SEEING AHEAD: PLUG-IN WORLD MODELS FOR PREDICTIVE REASONING IN FROZEN VLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Understanding a video involves recognizing the visible scene and anticipating how it may change. Video world models learn predictive representations; vision–language models (VLMs) connect visual evidence to questions and answers. We study how to combine these capabilities while retaining the VLM's own perception. We propose FoReST (Forecast Residual State Transfer), which adds predictive updates to a frozen VLM's visual tokens. One variant translates changes predicted by a pretrained world model; the other forecasts the update from that model's observed-video features. Only a compact head is trained from video continuations, without question–answer supervision. We introduce CLEVRER-Next, a next-event QA dataset built from existing CLEVRER videos and annotations, with explicit observation cutoffs and retained continuations for diagnosis. On CLEVRER-Next and ComPhy, latent transfer improves accuracy by 11.5 and 11.7 percentage points over the same frozen VLM given observed video. Against a trained forecaster of the VLM's own features, latent transfer provides additional gains on ComPhy, while direct forecasting improves CLEVRER-Next collision prediction. Controlled comparisons show that retaining native vision reduces degradation from imperfect forecasts and that prediction-input training can improve their translation.

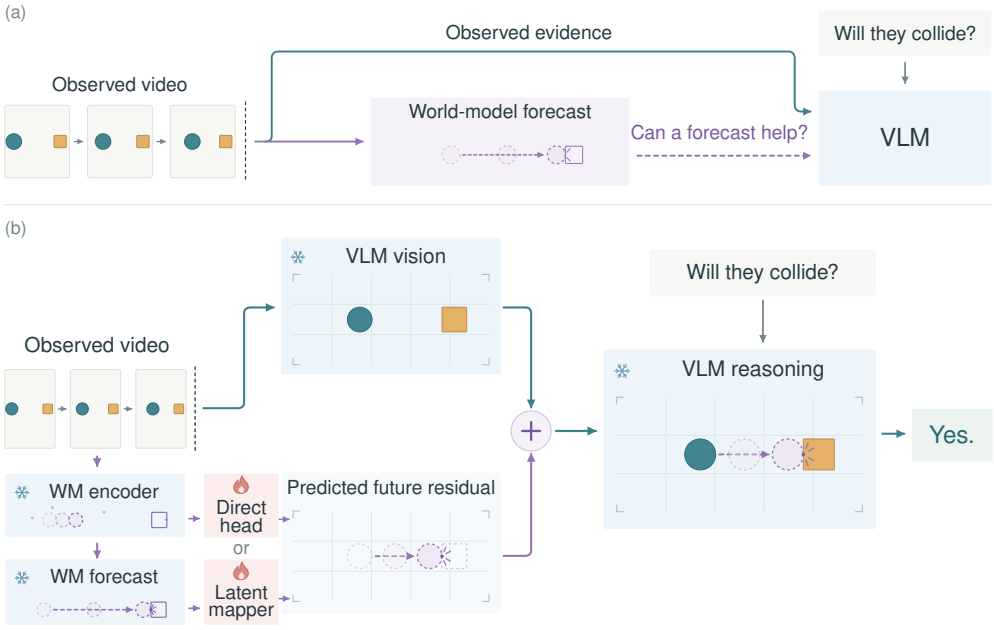


Figure 1: **FoReST: predictive updates with native perception.** (a) A video world model provides predictive information about an observed scene. (b) FoReST adds a predicted residual to the VLM's own visual tokens. FoReST-Direct forecasts it from observed world-model features; FoReST-Latent translates a pretrained forecast. These are alternative routes: only the chosen head is trained, while all pretrained components stay frozen. Trajectories and answers are schematic.

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.

1 INTRODUCTION

A video stops as one object approaches another. Will they collide, or will one leave the scene first? Answering requires connecting the observed scene to a prediction beyond it. Future-event video question answering makes this connection explicit (Yi et al., 2020; Chen et al., 2022).

This task brings together visual grounding and anticipation. Vision–language models (VLMs) connect questions to visual evidence (Li et al., 2023; Liu et al., 2023), while video world models learn predictive representations (Bardes et al., 2024; Assran et al., 2025). Their capabilities suggest a division of labor: the VLM relates the question to the observed objects, and the world model supplies information about their possible continuation (Figure 1(a)). FOREWARN uses latent forecasts in VLM-based robot policy steering (Wu et al., 2025); here we study a visual-input interface for future-event QA with the complete VLM frozen.

The world model forecasts in its own feature space, whereas the frozen VLM expects features from its native visual pathway. We therefore train the auxiliary branch to predict a change to a native visual reference rather than the entire continuation representation. This reference anchors the learned update in the receiver's coordinates.

To integrate predictive information into a frozen VLM while retaining its own perception, we propose **FoReST** (**Fo**recast **Re**sidual **S**tate **T**ransfer). FoReST adds a learned residual to native visual features constructed from the observed clip (Figure 1(b)). The framework has two variants: FoReST-Latent translates the change predicted by a pretrained world model into the VLM's feature space; FoReST-Direct learns the same update from observed world-model features.

To teach the head what information to add, we use the VLM's own visual features as supervision. The later frames of each training video show how its observed scene actually continued. We encode that continuation with the frozen VLM and compare it with the native features constructed from the observed clip. Their difference defines the update the head learns to predict using observed information alone. All pretrained components remain fixed, and no question–answer labels are used.

To evaluate predictive updates with precisely defined unseen events, we introduce **CLEVRER-Next**, a next-event question-answering dataset constructed from existing CLEVRER videos and simulator annotations (Yi et al., 2020). Each question asks which specific collision, entrance, or exit occurs first after an observation cutoff, with five candidate events to distinguish. Actual continuations are retained for diagnostic evaluation. We demonstrate FoReST's effectiveness on CLEVRER-Next and ComPhy's predictive questions (Chen et al., 2022): both variants improve the frozen VLM's observed-video performance. Controlled studies examine how preserving native vision and choosing fitting inputs affect the use of predictive information.

Our contributions are:

- We propose FoReST, which adds world-model predictive updates to a frozen VLM through latent-transfer and direct-forecasting heads trained without question–answer labels.
- We build CLEVRER-Next, a next-event QA dataset from CLEVRER videos and annotations, with five-choice questions and retained continuations for diagnostic evaluation.
- We demonstrate FoReST's effectiveness on both datasets and analyze native visual preservation, fitting inputs, and the use of video-dependent forecasts.

2 FORECST: FORECAST RESIDUAL STATE TRANSFER

FoReST gives a frozen VLM information about what may happen after the observed clip ends. It keeps the VLM's own visual features and trains a small head to predict an additive change, or *residual*, using a frozen world model. This residual represents information about the unseen continuation in the VLM's feature space. The VLM then answers the question using the observed evidence together with the predicted update. Figure 1(b) illustrates FoReST and its two alternative routes to this update.

2.1 PRESERVE NATIVE VISION AND PREDICT A CHANGE

Predictors of future visual features can be learned from unlabeled video (Vondrick et al., 2016). Here, the observed frames establish objects and motion, while future-event questions depend on the continuation. We forecast its visual features as a residual, learning how future information should change the VLM's representation while retaining a native visual base.

Let C be the observed clip. We extend it with copies of its final sampled frame to form a static reference continuation, then encode this input through the VLM's usual visual pathway. The resulting native visual tokens Q_R are the features supplied to its language model; F indexes the continuation block. The reference uses observations alone. Adding the forecast at these positions leaves the observed-video tokens unchanged:

$$\widetilde{Q}_F = Q_{R,F} + \widehat{\Delta Q}, \qquad \widehat{\Delta Q} = H(C). \tag{1}$$

Here $H(C)$ predicts the change to the native base, using only observed information. Observed-video tokens, reference-video tokens when present, and text embeddings remain unchanged before language processing. The update is question-independent and reusable across questions about the clip.

We implement H in two ways: ***FoReST-Latent*** translates a pretrained world-model forecast into a native residual, and ***FoReST-Direct*** forecasts it from observed world-model features.

2.2 FOREST-LATENT: TRANSLATE A PRETRAINED FORECAST

A pretrained world model already provides a forecast of the video's continuation. FoReST-Latent reuses that prediction so the learned head can focus on making it useful to the VLM. The head translates the predicted departure from the observed state into a change in the VLM's visual features.

Let P denote the world model's predicted future states. Let R be its encoding of the reference used to form Q_R, taken at the continuation positions. Subtracting R expresses the forecast relative to a known visual state. A learned mapper D_ϕ translates this difference into the native residual:

$$H_{\text{Latent}}(C) = D_\phi(P - R). \tag{2}$$

Both models thus use the same reference input, expressed in their own feature spaces. The subtraction defines the residual operationally; it does not guarantee a separation of appearance and dynamics.

The mapper applies a shared transformation to corresponding forecast positions and preserves $D_\phi(0) = 0$. If the world model predicts no change from the reference, the VLM's native base is left unchanged. Appendix A.1 specifies the zero-preserving layers and spatial alignment. The pretrained encoder and predictor remain fixed.

2.3 FOREST-DIRECT: LEARN THE FORECAST IN VLM COORDINATES

The second variant learns forecasting specifically for the update the VLM needs. The world model's encoder already represents the observed scene and its motion; a head trained on native residual targets can learn how to combine those features to anticipate the continuation. FoReST-Direct therefore learns prediction and translation together, without an intermediate forecast in world-model space.

Given observed features $O_V = E_V(C)$ from the same frozen world-model encoder, an attention head G_ψ predicts

$$H_{\text{Direct}}(C) = G_\psi(O_V). \tag{3}$$

Each output position attends to the observed features to predict its native update. This lets the head combine information across the observed clip for each continuation position. It does not call the released world-model predictor; Appendix A.3 specifies the attention implementation.

The variants place forecasting in different spaces: Latent reuses a pretrained forecast and learns its translation, while Direct learns the forecast in the VLM's coordinates. We evaluate the two variants separately with the same native base and residual target.

2.4 LEARN THE UPDATE FROM PAIRED CONTINUATIONS

The frozen VLM itself defines what the predicted update should represent. For a training clip with an observed prefix and an actual continuation, let Q_A be the VLM's encoding of the actual clip. The

difference from the observation-derived base at the continuation positions is

$$\Delta Q = Q_{A,F} - Q_{R,F}, \qquad \mathcal{L}(H) = \mathbb{E}\big[\ell_Q\big(H(C), \Delta Q\big)\big]. \tag{4}$$

The target is the change the actual continuation would contribute to the receiver's visual features. If the predicted update matches this target, adding it to the reference recovers the VLM's own actual-continuation features at the updated positions. The alignment loss compares token directions, normalized coordinates, and magnitudes; Appendix A.2 gives its exact reductions.

The actual continuation supplies supervision, not an input to H. In the primary FoReST-Latent recipe, the mapper is trained on $P - R$, the predicted states it will also receive at inference. It consequently learns to translate the predictor's outputs, rather than only idealized actual-future states. The actual-input and both-input alternatives test this choice in Section 5.3.

Only the forecasting or translation head is updated. All pretrained video and VLM components stay frozen, and no questions or answers train the head. Checkpoints are selected by alignment error on separate validation videos. At inference, the chosen head produces the update from the observed clip, Equation 1 inserts it, and the unchanged VLM answers the question.

3 Evaluation Protocols and Setup

3.1 CLEVRER-Next: Identify the Next Event

We construct CLEVRER-Next from CLEVRER videos and simulator annotations (Yi et al., 2020). Each five-choice question asks which specific collision, entrance, or exit occurs first after the observation cutoff. Each choice identifies an event and its participating objects, rather than only an event category. An alternative may be wrong because it occurs later, even if it eventually happens. Collision and exit objects are visible at the cutoff; an entrance concerns an object not yet visible. Appendix B.1 specifies the timing, unambiguous object descriptions, visibility filters, and distractors.

Each scene supplies two presentations with the same candidate meanings but different wording and order. Answer accuracy scores them individually; both-form accuracy requires both answers to be correct, not merely to agree. Retained continuations support diagnosis of the same question with predicted features, mapped actual-future features, or native real-future input. Actual-future inputs are labeled diagnostic conditions. Event analyses group scored questions by target event, keeping its type hidden and all choices intact.

3.2 ComPhy: Judge Predictive Candidates

ComPhy retains its official predictive questions and candidate statements (Chen et al., 2022). Each candidate is judged independently as True or False: True means that it correctly answers the question, including any negation. Candidate accuracy scores these judgments; all-options accuracy requires both judgments for a question to be correct. The VLM also receives four videos of separate experiments with the same objects, not continuations of the target. The forecasting head receives only the target video. Alignment pairs prefixes and suffixes within available clips; QA uses the complete observed target to ask about later events. Appendix B.2 gives exact sampling and timing.

3.3 Frozen Models and Evaluation Data

The primary system combines the released V-JEPA 2 ViT-L video model with Qwen3.5-27B (Assran et al., 2025; Qwen Team, 2026). Pretrained weights remain frozen. Heads are fitted on paired videos and selected by alignment validation, without question–answer labels. The main evaluation contains 1,000 CLEVRER-Next scenes and 297 ComPhy groups.

Primary learned routes are trained independently three times; tables report their mean and sample standard deviation. The additional receivers, Qwen2.5-VL-7B and InternVL3.5-8B (Bai et al., 2025; Wang et al., 2025), use one fitted head each on the same CLEVRER analysis subset. Appendix A.8 specifies their architectures and fitting.

3.4 Baselines and Evaluation

VLM (observed video) receives 16 real frames from the observed interval. *VLM + static continuation* instead uses 12 observed frames followed by four repeats of the last observation, matching FoReST's input layout without adding an update. *VLM-Forecast* learns the same residual target from the VLM's own observed tokens, removing the external video model. The three learned heads have approximately 12.6M parameters each and share training pairs, targets, and output interfaces.

Each FoReST variant's shuffled control substitutes another video's complete update, retaining the recipient's native video, references, and question. We score answers by fixed candidate likelihoods. Paired uncertainty estimates resample scenes or groups with all their answers. Appendices A and B specify architectures, training, prompts, randomization, and statistics.

