WORLDPATCH: RETAINED PREDICTIVE STATES FOR ANTICIPATORY MULTIMODAL DECODING

Anonymous authors
Paper under double-blind review

ABSTRACT

Predictive representation learning can use future observations to shape visual representations, yet the predicted state is often discarded before downstream decoding. Consequently, gains from future supervision do not reveal whether predictive training merely improves the representation of the present or whether the predicted state itself is useful when retained. We introduce WorldPatch, which predicts a short-horizon latent successor from the current image and retains it alongside the present representation for decoding. Paired future images supervise the successor during training but are unavailable at inference. We vary decoder access under matched predictive supervision to distinguish the effects of retaining the predicted successor from those of predictive training. On EPIC-KITCHENS-100, WorldPatch improves anticipatory decoding over an auxiliary-only model, with consistent gains on a second backbone and on Ego4D. Further analysis on the primary backbone shows that retaining the successor yields larger gains when current and future action labels differ. These findings show that future prediction can be useful beyond its role as a training objective: retaining its output can improve short-horizon multimodal decoding.

1 INTRODUCTION

Reasoning from visual observations often requires understanding not only what is present, but also what is likely to happen next. A single image can contain cues to an imminent interaction or state change—a hand poised over an object or a tool positioned for use. Multimodal language models can describe and reason about observed scenes (Liu et al., 2023; Li et al., 2023a; Zhu et al., 2023), yet their current-image interfaces typically expose only a representation of the observed scene to the decoder, not an explicit predicted successor state. The decoder is therefore left to infer any near-term continuation implicitly, even when the question itself concerns what comes next.

Predictive representation learning offers a natural way to make such continuations explicit. Joint-embedding predictive methods learn by predicting latent targets rather than reconstructing pixels (Assran et al., 2023), while video feature-prediction and latent-dynamics models extend this principle across time (Bardes et al., 2024; Wu et al., 2023). These methods show that latent prediction can shape useful visual representations. Future-L1 learns latent visual states using future-frame supervision and retains them to condition subsequent generation (Jiang et al., 2026). Video MLLMs make temporal evidence directly available to the decoder by conditioning on multiple observed frames at inference (Maaz et al., 2024; Lin et al., 2024a; Cheng et al., 2024). Unlike those frames, a short-horizon successor predicted from the current image does not add another observation. Its potential value lies in making a learned continuation directly available to the decoder. We ask whether retaining this prediction improves anticipatory decoding beyond what predictive training alone provides.

To study this question, we introduce WorldPatch (Figure 1), which keeps what the model observes separate from what it anticipates. The present state represents the observed scene, while a retained short-horizon successor represents the model's predicted continuation. Paired future images guide the learning of the successor but are withheld at inference, so the decoder receives a predicted continuation rather than a future observation.

Empirically, learning to predict and retaining the prediction are not interchangeable. On EPIC-KITCHENS-100 (Damen et al., 2022), WorldPatch with Slot-MLLM (Chi et al., 2025) increases the log-likelihood assigned to reference answers on anticipatory questions by an average of 0.032 over a

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

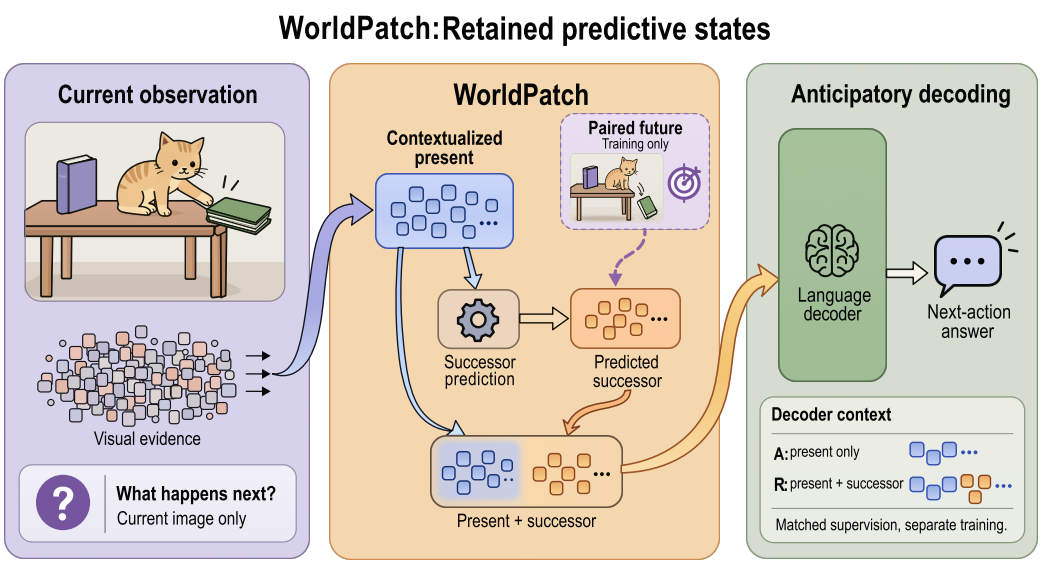


Figure 1: WorldPatch (R) retains a predicted successor alongside the contextualized present for anticipatory decoding. The auxiliary-only baseline (A) receives the same predictive supervision but decodes from the present alone.

matched auxiliary-only model trained with the same future-prediction objective. The corresponding gain is 0.026 with Qwen2.5-VL-7B-Instruct (Bai et al., 2025), and both backbones show improvements on Ego4D (Grauman et al., 2022). In the learned state space, predicted successors lie closer to their paired future states than the corresponding nonpredictive representations do. With WorldPatch's parameters held fixed, replacing its successor with a prediction from another example lowers the log-likelihood assigned to reference answers. More broadly, these findings identify decoder access to predicted states as a distinct design choice in predictive multimodal learning.

Our contributions are threefold. We introduce WorldPatch, a method for multimodal decoding that retains a predicted short-horizon successor alongside the present state. We show that retaining the successor improves anticipatory decoding beyond the benefits of predictive training alone. We find that the retained successor carries useful predictive content, with larger decoding gains when current and future action labels differ.

2 Related Work

2.1 Predictive Representations and Latent Dynamics

Predictive learning has served two related purposes: shaping visual representations and modeling how visual states evolve. Joint-embedding predictive methods learn transferable visual representations by predicting targets in latent space (Assran et al., 2023; Bardes et al., 2024; Li et al., 2026). World models learn to predict how observations or latent states evolve over time (Ha & Schmidhuber, 2018; LeCun, 2022; Bruce et al., 2024). Object-centric formulations represent this evolution over compact sets of entities (Kipf et al., 2021; Elsayed et al., 2022), supporting video prediction and planning (Wu et al., 2023). Across these lines of work, prediction is evaluated primarily through representation quality or rollout utility. Such evaluations leave open whether retaining the predicted state itself improves multimodal decoding from a single image beyond the benefits of predictive training alone.

2.2 Anticipation and Imagined Futures

Anticipation methods often predict future visual features as intermediate representations for recognition of upcoming actions (Girdhar & Grauman, 2021; Wu et al., 2021) or procedural-step forecasting and planning (Lin et al., 2025). Future-L1 introduces continuous latent visual spans that are aligned

with future-frame embeddings during training and retained to condition subsequent autoregressive generation (Jiang et al., 2026). It generates an interleaved, multi-step sequence of text and latent visual states from an observed video prefix, whereas WorldPatch predicts a single short-horizon successor from the current image and provides it to the decoder alongside the present state. Our focus is whether retaining this successor improves decoding under matched predictive supervision.

2.3 Visual Context in Multimodal Decoding

Multimodal language models provide visual context to language decoders through projected features, learned query tokens, or localized visual tokens (Liu et al., 2023; Li et al., 2023a; Ma et al., 2024). Object-centric variants organize observations into compact slots (Chi et al., 2025; Xu et al., 2024), while video MLLMs extend decoder context across multiple observed frames (Maaz et al., 2024; Lin et al., 2024a; Cheng et al., 2024). For video question answering, V-JEPA 2 (Assran et al., 2025) conditions a language model on projected features from a video encoder pretrained with a predictive objective. When the decoder context also includes a predicted state, however, improved performance does not by itself establish that the decoder benefits from the content of that prediction. The gain may instead reflect an improved present-state representation induced by predictive training or simply the presence of additional decoder context. We examine these explanations through matched model comparisons and interventions on the content of the retained successor.

3 Problem Formulation

Let I_t denote the current image, $I_{t+\Delta}$ its paired short-horizon future image, Q the query, and Y the target answer. During training, $I_{t+\Delta}$ is used only to construct the predictive target and is never included in the decoder context. The decoder is trained to predict Y from Q and states computed from I_t. At inference, the model receives only (I_t, Q), so any successor state must be predicted from the current image rather than obtained from a future observation. This information boundary distinguishes predictive supervision from future-frame conditioning and defines the setting we study.

At inference, the predicted successor $\widehat{S}_{t+\Delta}$ is a deterministic function of the current image and the learned parameters; it therefore provides no additional observation of the future. Its potential value is representational: future supervision may organize evidence already present in I_t into a short-horizon continuation that the decoder can use more readily. Let $\widetilde{S}_t$ denote the contextualized present state. We ask whether conditioning the decoder on both $\widetilde{S}_t$ and $\widehat{S}_{t+\Delta}$ improves prediction of Y under matched predictive supervision. A predictive-state interpretation requires evidence that the successor reflects the paired future rather than merely copying the present, and that its effect on decoding depends on its content.

To evaluate retained access under matched predictive supervision, let A denote an auxiliary-only condition that learns a successor but withholds it from the task decoder, and let R denote a matched condition that makes the predicted successor available during task training and inference. The two models are trained separately using the same backbone, data, and predictive objective. Let p_A and p_R denote the respective conditional response distributions. For $c \in \{A, R\}$, let GoldLL_c denote the length-normalized log-likelihood score for acceptable future answers under p_c, as defined in Section 5.1.

The retained-utility hypothesis, with the expectation taken over anticipatory examples, is

$$\Delta_{\mathrm{ret}} = \mathbb{E}[\mathrm{GoldLL}_R - \mathrm{GoldLL}_A] > 0. \tag{1}$$

We also examine whether this gain varies with agreement between current and future action labels. On current-action questions, we evaluate R against a present-only baseline P trained without predictive supervision, using a prespecified non-inferiority margin.

4 WorldPatch

Answering the question posed above requires the prediction to be represented as a state in its own right, distinct from the present. WorldPatch therefore preserves the contextualized present while introducing an additional state for its predicted short-horizon successor. Within this information

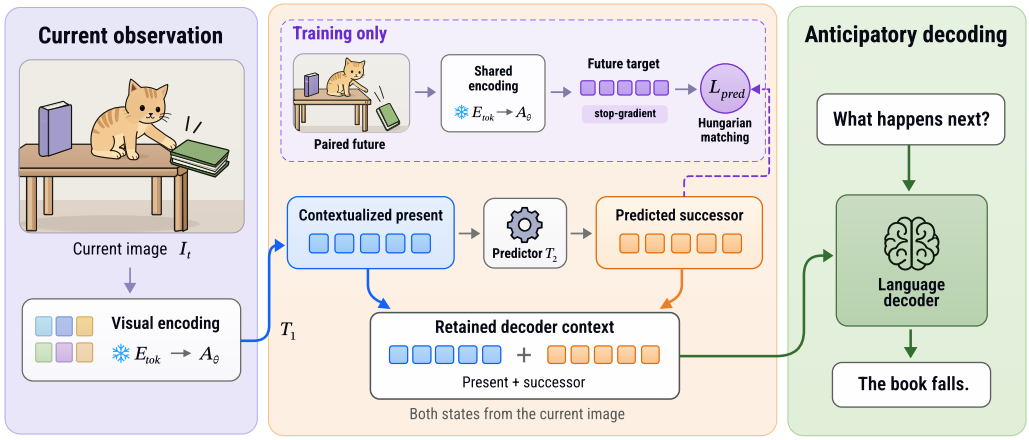


Figure 2: **Anticipatory decoding with WorldPatch.** WorldPatch retains a predicted short-horizon successor alongside the contextualized present state as decoder context. Both states are derived from the current image I_t; the paired future image is used only to form a stop-gradient training target.

boundary, future supervision guides the learning of the successor, whereas retaining it alongside the present makes it accessible to the decoder. WorldPatch thereby separates predictive supervision from predictive-state access, as summarized in Figure 2.

