $(m, n_{\text{eff}})$  or  $(m, \sigma_\tau)$ ) do not reproduce the full gains, and random head amplification fails to help. This indicates that the improvement is not explained by a generic increase in attention capacity; rather, it depends on jointly satisfying (i) *video gating* and (ii) *broad, long-range temporal support*. Together, these results validate our profiling metrics as identifying a functionally meaningful subset of heads that implement time-spanning evidence routing.

### C.3 Qualitative Examples

In this section, we present qualitative comparisons to intuitively demonstrate the effectiveness of our Temporal Head Amplification. As shown in Fig. 13 and Fig. 14, we display the key video frames, the input question, and the model's responses under two conditions: (1) the *Baseline* (without intervention), and (2) our *Intervened* model (with temporal head amplification). These cases highlight how our circuit-guided intervention effectively corrects temporal hallucinations and rectifies errors in chronological reasoning, enabling the model to accurately capture action sequences and event dependencies where the baseline fails.

Head Type	Frames=16	Frames=32
Baseline	73.6	74.6
(mass, neff)	73.6	74.2
(mass, $\sigma$ )	73.7	74.8
(neff, $\sigma$ )	73.4	74.1
Random	73.2	74.2
Temporal head	75.2	75.5

Table 6. Ablation study of temporal head type.

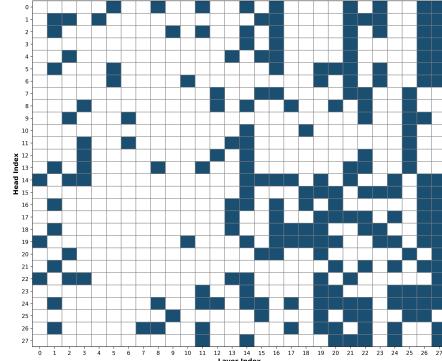


Figure 12. Layer head distribution heatmap.



**Question:** What is the direction of the camera motion relative to the aircraft carrier?

**Options:** A. counterclockwise B. downward C. upward D. clockwise

**Correct Answer:** A

**Qwen2.5VL:** The camera appears to be moving in a clockwise direction around the aircraft carrier, as we can see the carrier from different angles and perspectives throughout the video. Therefore, **the answer is D.**

**Qwen2.5VL+ Attention Intervention:** **The answer is A.** The aircraft carrier is moving in a counterclockwise direction. The aerial view shows the aircraft carrier from above, and the movement of the carrier can be observed by the changing position of the aircraft on the deck. The aircraft carrier is moving in a counterclockwise direction as it is shown

Figure 13. **Qualitative comparison on Visual Dynamics (Camera Motion).** The question asks for the direction of camera motion relative to the aircraft carrier. The **Baseline** model suffers from directional hallucination, incorrectly inferring a *clockwise* motion despite clear visual evidence. In contrast, our **Intervened** model correctly grounds the changing perspective over time, accurately identifying the *counterclockwise* motion (Option A). This demonstrates the efficacy of our intervention in rectifying errors related to continuous spatiotemporal changes.



**Question:** What is the order of events in the video?

**Options:**

- A. An apple is drying out and then a big wave is shown
- B. A big wave is shown and then an apple is drying out
- C. At the same time, an apple is drying out and a big wave is shown

**Correct Answer: C**

**Qwen2.5VL:** The correct answer is:\n\nA. An apple is drying out and then a big wave is shown\n\nThe video shows a sequence where first, an apple is depicted in various stages of drying out, and then it transitions to showing a large wave.

**Qwen2.5VL+ Attention Intervention:** The correct answer is:\n\nC. At the same time, an apple is drying out and a big wave is shown\n\nThe video shows a half of an apple that appears to be drying out, with the background showing a large wave.

**Figure 14. Qualitative comparison on the Sequence Task (Event Ordering).** The question probes the chronological order of two visual events: an apple drying out and a wave. The **Baseline** fails to recognize the split-screen simultaneity, hallucinating a sequential transition ("...and then a big wave is shown"). The **Intervened** model effectively suppresses this prior bias, correctly reasoning that the events occur "At the same time" (Option C). This highlights our method's ability to correct temporal dependency errors where the model defaults to a linear narrative.