4 Main Results

4.1 Predictive updates improve frozen-VLM question answering

Both FoReST variants improve future-event QA with the VLM's visual and language components frozen (Table 1). FoReST-Latent adds 11.5 answer-accuracy points on CLEVRER-Next and 11.7 candidate-accuracy points on ComPhy over observed-video inference. Gains also hold on the stricter joint metrics and against static continuation. Because that baseline shares the methods' input layout without a learned update, the benefit extends beyond rearranging observed frames.

Table 1: Future-event QA with one frozen VLM: all 1,000 CLEVRER-Next scenes, its 600-scene collision subset, and 297 ComPhy groups. Learned entries are mean ± sample SD of three fits. Collision retains the original five choices. Shuffling changes only the auxiliary update. Bold marks the highest unshuffled accuracy per column.

	CLEVRER-Next			ComPhy	
	All events		Collision subset	All groups	
Method	Answer	Both-form	Answer ($n = 600$)	Candidate	All-options
Observed video	46.2	41.5	48.1	69.0	47.8
Static continuation	50.3	46.1	51.6	70.5	49.5
VLM-Forecast	61.2 ± 0.1	56.8 ± 0.5	64.5 ± 0.6	76.0 ± 0.6	58.6 ± 1.2
FoReST-Latent	57.6 ± 0.5	52.3 ± 0.6	62.6 ± 0.3	$\mathbf{80.7 \pm 0.1}$	$\mathbf{66.2 \pm 0.2}$
FoReST-Direct	$\mathbf{63.2 \pm 0.5}$	$\mathbf{59.0 \pm 0.3}$	$\mathbf{69.1 \pm 0.6}$	77.3 ± 1.4	61.6 ± 2.0
Latent shuffled	51.7 ± 0.5	46.9 ± 0.4	52.7 ± 0.9	75.0 ± 0.8	57.4 ± 1.0
Direct shuffled	52.8 ± 0.3	47.6 ± 0.6	53.6 ± 1.1	72.9 ± 0.9	54.1 ± 1.6

4.2 What does an external forecasting route add?

VLM-Forecast is a strong comparator: it learns the same future-targeted update from the receiver's own visual features at a similar trainable-head budget. Its gains over both unchanged native inputs show that a learned prediction can help even without an external video model. The FoReST comparison therefore tests an external forecasting route beyond prediction from native features.

Two settings show a clear additional benefit. On ComPhy, FoReST-Latent improves candidate accuracy over VLM-Forecast by 4.7 percentage points and also improves all-options accuracy. On the 600 CLEVRER collision scenes, FoReST-Direct adds 4.5 answer-accuracy points over the same comparator. These questions require anticipating an interaction between objects already visible at the cutoff, while retaining all five candidate events and both question presentations. The full task and collision scores are reported together in Table 1.

The routes have different task profiles: Direct has the highest full-set CLEVRER mean, whereas Latent is strongest on ComPhy. Latent is below VLM-Forecast on full-set CLEVRER, and Direct's smaller aggregate advantage is less clearly resolved than its collision advantage. Appendix C.4 retains

the complete event-wise comparison, including the paired intervals. These results concern complete forecasting routes with a common output interface, rather than an isolated change of encoder.

Both variants also lose accuracy with another video's update (Table 1). Section 5.1 examines this scene correspondence and the role of temporal order.

5 Understanding Predictive Transfer

We first examine whether useful updates depend on the observed sequence. We then study two interface choices—retaining native vision and training on predicted inputs—before analyzing how future evidence is read and which events benefit across receivers.

5.1 Does the update use the observed sequence?

On all 1,000 CLEVRER scenes, we substitute another video's complete update (the shuffled control), fixing native input, question, and head (Figure 2).

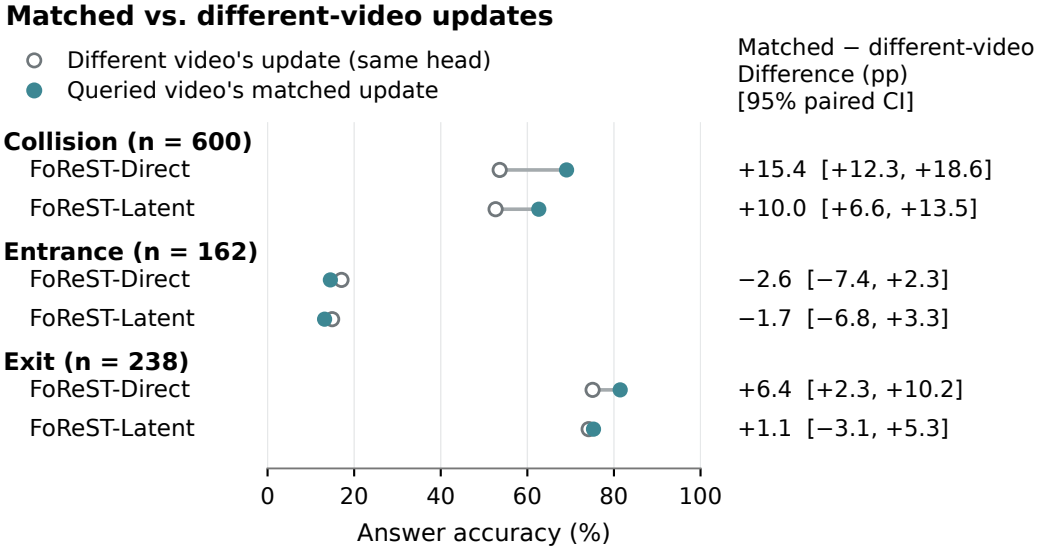


Figure 2: **Matched versus different-video updates.** Filled points use the queried video's update; open points use another video's update from the same head. Native input and question stay fixed, and all original questions and choices are retained. Accuracies average three fitted heads; the reported 95% intervals quantify paired scene-sampling uncertainty for those fixed heads.

Collision questions show the clearest benefit from video-matched updates for both variants; FoReST-Direct also benefits on exits. Entrance gains over the static reference can occur without an advantage over a different video's update, so improvement alone does not establish dependence on the scene.

Scene matching can reflect static content, so we also test temporal order. With one fixed FoReST-Direct head per dataset and unchanged native VLM input, reversing the external model's observed temporal blocks lowers CLEVRER answer accuracy by 4.1 points and ComPhy candidate accuracy by 5.75 points. Using one training-mean update for all videos lowers them by 9.2 and 6.25 points. These controls show sensitivity to temporal order and gains beyond a shared update (Appendix D.3).

5.2 Why add rather than replace native vision?

The following analyses use fixed development subsets of 500 CLEVRER scenes and, where applicable, 200 ComPhy groups (See Appendix B.3 for statistics).

Useful predictive content can still be imperfect. Both interfaces preserve the observed-video tokens; they differ only at the continuation positions. Addition predicts a change to the native reference,

whereas replacement predicts those tokens in full. The two heads are trained on the same actual absolute world-model states, with their respective output targets. Holding each head fixed, we interpolate its evaluation input between actual states T and forecasts P:

$$Z_\lambda = (1-\lambda)T + \lambda P, \qquad \lambda \in \{0, \tfrac{1}{2}, 1\}. \tag{5}$$

Actual and mixed inputs use future observations; the predicted endpoint uses only observed context. These diagnostic fits are separate from the primary prediction-input-trained head.

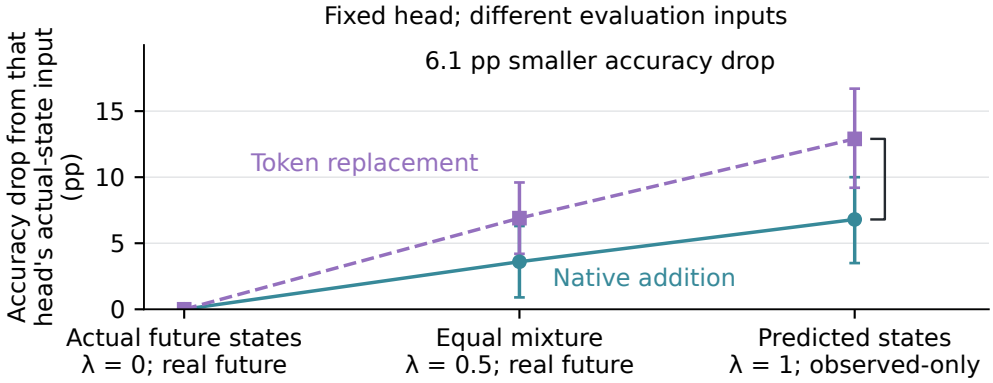


Figure 3: **Fixed heads under changing evaluation inputs.** Accuracy drops are relative to each head's actual-state endpoint: 57.2% for addition and 60.9% for replacement. Bars are paired 95% scene-bootstrap intervals on 500 CLEVRER analysis scenes. The 6.1-point contrast compares degradation, not final accuracy. Actual and mixed inputs use future observations; predicted inputs do not.

Replacement starts with higher actual-state accuracy but loses more as its input becomes a prediction (Figure 3). Native addition reduces this drop by 6.1 points and introduces fewer errors on questions the static baseline answered correctly. Retaining the native reference thus reduces deterioration within this controlled comparison: the forecast supplements an existing representation instead of supplying the entire continuation block. Appendix D.1 retains all endpoints and paired contrasts.

5.3 Why train the mapper on predictions?

The actual continuation defines what the update should represent, but the mapper will receive a forecast when it is used. We therefore compare training on actual world-model states, predicted states, or both input types. Architecture, paired clips, frozen models, and the checkpoint criterion remain fixed; every fitted head is evaluated on predicted states. Unlike the preceding interpolation experiment, this comparison changes training inputs, not the evaluation input of a fixed head. Both-input fitting averages two alignment losses rather than interpolating states.

Table 2: FoReST-Latent training inputs, with predictions at evaluation. Entries are mean accuracy (%) $\pm$ sample SD over three fits on the 500-scene/200-group analysis subsets. Both-input fitting averages two losses. Appendix C.2 reports both metrics and full comparisons.

Mapper training input	CLEVRER answer	ComPhy candidate
Actual states	50.33 ± 0.15	76.25 ± 0.50
Predicted states	50.70 ± 1.05	80.42 ± 0.14
Both input types	50.57 ± 0.21	79.50 ± 0.50

On ComPhy, training on predictions improves candidate accuracy by 4.2 percentage points over training on actual states; using both input types also helps (Table 2). Changing checkpoint selection alone yields much smaller differences. Actual continuations define the target; predicted inputs teach the mapper to translate the states it receives at deployment. CLEVRER answer accuracy changes little across recipes, and its both-form metric does not show the same improvement. Training a readout on predicted features has precedent in visual anticipation (Vondrick et al., 2016).

5.4 How much future evidence survives translation?

Can the continuation-token block carry useful real-future evidence? Replacing only this block with the VLM's native real-future features reproduces every evaluated actual-frame answer. These positions can therefore carry future evidence useful to the receiver.

Figure 4 compares native real-future input with mapped actual and predicted world-model states on the 500-scene analysis subset. Both mapped conditions use one fixed actual-input reference head, separate from primary fits. The primary receiver's native-to-mapped actual-future gap is 29.1 answer points. Since this gap already appears with actual states, forecast error alone cannot explain it; representation and translation also matter. The smaller Qwen shares this ordering (Appendix D.5).

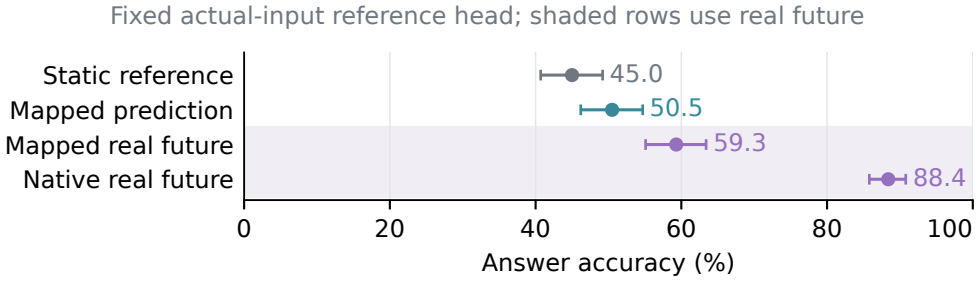
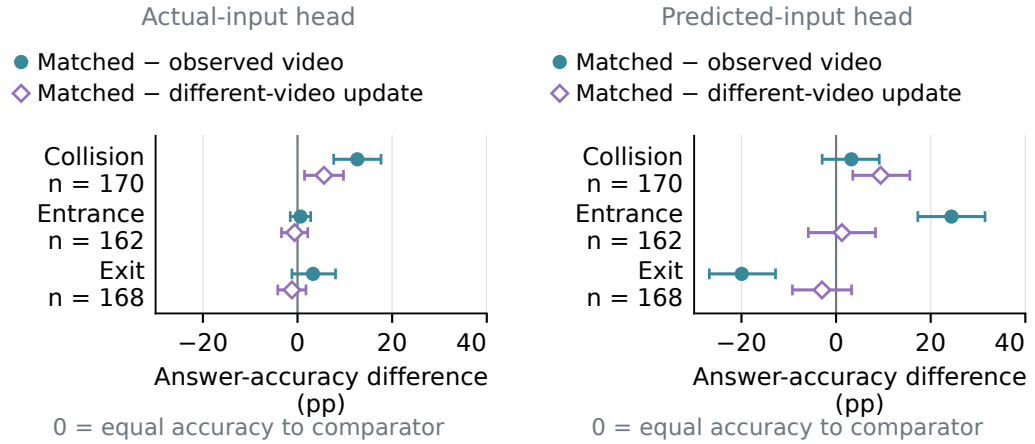


Figure 4: **Native future evidence versus its mapped representation.** Qwen3.5-27B uses one fixed actual-input reference head, separate from primary fits. The native endpoint combines identical actual-frame and exact-injection answers. Bars are marginal 95% scene-bootstrap accuracy intervals on 500 CLEVRER analysis scenes. Shaded rows use real future observations.