4.1 Present and Successor as Distinct States

The present and successor are defined asymmetrically: the former is derived from the current observation, whereas the latter is inferred from the former. Predicting a continuation first requires a coherent account of the configuration visible in the current image. WorldPatch therefore separates the formation of the present from the prediction of its continuation: observed evidence is contextualized to form the state on which prediction is based, and a separate successor is inferred from that state without replacing it. We form the contextualized present as

$$F_t = E_{\text{tok}}(I_t), \qquad S_t = A_\theta(F_t), \qquad \widetilde{S}_t = T_1(S_t). \tag{2}$$

The visual tokens F_t, produced by the frozen tokenizer E_{tok}, encode the evidence available in I_t. The structured representation S_t describes the present through a fixed-cardinality set of latent states, with learned queries determining how the visual evidence contributes to each state. Because the continuation concerns the joint configuration encoded by this set, T_1, a Transformer encoder without added slot-position embeddings, contextualizes each state with respect to the others. The resulting $\widetilde{S}_t$ represents the current configuration and forms the basis for inferring its successor. Given this state, the successor is defined as

$$\widehat{S}_{t+\Delta} = T_2(\widetilde{S}_t) = \text{LayerNorm}\Big(\widetilde{S}_t + \text{MLP}(\widetilde{S}_t)\Big)\,. \tag{3}$$

The residual form parameterizes $\widehat{S}_{t+\Delta}$ relative to $\widetilde{S}_t$, consistent with treating the successor as a short-horizon continuation of this configuration. The present remains explicit alongside this continuation, preserving the distinction between what is observed and what is predicted. This separation determines how the successor is represented, but not what predictive content it should encode.

4.2 Future Supervision under Current-Only Inference

Specifying the predictive content of the successor requires a reference to what follows, but the current-only interface excludes future evidence at inference. WorldPatch therefore derives this reference from a paired future observation during training while inferring the successor itself from I_t. The supervision is asymmetric: the paired future determines the target, whereas the current observation determines the prediction. The future reference is expressed in the shared state-abstraction space:

$$S^{+}_{t+\Delta} = \text{sg}[A_\theta(E_{\text{tok}}(I_{t+\Delta}))]\,, \tag{4}$$

where $S^{+}_{t+\Delta}$ is the fixed-cardinality state representation derived from the paired future and sg denotes stop-gradient. The predictive loss backpropagates through the successor branch, while gradients are stopped at the future reference. The shared abstraction module A_θ remains trainable through the current-observation pathway.

Because row order carries no prescribed semantics, supervision compares the predicted and target states as unordered collections rather than matching rows by index. Let $\hat{s}_i$ and s^{+}_i denote the respective rows of $\widehat{S}_{t+\Delta}$ and $S^{+}_{t+\Delta}$. We define

$$\pi^\star = \operatorname*{arg\,min}_{\pi \in \Pi_N} \sum_{i=1}^{N} \left\| \hat{s}_i - s^{+}_{\pi(i)} \right\|_2^2, \qquad \mathcal{L}_{\mathrm{pred}} = \frac{1}{Nd} \sum_{i=1}^{N} \left\| \hat{s}_i - s^{+}_{\pi^\star(i)} \right\|_2^2, \tag{5}$$

where Π_N is the set of permutations of the N rows, d is their dimensionality, and $\pi^\star$ is computed by the Hungarian algorithm (Kuhn, 1955). The resulting objective is invariant to row order and measures agreement between the two unordered state collections. Future supervision thus specifies what the successor is trained to encode while inference remains current-only.

4.3 Retaining the Successor for Decoding

A predicted successor used only in the predictive objective may shape the representation learned from the current image, but it does not enter the conditional distribution over responses. WorldPatch retains the prediction alongside $\widetilde{S}_t$, making both the contextualized present and its inferred short-horizon continuation explicit in the decoding context.

Although both states are retained, they play different temporal roles: $\widetilde{S}_t$ represents the contextualized present, whereas $\widehat{S}_{t+\Delta}$ represents its inferred continuation. We associate the two collections with learned type embeddings $e_{\mathrm{pres}}, e_{\mathrm{succ}} \in \mathbb{R}^d$:

$$C_t = \begin{bmatrix} \widetilde{S}_t + \mathbf{1}_N e_{\mathrm{pres}}^\top \\ \widehat{S}_{t+\Delta} + \mathbf{1}_N e_{\mathrm{succ}}^\top \end{bmatrix} \in \mathbb{R}^{2N \times d}, \tag{6}$$

where $\mathbf{1}_N$ is the N-dimensional vector of ones. The type embeddings distinguish the two collections without defining correspondences between their individual rows. The response is therefore conditioned on the joint present–successor context rather than on $\widetilde{S}_t$ alone. Its decoder-facing representation is given by

$$Z_t = P_{\mathrm{wp}}(C_t), \qquad H_t = P_{\mathrm{llm}}(Z_t), \tag{7}$$

where P_{wp} and P_{llm} denote the successive projection modules used to map C_t into the decoder embedding space. Given a query Q and a response sequence $Y = (y_1, \ldots, y_L)$, the conditional response distribution factorizes as

$$p(Y \mid Q, I_t) = \prod_{k=1}^{L} p(y_k \mid y_{<k}, Q, H_t)\,. \tag{8}$$

Training jointly aligns the successor with the future-derived reference and optimizes response likelihood conditioned on H_t. Let $\mathcal{L}_{\mathrm{task}}$ denote the autoregressive negative log-likelihood associated with Equation (8). The overall objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{task}} + \lambda_{\mathrm{pred}} \mathcal{L}_{\mathrm{pred}}, \tag{9}$$

where λ_{pred} weights the prediction loss. The future image determines the training target for the successor but is unavailable at inference. Both states retained for decoding are derived from I_t. Retaining the successor changes what the decoder conditions on, not what the model observes.

5 Experiments

Our experiments ask whether retaining a predicted successor improves anticipatory decoding beyond predictive training alone. We compare WorldPatch with an auxiliary-only baseline that receives the same predictive supervision but decodes from the present alone. To understand what the successor contributes, we examine whether it aligns with future target representations and how decoding changes when it is replaced by a prediction from another sample. We also assess whether retaining the successor yields larger gains when current and future action labels differ.

Table 1: Controlled comparisons on EPIC-KITCHENS-100 with Slot-MLLM. Scores are averaged over eight paired training seeds.

Condition	Predictive supervision	Second-state access	Direct task gradient	Future Gold-LL ↑
P	No	No	—	-1.222
A	Yes	No	—	-1.198
G	No	Yes	Yes	-1.210
R_{stop}	Yes	Yes	Stopped	-1.175
R	Yes	Yes	Yes	$\mathbf{-1.167}$

Access applies during training and evaluation; task gradients refer only to the second-state output.

5.1 Experimental Setup

Predictive training can shape the contextualized present even when the decoder does not receive the predicted successor. We therefore use an auxiliary-only baseline (A) that receives the same predictive supervision as WorldPatch (R). During both training and evaluation, A decodes from the present alone, whereas R decodes from both the present and the predicted successor. We make this comparison on both Slot-MLLM-7B-instruct (Chi et al., 2025) and Qwen2.5-VL-7B-Instruct (Bai et al., 2025), training A and R separately with matched data and training schedules within each backbone. Additional controls on Slot-MLLM examine the roles of predictive supervision and direct task gradients through the successor output (Table 1).

Both A and R answer questions about the future from the same current image, while future frames provide predictive supervision during training. On EPIC-KITCHENS-100 (Damen et al., 2022), each example pairs the last valid frame at least one second before an action begins with a reference frame from within that action. With the data split by video, the evaluation includes 8,327 frame pairs from 49 held-out videos. We test transfer to Ego4D v2 (Grauman et al., 2022) using the same models without further training or tuning. This transfer test covers 10,000 single-frame anticipation examples from 200 videos, derived from the FHO-LTA validation annotations.

We assess anticipatory decoding using both the likelihood assigned to acceptable future answers and the accuracy of generated answers. Future Gold-LL is a length-normalized log-likelihood score for acceptable future answers. Each answer is scored by its mean token log-probability, and scores for distinct acceptable answers are combined using log-mean-exp:

$$\ell(y) = \frac{1}{|y|} \sum_{r=1}^{|y|} \log p_\phi(y_r \mid I_t, Q, y_{<r}), \qquad \text{GoldLL} = \log \left(\frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}} \exp(\ell(y)) \right), \tag{10}$$

where $\mathcal{Y}$ denotes the set of distinct acceptable answers. Future-action exact match (EM) measures whether a generated answer matches any acceptable future reference. For both metrics, we average paired $R - A$ differences across eight training seeds per backbone, giving equal weight to participants on EPIC-KITCHENS-100 and to videos on Ego4D. Uncertainty in these differences is summarized by two-sided 95% confidence intervals from a paired hierarchical bootstrap.

5.2 Retained Access Beyond Predictive Training

Retaining the predicted successor improves anticipatory decoding beyond predictive training alone. On EPIC-KITCHENS-100, WorldPatch improves future Gold-LL over the auxiliary-only baseline by 0.032 on Slot-MLLM and 0.026 on Qwen2.5-VL (Table 2). The improvement is also reflected in generated answers: future-action EM increases by 6.0 and 4.0 percentage points on the two backbones, respectively. The gains extend to Ego4D without further training, with corresponding future Gold-LL improvements of 0.040 and 0.029. Future-action EM also increases by 2.5 percentage points on both backbones.

WorldPatch can improve anticipatory decoding without direct task gradients through the successor output. With this gradient path blocked (R_{stop}), it still outperforms the auxiliary-only baseline by

Table 2: Retained access across backbones and datasets. We report means and paired $R - A$ differences over eight training seeds per backbone, with two-sided 95% confidence intervals.

Backbone	Metric	A	R	$R - A$ [95% CI]
EPIC-KITCHENS-100				
Slot-MLLM	Future Gold-LL ↑	−1.198	−1.167	+0.032 [0.010, 0.055]
	Future-action EM (%) ↑	42.0	48.0	+6.0 [4.7, 7.3]
Qwen2.5-VL	Future Gold-LL ↑	−1.050	−1.024	+0.026 [0.011, 0.041]
	Future-action EM (%) ↑	50.0	54.0	+4.0 [2.1, 5.9]
Ego4D				
Slot-MLLM	Future Gold-LL ↑	−1.450	−1.410	+0.040 [0.022, 0.060]
	Future-action EM (%) ↑	32.0	34.5	+2.5 [1.7, 3.4]
Qwen2.5-VL	Future Gold-LL ↑	−1.300	−1.271	+0.029 [0.010, 0.048]
	Future-action EM (%) ↑	36.0	38.5	+2.5 [0.7, 4.2]

EM differences and their confidence intervals are in percentage points.

0.023 in future Gold-LL on Slot-MLLM (Table 1). Without predictive supervision, however, retaining a second state (G) yields a lower mean score than WorldPatch, suggesting that additional decoder context alone does not account for the gain. Alongside these gains in anticipation, WorldPatch meets the prespecified non-inferiority criterion for current-action EM relative to the present-only baseline P (Appendix G.5). We next examine whether the successor aligns with future targets and whether the decoder relies on its sample-specific content.

5.3 Predictive Content and Its Use

Closeness to a future target could reflect persistence of the present or a generic future representation. On Slot-MLLM, WorldPatch's retained successor is closer on average to its paired future target in the standardized target space than either the post-abstraction current state or a fixed training-target medoid. Prediction error also increases when the successor comes from another sample in the same video (Table 3(a)). These comparisons support a correspondence between the retained successor and its paired future target within the learned representation space.

Table 3: Future-target alignment and decoder use of the retained successor. Results are averaged over eight Slot-MLLM R checkpoints on EPIC-KITCHENS-100.

(a) Alignment with future targets

State	Mean distance ↓	Reference − native ↑	95% LCB
Native successor	**411.5**	—	—
Post-abstraction current state	457.2	+45.7	+22.8
Training-target medoid	454.5	+43.0	+2.4
Mismatched successor (same video)	464.0	+52.5	+8.2

(b) Content use in anticipatory decoding

State	Future Gold-LL ↑	Native − replacement ↑	95% LCB
Native successor	**−1.167**	—	—
Mismatched successor (same video)	−1.195	+0.028	+0.004

Distances are Hungarian-matched mean squared errors in the standardized target space; LCBs are one-sided 95% lower confidence bounds for paired differences.