These fixed-head comparisons distinguish native recognition, mapped actual evidence, and the further accuracy loss from prediction; they do not bound the separately trained primary head.

5.5 Which events benefit across receivers?

We test two smaller receivers on unchanged CLEVRER analysis questions. Event labels group scored questions without being revealed to the models.



(a) Qwen2.5-VL-7B (b) InternVL3.5-8B

Figure 5: **Matched forecasts against receiver-specific controls.** Panels show matched prediction minus observed video or another video's update on the same 170/162/168 collision/entrance/exit scenes. Bars are paired 95% scene-bootstrap intervals conditional on one head per receiver. Qwen2.5-VL-7B uses actual-input fitting; InternVL3.5-8B uses prediction-input fitting.

Both receivers show positive matched-versus-different-video effects on collisions (Figure 5). Gains over observed-video inference are clearer for Qwen than InternVL. Entrance accuracy can rise

without a clear matched-update benefit, while InternVL's exit mean is below observed video and its different-video control. Qwen's entrance and exit matched-content effects remain near zero.

Collision and exit objects are visible at the cutoff, so visibility alone cannot explain the pattern. Across the evaluated configurations, collision questions show the clearest shared use of video-matched updates. Appendix D.6 retains all event outcomes and both metrics.

6 Related Work

Predictive representations for downstream reasoning. Future-feature prediction supports anticipation from unlabeled video (Vondrick et al., 2016). SlotFormer and Causal-JEPA connect object-centric dynamics to supervised predictive QA (Wu et al., 2023; Nam et al., 2026). CLEVRER's neuro-symbolic model and ComPhy's Compositional Physics Learner also combine dynamic prediction with reasoning (Yi et al., 2020; Chen et al., 2022). Their original QA protocols differ from CLEVRER-Next, precluding direct accuracy comparisons. FoReST instead transfers predictive information into an independently pretrained, fully frozen VLM.

Connecting world-model outputs to VLMs. FOREWARN projects action-conditioned latent forecasts into a LoRA-adapted VLM using behavior-narration supervision (Wu et al., 2025). GeoWorld-VLM distills geometric world-model features into a trainable visual encoder and projector, freezing the language model and using spatial-answer supervision (Gu et al., 2026). PearlVLA uses plan-conditioned forecasts to iteratively refine latent action-plan tokens through residual updates, with action and reward objectives (Yang & Shan, 2026). Future-L1 interleaves visual latents and text during decoding, with future-feature supervision and reinforcement learning (Jiang et al., 2026). FoReST adds a question-independent update to native visual continuation tokens before language processing. The entire VLM stays frozen; only the head learns from paired video features.

External features and native perception. Flamingo and BLIP-2 establish learned interfaces between frozen visual and language components (Alayrac et al., 2022; Li et al., 2023). BRAVE combines multiple frozen vision encoders, although its VQA stage updates the connector and language model (Kar et al., 2024). FoReST preserves an existing receiver's visual pathway and trains a future-residual interface. Its VLM-Forecast comparison fixes the target and output interface to test external forecasting routes. Appendix E provides further context.

7 Limitations

Evaluation covers only CLEVRER-Next and ComPhy, with relatively simple synthetic scenes. Performance on complex interactions and natural videos remains untested.

Both FoReST routes use the same V-JEPA 2 world model. Their generality remains untested across world models with different architectures, predictive representations, or training objectives. The receiver comparisons involve different preprocessing and head-fitting configurations, so they characterize those systems rather than isolate the effect of model size.

8 Conclusion

FoReST connects a video world model to a frozen VLM through predictive updates to native visual features. Latent transfer and direct forecasting improve future-event QA. Event comparisons distinguish gains over native forecasting from benefits of matched updates. Native addition and prediction-input fitting help receivers use imperfect forecasts. The shared principle is to retain native perception and learn the update supplied by a forecast.

AI USE STATEMENT

In this work, we used generative AI tools to help develop and refine the conceptual framework and mathematical formulation of FoReST, propose and refine research hypotheses, design and provide feedback on the methodology and experiments, implement the methods and evaluation pipelines, prepare and reformat data, and analyze and interpret experimental results. This assistance included implementing the annotation-based construction of CLEVRER-Next questions from existing videos and simulator annotations. We have not used these research-assistance tools to synthesize the underlying video data or to substitute AI-generated answers for ground-truth labels. Developing and writing formal mathematical proofs are not applicable to this empirical study. Additionally, we used generative AI tools for identifying and summarizing relevant literature, preparing and revising scientific figures and tables. We have reviewed all AI-assisted work. Specifically, we checked references against the original sources, tested AI-assisted code, and verified reported results against experimental outputs. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikołaj Bińkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. Flamingo: a visual language model for few-shot learning. In *Advances in Neural Information Processing Systems*, volume 35, pp. 23716–23736. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/960a172bc7fbf0177ccccbb411a7d800-Abstract-Conference.html.

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. URL https://openaccess.thecvf.com/content/CVPR2023/html/Assran_Self-Supervised_Learning_From_Images_With_a_Joint-Embedding_Predictive_Architecture_CVPR_2023_paper.html.

Mahmoud 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. *arXiv preprint arXiv:2506.09985*, 2025. URL https://arxiv.org/abs/2506.09985.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *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, Mido Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=QaCCuDfBk2.

Delong Chen, Mustafa Shukor, Theo Moutakanni, Willy Chung, Jade Yu, Tejaswi Kasarla, Yejin Bang, Allen Bolourchi, Yann LeCun, and Pascale Fung. VL-JEPA: Joint embedding predictive architecture for vision-language, 2025. URL https://arxiv.org/abs/2512.10942.

Zhenfang Chen, Kexin Yi, Yunzhu Li, Mingyu Ding, Antonio Torralba, Joshua B. Tenenbaum, and Chuang Gan. ComPhy: Compositional physical reasoning of objects and events from videos. In

International Conference on Learning Representations, 2022. URL https://arxiv.org/abs/2205.01089.

Renjie Gu, Kaichen Zhou, Yan Luo, and Mengyu Wang. GeoWorld-VLM: Geometry from world models for vision-language models, 2026. URL https://arxiv.org/abs/2605.16713.

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.

Oğuzhan Fatih Kar, Alessio Tonioni, Petra Poklukar, Achin Kulshrestha, Amir Zamir, and Federico Tombari. BRAVE: Broadening the visual encoding of vision-language models. In *European Conference on Computer Vision*, 2024. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/html/2433_ECCV_2024_paper.php.

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 *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 19730–19742. PMLR, 2023. URL https://proceedings.mlr.press/v202/li23q.html.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/6dcf277ea32ce3288914faf369fe6de0-Abstract-Conference.html.

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

Heejeong Nam, Quentin Le Lidec, Lucas Maes, Yann LeCun, and Randall Balestriero. Causal-JEPA: Learning world models through object-level latent masking, 2026. URL https://arxiv.org/abs/2602.11389.

Cheng Qian, Emre Can Acikgoz, Bingxuan Li, Xiusi Chen, Yuji Zhang, Bingxiang He, Qinyu Luo, Dilek Hakkani-Tür, Gokhan Tur, Yunzhu Li, and Heng Ji. Current agents fail to leverage world model as tool for foresight, 2026. URL https://arxiv.org/abs/2601.03905.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. VideoMAE: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *Advances in Neural Information Processing Systems*, volume 35, pp. 10041–10054. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/416f9cb3276121c42eebb86352a4354a-Abstract-Conference.html.

Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Anticipating visual representations from unlabeled video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 98–106, 2016. URL https://www.cv-foundation.org/openaccess/content_cvpr_2016/html/Vondrick_Anticipating_Visual_Representations_CVPR_2016_paper.html.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025. URL https://arxiv.org/abs/2508.18265.

Yilin Wu, Thomas Tian, Gokul Swamy, and Andrea Bajcsy. From foresight to forethought: VLM-in-the-loop policy steering via latent alignment. In *Proceedings of Robotics: Science and Systems*, 2025. doi: 10.15607/RSS.2025.XXI.076. URL https://www.roboticsproceedings.org/rss21/p076.html.

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://arxiv.org/abs/2210.05861.

Bochen Yang and Lianlei Shan. PearlVLA: Progressive embodied action-plan refinement in latent space, 2026. URL https://arxiv.org/abs/2606.17924.

Chen Yang, Shenxiang Zeng, Haoyang Zhao, Zhouyuan Xu, Youquan He, Haoyu Li, Mingyi Deng, Jiansheng Fan, and Chen Wang. PhysMind: From video to executable worlds for training-free physical reasoning, 2026. URL https://arxiv.org/abs/2608.04575.

Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B. Tenenbaum. CLEVRER: CoLlision Events for Video REpresentation and Reasoning. In *International Conference on Learning Representations*, 2020. URL https://arxiv.org/abs/1910.01442.

Haichao Zhang, Yijiang Li, Shwai He, Tushar Nagarajan, Mingfei Chen, Jianglin Lu, Ang Li, and Yun Fu. ThinkJEPA: Empowering latent world models with large vision-language reasoning model, 2026. URL https://arxiv.org/abs/2603.22281.

A ARCHITECTURES, LOSSES, AND FITTING

A.1 NATIVE BOUNDARIES AND VIDEO GEOMETRY

Native base and residual targets. The native base in Section 2.1 is built entirely from observed frames. For twelve observed samples $C = (c_1, \ldots, c_{12})$, we form $C_R = (c_1, \ldots, c_{12}, c_{12}, c_{12}, c_{12}, c_{12})$. This reference holds the last sampled observation fixed for four continuation positions; the controls call it a *static continuation* or persistence. The frozen VLM's visual pathway encodes C_R as Q_R, and F indexes the native tokens for the four appended frames. The frozen world-model encoder processes the same reference video; its final two temporal slots define R. Its predicted states P refer to those same slots. Actual training continuations replace the four repeated frames with four later observations to produce Q_A, aligned with Q_R at F. Both FoReST variants learn the target $Q_{A,F} - Q_{R,F}$ and add their prediction to $Q_{R,F}$. Thus the reference is part of the residual construction for both variants; the static-continuation baseline uses the same input with no update.

V-JEPA uses a 256×256 canvas, patch size 16, and temporal tubelets of size 2. Twelve observed frames produce six 16×16 grids (1536×1024 tokens). The released predictor jointly produces slots 6 and 7 (512×1024 states), without recursive feedback. Actual and persistence encodings contain eight slots; their last two define future targets. The context encoder sees only observations.

For Qwen injection, reshape a future tensor as $[B, 2, 8, 2, 8, 2, 1024]$, average the two axes of size 2 in FP32, and flatten to $[B, 128, 1024]$. Affine-free LayerNorm with $\epsilon = 10^{-5}$ precedes a bias-free $1024 \to 2048 \to d_Q$ MLP with GELU. For the primary receiver $d_Q = 5120$, giving $1024 \cdot 2048 + 2048 \cdot 5120 = 12{,}582{,}912$ learned parameters. Pooling, normalization, and the MLP all send zero to zero, so the pointwise residual mapper satisfies $D(0) = 0$ for every fitted checkpoint.

A Qwen3.5-27B target video contributes 512 native post-merge tokens: 384 observed tokens and 128 continuation tokens. Each eight-frame ComPhy reference contributes 256; the four references plus target contain 1,536 visual tokens. Processor visual masks and grid metadata locate each video block, independently of prompt offsets. Injection preserves observed/reference tokens, text, boundaries, positions, and attention. The 12-frame observed baseline has 384 tokens without padding; observed16 has 512 real observed tokens.

Qwen2.5-VL-7B-Instruct uses a 224×224 canvas, visual patch size 14, spatial merge factor 2, temporal tubelet size 2, and output dimension 3,584. Its resulting 8×8 post-merge grid gives the same 512/128 target/future token counts. Its independently fitted mapper has $1024 \cdot 2048 + 2048 \cdot 3584 = 9{,}437{,}184$ parameters.

A.2 LOSS REDUCTIONS

For $x, y \in \mathbb{R}^{B \times N \times d}$, let i index the BN tokens. The cosine term is

$$L_c(x, y) = \frac{1}{BN} \sum_i \left(1 - \frac{\langle x_i, y_i \rangle}{\max(\|x_i\|_2, 10^{-8}) \max(\|y_i\|_2, 10^{-8})} \right). \tag{6}$$

Let $\mathcal{N}$ center each token and divide by $\sqrt{\mathrm{Var}(x_i) + 10^{-5}}$, using variance denominator d and no learned affine transform. The normalized Huber term averages over tokens *and* coordinates:

$$L_h(x, y) = \frac{1}{BNd} \sum_{i,j} h(\mathcal{N}(x_i)_j - \mathcal{N}(y_i)_j), \quad h(a) = \begin{cases} \frac{1}{2}a^2, & |a| < 1, \\ |a| - \frac{1}{2}, & |a| \geq 1. \end{cases} \tag{7}$$

The token-norm term is $L_n(x, y) = (BN)^{-1} \sum_i |\|x_i\|_2 - \|y_i\|_2|$. The common objective is $\ell_Q = L_c + 0.10 L_h + 0.01 L_n$, with FP32 reductions. The original pointwise implementation also computes $0.01 L_0$, where $L_0 = N^{-1} \sum_n \|D(0)_n\|_2^2$; it is structurally zero. Forward activations use BF16 with FP32 learned parameters and optimizer state.

A.3 FORECST-DIRECT AND CONTROLLED INJECTION INTERFACES

FoReST-Direct is the framework's learned forecasting variant: it maps observed features from the same frozen video encoder directly to the native residual, without calling the released predictor. It

pools all six observed grids to 384 tokens, projects them from 1,024 to 512 dimensions, and attends to this memory with 128 factorized temporal/row/column future queries. Three pre-normalized cross-attention blocks use width 512, eight heads, and $512 \to 2048 \to 512$ feed-forward layers. A final projection produces 128 receiver-width residuals. The head has 12,620,800 parameters. Its target is ΔQ, and it receives no predicted or actual-future latent input.

Controlled addition and replacement. The addition/replacement comparison keeps each interface's meaning fixed. DELTA-add maps pooled $T - R$ to ΔQ during fitting, and pooled $P - R$ at inference. ABS-add maps pooled T to ΔQ and deploys on pooled P. ABS-replace maps pooled T to absolute native future tokens $Q_{A,F}$ and replaces the future block with its output. ABS-add is bias-free; replacement includes both linear biases, giving 12,590,080 parameters. All retain the native observed block. The interpolation comparison uses the original actual-input-selected seed-13 fits for these interfaces, separately from deployment-selected T/P/M and the earlier actual-input FoReST-Latent reference configuration.

Observed augmentation. Observed augmentation maps the final two observed V-JEPA blocks to native features encoded from a standalone clip of the last four observed samples. Its biased MLP has 12,590,080 parameters. Inference appends a separately labeled observed-video segment containing 128 positions; the main persistence video remains intact. Its native companion uses the same extra segment and receiver-native features directly. Both have 640 visual tokens on CLEVRER and 1,664 on ComPhy, rather than the 512/1,536 of future-block addition. The extra segment is labeled "Additional observed clip (same scene, already observed):" for CLEVRER and "Additional observed target clip (already observed):" for ComPhy.

A.4 VLM-FORECAST: FORECASTING FROM NATIVE OBSERVED TOKENS

VLM-Forecast uses the Qwen3.5-27B post-merge visual boundary used by both FoReST variants. We encode the static-continuation clip: 12 observed samples followed by four copies of the last sampled observation. Its 512 target-video tokens are ordered by the processor's prepared time/row/column metadata. The first six 8×8 blocks form $O_Q \in \mathbb{R}^{384\times5120}$; the final two blocks supply $Q_{R,F}$. We never extract the observed slice from an actual-future clip, whose encoding could depend on future frames. The head receives detached target-video tokens only. ComPhy reference videos remain in the receiver's ordinary input but are absent from this forecasting memory.

Fixed architecture. An affine-free LayerNorm with $\epsilon = 10^{-5}$ and a bias-free $5120 \to 512$ projection produce the memory. Fixed sinusoidal time/row/column encodings occupy 170/170/172 dimensions, respectively. Observed time indices are 0–5. Future queries sum learned temporal (2×512), row (8×512), and column (8×512) embeddings to form 128 query vectors, adding the fixed position code for temporal indices 6–7. Three width-512 cross-attention blocks have eight heads and no query self-attention. For query X and memory M, each block computes

$$A = \mathrm{Attention}(W_q\,\mathrm{LN}(X), W_k\,\mathrm{LN}(M), W_v\,\mathrm{LN}(M))\,, \tag{8}$$

$$X' = X + W_o A, \qquad X_{\mathrm{next}} = X' + W_2\,\mathrm{GELU}(W_1\,\mathrm{LN}(X'))\,. \tag{9}$$

The four attention projections are $512 \to 512$, and the feed-forward pair is $512 \to 1360 \to 512$. After three blocks, affine-free LayerNorm and a bias-free $512 \to 5120$ projection produce $G_Q(O_Q) \in \mathbb{R}^{128\times5120}$. Every normalization is affine-free, every linear map is bias-free, and dropout is zero. Query embeddings use normal initialization with standard deviation 0.02; projections use Xavier initialization. The output projection is not initialized to zero.

The input and output projections each contain 2,621,440 parameters, attention projections contain 3,145,728, feed-forward layers contain 4,177,920, and factorized queries contain 9,216. The total is **12,575,744**, versus FoReST-Latent's 12,582,912: 7,168 fewer, or 0.057%. All counted parameters participate in prediction. The feed-forward width is smaller than FoReST-Direct's because the 5,120-dimensional native input consumes more of the matched learned budget.

Target, fitting, and deployment. The target remains $\Delta Q = Q_{A,F} - Q_{R,F}$ and the loss is the same three-term ℓ_Q defined above. Learned queries can yield nonzero outputs even with zero visual memory; FoReST-Latent's structural $D(0) = 0$ property and its identically zero penalty do

not apply. Three independent heads (seeds 13, 17, and 42) are fitted per dataset on the existing 1,935/321 CLEVRER windows and 600/120 ComPhy groups. AdamW, BF16 activations, FP32 learned parameters and reductions, effective batch 16, validation milestones, extension criterion, and alignment-only checkpoint selection follow Appendix A.5. Input features are stored once per window and streamed; fitting does not require a resident language transformer.

At inference only the final 128 target-video token values change:

$$\widetilde{Q}_F = Q_{R,F} + G_Q(O_Q).$$

The coefficient is one. The head makes no V-JEPA call, and it receives no question, candidate, reference video, or actual future. Observed tokens and the receiver's position and attention metadata remain unchanged. One residual per source is reused across its two questions or candidates. Evaluation uses the main 1,000 CLEVRER scenes and 297 ComPhy groups, with the original prompts and independent answer scoring. Paired comparisons on the same scenes or groups use VLM (observed video), VLM + static continuation, and the corresponding training seed of FoReST-Latent and FoReST-Direct. Mean and sample SD summarize the three fits; each scene-bootstrap interval conditions on one selected fit. No shuffled VLM-Forecast condition was evaluated.

Selected fits and training behavior. All three CLEVRER fits meet the validation-improvement criterion for extension and finish at update 4,000. Seeds 13/17/42 select updates 4,000/4,000/2,500, with minimum validation losses 0.813275/0.797847/0.778582 and selected 64-window training-monitor losses 0.552871/0.533837/0.625875. No ComPhy fit meets the extension threshold: all finish at update 2,000 and select 2,000/2,000/1,500, with validation losses 0.977463/0.970273/0.973305 and training-monitor losses 0.719438/0.723028/0.777223. No fit stops early. Seeds 13 and 17 remain improving at their maximum budgets on both tasks; seed 42 selects an earlier validation minimum before subsequent deterioration. Each selected checkpoint was fixed before its QA evaluation.

A.5 Common optimization and selection

The FoReST-Latent fitting ablation distinguishes actual-input (T), predicted-input (P), and mixed-input (M) training. Actual-input fitting supplies actual future video states T, while primary predicted-input fitting supplies the original predictor's P. Mixed-input fitting averages their losses:

$$\mathcal{L}_T = \ell_Q(D(T-R), \Delta Q), \qquad \mathcal{L}_P = \ell_Q(D(P-R), \Delta Q),$$
$$\mathcal{L}_M = \tfrac{1}{2}\mathcal{L}_T + \tfrac{1}{2}\mathcal{L}_P. \tag{10}$$

All three deploy on $P - R$ and share initialization, data, optimizer, and checkpoint criterion within each seed.

The controlled T/P/M and FoReST-Direct fits use AdamW (Loshchilov & Hutter, 2019) with constant learning rate 10^{-4}, weight decay 10^{-4}, betas $(0.9, 0.999)$, epsilon 10^{-8}, effective batch size 16, and gradient clipping at 1.0. Full alignment-validation evaluation occurs at updates 0, 250, 500, 1,000, 1,500, and 2,000. A validation-loss decrease of at least 1% from 1,500 to 2,000 permits one extension to 4,000, checked every 500 updates. Three consecutive validation increases with decreasing comparable training loss permit stopping and retaining the earlier minimum. Selection includes initialization and never uses QA accuracy.

T/P/M select on deployment-input loss with $P - R$; FoReST-Direct selects on observed-context loss. Six actual-input replays recover intermediate T checkpoints, separately from original actual-input-selected fits. The 24 controlled trajectories comprise 18 T/P/M fits and six FoReST-Direct fits, including two retained seed-13 FoReST-Direct fits. All 22 new fits finish at update 2,000 without extension: ten remain improving, twelve select earlier minima after divergence.

The FoReST-Latent recipe for the main evaluation is selected by the mean relative deployment-validation loss across datasets d and seeds s:

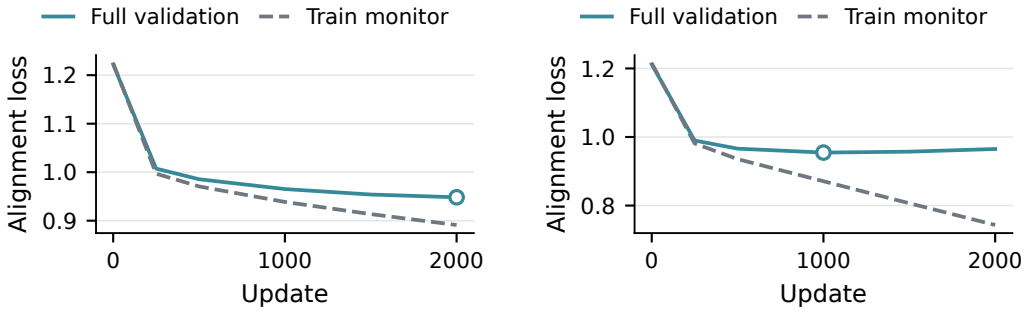
$$S(r) = \frac{1}{6}\sum_{d,s} \frac{J_{d,s}(r)}{\max(J_{d,s}(T), 10^{-8})}, \quad r \in \{T, P, M\}, \tag{11}$$

where each J uses that recipe's selected checkpoint. Relative ties within 0.5% prefer T, then P, then M. The measured values are $S(T) = 1$, $S(P) = 0.918026$, and $S(M) = 0.923973$; P is selected. This choice preceded main-evaluation scoring.

Table 3: Actual-input alignment for Qwen2.5-VL-7B. Each earlier scratch fit uses seed 13 and selects the minimum full alignment-validation objective, including initialization. Selected and final losses describe alignment training, not QA performance.

Dataset	Parameters	Selected update	Final update	Best loss	Final loss
CLEVRER-Next	9437184	2000	2000	0.815630	0.815630
ComPhy	9437184	1000	2000	0.891797	0.904053

Train/validation windows: CLEVRER-Next 1935/321; ComPhy 600/120. Update extension (CLEVRER/ComPhy): No/No; early stopping: No/No.



(a) CLEVRER (b) ComPhy

Figure 6: Qwen3.5-27B FoReST-Latent, seed 13: full alignment-validation and fixed 64-window training-monitor losses use identical deployment inputs and reductions. The monitor differs from stochastic minibatch loss. Hollow markers identify selected checkpoints. Improving at the budget does not establish convergence.

A.6 Reference configurations and measured training behavior

The seed-13 CLEVRER reference starts from scratch: 1,200 updates, learning rate 3×10^{-4}, batch 16, weight decay 10^{-4}. Validation at 400/800/1,200 selects 800, with objective 0.920385 versus final 0.921831. ComPhy initializes from that mapper and trains 600 updates at 10^{-4}; validation at 0/300/600 selects 300, with objective 0.930970 versus final 0.932333. Their separate QA results appear in Appendix C.

Both actual-input FoReST-Latent mappers for Qwen2.5-VL-7B use seed-13 scratch initialization, actual future inputs, the common 10^{-4} recipe, and actual-input selection (Table 3).

CLEVRER P/M and new FoReST-Direct fits select update 2,000; T selects 2,000/2,000/1,500 for seeds 13/17/42. ComPhy selects T at 500, P at 1,000, M at 1,500/1,500/1,000, and new FoReST-Direct at 1,000/1,500. Figure 6 shows representative trends.

A.7 Context-conditioned comparison

CTX-P and CTX-O are controlled contextual heads, separate from the two FoReST variants. Both instantiate FoReST-Direct's width-512, eight-head, three-block backbone from scratch with the same seed. A zero-initialized bias-free projection $W_{\text{aux}} : 1024 \to 512$ changes the queries to

$$q = q_{\text{Direct}} + W_{\text{aux}} \, \text{LN}(X), \tag{12}$$

with affine-free normalization. CTX-P uses $X = \text{pool}(P - R)$; CTX-O uses the 128 pooled tokens of the last two observed blocks. Both retain 384 observed memory tokens and FoReST-Direct's output projection. Each has 13,145,088 active parameters: exactly matched to each other and near-matched to FoReST-Direct.

Both heads use seeds 13/17 on each dataset, objective $\ell_Q(H(O, X), \Delta Q)$, and the common optimizer, validation, extension, and selection rules. CTX-P fits predicted auxiliary inputs; actual future supplies targets only. QA uses the analysis populations.

Auxiliary shuffling transfers the donor's $P_d - R_d$ while fixing recipient memory, native base, references, and text; auxiliary zero supplies zeros. CTX-O analogously replaces only auxiliary observed tokens. Both responses share a donor. These fixed-context interventions differ from the two FoReST variants' whole-update shuffling. Paired comparisons test CTX-P against pointwise FoReST-Latent, FoReST-Direct, CTX-O, its auxiliary shuffle, and zero; CTX-O is also compared with FoReST-Direct. Baselines match seeds 13/17; primary tables retain three seeds. The controls separate added context/capacity from predictive-query computation, which adds no new observation.

All eight fits finish at update 2,000 without extension or early stopping. CLEVRER selects 2,000 for both heads/seeds while validation improves. ComPhy selects 1,500 for CTX-P and 1,000 for CTX-O at both seeds; subsequent validation worsens while the fixed 64-window training monitor improves. Complete-head and auxiliary-projection gradient norms remain finite and nonzero in every logged interval.

A.8 INTERNVL3.5-8B NATIVE INTERFACE

Receivers and visual geometry. The additional receiver is `OpenGVLab/InternVL3_5-8B` (Wang et al., 2025), with approximately 8.5B total parameters and language width 4,096. All receiver components remain frozen in BF16, including its InternViT vision encoder and released visual projector. Each source frame retains its full field of view with aspect-preserving square padding, a single 448×448 tile, and the released ImageNet normalization (means 0.485/0.456/0.406; standard deviations 0.229/0.224/0.225). There are no extra thumbnails or dynamic crops. The released pixel shuffle with `ps_version=v2` and `mlp1` produce 256 language-width tokens per frame. A 16-frame target contributes 4,096 tokens; its final four frames contribute 1,024. ComPhy's four eight-frame references increase the total to 12,288.