This correspondence alone does not establish that the decoder uses the successor's content. With WorldPatch's weights and current state held fixed, replacing the successor with a prediction from another sample in the same video lowers future Gold-LL by 0.028 on average (Table 3(b)). The

decoder therefore benefits from a successor matched to the current observation. In Figure 3(a,b), WorldPatch correctly predicts opening a drawer and cutting an onion, whereas A predicts opening a cupboard and inserting an onion. In the onion example, a mismatched successor changes WorldPatch's answer from "cut onion" to "put spatula" (Figure 3(c)).

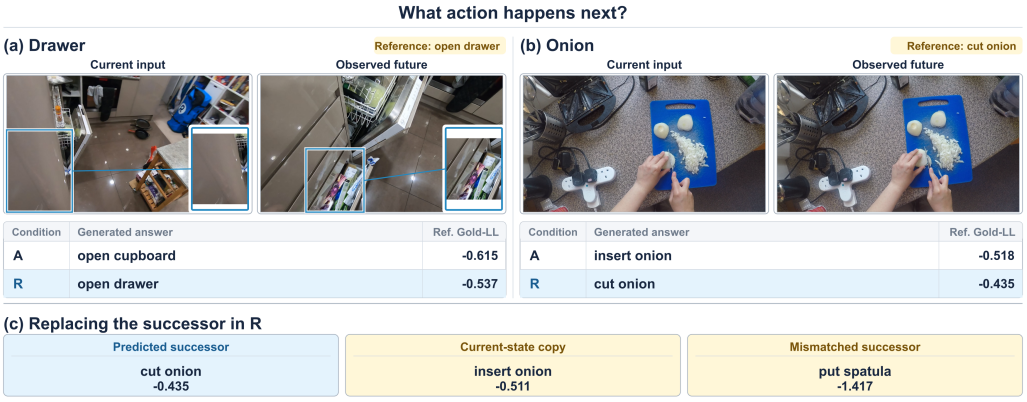


Figure 3: Next-action predictions with Slot-MLLM on two EPIC-KITCHENS-100 development examples. (a,b) Answers from separately trained A and R. (c) Successor replacements with R's weights and current state fixed. Future frames are for reference only; scores are Gold-LL values for the reference answers.

5.4 Retained Utility and Action-Label Agreement

Retaining the predicted successor yields larger gains when current and future action labels differ. We group EPIC-KITCHENS-100 examples by whether their current and future reference answers share the same verb–noun label. The analysis includes 7,524 of the 8,327 frame pairs, each with a single current action label and a single future action label. We compare the groups' $R - A$ gains within each of the 49 videos and average the differences using the participant weighting described in Section 5.1. The gain in future Gold-LL is 0.027 larger when the labels differ than when they match (Table 4; Appendix G.6). Future-action EM shows the same pattern, with a 4.0 percentage-point larger gain when the labels differ.

Table 4: Retained utility by action-label agreement on EPIC-KITCHENS-100. Results are averaged over eight paired training seeds for Slot-MLLM.

Metric	Current–future labels	A	R	$R - A$	95% CI
Future Gold-LL	Same	-1.080	-1.065	$+0.015$	$[0.0004, 0.0280]$
	Different	-1.270	-1.228	$+0.042$	$[0.0214, 0.0627]$
	Different − same	—	—	$+0.027$	$[0.0120, 0.0415]$
Future-action EM	Same	52.0	55.0	$+3.0$	$[1.1, 5.3]$
	Different	37.0	44.0	$+7.0$	$[4.4, 9.3]$
	Different − same	—	—	$+4.0$	$[1.2, 6.6]$

5.5 Broader Multimodal Evaluation

WorldPatch's gains extend beyond anticipatory decoding to image understanding, generation, and editing. In a separate single-seed evaluation, WorldPatch outperforms the full Slot-MLLM model on all eight image-understanding benchmarks (Table 5). The gain on GQA (Hudson & Manning, 2019) is 2.3 percentage points. The retained successor also contributes to image understanding. With WorldPatch's weights fixed, replacing it with a copy of the present lowers scores on GQA, MME-P (Fu et al., 2023), and NaturalBench (Li et al., 2024b) while preserving context length (Appendix A.1).

Table 5: Image understanding, generation, and editing results. Evaluation details and result sources are in Appendices C and D.

Image understanding								
Method	CIDEr	GQA	MME-P	SEED-Bench	POPE F1	MM-Vet	MMMU	NaturalBench
LLaVA-1.5	120.8	55.0	1101.6	46.3	78.6	20.1	27.0	57.9
MiniGPT-4	117.8	51.9	1081.4	44.0	77.3	19.6	25.0	55.0
SEED-LLaMA	122.5	52.4	1123.9	48.6	78.0	21.3	26.8	57.5
VILA-U	90.7	58.3	**1336.2**	**61.9**	**83.9**	**27.7**	29.2	**63.1**
Slot-MLLM-base	119.4	53.8	1128.4	44.7	78.3	17.8	29.0	59.0
Slot-MLLM	126.8	58.1	1202.6	47.4	79.8	20.6	28.0	60.5
WorldPatch	**129.4**	**60.4**	1242.0	49.6	81.2	22.5	**30.0**	62.4

	T2I-CompBench			GenEval	GenAI-Bench		MagicBrush		EVR	
Method	Color	Shape	Texture		Basic	Advanced	CLIP-I	L1 ↓	CLIP-I	L1 ↓
LLaVA-Interactive	0.210	0.180	0.240	0.270	0.50	0.46	75.0	30.0	65.0	32.0
LaViT	0.273	**0.319**	**0.399**	**0.480**	**0.78**	**0.62**	78.5	28.6	**69.1**	30.5
SEED-LLaMA	0.174	0.203	0.236	0.290	0.52	0.49	77.9	28.8	68.6	31.1
Slot-MLLM-base	0.151	0.177	0.212	0.240	0.46	0.43	77.2	28.2	67.4	30.2
Slot-MLLM	0.276	0.199	0.288	0.310	0.58	0.50	82.2	18.6	66.9	20.8
WorldPatch	**0.289**	0.215	0.301	0.330	0.60	0.53	**83.1**	**17.9**	67.5	**20.2**

The comparison with Slot-MLLM shows a similar pattern for text-to-image generation and instruction-guided editing (Table 5). Relative to the published Slot-MLLM results, WorldPatch scores 0.330 versus 0.310 on GenEval (Ghosh et al., 2023) and 83.1 versus 82.2 on MagicBrush (Zhang et al., 2023) CLIP-I (Radford et al., 2021). In the robot example, WorldPatch replaces the head with Doraemon's as instructed (Figure 4(c)). When asked to give the original robot tyrannosaur-like legs, however, WorldPatch also introduces a dinosaur head, torso, and tail (Figure 4(d)). The edit thus extends beyond the requested region, a limitation also observed in existing image-editing methods (Couairon et al., 2022).

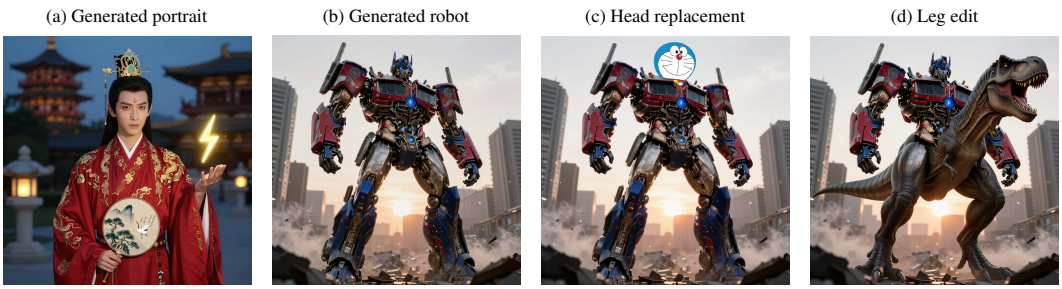


Figure 4: Image generation and editing with WorldPatch. (a,b) Generated images. (c,d) Independent edits of the robot in (b).

6 Conclusion

We investigated whether retaining a learned prediction improves anticipatory multimodal decoding beyond predictive training alone. WorldPatch retains the predicted short-horizon successor alongside the contextualized present, making both states available to the decoder. With predictive supervision held fixed, retaining the successor improves anticipation from the current image. Further analyses indicate that the retained state carries future-relevant content and contributes to decoding. These findings support retaining learned predictions as part of the context for anticipatory decoding.

AI USE STATEMENT

We used generative AI tools to assist with manuscript editing and translation and to identify relevant literature. The authors take responsibility for the final content of the paper, including its references and claims.

REPRODUCIBILITY STATEMENT

Appendix G documents the model and training settings, data construction, scoring, and statistical analyses. Evaluation protocols for the broader multimodal study are detailed in Appendices C, D, and E. Our planned release and data-use restrictions are described in Appendix G.7.

REFERENCES

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15619–15629, 2023.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-JEPA 2: Self-supervised video models enable understanding, prediction and planning, 2025. URL https://arxiv.org/abs/2506.09985.

Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. ScanQA: 3d question answering for spatial scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19129–19139, 2022. doi: 10.1109/CVPR52688.2022.01854. URL https://openaccess.thecvf.com/content/CVPR2022/html/Azuma_ScanQA_3D_Question_Answering_for_Spatial_Scene_Understanding_CVPR_2022_paper.html.

Shuai Bai, Keqin Chen, Xuejing Liu, et al. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025. URL https://arxiv.org/abs/2502.13923.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *arXiv preprint arXiv:2404.08471*, 2024.

Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *Forty-first International Conference on Machine Learning*, 2024.

Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11621–11631, 2020. URL https://openaccess.thecvf.com/content_CVPR_2020/html/Caesar_nuScenes_A_Multimodal_Dataset_for_Autonomous_Driving_CVPR_2020_paper.html.

Joya Chen, Ziyun Zeng, Yiqi Lin, Wei Li, Zejun Ma, and Mike Zheng Shou. LiveCC: Learning video LLM with streaming speech transcription at scale. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 29083–29095, 2025.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. ShareGPT4V: Improving large multi-modal models with better captions. In *Computer Vision – ECCV 2024*, pp. 370–387. Springer, 2024. doi: 10.1007/978-3-031-72643-9_22. URL https://doi.org/10.1007/978-3-031-72643-9_22.

Wei-Ge Chen, Irina Spiridonova, Jianwei Yang, Jianfeng Gao, and Chunyuan Li. LLaVA-Interactive: An all-in-one demo for image chat, segmentation, generation and editing. *arXiv preprint arXiv:2311.00571*, 2023. URL https://arxiv.org/abs/2311.00571.

Chuanqi Cheng, Jian Guan, Wei Wu, and Rui Yan. Scaling video-language models to 10k frames via hierarchical differential distillation. *arXiv preprint arXiv:2504.02438*, 2025.

Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, et al. VideoLLaMA 2: Advancing spatial-temporal modeling and audio understanding in video-LLMs. *arXiv preprint arXiv:2406.07476*, 2024.

Donghwan Chi, Hyomin Kim, Yoonjin Oh, Yongjin Kim, Donghoon Lee, Daejin Jo, Jongmin Kim, Junyeob Baek, Sungjin Ahn, and Sungwoong Kim. Slot-MLLM: Object-centric visual tokenization for multimodal LLM. *arXiv preprint arXiv:2505.17726*, 2025.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality, 2023. URL https://lmsys.org/blog/2023-03-30-vicuna/.

Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. DiffEdit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*, 2022. doi: 10.48550/arXiv.2210.11427. URL https://arxiv.org/abs/2210.11427.

Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5828–5839, 2017. doi: 10.1109/CVPR.2017.261. URL https://openaccess.thecvf.com/content_cvpr_2017/html/Dai_ScanNet_Richly-Annotated_3D_CVPR_2017_paper.html.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision: Collection, pipeline and challenges for EPIC-KITCHENS-100. *International Journal of Computer Vision*, 130:33–55, 2022. doi: 10.1007/s11263-021-01531-2.

Gamaleldin Elsayed, Aravindh Mahendran, Sjoerd Van Steenkiste, Klaus Greff, Michael C Mozer, and Thomas Kipf. SAVi++: Towards end-to-end object-centric learning from real-world videos. *Advances in Neural Information Processing Systems*, 35:28940–28954, 2022.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. MME: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-MME: The first-ever comprehensive evaluation benchmark of multi-modal LLMs in video analysis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 24108–24118, 2025.

Yuying Ge, Sijie Zhao, Ziyun Zeng, Yixiao Ge, Chen Li, Xintao Wang, and Ying Shan. Making llama see and draw with seed tokenizer. *arXiv preprint arXiv:2310.01218*, 2023.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. GenEval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 36: 52132–52152, 2023.

Rohit Girdhar and Kristen Grauman. Anticipative video transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13505–13515, 2021. URL https://openaccess.thecvf.com/content/ICCV2021/html/Girdhar_Anticipative_Video_Transformer_ICCV_2021_paper.html.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4D: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18995–19012, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Grauman_Ego4D_Around_the_World_in_3000_Hours_of_Egocentric_Video_CVPR_2022_paper.html.

David Ha and Jürgen Schmidhuber. World models. arXiv:1803.10122, 2018. URL https://arxiv.org/abs/1803.10122.

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. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2I-CompBench: A comprehensive benchmark for open-world compositional text-to-image generation. *Advances in Neural Information Processing Systems*, 36:78723–78747, 2023.

Drew A Hudson and Christopher D Manning. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6700–6709, 2019.

Tianxiang Jiang, Linquan Wu, Sheng Xia, Songze Li, Ziang Yan, Haoyu Yang, Yu Qiao, and Yi Wang. Imagine before you predict: Interleaved latent visual reasoning for video event prediction. *arXiv preprint arXiv:2606.05769*, 2026. URL https://arxiv.org/abs/2606.05769.

Yang Jin, Kun Xu, Kun Xu, Liwei Chen, Chao Liao, Jianchao Tan, Quzhe Huang, Bin Chen, Chenyi Lei, An Liu, Chengru Song, et al. Unified language-vision pretraining in LLM with dynamic discrete visual tokenization. *arXiv preprint arXiv:2309.04669*, 2023.

Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015. URL https://cs.stanford.edu/people/karpathy/cvpr2015.pdf.

Thomas Kipf, Gamaleldin F Elsayed, Aravindh Mahendran, Austin Stone, Sara Sabour, Georg Heigold, Rico Jonschkowski, Alexey Dosovitskiy, and Klaus Greff. Conditional object-centric learning from video. *arXiv preprint arXiv:2111.12594*, 2021.

H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1–2):83–97, 1955. doi: 10.1002/nav.3800020109.

Yann LeCun. A path towards autonomous machine intelligence. OpenReview, 2022. URL https://openreview.net/forum?id=BZ5a1r-kVsf. Version 0.9.2, 2022-06-27.

Baiqi Li, Zhiqiu Lin, Deepak Pathak, Jiayao Li, Yixin Fei, Kewen Wu, Tiffany Ling, Xide Xia, Pengchuan Zhang, Graham Neubig, et al. GenAI-Bench: Evaluating and improving compositional text-to-visual generation. *arXiv preprint arXiv:2406.13743*, 2024a.

Baiqi Li, Zhiqiu Lin, Wenxuan Peng, Jean de Dieu Nyandwi, Daniel Jiang, Zixian Ma, Simran Khanuja, Ranjay Krishna, Graham Neubig, and Deva Ramanan. NaturalBench: Evaluating vision-language models on natural adversarial samples. *Advances in Neural Information Processing Systems*, 37:17044–17068, 2024b.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. LLaVA-OneVision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024c.

Bohao Li, Yuying Ge, Yixiao Ge, Guangzhi Wang, Rui Wang, Ruimao Zhang, and Ying Shan. SEED-Bench: Benchmarking multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13299–13308, 2024d.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 19730–19742. PMLR, 23–29 Jul 2023a. URL https://proceedings.mlr.press/v202/li23q.html.

Rui Li, Xiaohan Wang, Yuhui Zhang, Orr Zohar, Zeyu Wang, and Serena Yeung-Levy. Temporal preference optimization for long-form video understanding. *arXiv preprint arXiv:2501.13919*, 2025.

Xianhang Li, Chen Huang, Chun-Liang Li, Eran Malach, Joshua M. Susskind, Vimal Thilak, and Etai Littwin. Rethinking JEPA: Compute-efficient video SSL with frozen teachers. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=3cB9243E9i.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 292–305, 2023b.

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

Han Lin, Tushar Nagarajan, Nicolas Ballas, Mahmoud Assran, Mojtaba Komeili, Mohit Bansal, and Koustuv Sinha. VEDIT: Latent prediction architecture for procedural video representation learning. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=LDAj4UJ4aL.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *Computer Vision – ECCV 2014*, pp. 740–755. Springer International Publishing, 2014. doi: 10.1007/978-3-319-10602-1_48. URL https://doi.org/10.1007/978-3-319-10602-1_48.

Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating Text-to-Visual Generation with Image-to-Text Generation. *arXiv preprint arXiv:2404.01291*, 2024b. URL https://arxiv.org/abs/2404.01291.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/6dcf277ea32ce3288914faf369fe6de0-Paper-Conference.pdf.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 26296–26306, 2024.

Zhijian Liu, Ligeng Zhu, Baifeng Shi, Zhuoyang Zhang, Yuming Lou, Shang Yang, Haocheng Xi, Shiyi Cao, Yuxian Gu, Dacheng Li, et al. NVILA: Efficient frontier visual language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4122–4134, 2025.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *Advances in neural information processing systems*, 33:11525–11538, 2020.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://arxiv.org/abs/1711.05101.

Yongdong Luo, Wang Chen, Weizhong Huang, Shukang Yin, Haojia Lin, Jinfa Huang, Chaoyou Fu, Jiayi Ji, Xiawu Zheng, and Jiebo Luo. QuoTA: Query-oriented token assignment via CoT query decouple for long video comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 24160–24168, 2026.

Chuofan Ma, Yi Jiang, Jiannan Wu, Zehuan Yuan, and Xiaojuan Qi. Groma: Localized visual tokenization for grounding multimodal large language models. In *European Conference on Computer Vision*, pp. 417–435. Springer, 2024.

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

OpenBMB MiniCPM-o Team. MiniCPM-o 2.6: A GPT-4o level MLLM for vision, speech, and multimodal live streaming on your phone, 2025. URL https://huggingface.co/openbmb/MiniCPM-o-2_6. Accessed January 31, 2025.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision, 2023. URL https://arxiv.org/abs/2304.07193.

Tianwen Qian, Jingjing Chen, Linhai Zhuo, Yang Jiao, and Yu-Gang Jiang. NuScenes-QA: A multimodal visual question answering benchmark for autonomous driving scenario. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 4542–4550, 2024. doi: 10.1609/aaai.v38i5.28253. URL https://ojs.aaai.org/index.php/AAAI/article/view/28253.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Yunhang Shen, Chaoyou Fu, Shaoqi Dong, Xiong Wang, Yi-Fan Zhang, Peixian Chen, Mengdan Zhang, Haoyu Cao, Ke Li, Shaohui Lin, et al. Long-vita: Scaling large multi-modal models to 1 million tokens with leading short-context accuracy. *arXiv preprint arXiv:2502.05177*, 2025.

Hao Tan, Franck Dernoncourt, Zhe Lin, Trung Bui, and Mohit Bansal. Expressing visual relationships via language. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1873–1883, 2019.

Canhui Tang, Zifan Han, Hongbo Sun, Sanping Zhou, Xuchong Zhang, Xin Wei, Ye Yuan, Huayu Zhang, Jinglin Xu, and Hao Sun. TSPO: Temporal sampling policy optimization for long-form video language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 9368–9376, 2026.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. CIDEr: Consensus-Based Image Description Evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. URL https://openaccess.thecvf.com/content_cvpr_2015/html/Vedantam_CIDEr_Consensus-Based_Image_2015_CVPR_paper.html.

Video-MME Team. Video-MME Leaderboard. https://video-mme.github.io/home_page.html, 2026. Accessed September 7, 2026.

Han Wang, Yuxiang Nie, Yongjie Ye, Yanjie Wang, Shuai Li, Haiyang Yu, Jinghui Lu, and Can Huang. Dynamic-VLM: Simple dynamic visual token compression for videollm. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20812–20823, 2025.

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

Yecheng Wu, Zhuoyang Zhang, Junyu Chen, Haotian Tang, Dacheng Li, Yunhao Fang, Ligeng Zhu, Enze Xie, Hongxu Yin, Li Yi, et al. VILA-U: a unified foundation model integrating visual understanding and generation. *arXiv preprint arXiv:2409.04429*, 2024.

Yu Wu, Linchao Zhu, Xiaohan Wang, Yi Yang, and Fei Wu. Learning to anticipate egocentric actions by imagination. *IEEE Transactions on Image Processing*, 30:1143–1152, 2021. doi: 10.1109/TIP.2020.3040521.

Ziyi Wu, Nikita Dvornik, Klaus Greff, Thomas Kipf, and Animesh Garg. SlotFormer: Unsupervised visual dynamics simulation with object-centric models. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=TFbwV6I0VLg.

Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. NExT-QA: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9777–9786, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/html/Xiao_NExT-QA_Next_Phase_of_Question-Answering_to_Explaining_Temporal_Actions_CVPR_2021_paper.html.

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*, pp. 13204–13214, 2024. doi: 10.1109/CVPR52733.2024.01254. URL https://arxiv.org/abs/2309.01327.

Jiaqi Xu, Cuiling Lan, Wenxuan Xie, Xuejin Chen, and Yan Lu. Slot-VLM: Object-event slots for video-language modeling. *Advances in Neural Information Processing Systems*, 37:632–659, 2024.

Biao Yang, Bin Wen, Boyang Ding, Changyi Liu, Chenglong Chu, Chengru Song, Chongling Rao, Chuan Yi, Da Li, Dunju Zang, et al. Kwai Keye-VL 1.5 technical report. *arXiv preprint arXiv:2509.01563*, 2025.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. MM-Vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*, 2023.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9556–9567, 2024.

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

Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. MagicBrush: A manually annotated dataset for instruction-guided image editing. *Advances in Neural Information Processing Systems*, 36:31428–31449, 2023.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. MiniGPT-4: Enhancing vision-language understanding with advanced large language models, 2023. URL https://arxiv.org/abs/2304.10592.

A Retained-State Interface Analyses

These analyses examine how the content and organization of the present–successor context affect multimodal decoding. All comparisons use the same Full model from the single-seed multimodal evaluation (seed 42), with its weights fixed.

All interface modifications are applied before P_{WP}. For GenEval, they affect only the draft-derived states used in the second generation pass. All conditions share identical first-pass image tokens, prompts, decoding settings, and random seeds.

Table 6: Inference-time modifications of the retained-state interface. Full uses the native present–successor concatenation. All variants share the same multimodal checkpoint. Higher is better for all metrics; NaturalBench reports per-question accuracy. Dashes indicate unreported results.

Variant	GQA	MME-P	NaturalBench Acc. (%)	GenEval	MagicBrush CLIP-I
Full (native)	60.4	1242.0	62.4	0.33	83.1
Copy control					
Copied present	57.5	1171.3	60.2	—	—
Removal and slot permutation					
Present only	57.6	1174.8	—	0.29	80.8
Zero successor	57.3	1167.9	—	0.28	80.2
Permuted successor slots	56.7	1155.1	—	0.27	79.4
Fusion					
Sum	57.2	1164.8	60.0	—	—
MeanPool	56.9	1157.5	59.8	—	—

A.1 Predicted Successor versus Copied Present

To test whether the predicted successor improves decoding beyond repeating the present, we replace it with a second copy of $\widetilde{S}_t$ while retaining the remaining decoding path. Both variants therefore supply $2N$ state tokens before projection and are evaluated on the same samples with identical prompts, decoding settings, and evaluators. The native successor yields higher scores on all three evaluated benchmarks (Table 6). The gains are 2.9 percentage points on GQA (95% CI: [1.5, 4.2]), 70.7 score points on MME-P ([44.6, 95.9]), and 2.2 percentage points on NaturalBench ([0.9, 3.5]). For this trained model, the predicted successor provides more useful context than an additional copy of the present.

For this comparison, we report two-sided 95% percentile intervals from 10,000 paired cluster bootstrap resamples per benchmark. The sampling units are images with all their questions for GQA, complete image units within each MME-P perception category, and complete four-question groups for NaturalBench. Both variants use the same resampled units. Benchmark scores are recomputed in every resample, including category-level ACC and ACC+ before aggregation for MME-P and per-question accuracy for NaturalBench. These intervals quantify evaluation-sample uncertainty conditional on this checkpoint.