Residual translation and temporal lift. The mapper receives the original predictor's unpooled residual $P - R$, reshaped as $[2, 16, 16, 1024]$. The shared pointwise network uses affine-free LayerNorm, bias-free $1024 \to 2048 \to d_Q$ layers, and GELU. It has 10,485,760 trainable parameters. Let U_0, U_1 be the two mapped spatial grids. The four frame updates are

$$U_0, \qquad \tfrac{3}{4}U_0 + \tfrac{1}{4}U_1, \qquad \tfrac{1}{4}U_0 + \tfrac{3}{4}U_1, \qquad U_1.$$

This fixed center-aligned temporal interpolation preserves the receiver's spatial order and uses no spatial pooling. Both the mapper and lift send zero to zero; the update is added with coefficient one. Targets are the receiver's native actual-future features minus its repeated-last-observed-frame features, extracted through the released `extract_feature` boundary. Losses average over the resulting 1,024 future tokens. The protocol specifies one dataset-specific scratch FoReST-Latent fit at seed 13, using the complete existing alignment populations, common optimizer, validation milestones, and extension rule. No QA labels train or select a mapper.

Recipients, controls, and scoring. CLEVRER uses all 500 analysis scenes: 170 collision, 162 entrance, and 168 exit scenes. The initial 300-scene evaluation was completed with the remaining 200 scenes using the same selected mapper, receiver, prompt, and scoring protocol; its original scores are retained. ComPhy retains all 200 analysis groups. Both original forms or candidates remain, as do all four ComPhy references. The unchanged parent-population seed-13 donor map supplies shuffled residuals; all CLEVRER recipients and donors now belong to the complete 500-scene analysis population.

The controls are VLM (observed video), VLM + static continuation, matched FoReST-Latent, and shuffled FoReST-Latent; CLEVRER additionally has actual future pixels. Native features occupy the actual expanded image placeholders in the official conversation template. Only the final four target-frame blocks receive residuals. Frame labels contain the task timestamps, and each reference has its own timeline. Neutral CLEVRER and independent-candidate ComPhy questions use the receiver's default non-thinking system prompt. Each tokenizer defines the answer continuations; single-token alternatives share one forward, while a multi-token alternative uses its full summed continuation log likelihood without length normalization. No answer sampling is used. All conditions use the same label and scoring rules. Paired 5,000-resample scene/group intervals compare matched FoReST-Latent with each native control and its own shuffle, and actual pixels with static continuation on CLEVRER. A single fit does not measure training-seed robustness.

Fitting records. Both InternVL3.5-8B mappers finish at 2,000 updates with finite, nonzero gradients and no extension or early stopping. CLEVRER selects update 2,000 with validation loss 1.005868; ComPhy selects 1,500 with loss 0.982489 before a small subsequent increase. Losses average over all 321 validation windows or 120 groups, respectively. The CLEVRER fit remains improving at the update budget.

B TASKS, POPULATIONS, AND SCORING

B.1 CLEVRER-NEXT QUESTION CONSTRUCTION

Object descriptions use color, material, and shape from the CLEVRER simulator annotations; scenes with ambiguous descriptions are excluded. Collisions use annotated object pairs. Entrances and exits are changes in camera visibility at 25 frames per second. Duplicate event annotations are removed. The target is the unique next eligible event after cutoff $c = t_{\text{event}} - 0.5$, with $2.0 \leq c \leq 3.5$ seconds. No event intervenes between cutoff and target, and the nearest competing event is more than 0.20 seconds away. Collision and exit objects must be visible at cutoff; entrance objects must be absent at cutoff and identifiable at the event.

The future interval ends at $\min(c + 2, 5)$ seconds. Four distractors use event signatures compatible with the cutoff visibility: collisions among visible pairs, exits of visible objects, and entrances of absent objects. A distractor cannot occur before or at the target. The deterministic mixture includes a same-type alternative, a different-type alternative, and one absent from the entire future interval; a fixed ranking fills remaining choices. Only candidate text conveys object and event identities to the receiver. Trajectories, simulator state, and answer labels are not receiver inputs.

Each scene contributes two fixed forms sharing five candidate meanings but changing stem and option order. The stems are "Which event happens next after the observed clip?" and "What is the next event involving the objects in the scene?" Form accuracy averages the two indicators; both-form correctness is their conjunction. The historical 100-scene population balances correct letters within each form; its results remain separate from the later populations.

Frame sampling and alignment. Samples are cell centers $t_i = a + (i + \frac{1}{2})(b - a)/n$. CLEVRER uses 12 observed samples on $[0, c]$ and four continuation samples on $(c, \min(c + 2, 5)]$. VLM + static continuation repeats the last *sampled* observed frame at the latter positions; this technical reference is also called persistence. VLM (observed video) samples 16 times in $[0, c]$. Alignment cutoffs use the fixed grid $\{0.5, 1.0, \ldots, 3.5\}$ seconds, retaining a unique first event in the continuation interval and at least 0.5 seconds between chosen cutoffs per source. This yields 1,935 training windows from 302 scenes and 321 validation windows from 50 other scenes. Event metadata chooses windows; no answer text enters fitting.

B.2 COMPHY CANDIDATE SEMANTICS AND SAMPLING

We retain official `predictive_multiple_choice` records, including question IDs, candidate IDs, text, negation, and labels. A candidate labeled `correct` maps to True and `wrong` to False. Each evaluated group has two candidates, one of each label, but inference imposes no one-True constraint. The judgments are independent. All four reference experiments and the target remain in their original group.

References are labeled "Reference [ID] (separate experiment):" and the target "Target video:". Each reference has eight cell-center samples on its own $[0, 2]$-second timeline. QA target observations cover $[0, 5]$; persistence appends four copies of its final sampled observed frame. Alignment uses a $[0, 3]$ prefix and $(3, 5]$ suffix, one window per group. Decoding at 25 fps uses frame index $\lfloor 25t \rfloor$ with aspect-preserving square padding. The forecasting branches see only the target; reference experiments are receiver inputs. There is no actual-future RGB oracle for the later QA interval in the acquired release.

B.3 MAIN EVALUATION AND ANALYSIS SUBSETS

The CLEVRER-Next main evaluation contains 1,000 scenes and 2,000 question forms: 600 collision, 162 entrance, and 238 exit scenes. Its development (analysis) subset contains 500 scenes and 1,000 forms: 170 collision, 162 entrance, and 168 exit scenes.

The ComPhy main evaluation contains 297 groups and 594 candidate judgments. Its development (analysis) subset contains 200 groups and 400 candidate judgments.

B.4 EXACT PROMPTS AND ANSWER SCORING

The CLEVRER system message is "You are a careful video multiple-choice question answering assistant." The condition-neutral user prompt is:

> The observation cutoff for this question is {`cutoff:.2f`} seconds.
> In the question, the observed clip means the part ending at this cutoff.
> Using the provided visual input, choose the first event that happens after this cutoff from the answer options.
> Choose the single best answer option for the multiple-choice question.
> Return exactly one line in this format: Your Answer Letter: X
> X must be one of A, B, C, D, or E. Do not explain.
> `[original question and choices]`
> Final answer:

The scorer appends "Your Answer Letter:" and compares the valid continuation likelihoods in one forward when each continuation is a single token. The primary receiver's A–E token IDs are 357, 417, 351, 414, and 458. Receiver-specific tokenizers determine other receivers' continuations. Inference is deterministic, with frozen weights in evaluation mode and no answer sampling.

The earlier CLEVRER prompt described every input as containing future-position frames. Its historical results retain their prompt identity and are not pooled with neutral-prompt comparisons.

The ComPhy system message is "You are a careful video reasoning assistant." After the four references and target, each candidate is scored with:

> The reference videos show separate experiments with the same objects. They are not a continuation of the target video. Answer the predictive question about the target video. Judge the candidate independently, including any negation in the question. True means this candidate correctly answers the given question; False means it does not. Respond with exactly True or False.
> Question: [original question]
> Candidate: [original candidate]

The primary receiver compares full-vocabulary likelihoods of the single tokens True and False (IDs 2434 and 3913). Thinking is disabled in its chat template. Removing references removes both their video segments and descriptive sentence; the candidate semantics are unchanged. No hidden physical properties, programs, trajectories, or answer labels enter either task's input.

B.5 DONORS, INTERVENTIONS, AND UNCERTAINTY

Whole-update shuffling transfers $D(P_d - R_d)$ or $G(O_d)$ from a different source, preserving the recipient's native context, references, token layout, and question. A donor's prediction is always paired with that donor's own persistence. Both forms/candidates share a donor. CLEVRER donor cycles are formed within event family and future-duration bin by shuffling sorted question IDs using the map seed plus 400. ComPhy shuffles a permutation until it has no fixed points. The primary donor seed is 13; additional historical maps use 17 and 42. Donor-map variation and training-seed variation are separate quantities. The main evaluation preserves the original fixed donor assignments within each dataset's two original acquisition blocks. Pooling their scores leaves every recipient–donor pair unchanged; it does not draw a new permutation across blocks. The donor intervention tests dependence on the matched auxiliary update. It also changes scene compatibility and therefore does not isolate physical prediction accuracy alone.

Temporal reversal changes the six observed tubelets to order $(5, 4, 3, 2, 1, 0)$, retaining frame order within each tubelet, then re-encodes the reordered RGB. The recipient's native stream and video-model static-reference representation retain their original order. FoReST-Direct keeps its future queries. The fixed training-mean control first averages predicted updates across windows within each training scene, then across scenes, preserving all output positions. ComPhy has one window per group. This mean uses neither future targets nor QA labels.

Historical appearance controls use nearest object-attribute matches on CLEVRER (256 donors for 500 recipients; no exact matches) and last-observed-frame HSV histograms on ComPhy (133 donors for 200 recipients). These controls differ from random whole-update shuffling.

For a paired accuracy difference, resample source scenes or groups 5,000 times with seed 13, keeping all forms/candidates attached and taking percentile 95% intervals. This preserves the dependence between conditions. Correction counts require a baseline-incorrect group to become fully correct; harms count previously fully correct groups becoming incorrect. Training-seed means use sample SD across fits, rather than interpreting seeds as additional independent test examples. Event-specific comparisons retain all three families and their own denominators.

Event effects averaged over fixed fits. The event comparisons in Figures 9 and 2 use the complete CLEVRER population, grouped by the annotated next event without changing any scored question. For scene i, method m, and retained fit s, let $a_{m,s,i} = (c_{m,s,i,1} + c_{m,s,i,2})/2$ be answer accuracy and $j_{m,s,i} = c_{m,s,i,1}c_{m,s,i,2}$ be both-form correctness. For a learned comparator b, the paired scene effect is

$$d_i = \frac{1}{3} \sum_{s \in \{13,17,42\}} (a_{m,s,i} - a_{b,s,i}).$$

A fixed native baseline supplies the same scene score in each summand; each own-shuffle comparison uses the shuffled output of that same fitted head. The reported paired difference is $100\,\overline{d}$. We resample complete scenes 5,000 times with seed 13, preserving the original acquisition-block counts within each event, and report the 2.5th and 97.5th percentiles. Both-form effects substitute j for a. These intervals quantify source uncertainty conditional on the three retained fits; they are neither uncertainty over retraining nor accuracy of an ensemble. The fits are never treated as additional scenes. Table means and sample SD retain their separate meaning. The previously reported per-fit whole-population intervals in Appendix C.3 use their original unstratified source resamples and remain unchanged.

B.6 Small-receiver event comparisons

Both smaller receivers use the same 500 CLEVRER analysis scenes: 170 collision, 162 entrance, and 168 exit scenes. Qwen2.5-VL-7B uses a scratch actual-input mapper; InternVL3.5-8B uses a scratch predicted-input mapper. Both retain the neutral five-choice questions and their original answer orders. Event labels classify already scored answers; they do not change the prompt, choices, checkpoint, or inference task. Each receiver is compared with its own VLM (observed video), VLM + static continuation, and shuffled-update endpoints on exactly the same sources. Full form and both-form outcomes are reported for collision, entrance, exit, and overall. Source-paired intervals condition on one selected head and donor map. The evaluation sources are shared and the fitting recipes differ, so this is an analysis of event dependence across two configurations.

C Primary outcomes, fitting comparisons, and event analysis

Table 4 reports each fitted seed on the main populations of 1,000 CLEVRER scenes and 297 ComPhy groups. Within each comparison, receiver, population, prompt, and donor map are fixed. Source-bootstrap intervals and training-seed SD describe different variation.

C.1 Main paired results

On CLEVRER, every FoReST-Latent and FoReST-Direct seed exceeds both native baselines and its own shuffle in both metrics, with positive paired intervals. FoReST-Direct's advantage over FoReST-Latent ranges from 5.05–6.45 answer points and 6.20–7.70 both-form points, again with

Table 4: Individual-seed matched accuracy (%) on the main evaluation: CLEVRER 1,000 scenes and ComPhy 297 groups. FoReST-Latent uses the validation-selected predicted-input recipe P. Both-form/all-options requires both responses within a source correct. ComPhy uses four references; CLEVRER uses the neutral cutoff prompt. Fitting-recipe and other diagnostic ablations use the analysis subsets.

Route	Seed	C answer	C both	M candidate	M all
VLM-Forecast	13	61.10	56.40	75.93	57.91
VLM-Forecast	17	61.35	57.30	76.60	59.93
VLM-Forecast	42	61.10	56.70	75.42	57.91
FoReST-Latent (P)	13	58.20	52.70	80.81	66.33
FoReST-Latent (P)	17	57.20	51.60	80.64	66.33
FoReST-Latent (P)	42	57.50	52.50	80.64	65.99
FoReST-Direct	13	63.25	58.90	78.79	63.64
FoReST-Direct	17	63.65	59.30	77.10	61.62
FoReST-Direct	42	62.60	58.80	76.09	59.60

Table 5: Earlier fixed reference mappers on the main evaluation. These single-seed fits are separate from the three-seed FoReST-Latent comparison in Table 1; their training definitions are given in the fitting protocol. All values are accuracy (%); no seed SD is assigned. Shuffled updates retain the original within-population donor maps.