A.2 Successor Removal and Slot Permutation

We next examine how decoding changes when the successor is removed or zeroed. Removing it shortens the state sequence from $2N$ to N, whereas zeroing the successor tensor preserves the original context length. Both interventions lower scores across the four evaluated metrics in Table 6. The decline under zeroing shows that retaining the second block's size alone does not preserve the native scores.

We also permute the successor slots independently within each sample, leaving the present and the set of successor vectors unchanged. The lower scores under this permutation indicate that the trained decoder interface is sensitive to slot order. This probes a different property from the cross-sample replacement in Section 5.3, which changes the successor content.

A.3 Present–Successor Fusion

We also compare concatenation with Sum and MeanPool. Sum adds the present and predicted-successor states element-wise at corresponding slot indices, whereas MeanPool averages the two states at each index. Both operations precede the shared token-wise projection and yield N tokens, compared with $2N$ under concatenation. Both variants score lower on GQA, MME-P, and NaturalBench (Table 6). These comparisons use a fixed model trained with concatenation and characterize its response to changes in the inference-time interface.

B Component and Training-Objective Analyses

We examine how state construction and auxiliary objectives affect the multimodal performance reported in Section 5.5, and how this performance varies with slot count and prediction offset. Full is trained in the single-seed multimodal setting (seed 42) with predictive supervision and a cross-view consistency objective. Each ablated variant is retrained with the same seed and training schedule and evaluated using the same task protocols as Full.

Table 7: Component and training-objective ablations on image understanding, generation, and editing. Higher is better for all metrics. The joint auxiliary-objective ablation removes both $\mathcal{L}_{\text{pred}}$ and $\mathcal{L}_{\text{cons}}$ while retaining the complete state pathway.

Variant	GQA	MME-P	GenEval	MagicBrush CLIP-I
Full	60.4	1242.0	0.33	83.1
w/o State Modeling	57.8	1145.0	0.30	81.0
w/o State Transition	54.2	1130.0	0.28	79.5
w/o $\mathcal{L}_{\text{pred}}$	56.8	1148.9	0.29	82.0
w/o $\mathcal{L}_{\text{cons}}$	57.4	1166.7	0.30	82.4
w/o Auxiliary Objectives	53.5	980.2	0.26	80.0

B.1 State Modeling and Transition

The state-modeling ablation removes the structured-state pathway, including state abstraction A and contextualization T_1. The visual encoder and matched non-state decoder-conditioning path are retained under the same decoder budget. Since this variant does not form a contextualized present, T_2 and both structured-state auxiliary objectives are omitted.

The state-transition ablation retains A and T_1 to construct $\widetilde{S}_t$, but removes T_2. The decoder receives a single N-token present-state block. The predictive loss is omitted, while the cross-view consistency objective remains unchanged.

Both variants perform worse than Full on the reported understanding, generation, and editing metrics (Tables 7 and 8). Without state transition, GQA falls from 60.4 to 54.2 and MagicBrush CLIP-I from 83.1 to 79.5.

Table 8: Additional generation and editing metrics for the component and joint-objective ablations in Table 7. Color, shape, and texture are T2I-CompBench scores; MagicBrush L1 is pixelwise MAE on RGB $[0, 1]$, multiplied by 100. Arrows indicate the preferred direction.

Metric	Full	w/o State Modeling	w/o State Transition	w/o Auxiliary Objectives
Color ↑	0.289	0.258	0.235	0.220
Shape ↑	0.215	0.196	0.173	0.173
Texture ↑	0.301	0.274	0.260	0.237
GenAI-B ↑	0.60	0.55	0.52	0.50
MagicBrush L1 ↓	17.9	20.5	23.0	21.8

B.2 Predictive and Cross-View Objectives

The structural ablations also change auxiliary supervision. To examine the training objectives with the state pathway retained, we remove $\mathcal{L}_{\text{pred}}$ and $\mathcal{L}_{\text{cons}}$ separately and jointly. Full assigns weight 0.1 to each auxiliary loss; ablated losses have weight zero. Removing either objective lowers all four task scores, with a further decline when both are removed (Table 7).

B.3 Slot Count and Prediction Offset

We vary the slot count N and prediction offset k separately, keeping the other settings fixed (Table 9). For the slot-count comparison, we vary the number of learned queries in the trainable state resampler

used by Full (Section G.1). Among the tested values, $N = 32$ and $k = 10$ attain the highest scores on all four metrics in their respective comparisons. Increasing the slot count beyond 32 does not improve scores, and both shorter and longer offsets score below the 10-frame default.

Table 9: Slot count and prediction offset varied separately. The offset k is measured in frames. The first row is the shared Full reference from Table 7. Higher is better for all metrics.

N	k	GQA	MME-P	GenEval	MagicBrush CLIP-I
32	10	60.4	1242.0	0.33	83.1
16	10	58.3	1183.1	0.30	81.9
24	10	59.8	1224.0	0.32	82.6
48	10	60.0	1228.9	0.32	82.7
64	10	58.9	1206.0	0.31	82.4
32	5	58.9	1221.9	0.31	82.5
32	15	59.4	1235.7	0.32	82.7
32	20	58.5	1208.0	0.30	81.8

C Image Understanding Evaluation

Table 10 summarizes the evaluation sets and metrics used to compute WorldPatch's image understanding scores in Table 5.

C.1 Benchmarks and Evaluation Protocols

Table 10: Evaluation sets and metrics for WorldPatch's image understanding results in Table 5.

Benchmark	Evaluation set	Reported metric
COCO (Lin et al., 2014)	Karpathy test split	CIDEr
GQA (Hudson & Manning, 2019)	Full balanced Test-Dev	Accuracy (%)
MME (Fu et al., 2023)	Perception categories	MME-P score
SEED-Bench (Li et al., 2024d)	SEED-Bench-1 Image subset	Accuracy (%)
POPE (Li et al., 2023b)	COCO val2014, three settings	Mean F1 (%)
MM-Vet (Yu et al., 2023)	Full v1 set	Mean graded score
MMMU (Yue et al., 2024)	Original validation set	Accuracy (%)
NaturalBench (Li et al., 2024b)	Natural adversarial questions	Binary VQA accuracy (%)

COCO captions are evaluated on the Karpathy test split (Karpathy & Fei-Fei, 2015) using CIDEr (Vedantam et al., 2015), which measures agreement with human reference captions.

GQA uses the full balanced Test-Dev split and the official v1.1 evaluator. Accuracy is computed over individual questions, with no additional subsampling.

MME-P aggregates the perception categories of MME. Each category contributes the sum of question-level accuracy (ACC) and image-level accuracy (ACC+), where ACC+ requires both questions associated with an image to be answered correctly. Both terms are expressed as percentages before summation.

SEED-Bench uses the complete SEED-Bench-1 Image subset across all its evaluation dimensions. We select the candidate answer with the highest conditional likelihood and report question-level accuracy.

POPE uses COCO val2014 images and the fixed official question files for the random, popular, and adversarial settings. F1 is computed separately for each setting; the value in Table 5 is the unweighted mean of these three F1 scores.

MM-Vet uses the complete v1 evaluation set with the official grading prompt and evaluator. Answers are graded with `gpt-4-0613` over five grading runs, and the reported result is the mean of the five overall scores.

MMMU uses the original validation set of 900 questions across 30 subjects under zero-shot evaluation, with the official answer parser and evaluator. The reported accuracy weights each question equally across the full set.

For NaturalBench, Acc. (%) is the number of correctly answered questions divided by the total number of evaluated questions, multiplied by 100. Each question contributes separately to this score, including questions that belong to the same image–question group.

C.2 Comparison Methods and Result Sources

The comparison includes LLaVA-1.5 (Liu et al., 2024), MiniGPT-4 (Zhu et al., 2023), SEED-LLaMA (Ge et al., 2023), VILA-U (Wu et al., 2024), and both Slot-MLLM-base and Slot-MLLM (Chi et al., 2025). We evaluate LLaVA-1.5 and MiniGPT-4 using the image-understanding protocols described in this appendix. Results for SEED-LLaMA, VILA-U, Slot-MLLM-base, and Slot-MLLM are taken from Table 2 of Chi et al. (2025). Slot-MLLM denotes the full model. Slot-MLLM-base is its tokenizer ablation, which omits Slot Attention (Locatello et al., 2020) and trains the visual tokenizer with the CLIP image-embedding matching objective.

WorldPatch uses Vicuna-7B (Chiang et al., 2023), with scores from a single training run (seed 42). Table 5 provides a comparison with existing multimodal systems; the matched retained-state comparisons are reported separately in Section 5.2.

D Image Generation and Editing Evaluation

The protocols below specify how WorldPatch's image generation and editing scores in Table 5 are computed.

D.1 Text-to-Image Generation

Text-to-image generation uses a two-pass interface. Given a text prompt, the LLM first generates 128 image tokens, which are reshaped into 32×4 quantization indices. The frozen stage-2 tokenizer mapping converts these indices into a draft state of shape 32×768. WorldPatch first resamples this draft to N slots, then processes it successively through T_1 and T_2, retaining both outputs and projecting them through P_{WP} and P_{LLM}. The projected states, together with the same prompt, condition a second generation pass that also produces 128 image tokens. The resulting output is decoded into an RGB image using the official Stable-UnCLIP small checkpoint.[1]

For attribute binding, we use the original T2I-CompBench (Huang et al., 2023), with 300 prompts from each of `color_val.txt`, `shape_val.txt`, and `texture_val.txt`. Each prompt produces ten images using sampling seeds 42–51, for 9,000 images in total. The official BLIP-VQA evaluator scores each image. Scores are first averaged over the ten images for each prompt and then over the 300 prompts in each category, yielding the Color, Shape, and Texture columns in Table 5.

For GenEval (Ghosh et al., 2023), we generate four images for each of the original 553 prompts, using four fixed sampling seeds (2,212 images in total). Evaluation follows the official scripts with Mask2Former and MMDetection 2.x. Within each of the six task categories, the score is the fraction of generated images that satisfy their prompt's requirements according to the evaluator. The reported GenEval score is the unweighted mean of the six category scores.

GenAI-Bench (Li et al., 2024a) uses all 1,600 prompts, with one image per prompt and a fixed seed assigned to each prompt. Each generated image is scored directly, without candidate selection. VQAScore (Lin et al., 2024b) is computed with `clip-flant5-xxl` from the likelihood assigned to the answer `Yes`. The question template is `Does this figure show "{}"? Please answer yes or no.`, with the prompt substituted for {}. GenAI-B and GenAI-A average the image scores over the prompt indices in the official `basic` and `advanced` tag lists, respectively.

[1] `https://huggingface.co/stabilityai/stable-diffusion-2-1-unclip-small`

D.2 Instruction-Guided Image Editing

WorldPatch is evaluated on MagicBrush and EVR in the mask-free, single-turn setting. For each source image and its original edit instruction, the model generates one edited image.

MagicBrush (Zhang et al., 2023) uses all 1,053 edit turns from the 535 sessions in the official test split. Each turn starts from its ground-truth predecessor image, including turns later in a session. The target image is used only for scoring; masks and local descriptions are not supplied to the model. Each metric is averaged over edit turns.

EVR uses the official ImgEdit test split from Tan et al. (2019). All three requests associated with each source–target image pair are evaluated separately. For each metric, we first average over the three requests for a pair and then take the unweighted mean across pairs, so that each image pair contributes equally.

CLIP-I measures similarity between the generated image and the target edited image using OpenAI CLIP ViT-B/32 (Radford et al., 2021). Both images undergo the official preprocessing: bicubic resizing, a 224×224 center crop, RGB conversion, and channel normalization. Let $f(I)$ denote the resulting CLIP image embedding and $\bar{f}(I) = f(I)/\|f(I)\|_2$. The reported score is

$$\text{CLIP-I}(I^{\text{out}}, I^{\text{target}}) = 100\, \bar{f}(I^{\text{out}})^\top \bar{f}(I^{\text{target}}). \tag{11}$$

For L1 evaluation, both images are converted to RGB. The target image retains its native resolution, while the generated image is resized directly to the target dimensions using bicubic interpolation, without cropping or padding. The resize operation does not enforce preservation of the generated image's aspect ratio. Both images are then converted to tensors with values in $[0, 1]$. For a target image of size $H \times W$, we report the mean absolute error over all pixels and three channels, multiplied by 100:

$$\text{L1}(I^{\text{out}}, I^{\text{target}}) = \frac{100}{3HW} \sum_{h=1}^{H} \sum_{w=1}^{W} \sum_{c=1}^{3} \left| I^{\text{out}}_{hwc} - I^{\text{target}}_{hwc} \right|. \tag{12}$$

Higher CLIP-I and lower L1 indicate greater agreement with the target in feature space and pixel space, respectively.

D.3 Comparison Methods and Result Sources

WorldPatch uses a Slot Q-Former with the Stable-UnCLIP (Stable Diffusion 2.1 UnCLIP, small) image decoder. WorldPatch's results in Table 5 come from a single training run with seed 42; the multiple sampling seeds above vary image generation at inference.

The comparison includes LLaVA-Interactive (Chen et al., 2023), LaViT (Jin et al., 2023), SEED-LLaMA (Ge et al., 2023), and both Slot-MLLM-base and Slot-MLLM (Chi et al., 2025). LLaVA-Interactive combines LLaVA for visual dialogue, GLIGEN for image generation and editing, and SEEM for segmentation and region operations. We evaluate LLaVA-Interactive using the generation and editing protocols described in this appendix. Results for LaViT, SEED-LLaMA, Slot-MLLM-base, and Slot-MLLM are taken from Table 3 of Chi et al. (2025) and retain their source evaluation protocols. Table 5 compares performance across systems, while Section 5.2 assesses retained predictive-state access under matched predictive supervision. Component and training-objective ablations are reported in Appendix B.

E Multi-Frame Video Understanding

We extend the image-understanding evaluation in Section 5.5 to video question answering on Video-MME. This evaluation uses multiple observed frames to examine WorldPatch's performance on videos of different durations.

E.1 Evaluation Protocol

We use the original Video-MME benchmark (Fu et al., 2025), comprising 900 videos and 2,700 four-choice questions. The short, medium, and long groups each contain 300 videos and 900 questions.

Table 11: Reported Video-MME accuracy (%; higher is better). Comparison methods use subtitles; WorldPatch uses 128 frames without subtitles or audio. Parameter counts and frame-sampling budgets follow the source reports. The 1 fps settings are capped at 600 frames for ViLAMP and 256 for Long-VITA; LiveCC's 2 fps setting is capped at 480 frames.

Method	Params	Frames / Sampling	Overall	Short	Medium	Long
Comparison methods with subtitles						
Keye-VL-1.5 (Yang et al., 2025)	8B	64	74.0	84.0	71.9	66.2
TSPO (Tang et al., 2026)	72B	128	73.7	83.4	74.2	63.6
ViLAMP (Cheng et al., 2025)	7B	1 fps	73.5	81.2	71.2	68.1
Long-VITA (Shen et al., 2025)	14B	1 fps	72.0	77.0	73.0	66.0
TPO (Li et al., 2025)	7B	96	71.5	78.7	69.4	66.4
VideoLLaMA 3 (Zhang et al., 2025)	7B	180	70.3	80.2	69.6	61.0
LiveCC (Chen et al., 2025)	7B	2 fps	70.3	76.6	70.3	64.1
QuoTA (Luo et al., 2026)	7B	64	70.0	79.0	68.0	62.9
NVILA (Liu et al., 2025)	7B	1024	70.0	77.6	69.0	63.3
LLaVA-OneVision (Li et al., 2024c)	72B	32	69.6	79.3	66.9	62.4
ReTaKe (Wang et al., 2024)	7B	1024	68.9	74.3	69.3	62.9
ByteVideoLLM (Wang et al., 2025)	14B	100	68.8	77.1	69.1	60.2
MiniCPM-o 2.6 (OpenBMB MiniCPM-o Team, 2025)	8B	64	67.9	78.3	69.1	56.3
WorldPatch without subtitles or audio						
WorldPatch	7B	128	77.1	85.6	76.1	69.8

WorldPatch receives 128 frames per video, without subtitles or audio. We divide the full decodable duration into 128 equal intervals and select the frame whose presentation timestamp is nearest to each interval's midpoint. Sampling is deterministic and independent of the question; all three questions for a video use the same frames. Each frame undergoes the model's existing 256×256 visual preprocessing.

We use the same terminal WorldPatch-Full checkpoint (7B, seed 42) as in Section 5.5, with all weights frozen and no additional training or video adaptation. Each sampled frame is processed independently by the existing state pathway to obtain a present state $\widetilde{S}_i$ and its predicted successor $\widehat{S}_i = T_2(\widetilde{S}_i)$. To assemble the video representation, we divide the sampled frames in temporal order into 16 blocks of eight. Within block b, the two states are averaged separately across frames at corresponding slot indices:

$$\overline{S}_b = \frac{1}{8} \sum_{i \in \mathcal{B}_b} \widetilde{S}_i, \qquad \overline{S}_b^{\mathrm{succ}} = \frac{1}{8} \sum_{i \in \mathcal{B}_b} \widehat{S}_i, \tag{13}$$

where $\mathcal{B}_b$ contains the eight frame indices in block b. Both averages retain all $N = 32$ slots. We add the existing present and successor type embeddings, concatenate the two states within each block, and arrange the blocks in temporal order. The resulting $16 \times (32 + 32) = 1{,}024$ state tokens pass through the existing token-wise projections and form continuous context for the Vicuna-7B decoder. The decoder uses its standard sequence positions; no learnable temporal adapter is introduced.

Each question and its four options are accompanied by the instruction: "Select the best answer based on the video. Respond with only the option letter." Answers are decoded greedily. We extract option letters using the benchmark's answer parser, counting empty, unparseable, or ambiguous responses as incorrect. Accuracy is computed over all 900 questions in each duration group. Overall accuracy uses the total number of correct answers across all 2,700 questions.

E.2 Results and Comparisons

Table 11 reports overall and duration-specific accuracy on Video-MME. Comparison results are taken from the Video-MME leaderboard (Video-MME Team, 2026), retaining the reported model sizes and frame-sampling budgets. These methods use subtitles, whereas WorldPatch uses 128 frames without subtitles or audio.

WorldPatch's reported accuracy decreases from 85.6% on short videos to 76.1% on medium videos and 69.8% on long videos under the same 128-frame input budget. The comparison provides context across systems with different input conditions; the controlled comparisons in Section 5.2 address the contribution of retained predictive-state access.



(a) Source image

(b) Head replacement
replace the transformer's head with Doraemon.

(c) Leg edit
turn the transformer's legs into tyrannosaur-like legs.

Figure 5: Instruction-guided editing with WorldPatch. The outputs in (b) and (c) are generated independently from the shared source image in (a), using the instructions shown below each output.

F Qualitative Examples

Figure 5 provides larger views of the editing examples in Figure 4(c,d), together with their shared source image.

For the head-replacement instruction, the output replaces the robot's head with Doraemon's while retaining a similar overall composition. The leg edit also introduces a dinosaur head, neck, torso, and tail, extending the transformation beyond the requested legs.

G Implementation and Evaluation Details

The controlled anticipation study and the broader multimodal evaluations use different training settings. We describe the model configurations, data construction, scoring, and statistical analyses for these evaluations below.

G.1 Model and Training Settings

Slot-MLLM anticipation. All five conditions use the `KU-AGI/Slot-MLLM-7B-instruct` backbone (Chi et al., 2025). The DINOv2 visual encoder (Oquab et al., 2023), Slot Q-Former tokenizer, and base language-model weights are frozen. The current observation is processed at 448×448 resolution. The trainable abstraction module A_θ produces $N = 32$ states of dimension $d = 768$. The contextualization module T_1 is a two-layer Transformer with 12 attention heads, feed-forward width 2,048, and zero dropout; T_2 is a residual MLP with widths $768 \to 2048 \to 768$. Branch-specific type embeddings and the projections map the retained states into the decoder context, as defined in Section 4.3.

Stage 0 trains the WorldPatch modules enabled by each condition, the projections, and the state-type embeddings for 5,000 updates, with the decoder and its LoRA (Hu et al., 2022) parameters frozen. Stage 1 starts from the terminal Stage 0 checkpoint and continues training these components for 50,000 updates while also optimizing decoder LoRA. Both stages use one source visual pair per update. AdamW (Loshchilov & Hutter, 2019) uses learning rates of 10^{-4} for the WorldPatch-side parameters and 10^{-5} for decoder LoRA. The LoRA rank is 16, its scaling parameter is 32, and its dropout is 0.05. Weight decay is 0.01, the gradient-norm limit is 1.0, and training uses BF16. Each condition is trained separately with the same eight paired seeds, giving 40 training runs. All conditions and seeds are evaluated at the fixed Stage 1 50K terminal checkpoint, with no checkpoint selection based on development-set scores.

Conditions A, $\mathrm{R}_{\mathrm{stop}}$, and R use prediction-loss weight 0.1; P and G omit that objective. All five conditions omit cross-view consistency. In $\mathrm{R}_{\mathrm{stop}}$, the successor is detached only on its route to the

task loss. The prediction objective continues to train the current-side abstraction and transition, while the present and shared components retain their task gradients.

Qwen adaptation. We additionally train A and R on `Qwen/Qwen2.5-VL-7B-Instruct`, keeping its native visual encoder, visual merger, and base language-model weights frozen. A trainable projection maps merged visual features from 3,584 to 768 dimensions before a 32-state cross-attention resampler. T_1 and T_2 follow the controlled architectures above, with a LayerNorm at the output of T_2. Within each arm, the current and future branches share the input projection and resampler; future targets are detached after resampling. After adding type embeddings, decoder alignment maps the retained states to 3,584 dimensions, providing A with 32 present tokens and R with 32 present plus 32 successor tokens. Native visual patch tokens do not enter the decoder through a separate path.

The state prefixes use one-dimensional sequence positions replicated across the three mRoPE axes, without image-grid coordinates. Training, scoring, and generation use the same position convention. Current and future images are processed at 448×448 with the Qwen processor; text uses the Qwen tokenizer, chat template, and EOS convention.

We train eight A/R pairs using the two-stage training budget and optimizer settings above, with prediction-loss weight 0.1 and no cross-view consistency. Within each pair, common modules have matching initializations, and data order and future-target sampling are shared. Using the LoRA hyperparameters above, we adapt only the self-attention `q_proj`, `k_proj`, `v_proj`, and `o_proj` projections in every Qwen decoder layer, with `bias="none"`. LoRA remains frozen in Stage 0 and is trained in Stage 1. For active parameter groups, the learning-rate schedule restarts at each stage with 5% linear warmup (250 updates in Stage 0 and 2,500 in Stage 1), followed by cosine decay to 10% of the group's peak learning rate. Evaluation uses each run's fixed Stage 1 50K terminal checkpoint.

Multimodal evaluations. The WorldPatch models used in Section 5.5 and Appendices A–D follow the separate single-seed setting summarized in Table 12. Here, the frozen tokenizer's stage-2 mapping produces 32 slots of dimension 768. The trainable abstraction module A_θ is a cross-attention resampler whose N learned queries map these slots to N slots before T_1. The current and future branches share this resampler, with future targets detached after resampling. The two-layer T_1 uses pre-layer normalization, 12 heads, feed-forward width 2,048, dropout 0.1, and no explicit positional encoding. The transition is

$$\widehat{S}_{t+k} = \mathrm{LayerNorm}\Big(\widetilde{S}_t + \mathrm{MLP}(\widetilde{S}_t)\Big), \tag{14}$$

where the MLP has widths $768 \to 2048 \to 768$ and a ReLU activation. With $N = 32$, concatenation gives 64 state tokens and the Vicuna-7B decoder receives conditioning features of shape 64×4096. Image generation and editing use the Slot Q-Former and Stable-UnCLIP image-decoding path described in Appendix D.

Visual inputs are resized to 256×256 and normalized channelwise with mean $(0.485, 0.456, 0.406)$ and standard deviation $(0.229, 0.224, 0.225)$. The same preprocessing is applied to current images, future images, and paired views used for cross-view consistency. Multimodal pretraining and instruction tuning each last 50,000 steps. AdamW uses separate decoder-side and WorldPatch parameter groups, with learning rates 10^{-5} and 10^{-4}, respectively. Training uses batch size 2, weight decay 0.01, gradient clipping at norm 1.0, and FP16 with gradient scaling. These runs use seed 42.

Multimodal pretraining draws equally from four source groups: Ego4D (Grauman et al., 2022), NExT-QA/NExT-GQA (Xiao et al., 2021; 2024), nuScenes/nuScenes-QA (Caesar et al., 2020; Qian et al., 2024), and ScanNet/ScanQA (Dai et al., 2017; Azuma et al., 2022). Each group accounts for 25% of sampled examples. During instruction tuning, LLaVA-Instruct and ShareGPT4V each account for 25% of sampled examples (Liu et al., 2023; Chen et al., 2024), and each of the four paired-data groups accounts for 12.5%. Both stages use frozen source-balanced sampling sequences.