	CLEVRER (1,000)		ComPhy (297)	
Method	Answer acc.	Both-form	Candidate acc.	All-options
FoReST-Latent reference	59.10	54.60	78.11	63.30
FoReST-Latent reference shuffled	52.90	48.40	73.91	57.24

positive intervals. The earlier actual-input reference fits in Table 5 retain their separate learning-rate and selection histories (Appendix A).

On ComPhy, both FoReST variants also exceed both native baselines and their own shuffles at every seed in both metrics. FoReST-Latent exceeds FoReST-Direct by 2.02–4.55 candidate and 2.69–6.40 all-options points, with positive paired intervals throughout. Thus the relative strength of the two routes changes across tasks even though both benefit from the video-matched update.

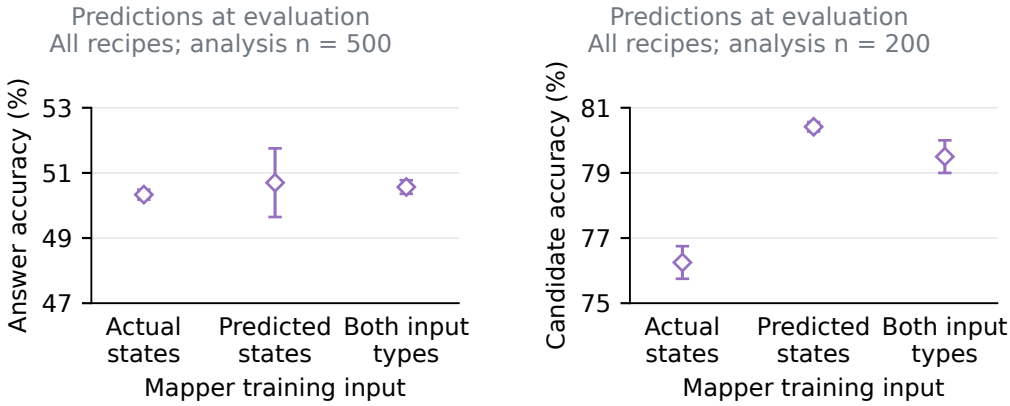
C.2 Fitting-input and contextual comparisons

Table 6 retains FoReST-Latent actual-input (T), predicted-input (P), and both-input (M) fitting on both analysis sets, followed by the separately matched two-seed contextual comparison. On CLEVRER, FoReST-Direct exceeds every pointwise recipe at every seed in both metrics. P/M do not consistently improve on T; P seed 17 decreases both-form correctness by 3.2 points $[-6.2,-0.2]$. On ComPhy, P exceeds T and FoReST-Direct in both metrics at every seed, and M also exceeds T. Changing only T's checkpoint-selection objective gives -0.50, $+0.75$, and -0.25 option points, with intervals including zero. The fitting-input benefit exceeds the measured effect of reselection alone.

Figure 7 shows the three recipes evaluated on predictions; Table 2 in Section 5.3 summarizes their primary metrics.

Table 6: Fitting inputs and contextual translation on the analysis sets: 500 CLEVRER scenes and 200 ComPhy groups. Accuracy (%); learned rows give mean ± sample SD. The upper panel uses three seeds; the lower panel matches seeds 13/17 for every learned route. FoReST-Latent T/P/M train on actual states, predicted states, or both input types and select by deployed-input validation loss. M averages the two alignment losses; it does not interpolate states. CTX-P adds predicted auxiliary input to observed-context attention; CTX-O instead adds observed queries.

	CLEVRER		ComPhy	
Method	Answer acc.	Both-form	Candidate acc.	All-options
Fitting-input comparison: seeds 13, 17, and 42				
FoReST-Latent (T)	50.33 ± 0.15	46.67 ± 0.31	76.25 ± 0.50	61.33 ± 1.04
FoReST-Latent (P)	50.70 ± 1.05	45.20 ± 1.22	80.42 ± 0.14	66.83 ± 0.29
FoReST-Latent (M)	50.57 ± 0.21	45.27 ± 0.12	79.50 ± 0.50	65.50 ± 1.00
FoReST-Direct	55.63 ± 1.08	51.87 ± 1.14	76.75 ± 0.75	61.50 ± 1.00
Contextual comparison: seeds 13 and 17 for every learned route				
VLM (observed video)	39.70	35.80	67.75	46.00
VLM + static continuation	45.00	41.40	69.50	48.50
FoReST-Latent (P)	50.75 ± 1.48	44.90 ± 1.56	80.38 ± 0.18	66.75 ± 0.35
FoReST-Direct	56.25 ± 0.21	52.50 ± 0.42	77.12 ± 0.53	62.00 ± 0.71
CTX-P	54.50 ± 0.14	50.20 ± 0.28	77.25 ± 0.35	62.50 ± 0.00
CTX-O	54.80 ± 1.56	50.70 ± 2.40	76.38 ± 1.59	60.50 ± 3.54



(a) CLEVRER-Next: answer accuracy

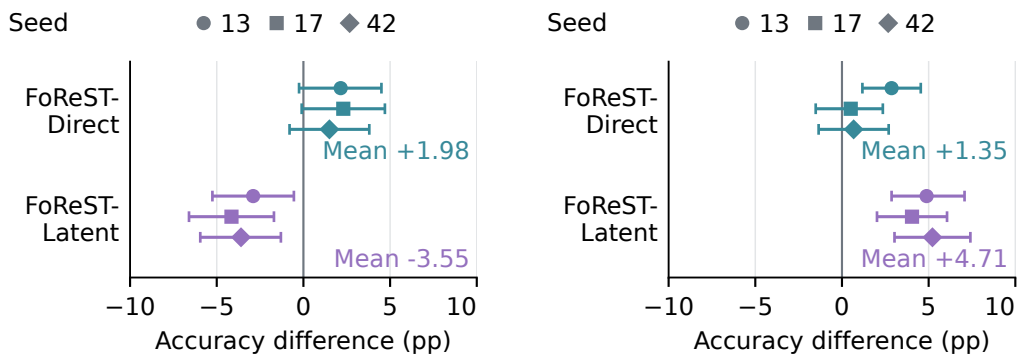
(b) ComPhy: candidate accuracy

Figure 7: **Different mapper training inputs; predictions at evaluation.** Actual-input, predicted-input, and both-input recipes share the architecture, paired videos, and prediction-based checkpoint criterion. Points and bars show mean ± sample standard deviation across three fits on the CLEVRER (500 scenes) and ComPhy (200 groups) analysis subsets. Both-input fitting averages two alignment losses, not latent states. Both panels use the same vertical span, so equal point separations represent equal percentage-point changes.

C.3 Forecasting from native and external representations

The three learned routes use seeds 13, 17, and 42 on the same main populations: 1,000 CLEVRER-Next scenes and 297 ComPhy groups. Table 7 reports every seed's absolute accuracy; Table 8 gives same-seed paired comparisons. Sample SD summarizes variation across fits; each source-bootstrap interval conditions on the two fitted models being compared. The tables also retain comparisons with both fixed native baselines.

VLM-Forecast reaches 61.18 ± 0.14 answer and 56.80 ± 0.46 both-form accuracy on CLEVRER-Next, and 75.98 ± 0.59 candidate and 58.59 ± 1.17 all-options accuracy on ComPhy. Its gains over both native baselines have positive paired intervals at every seed in both metrics.



(a) CLEVRER-Next (b) ComPhy

Figure 8: **Fit-specific native-versus-external comparisons.** Each point is FoReST minus VLM-Forecast on the complete main evaluation. Bars are paired 95% source-bootstrap intervals conditional on each pair of fits; annotated mean differences do not inherit these intervals.

FoReST-Direct's CLEVRER answer gains over VLM-Forecast are 2.15 [−0.25, 4.50], 2.30 [−0.10, 4.70], and 1.50 [−0.80, 3.80] points for seeds 13, 17, and 42. Both-form gains also have intervals crossing zero. On ComPhy, the corresponding candidate gains are 2.86 [1.18, 4.55], 0.51 [−1.52, 2.36], and 0.67 [−1.35, 2.69]; only seed 13 has positive intervals in both metrics. The mean Direct-minus-native differences are 1.98/2.20 points on CLEVRER and 1.35/3.03 on ComPhy, for the primary/joint metrics.

FoReST-Latent averages 3.55 answer and 4.53 both-form points below VLM-Forecast on CLEVRER, but 4.71 candidate and 7.63 all-options points above it on ComPhy. All same-seed paired intervals exclude zero in these respective directions. These source-dependent effects do not support an unconditional advantage for external representations. No shuffled VLM-Forecast condition was evaluated.

Figure 8 retains the complete fit-specific comparison, including both favorable and unfavorable contrasts.

Table 7: Absolute accuracy (percent) on the main evaluation: 1,000 CLEVRER scenes and 297 ComPhy groups. Fixed baselines are shared by all fits. Every learned route is shown at each training seed on the same sources. Primary denotes answer accuracy on CLEVRER and candidate accuracy on ComPhy; joint requires both original responses correct. Three-seed means and sample SDs appear in Table 1.

Training seed	Method	Primary	Joint
CLEVRER-Next (1,000 scenes); primary: answer; joint: both-form			
Fixed	VLM (observed video)	46.15	41.50
Fixed	VLM + static continuation	50.30	46.10
13	VLM-Forecast	61.10	56.40
13	FoReST-Latent	58.20	52.70
13	FoReST-Direct	63.25	58.90
17	VLM-Forecast	61.35	57.30
17	FoReST-Latent	57.20	51.60
17	FoReST-Direct	63.65	59.30
42	VLM-Forecast	61.10	56.70
42	FoReST-Latent	57.50	52.50
42	FoReST-Direct	62.60	58.80
ComPhy (297 groups); primary: candidate; joint: all-options			
Fixed	VLM (observed video)	69.02	47.81
Fixed	VLM + static continuation	70.54	49.49

Continued on next page

Training seed	Method	Primary	Joint
13	VLM-Forecast	75.93	57.91
13	FoReST-Latent	80.81	66.33
13	FoReST-Direct	78.79	63.64
17	VLM-Forecast	76.60	59.93
17	FoReST-Latent	80.64	66.33
17	FoReST-Direct	77.10	61.62
42	VLM-Forecast	75.42	57.91
42	FoReST-Latent	80.64	65.99
42	FoReST-Direct	76.09	59.60

Table 8: Paired effects on the main evaluation (CLEVRER: 1,000 scenes; ComPhy: 297 groups). Every cell is VLM-Forecast minus the column comparator, in percentage points [95% source-bootstrap interval]. Learned comparisons match training seeds; native baselines are fixed. Each interval uses 5,000 scene/group resamples with seed 13, retaining both responses together, and conditions on the selected fit. These are not intervals for three-seed means; the fit-specific figure in Appendix C.3 (Figure 8) uses the reverse direction for FoReST comparisons.

Seed	VLM (observed video)	VLM + static continuation	FoReST-Latent	FoReST-Direct
CLEVRER-Next (1,000 scenes): answer accuracy				
	+14.95	+10.80	+2.90	-2.15
13	[+12.35, +17.75]	[+8.25, +13.50]	[+0.55, +5.25]	[-4.50, +0.25]
	+15.20	+11.05	+4.15	-2.30
17	[+12.60, +17.90]	[+8.25, +13.75]	[+1.70, +6.60]	[-4.70, +0.10]
	+14.95	+10.80	+3.60	-1.50
42	[+12.30, +17.70]	[+8.05, +13.55]	[+1.30, +5.95]	[-3.80, +0.80]
CLEVRER-Next (1,000 scenes): both-form accuracy				
	+14.90	+10.30	+3.70	-2.50
13	[+12.00, +18.00]	[+7.40, +13.30]	[+1.10, +6.40]	[-5.10, +0.20]
	+15.80	+11.20	+5.70	-2.00
17	[+12.90, +18.90]	[+8.10, +14.20]	[+2.90, +8.60]	[-4.60, +0.70]
	+15.20	+10.60	+4.20	-2.10
42	[+12.30, +18.20]	[+7.60, +13.70]	[+1.50, +7.00]	[-4.70, +0.50]
ComPhy (297 groups): candidate accuracy				
	+6.90	+5.39	-4.88	-2.86
13	[+4.04, +9.76]	[+3.20, +7.58]	[-7.07, -2.86]	[-4.55, -1.18]
	+7.58	+6.06	-4.04	-0.51
17	[+4.71, +10.44]	[+3.70, +8.42]	[-6.06, -2.02]	[-2.36, +1.52]
	+6.40	+4.88	-5.22	-0.67
42	[+3.54, +9.26]	[+2.69, +7.07]	[-7.41, -3.03]	[-2.69, +1.35]
ComPhy (297 groups): all-options accuracy				
	+10.10	+8.42	-8.42	-5.72
13	[+5.39, +14.81]	[+4.38, +12.46]	[-12.12, -5.05]	[-8.75, -2.69]
	+12.12	+10.44	-6.40	-1.68
17	[+7.41, +16.84]	[+6.06, +14.81]	[-10.10, -3.03]	[-5.05, +1.68]
	+10.10	+8.42	-8.08	-1.68
42	[+5.39, +14.81]	[+4.71, +12.46]	[-11.78, -4.38]	[-5.05, +1.68]

Table 9 also retains differences of the three-seed mean accuracies, separately from the conditional paired intervals.

Table 9: Differences in mean accuracy on the main evaluation populations (CLEVRER 1,000 scenes; ComPhy 297 groups): first minus second, in percentage points. Learned methods are averaged over seeds 13, 17, and 42; the observed-video baseline is fixed. Per-seed paired intervals appear in Appendix C.3.