The Ego4D group uses official training videos and training annotations to construct temporal pairs. EgoSchema questions and answers are excluded from training.

The video evaluation in Appendix E uses the same terminal WorldPatch-Full (7B, seed 42) checkpoint from this multimodal training run. The checkpoint is identified in the experiment records by its

SHA-256 digest. The 128-frame input changes only the organization of inference-time observations; all weights remain frozen, with no additional training or video adaptation.

Table 12: Training settings for the Slot-MLLM controlled study and the separate multimodal evaluations. The prediction offset in the controlled study varies with the selected timestamps; the multimodal setting uses a fixed frame offset.

Setting	Controlled anticipation (Slot-MLLM)	Multimodal evaluations
Input resolution	448×448	256×256
State shape	32×768	32×768
Abstraction	Trainable A_θ	Trainable slot resampler
T_1 dropout	0	0.1
Training budget	5K + 50K updates	50K + 50K steps
Training unit	One visual pair per update	Batch size 2
WorldPatch learning rate	10^{-4}	10^{-4}
Decoder learning rate	10^{-5} for LoRA	10^{-5} for decoder-side parameters
Precision	BF16	FP16 with gradient scaling
Prediction supervision	Weight 0.1 in A, $\mathrm{R}_{\mathrm{stop}}$, R	Weight 0.1 in Full
Cross-view consistency	Omitted	Weight 0.1 in Full
Prediction offset	Timestamp-dependent	$k = 10$ frames
Training repetitions	Eight paired seeds per condition	Seed 42

The Full multimodal model combines task supervision with two auxiliary objectives, prediction and cross-view consistency, each weighted by 0.1. Temporal pairs are drawn from the same source sequence at frame offset $k = 10$. Prediction uses an independent Hungarian assignment for each source visual pair with legal future supervision. The matching cost sums squared errors across all 768 feature dimensions. For pair i, let κ_i index a target sampled uniformly from its deduplicated legal future targets. Its prediction loss is

$$\ell_{\mathrm{pred},i}(\kappa_i) = \frac{1}{Nd} \min_{\pi_i \in \mathfrak{S}_N} \sum_{n=1}^{N} \left\| \widehat{S}_{i,n} - S^{+}_{i,\kappa_i,\pi_i(n)} \right\|_2^2, \tag{15}$$

where $N = 32$ by default, $d = 768$, and every S^+ is detached. The loss averages over matched elements. Source pairs receive equal prediction-loss weight, independently of their number of question rows. Gradients pass through the prediction side only. Sampling one target per pair gives an unbiased estimate of the objective that averages equally over all its legal targets.

The batch prediction loss averages these per-pair losses over the B_{pred} visual pairs with legal future supervision. When $B_{\mathrm{pred}} = 0$, the prediction term is skipped and contributes zero to the total loss.

Cross-view consistency uses image pairs from the official training splits of ScanNet v2 and nuScenes v1.0-trainval. ScanNet pairs contain two RGB frames from the same scene with different camera poses and overlapping fields of view. nuScenes pairs contain keyframe images from adjacent cameras with overlapping fields of view, associated with the same `sample_token`. The pairs are fixed before training.

For each valid pair i, the two views are encoded separately to obtain $\widetilde{S}_i^{(a)}$ and $\widetilde{S}_i^{(b)}$. Hungarian matching minimizes the sum of squared Euclidean slot distances, giving the per-pair consistency loss

$$\ell_{\mathrm{cons},i} = \frac{1}{Nd} \sum_{n=1}^{N} \left\| \widetilde{S}^{(a)}_{i,n} - \widetilde{S}^{(b)}_{i,\pi_i^*(n)} \right\|_2^2. \tag{16}$$

The per-pair consistency loss averages matched squared errors over all slot and feature dimensions. The batch loss ($\mathcal{L}_{\mathrm{cons}}$) averages these per-pair losses over valid view pairs. If a batch contains no valid view pair, the consistency term is skipped and contributes zero. The consistency term depends on T_1 outputs and does not require a predicted successor, which permits its retention in the state-transition ablation in Appendix B.

Table 13 gives the additional trainable parameters in A_θ, T_1, T_2, and decoder alignment, including the type embeddings. The shared visual tokenizer remains frozen.

Table 13: Additional trainable WorldPatch parameters in the default multimodal implementation. These counts exclude the shared backbone and do not give the total trainable parameter count of the controlled anticipation study.

Component	Instantiation	Parameters
Abstraction	Cross-attention resampler, $N = 32$	2,391,552
T_1	Two-layer Transformer, $d = 768$, $h = 12$, $d_{\text{ff}} = 2048$	11,029,504
T_2	$768 \to 2048 \to 768$ MLP with LayerNorm	3,150,080
Decoder alignment	Projection and type embeddings	3,151,360
Subtotal		19,722,496

G.2 Construction of Anticipation Examples

The controlled study uses the frozen EK100 V4 protocol on EPIC-KITCHENS-100 (Damen et al., 2022), identified as `worldpatch_v4_ek100_1s_gap_target0p5_onset_ge1_v2`. The protocol contains 493 videos and 66,033 valid events, yielding 65,971 deduplicated source visual pairs. Complete videos form the units of the `train`, `dev_tune`, and `dev_select` splits (Table 14). Questions and acceptable answers are read directly from the frozen sidecar files without regeneration.

Table 14: Video and source-pair counts in the frozen EK100 V4 protocol. Split names follow the frozen data manifest.

Split	Videos	Source visual pairs
`train`	395	50,522
`dev_tune`	49	7,122
`dev_select`	49	8,327
Total	493	65,971

The `dev_tune` split is used only to select and freeze $\lambda_{\text{pred}} = 0.1$ and the training, numerical, and evaluation settings. It is excluded from the final reported means and confidence intervals.

Final evaluation on EPIC-KITCHENS-100 uses the frozen `dev_select` split, which contains 49 videos and 8,338 valid events yielding 8,327 source visual pairs. Evaluation is conducted after the settings and rules have been fixed; the results are not used for further tuning, rule changes, or checkpoint selection. The source pairs are constructed using the prespecified eligibility filters.

For each eligible action, let a be its onset time, τ its duration, and $\mathcal{T}_v$ the valid presentation timestamps in its video. The source timestamp is

$$t_{\text{src}} = \max\{u \in \mathcal{T}_v : u \le a - 1\}, \tag{17}$$

with times measured in seconds. The target is the valid in-action timestamp nearest to

$$a + \min(0.5, \tau/2), \tag{18}$$

using the earlier timestamp in an exact tie. The realized horizon is $t_{\text{tgt}} - t_{\text{src}}$. Thus, the one-second source gap does not imply a fixed source–target interval. The current image is the visual input to anticipatory decoding; the future image supplies supervision and the reference for target-alignment evaluation.

A source visual pair can have multiple questions and multiple legal future targets. Both acceptable answers and legal targets are deduplicated before their respective reductions. If pair i has K_i future targets, its prediction objective averages the matched loss over those targets:

$$\mathcal{L}_{\text{pred},i} = \frac{1}{K_i} \sum_{k=1}^{K_i} \frac{1}{Nd} \min_{\pi \in \mathfrak{S}_N} \sum_{n=1}^{N} \left\| \widehat{S}_{i,n} - S^{+}_{ik,\pi(n)} \right\|_2^2. \tag{19}$$

Source visual pairs receive equal prediction-loss weight, independently of their number of question rows. Uniformly sampling one legal target per pair and update gives an unbiased estimate of this objective. The standardized diagnostic distance used in Table 3 is defined separately in Section G.4.

Ego4D external evaluation. We evaluate a randomly sampled subset of the Ego4D v2 FHO-LTA validation annotations. Action onset times are expressed on the original video's timeline, and the current image is the last decodable frame at least one second before onset, selected using presentation timestamps. Records with invalid action labels, missing videos, or no eligible input frame are excluded.

The resulting evaluation set contains 10,000 examples from 200 original videos.

Each example consists of a unique (`video_uid`, `current_frame_index`) and the question "What action happens next?" Annotations sharing the same current frame are merged before evaluation. Their official verb–noun phrases, checked against the action taxonomy, form a deduplicated set of acceptable answers. Gold-LL combines the reference scores by log-mean-exp as in Equation (10); future-action EM follows the normalization and validity rules in Section G.3.

All clips from the same original video share one `video_uid`. Scores are averaged over examples within each video, then equally across videos and the eight paired training seeds within each backbone. The A and R checkpoints and evaluation rules are fixed before evaluating this set. FHO-LTA validation is excluded from the controlled training, tuning, checkpoint selection, and selection of prompts, decoding settings, and scoring rules.

G.3 Task Scoring

Future Gold-LL. For each question, every distinct acceptable answer is scored by teacher forcing. The score averages log probabilities over its answer tokens and exactly one EOS; question, visual-prefix, prompt, and padding positions are excluded. The answer-level scores are combined by log-mean-exp, as in the main-text definition. This gives one score per question while allowing multiple acceptable answers.

On EPIC-KITCHENS-100, for seed s and condition c, let q_{scpvij} be the score for question j of pair i in video v from participant p. The reported within-seed mean is

$$\mu_{sc} = \frac{1}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} \frac{1}{|\mathcal{V}_p|} \sum_{v \in \mathcal{V}_p} \frac{1}{|\mathcal{I}_{pv}|} \sum_{i \in \mathcal{I}_{pv}} \frac{1}{|\mathcal{Q}_i|} \sum_{j \in \mathcal{Q}_i} q_{scpvij}. \tag{20}$$

Each averaging step gives equal weight to the units at that level. In particular, additional questions do not increase a pair's weight, and participants contribute equally to the final within-seed score. Within each backbone, the primary effect is the mean of $\mu_{sR} - \mu_{sA}$ across the eight paired seeds.

For all EM evaluations, we use greedy decoding with a maximum of 12 generated tokens, including EOS, and stop at the first EOS.

Current-action EM. For current-action evaluation with Slot-MLLM, the model receives the current image and the prompt *What's happening now?* The generated response is compared with the full set of current-action reference answers for its visual pair. Predictions and references undergo Unicode NFKC normalization, case folding, trimming, and consecutive-whitespace collapse; normalized references are then deduplicated. A prediction receives credit if it matches any reference exactly.

Responses without EOS, longer than 12 tokens (including EOS), or outside the permitted verb–noun action-phrase format receive zero credit and remain in the denominator. Each visual pair has equal weight within a seed, and scores are averaged equally across seeds. The R–P comparison tests whether WorldPatch is non-inferior to the present-only baseline on current-action questions under the prespecified margin. Scores are reported as percentages, and the R–P contrast and its non-inferiority margin are reported in percentage points.

Future-action EM. We compare each generated answer with its acceptable future references using the same normalization and validity rules as current-action EM. A response receives credit if it matches any reference exactly. On EPIC-KITCHENS-100, we average question scores within pairs, pair scores within videos, and video scores within participants. On Ego4D, we average scores within each original video and then equally across videos. Both evaluations average equally across the eight training seeds. Scores are reported as percentages, and R–A differences are in percentage points.

G.4 Target Alignment and State Replacement

Standardized target alignment. Each of the eight Slot-MLLM R checkpoints has its own feature-wise population mean μ_s and standard deviation σ_s, estimated only from its detached training-split future-target slots. The same transform

$$z_s(X)_{nd} = \frac{X_{nd} - \mu_{s,d}}{\sigma_{s,d}} \tag{21}$$

is applied to predictions, evaluation targets, and all reference states. If any feature standard deviation is at most 10^{-12}, target-alignment evaluation stops for that checkpoint.

For a candidate state X and target S^+, the diagnostic distance is

$$d_s(X, S^+) = \frac{1}{Nd} \min_{\pi \in \mathfrak{S}_N} \sum_{n=1}^{N} \left\| z_s(X)_n - z_s(S^+)_{\pi(n)} \right\|_2^2, \tag{22}$$

where Hungarian matching finds the minimizing assignment. The squared Euclidean cost determines the matching, and the reported distance averages over all $N \times d$ elements. For multiple future targets, each candidate is matched independently to each target before averaging:

$$D_{s,i}(X) = \frac{1}{K_i} \sum_{k=1}^{K_i} d_s(X_i, S_{ik}^+), \qquad g_{s,i,b} = D_{s,i}(b) - D_{s,i}(\widehat{S}). \tag{23}$$

This uses all legal targets with equal weight within a pair. Pair distances and gains are then averaged within videos, within participants, and across seeds, following the hierarchy used for future Gold-LL after its question-level reduction.