Contrast	CLEVRER answer	ComPhy candidate
VLM-Forecast − observed VLM	$+15.03$	$+6.96$
FoReST-Latent − VLM-Forecast	-3.55	$+4.71$
FoReST-Direct − VLM-Forecast	$+1.98$	$+1.35$

C.4 Complete main event results

The complete main CLEVRER-Next evaluation contains 600 collision, 162 entrance, and 238 exit scenes. Table 1 reports collision accuracy alongside full-task scores. Figure 9 gives the complete comparison with VLM-Forecast; Figure 2 in the main text examines dependence on each route's matched update. All original questions, candidate choices, fitted heads, and donor assignments remain fixed. The collision emphasis follows the earlier event analysis; the full-task evaluation and every event family remain reported. These populations differ from the 170/162/168-scene analysis families in Appendix D.3; pooling does not create a new independent test.

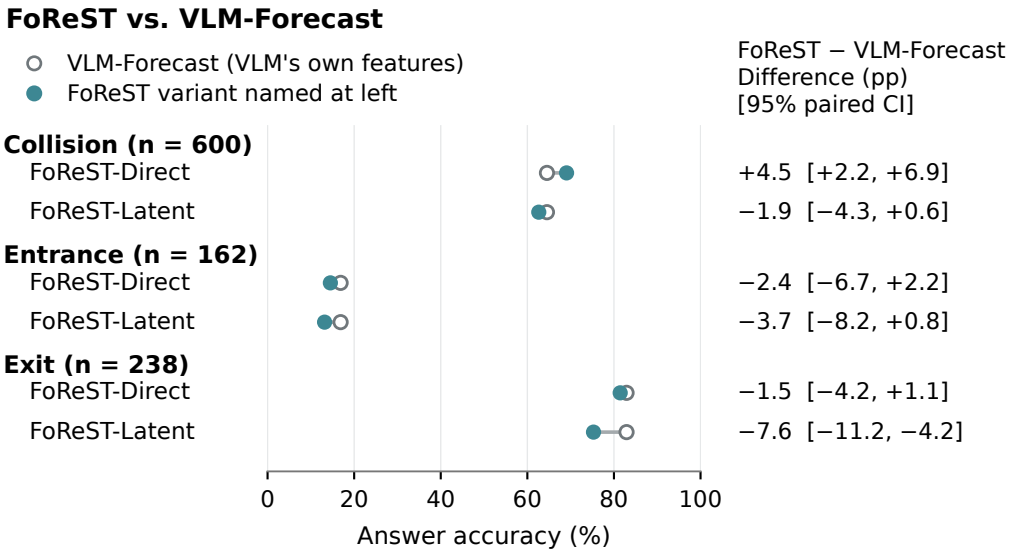


Figure 9: **Event-wise comparison with VLM-native forecasting.** Open points show VLM-Forecast and filled points the FoReST variant named in each row. The complete CLEVRER-Next evaluation is partitioned into collision ($n = 600$), entrance ($n = 162$), and exit ($n = 238$) scenes without changing candidates or presentations. Accuracies average three fixed fits; the right-hand column gives FoReST minus VLM-Forecast and its paired 95% scene-bootstrap interval, conditional on those fits.

Table 10 reports both metrics and fitting variability for all seven displayed conditions. Table 11 retains every fitted-head endpoint, and Table 12 gives the fixed-fit-average source intervals defined in Appendix B.5. Every condition covers the same original source and question IDs.

On collisions, FoReST-Direct exceeds VLM-Forecast by 4.53 answer points [2.19, 6.89], whereas FoReST-Latent differs by -1.89 [-4.33, 0.61]. Both routes have positive matched-minus-shuffled effects, 9.97 [6.56, 13.50] and 15.44 [12.33, 18.61] points for Latent and Direct. The Direct advantage connects its collision gain to the external forecasting route rather than only to a learned native update; latent transfer does not establish that advantage. Entrance intervals for these native and matched-content contrasts cross zero. On exits, Latent remains below VLM-Forecast by 7.63 points [4.20, 11.20], while Direct retains a positive matched-content effect but an uncertain native-comparator difference.

Table 10: Main CLEVRER-Next event accuracies (percent). All 1,000 scenes and both original five-choice presentations are retained. Learned entries are mean ± sample SD across three fitted heads. These 600/162/238-scene families differ from the earlier analysis subset.

Method	Answer	Both-form
Collision (600 scenes)		
Observed video	48.08	41.83
Static continuation	51.58	46.00
VLM-Forecast	64.53 ± 0.61	59.39 ± 1.00
FoReST-Latent	62.64 ± 0.32	56.78 ± 0.42
FoReST-Direct	69.06 ± 0.55	64.39 ± 1.13
Latent shuffled	52.67 ± 0.87	46.67 ± 1.01
Direct shuffled	53.61 ± 1.13	47.56 ± 1.08
Entrance (162 scenes)		
Observed video	0.93	0.62
Static continuation	1.23	0.62
VLM-Forecast	16.87 ± 2.70	12.35 ± 3.27
FoReST-Latent	13.17 ± 1.46	8.23 ± 1.28

Continued on next page

Method	Answer	Both-form
FoReST-Direct	14.51 ± 1.88	10.49 ± 1.63
Latent shuffled	14.92 ± 0.99	11.11 ± 1.23
Direct shuffled	17.08 ± 2.62	12.14 ± 3.40
Exit (238 scenes)		
Observed video	72.06	68.49
Static continuation	80.46	77.31
VLM-Forecast	82.91 ± 1.06	80.53 ± 1.89
FoReST-Latent	75.28 ± 0.44	70.87 ± 0.64
FoReST-Direct	81.44 ± 1.59	78.43 ± 2.43
Latent shuffled	74.16 ± 0.36	71.71 ± 0.49
Direct shuffled	75.07 ± 1.95	71.85 ± 1.83

Table 11: Individual fitted-head accuracy (percent) for every original main-evaluation condition and event family. A/B denotes answer/both-form accuracy. Fixed native baselines do not have a training-seed identity.

Method	Seed	Collision (600)		Entrance (162)		Exit (238)	
		A	B	A	B	A	B
Observed video	–	48.08	41.83	0.93	0.62	72.06	68.49
Static continuation	–	51.58	46.00	1.23	0.62	80.46	77.31
VLM-Forecast	13	63.83	58.33	19.14	13.58	82.77	80.67
VLM-Forecast	17	65.00	60.33	17.59	14.81	81.93	78.57
VLM-Forecast	42	64.75	59.50	13.89	8.64	84.03	82.35
FoReST-Latent	13	63.00	57.17	14.81	9.26	75.63	71.01
FoReST-Latent	17	62.42	56.33	12.04	6.79	74.79	70.17
FoReST-Latent	42	62.50	56.83	12.65	8.64	75.42	71.43
FoReST-Direct	13	68.42	63.50	15.74	11.11	82.56	79.83
FoReST-Direct	17	69.33	64.00	15.43	11.73	82.14	79.83
FoReST-Direct	42	69.42	65.67	12.35	8.64	79.62	75.63
Latent shuffled	13	52.25	46.00	16.05	12.35	74.58	72.27
Latent shuffled	17	53.67	47.83	14.20	9.88	73.95	71.43
Latent shuffled	42	52.08	46.17	14.51	11.11	73.95	71.43
Direct shuffled	13	54.50	48.67	14.51	9.88	75.63	72.69
Direct shuffled	17	52.33	46.50	16.98	10.49	76.68	73.11
Direct shuffled	42	54.00	47.50	19.75	16.05	72.90	69.75

Table 12: Main event-family paired effects (percentage points [95% scene-bootstrap interval]). Differences first average source-level correctness over the three matched fitted heads. The 5,000 paired resamples preserve original acquisition-block counts within each event and keep both question forms together. Intervals condition on the retained heads and do not quantify retraining uncertainty. No VLM-Forecast shuffle was measured.

Contrast	Answer	Both-form
Collision (600 scenes)		
VLM-Forecast − Observed video	$+16.44$ $[+13.17, +19.92]$	$+17.56$ $[+13.83, +21.44]$
VLM-Forecast − Static continuation	$+12.94$ $[+9.56, +16.39]$	$+13.39$ $[+9.67, +17.11]$
FoReST-Latent − Observed video	$+14.56$ $[+10.75, +18.42]$	$+14.94$ $[+10.78, +19.22]$
FoReST-Latent − Static continuation	$+11.06$ $[+7.42, +14.64]$	$+10.78$ $[+6.83, +14.83]$
FoReST-Latent − VLM-Forecast	-1.89 $[-4.33, +0.61]$	-2.61 $[-5.44, +0.28]$
FoReST-Latent − own shuffled	$+9.97$ $[+6.56, +13.50]$	$+10.11$ $[+6.50, +13.78]$
FoReST-Direct − Observed video	$+20.97$ $[+17.22, +24.72]$	$+22.56$ $[+18.50, +26.61]$
FoReST-Direct − Static continuation	$+17.47$ $[+13.78, +21.19]$	$+18.39$ $[+14.44, +22.44]$
FoReST-Direct − VLM-Forecast	$+4.53$ $[+2.19, +6.89]$	$+5.00$ $[+2.55, +7.61]$
FoReST-Direct − own shuffled	$+15.44$ $[+12.33, +18.61]$	$+16.83$ $[+13.50, +20.22]$

Continued on next page

Contrast	Answer	Both-form
Entrance (162 scenes)		
VLM-Forecast − Observed video	$+15.95$ $[+12.14, +20.06]$	$+11.73$ $[+8.43, +15.43]$
VLM-Forecast − Static continuation	$+15.64$ $[+11.93, +19.75]$	$+11.73$ $[+8.43, +15.43]$
FoReST-Latent − Observed video	$+12.24$ $[+8.54, +16.56]$	$+7.61$ $[+4.32, +11.52]$
FoReST-Latent − Static continuation	$+11.93$ $[+7.92, +16.36]$	$+7.61$ $[+4.32, +11.52]$
FoReST-Latent − VLM-Forecast	-3.70 $[-8.23, +0.82]$	-4.12 $[-8.64, +0.21]$
FoReST-Latent − own shuffled	-1.75 $[-6.79, +3.29]$	-2.88 $[-7.82, +1.85]$
FoReST-Direct − Observed video	$+13.58$ $[+10.19, +17.39]$	$+9.88$ $[+6.79, +13.58]$
FoReST-Direct − Static continuation	$+13.27$ $[+9.77, +17.18]$	$+9.88$ $[+6.79, +13.58]$
FoReST-Direct − VLM-Forecast	-2.37 $[-6.69, +2.16]$	-1.85 $[-5.97, +2.47]$
FoReST-Direct − own shuffled	-2.57 $[-7.41, +2.26]$	-1.65 $[-6.38, +3.09]$
Exit (238 scenes)		
VLM-Forecast − Observed video	$+10.85$ $[+6.51, +15.41]$	$+12.04$ $[+7.14, +16.95]$
VLM-Forecast − Static continuation	$+2.45$ $[-1.19, +6.30]$	$+3.22$ $[-1.26, +7.84]$
FoReST-Latent − Observed video	$+3.22$ $[-1.61, +8.12]$	$+2.38$ $[-2.94, +7.70]$
FoReST-Latent − Static continuation	-5.18 $[-9.73, -0.91]$	-6.44 $[-11.76, -1.26]$
FoReST-Latent − VLM-Forecast	-7.63 $[-11.20, -4.20]$	-9.66 $[-14.15, -5.46]$
FoReST-Latent − own shuffled	$+1.12$ $[-3.08, +5.25]$	-0.84 $[-5.74, +4.06]$
FoReST-Direct − Observed video	$+9.38$ $[+4.90, +13.94]$	$+9.94$ $[+4.76, +14.85]$
FoReST-Direct − Static continuation	$+0.98$ $[-3.01, +4.90]$	$+1.12$ $[-3.64, +5.74]$
FoReST-Direct − VLM-Forecast	-1.47 $[-4.20, +1.12]$	-2.10 $[-5.32, +0.84]$
FoReST-Direct − own shuffled	$+6.37$ $[+2.31, +10.22]$	$+6.58$ $[+2.10, +10.92]$

D Interface, content, and receiver diagnostics

D.1 Actual-to-predicted interpolation

Each fixed actual-input-selected seed-13 interface receives $Z_\lambda = (1-\lambda)T + \lambda P$ at $\lambda = 0, 1/2, 1$. DELTA-add subtracts the recipient's persistence before mapping; ABS-add and ABS-replace retain absolute input and their own fitted residual or absolute-native targets. Actual and midpoint inputs use future information. Table 13 retains all accuracy endpoints and both metrics. Figure 3 instead measures loss from the same fitted interface's actual endpoint; its paired intervals resample scene-level differences, not marginal endpoint intervals.

Replacement's excess actual-to-predicted loss is 6.1 form points [2.0, 10.2] and 9.0 both-form points [4.2, 13.8] versus ABS-add, or 4.3 [0.0, 8.9] and 6.4 [1.2, 11.8] versus DELTA-add. All midpoint excess-loss intervals include or touch zero. At the predicted endpoint, ABS-add exceeds replacement by 2.4 form points [-1.1, 6.0] and 6.0 both-form points [2.2, 10.0]. DELTA-add minus ABS-add is 0.0 [-2.4, 2.5] and -1.4 [-4.4, 1.6]. Less degradation is therefore distinct from always higher endpoint accuracy.

Among 293 persistence-incorrect scenes, DELTA-add corrects $92 \to 73 \to 49$ along the interpolation; ABS-add corrects $82 \to 57 \to 44$; replacement corrects $111 \to 80 \to 48$. Harms among 207 baseline-fully-correct scenes are $28 \to 29 \to 26$, $24 \to 19 \to 14$, and $38 \to 46 \to 48$. Predicted teacher-available recovery is 36/92, 31/82, and 34/111 for the three interfaces. These conditional denominators are mapper-specific, not a common test population or an evaluated router.