The three references in Table 3(a) are the post-abstraction current state S_t, a fixed medoid constructed only from the same checkpoint's training targets, and a predicted successor drawn from another pair in the same video. The current-state reference precedes T_1. Derangement uses a frozen one-to-one mapping with no fixed points, so no pair receives its own successor. All comparisons are formed within a checkpoint before aggregation across seeds.

Medoid construction uses one primary future image per training pair, as specified in the frozen manifest. When multiple legal future images are available, the primary image is the one closest to the nominal future timestamp, with ties resolved successively by PTS and frame SHA-256. This pair-to-image mapping is shared across all R checkpoints; each checkpoint encodes these images in its own detached target-state space.

We rank training pairs by the protocol's frozen salted SHA-256 ordering and retain the first 256, or all pairs if fewer are available. Their primary target-state sets form both the candidate and reference sets. Within this fixed sample, each candidate is matched independently to every reference target set using Hungarian assignment and squared-Euclidean cost. We choose the candidate with the smallest total distance, resolving ties by the protocol's frozen hash-based rule. This is the exact medoid within the selected sample and a deterministic approximation to the full training-set medoid. Selection takes place in the checkpoint's original target-state space.

For target-alignment evaluation, the selected medoid is then transformed with the same train-target standardizer used for predictions, other references, and evaluation targets. The primary-image rule specifies medoid inputs; the training objective and the alignment diagnostic retain their respective reductions over all legal future targets.

All eight R checkpoints pass the finite/nonconstant check. This check excludes non-finite or numerically constant representations but does not rule out low-rank variation.

Fixed-model content replacement. For Table 3(b), each of the eight R checkpoints is evaluated with its weights fixed. Native conditioning uses the recipient's predicted successor; deranged conditioning substitutes another pair's predicted successor through a frozen within-video one-to-one mapping without fixed points. The recipient present state, prefix length, type and position embeddings, projections, question, decoder, decoding settings, and evaluation cohort are held fixed. Both modes use the Gold-LL scoring and aggregation defined above. The paired effect is native minus deranged Gold-LL. The native scores are the same R scores used in Table 1. Appendix A describes the separate multimodal copy and slot-order interventions.

G.5 Statistical Analysis and Per-Seed Results

All A/R estimates and confidence intervals are computed separately within each backbone. The R_{stop} comparison, current-action evaluation, target-alignment analysis, and fixed-model state replacement use Slot-MLLM.

For EPIC-KITCHENS-100, uncertainty is estimated with 10,000 crossed-bootstrap replicates using `numpy.random.PCG64` and base seed `8230983788122896115`. Each replicate samples eight paired training seeds with replacement, samples participants with replacement, and then samples videos with replacement within each selected participant occurrence. The same participant and video draws are used for every sampled seed and paired condition or intervention mode. All pair-level observations within each selected video remain together. Each replicate recomputes the paired contrast using its original averaging rule. Future Gold-LL, future-action EM, and target-alignment gains retain their video–participant hierarchy. Current-action EM retains equal weight per visual pair, including multiplicities induced by the resampled videos, before averaging over the sampled seeds.

The R–A comparisons in future Gold-LL and future-action EM, and the R_{stop}–A comparison in future Gold-LL, use two-sided 95% percentile intervals. The R_{stop}–A comparison is tested only if the primary Slot-MLLM R–A Gold-LL criterion is met. The current-action EM comparison, each target-alignment gain, and native–deranged Gold-LL use one-sided 95% lower bounds. Two-sided intervals use the 2.5th and 97.5th percentiles; one-sided lower bounds use the 5th percentile. All quantiles are computed with `np.quantile(method="linear")`.

Support for a positive R–A effect requires its lower confidence limit to exceed zero. Current-action EM non-inferiority uses a lower bound greater than -2 percentage points. Target-alignment support requires positive lower bounds for all three reference-minus-native gains together with the finite/nonconstant check. Positive-seed counts describe the observed direction across runs.

With Slot-MLLM on EPIC-KITCHENS-100, R_{stop} improves future Gold-LL over A by 0.023 (95% CI $[0.003,\ 0.043]$). Current-action EM is 71.58% for P and 71.69% for R, with a paired R–P difference of 0.11 percentage points. The one-sided 95% lower confidence bound is -0.52 percentage points, above the prespecified non-inferiority margin of -2 percentage points.

For Ego4D, we form 10,000 paired bootstrap replicates by independently resampling the eight seed pairs and the original videos with replacement. Each replicate uses the same video draws for all sampled seeds and both conditions, retaining all examples within each sampled video. Scores are recomputed with equal weighting across videos and seeds. Both metrics use the same resamples, and the R–A contrasts are summarized by two-sided 95% percentile intervals.

Tables 15 and 16 report per-seed Slot-MLLM scores and effects on EPIC-KITCHENS-100. S1–S8 identify the same paired runs in both tables and in the action-label analysis in Section G.6. Table 17 reports Qwen results; each Q1–Q8 identifier denotes the same checkpoint pair across EPIC-KITCHENS-100 and Ego4D. All contrasts are computed before rounding. Uncertainty for the fixed-checkpoint multimodal comparisons is described in Appendix A.

Table 15: Per-seed task scores on EPIC-KITCHENS-100 with Slot-MLLM. Gold-LL and EM use their respective aggregation rules in Section G.3.

Condition	S1	S2	S3	S4	S5	S6	S7	S8
Future Gold-LL								
P	−1.225	−1.261	−1.259	−1.190	−1.188	−1.236	−1.174	−1.246
A	−1.197	−1.209	−1.269	−1.114	−1.195	−1.201	−1.186	−1.215
G	−1.200	−1.232	−1.300	−1.122	−1.173	−1.218	−1.190	−1.242
R_{stop}	−1.138	−1.230	−1.215	−1.100	−1.142	−1.174	−1.181	−1.221
R	−1.144	−1.119	−1.282	−1.079	−1.150	−1.167	−1.197	−1.195
Current-action EM (%)								
P	70.11	73.15	70.81	72.19	71.27	72.64	70.48	71.96
R	69.20	74.10	70.95	71.80	71.60	73.10	70.27	72.51

Table 16: Per-seed paired effects on EPIC-KITCHENS-100 with Slot-MLLM. EM differences are in percentage points; target-alignment gains use the standardized mean-squared distance in Equation (22). Action-label contrasts are defined in Section G.6.

Comparison	S1	S2	S3	S4	S5	S6	S7	S8
Future Gold-LL								
R–A	+0.053	+0.090	−0.013	+0.035	+0.045	+0.034	−0.011	+0.020
R_{stop}–A	+0.059	−0.021	+0.054	+0.014	+0.053	+0.027	+0.005	−0.006
Current-action EM (pp)								
R–P	−0.91	+0.95	+0.14	−0.38	+0.32	+0.46	−0.22	+0.55
Target-alignment gain: reference minus native								
Current state	+76.6	+45.5	+6.8	+58.9	+77.3	+44.8	−7.5	+62.8
Medoid	+144.7	+76.0	−5.3	−14.4	+97.1	+31.9	+25.9	−11.7
Deranged	+105.6	+8.9	−10.6	+99.1	+125.8	−4.3	+72.7	+22.7
Fixed-R content use: native minus deranged Gold-LL								
Native–deranged	+0.068	+0.051	−0.014	+0.024	+0.040	+0.034	−0.008	+0.029
Action-label agreement: different–same gains								
Future Gold-LL	+0.026	+0.043	+0.010	+0.036	+0.039	+0.046	−0.007	+0.023
Future-action EM (pp)	+4.8	+6.9	−0.6	+5.5	+5.8	+4.6	−0.3	+5.3

Table 17: Per-run Qwen A/R scores on EPIC-KITCHENS-100 and Ego4D. The same run identifiers denote the same checkpoint pairs across datasets.

	Future Gold-LL			Future-action EM		
Run	A	R	$R-A$	A (%)	R (%)	$R-A$ (pp)
EPIC-KITCHENS-100						
Q1	−1.056	−1.042	+0.014	50.7	55.5	+4.8
Q2	−1.066	−1.025	+0.041	49.0	49.9	+0.9
Q3	−1.033	−1.002	+0.031	50.1	56.7	+6.6
Q4	−1.077	−1.075	+0.002	49.5	53.0	+3.5
Q5	−1.054	−1.005	+0.049	48.5	54.5	+6.0
Q6	−1.020	−0.991	+0.029	51.1	52.4	+1.3
Q7	−1.059	−1.039	+0.020	50.9	55.3	+4.4
Q8	−1.036	−1.014	+0.022	50.3	55.0	+4.7
Mean	−1.050	−1.024	+0.026	50.0	54.0	+4.0
Ego4D						
Q1	−1.330	−1.268	+0.062	35.7	35.4	−0.3
Q2	−1.311	−1.275	+0.036	36.1	34.0	−2.1
Q3	−1.292	−1.236	+0.056	37.0	40.2	+3.2
Q4	−1.319	−1.337	−0.018	35.9	39.4	+3.5
Q5	−1.314	−1.289	+0.025	34.1	39.4	+5.3
Q6	−1.258	−1.214	+0.044	37.2	40.0	+2.8
Q7	−1.297	−1.290	+0.007	35.5	40.4	+4.9
Q8	−1.280	−1.260	+0.020	36.5	39.4	+2.9
Mean	−1.300	−1.271	+0.029	36.0	38.5	+2.5

G.6 Action-Label Agreement

The secondary analysis in Table 4 reuses the eight paired Slot-MLLM A/R runs and the EPIC-KITCHENS-100 `dev_select` split. Current-action references come from the frozen data protocol and are verified against the actual input frame and timestamp. Future references include every acceptable answer across all future questions for each pair. Using the normalization in Section G.3, we map references to canonical verb–noun classes in the original V4 category tables. All references must map unambiguously, and each side must yield exactly one class for a pair to be eligible. Synonymous references may map to the same class. We fixed the mapping and groups before

Table 18: Coverage of the action-label analysis on EPIC-KITCHENS-100.

Category	Pairs
Included	
Same labels	2,106
Different labels	5,418
Excluded	
Unverifiable current-reference source	48
Ambiguous canonical mapping	124
Multiple current action classes	623
Multiple future action classes	8
Total	8,327

examining groupwise gains. This mapping is used only for grouping; scoring retains the original reference answers.

Table 18 summarizes eligibility. Excluded pairs are assigned the first applicable reason in the order shown. The 7,524 eligible pairs span all 49 videos and 27 participants. Every video contains both groups, so the common-coverage requirement excludes no additional pairs.

After averaging question scores within each pair as in Section G.3, let Δ_{spvg} denote the mean pair-level R−A gain for seed s, participant p, video v, and label group g. Here $g = 0$ denotes matching labels and $g = 1$ different labels. The primary contrast is

$$D = \frac{1}{8}\sum_{s=1}^{8} \frac{1}{|\mathcal{P}^*|} \sum_{p\in\mathcal{P}^*} \frac{1}{|\mathcal{V}_p^*|} \sum_{v\in\mathcal{V}_p^*} (\Delta_{spv1} - \Delta_{spv0}) \,, \tag{24}$$

where $\mathcal{V}_p^*$ contains participant p's videos with both groups and $\mathcal{P}^*$ contains participants with at least one such video. Group scores and gains use these same outer weights. Future-action EM uses the same cohort and averaging rule.

We use the 10,000 crossed-bootstrap replicates described in Section G.5, keeping the label groups and analysis cohort fixed. The sampled seed pairs, participants, and videos are shared across both groups, A/R, and both metrics. Each replicate recomputes both group gains and their difference. We report two-sided 95% percentile confidence intervals. The per-seed gain differences are included in Table 16.

The matching- and different-label groups contain 318 and 742 future action classes, respectively, with 267 shared classes. Prediction horizons lie in $[1.45, 1.55)$ seconds for 80.3% and 74.1% of their target weight, respectively; each pair has equal total weight, divided equally among all its legal targets. The comparison describes retained utility across observed label groups that also differ in action composition and prediction horizon.

G.7 Ethics and Release

We plan to release resolved configurations, immutable manifests, source and feature hashes, normalization statistics, evaluators, label mappings, per-pair group assignments, and per-run outputs, subject to the terms of the source datasets. Licensed videos and extracted frames will not be redistributed. No identity inference is attempted.