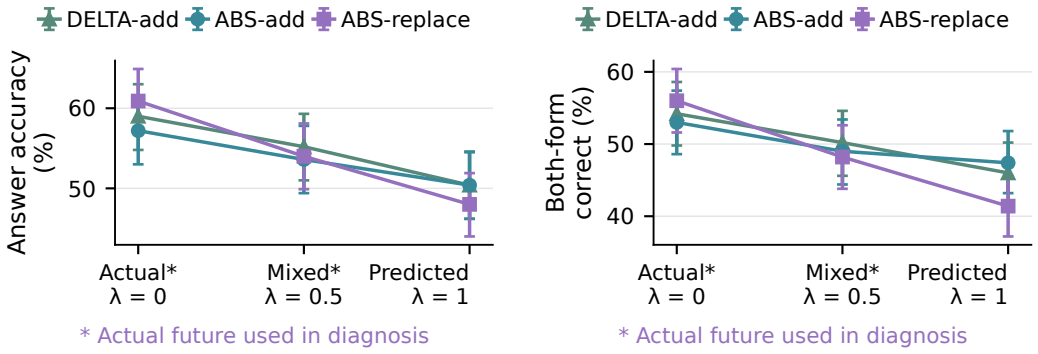
D.2 Context-conditioned readout

CTX-P and CTX-O share FoReST-Direct's observed memory and a 13,145,088-parameter architecture, differing in predictive versus observed auxiliary queries (Appendix A.7). Table 6 reports their two-seed absolute results. Comparisons match seeds 13/17 on the 500-scene/200-group analysis populations. Auxiliary shuffle and zero preserve recipient memory, native tokens, references, and text; they differ from whole-update FoReST-Latent/FoReST-Direct shuffling.

On CLEVRER, CTX-P exceeds pointwise FoReST-Latent in both metrics at both seeds, while differences from FoReST-Direct and CTX-O remain uncertain. On ComPhy, it falls below FoReST-Latent at both seeds and differs uncertainly from FoReST-Direct. Its all-options advantage over CTX-O is positive only at seed 13; that seed's option interval touches zero, and both seed-17 intervals

Table 13: Fixed-fit actual-to-predicted interpolation on 500 CLEVRER analysis scenes: accuracy (%) [95% scene-bootstrap interval]. Actual and midpoint require future information. Each curve holds its interface and actual-input reference seed-13 mapper fixed.

Interface	Input	Answer acc.	Both-form
DELTA-add	Actual	59.00 [54.80, 63.00]	54.20 [49.80, 58.60]
DELTA-add	Midpoint	55.20 [51.00, 59.30]	50.20 [45.60, 54.60]
DELTA-add	Predicted	50.40 [46.20, 54.50]	46.00 [41.60, 50.20]
ABS-add	Actual	57.20 [53.00, 61.30]	53.00 [48.60, 57.40]
ABS-add	Midpoint	53.60 [49.40, 57.80]	49.00 [44.40, 53.40]
ABS-add	Predicted	50.40 [46.20, 54.60]	47.40 [43.20, 51.80]
ABS-replace	Actual	60.90 [56.80, 64.90]	56.00 [51.60, 60.40]
ABS-replace	Midpoint	54.00 [49.90, 58.10]	48.20 [43.80, 52.60]
ABS-replace	Predicted	48.00 [44.00, 51.90]	41.40 [37.20, 45.60]



(a) Answer accuracy

(b) Both-form correctness

Figure 10: Raw accuracy along actual-to-predicted interpolation for all three fixed actual-input seed-13 interfaces on the same 500 CLEVRER analysis scenes. The panels retain form and both-form accuracy, with 95% scene-bootstrap intervals for each endpoint. Each curve keeps its mapper and injection operation fixed. Actual and midpoint inputs use future information; predicted inputs are observed-only. These are marginal accuracy intervals, distinct from the paired degradation intervals in Figure 3.

include zero. CTX-O falls below FoReST-Direct at CLEVRER seed 17 and ComPhy seed 13. Thus these controls do not establish a consistent predictive-query advantage over the observed-only alternatives.

ComPhy CTX-P exceeds auxiliary shuffle and zero in both metrics at both seeds. CLEVRER's positive effects are limited to shuffle/form at seed 13 and zero/both-form at seed 17, whose form interval touches zero. An auxiliary branch can matter without improving on FoReST-Direct.

D.3 Event selectivity, corrections, and content controls

The analyses in this subsection use the fixed development subsets of 500 CLEVRER scenes and 200 ComPhy groups (Appendix B.3). Table 14 uses all three seeds consistently for the primary event means. Entrance gains over persistence coexist with nonpositive mean matched-content differences. The analysis set retains every event family, independently of these outcomes.

The temporal-reversal and training-mean controls use one fixed FoReST-Direct head per dataset (seed 13) on these subsets (Figure 11). Temporal reversal reorders only the external encoder's observed blocks; the mean update averages predictions from training contexts. Neither changes the recipient VLM's native video or question. The different-video condition is the existing whole-update shuffle, using the same fitted head.

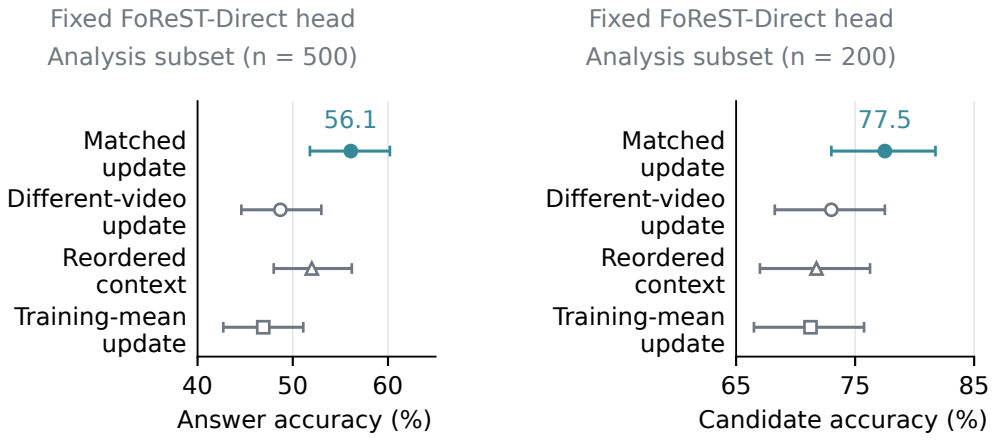
On ComPhy analysis, all T/P/M/FoReST-Direct seeds exceed their own whole-update shuffles in both metrics. FoReST-Direct seed 13 exceeds reversal by 5.75 option points [3.25, 8.25] and 9.50

Table 14: Video-specific content on the 500-scene CLEVRER analysis set: form-accuracy gains over static continuation and own-shuffled update (percentage points). Cells give mean ± sample SD over three training seeds, not confidence intervals. Each donor supplies the complete update while the recipient native input and frozen receiver stay fixed.

Event	N	Route	Over static continuation	Over shuffle
Collision	170	FoReST-Latent	9.90 ± 1.51	9.12 ± 3.28
Collision	170	FoReST-Direct	16.18 ± 1.35	16.57 ± 1.89
Entrance	162	FoReST-Latent	11.93 ± 1.46	-1.75 ± 0.47
Entrance	162	FoReST-Direct	13.27 ± 1.88	-2.57 ± 4.41
Exit	168	FoReST-Latent	-4.56 ± 0.34	0.30 ± 0.52
Exit	168	FoReST-Direct	2.48 ± 2.42	5.36 ± 0.89

Table 15: Seed-13 content controls on the analysis sets (500 CLEVRER scenes, 200 ComPhy groups), accuracy (%). The actual-input reference is an earlier fixed FoReST-Latent mapper, separate from both the controlled fitting ablation and primary predicted-input fit. Reversal changes the external observed sequence; training mean replaces the external update with its scene-balanced training mean. Native recipient input and references stay fixed.

Condition	C answer	C both	M candidate	M all
FoReST-Latent reference (actual-input)				
Matched	50.40	46.00	77.25	61.50
Shuffled	46.60	42.40	71.50	53.00
Reversed	47.40	42.60	71.75	53.00
Training mean	45.40	41.80	69.75	48.50
FoReST-Direct				
Matched	56.10	52.20	77.50	62.50
Shuffled	48.70	44.80	73.00	56.00
Reversed	52.00	47.60	71.75	53.00
Training mean	46.90	43.80	71.25	52.50



(a) CLEVRER-Next (b) ComPhy

Figure 11: **Sequence-dependent updates versus a shared adjustment.** One fixed FoReST-Direct head per dataset is evaluated on the CLEVRER (500 scenes) and ComPhy (200 groups) analysis subsets. The receiver's native input is unchanged. The external path uses its original context, a different video's update, a reordered context, or a fixed training-mean update. Error bars are the existing marginal 95% accuracy intervals conditional on these heads; the paired contrasts reported below quantify differences between conditions. Reordering is an input intervention, not a measurement of physical prediction error in isolation.

all-options points [5.50, 14.00], and the training mean by 6.25 [4.00, 8.75] and 10.00 [6.00, 14.50]. The earlier actual-input reference mapper also exceeds these controls; its 50.40% CLEVRER and 77.25% ComPhy analysis accuracies belong to that separate fit, not the primary FoReST-Latent configuration. On CLEVRER analysis, FoReST-Latent has a positive own-shuffle form interval only at seed 13, with every both-form interval crossing zero; the main evaluation uses a larger, more collision-heavy population.

ComPhy seed-13 FoReST-Latent corrects 38 of 103 persistence-incorrect groups and harms two of 97 fully correct groups; FoReST-Direct corrects 29 and harms one. FoReST-Latent alone is fully correct on ten groups, FoReST-Direct alone on two, both on 123, and neither on 65. Oracle unions use correctness labels and are not deployed ensembles.

D.4 Representation error and downstream accuracy

Native errors compare 128×5120 updates with ΔQ; replacement is converted to a residual by subtracting native persistence. Pooled-input errors instead compare 128×1024 inputs with their own actual-future target. Cosine, coordinate MSE, scene-wise relative MSE, token-norm error, and normalized Huber retain separate reductions. Relative MSE divides each scene's error by its target mean squared magnitude, floored at 10^{-8}, before averaging scene ratios.

P/M lower cosine, Huber, and norm errors versus T for every seed, while MSE decreases for P only at seed 13 and for M at none. Within each of 32 endpoints, per-scene error/raw-QA associations satisfy $|\rho| \leq 0.106$; associations with gain over persistence reach 0.174. Per-scene error-change/QA-change associations within seven paired updates satisfy $|\rho| \leq 0.088$. These are not correlations across aggregate method means. ABS-add relative native MSE decreases $1.1719 \rightarrow 1.1159 \rightarrow 1.1128$ while form accuracy falls $57.2 \rightarrow 53.6 \rightarrow 50.4\%$; a single representation distance does not consistently order QA.

D.5 Qwen actual-future diagnosis

The Qwen real-future comparison fixes 500 CLEVRER analysis scenes and the neutral prompt. Actual pixels supply the four real continuation frames; exact injection replaces only persistence's native future block with the native actual block. Actual-latent and predicted-latent inputs share each receiver's scratch actual-input mapper, distinct from primary predicted-input FoReST-Latent. Pixel and exact-block answers agree on every evaluated form under both prompts. Tables 16 and 17 retain both metrics and the separate oracle endpoints.

Actual-pixel gains are positive in every event family for both receivers. The 27B collision/entrance/exit form gains are 30.3/89.8/11.9 points; 7B gains are 14.4/66.7/17.3. Translation gaps also vary: mapped actual input falls 60.5 form points below native actual input on 27B entrances and

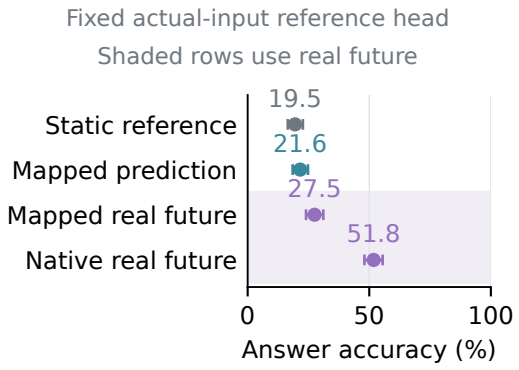


Figure 12: **Future-evidence diagnosis in Qwen2.5-VL-7B.** As in Figure 4, mapped actual and predicted states share one fixed actual-input reference head. Native pixels and exact native injection produce identical answers and share an endpoint. Bars are marginal 95% scene-bootstrap accuracy intervals on the same 500 scenes; shaded conditions access real future observations.

Table 16: Real-future and latent-input QA with fixed receiver-specific mappers. Accuracy (%) on the same 500 CLEVRER analysis scenes. The actual-pixel, exact-token, and actual-latent rows use future information; mapper conditions retain each receiver's fixed actual-input reference fit. Pixel and exact-token conditions produce identical answers on all 1,000 forms for each receiver.

Condition	27B answer	27B both	7B answer	7B both
VLM (observed video)	39.70	35.80	16.00	10.80
VLM + static continuation	45.00	41.40	19.50	14.60
Actual future pixels	88.40	85.00	51.80	39.40
Exact future tokens	88.40	85.00	51.80	39.40
Actual-latent mapper	59.30	55.20	27.50	19.60
Predicted-latent mapper	50.50	47.40	21.60	15.20
Shuffled prediction	46.70	42.40	20.30	13.80

Table 17: Receiver diagnosis: paired differences in percentage points [95% scene-bootstrap interval], 500 analysis scenes and 5,000 resamples. Each comparison keeps receiver, prompt, and the earlier actual-input reference mapper fixed.

Difference	Answer acc.	Both-form
Qwen3.5-27B		
Pixels − static continuation	43.40 [38.80, 48.00]	43.60 [38.80, 48.60]
Pixels − exact	0.00 [0.00, 0.00]	0.00 [0.00, 0.00]
Mapped actual − exact	-29.10 [-33.50, -24.60]	-29.80 [-34.80, -24.80]
Prediction − mapped actual	-8.80 [-12.10, -5.50]	-7.80 [-11.40, -4.20]
Prediction − shuffle	3.80 [0.90, 6.80]	5.00 [1.80, 8.40]
Qwen2.5-VL-7B		
Pixels − static continuation	32.30 [28.30, 36.20]	24.80 [20.40, 29.00]
Pixels − exact	0.00 [0.00, 0.00]	0.00 [0.00, 0.00]
Mapped actual − exact	-24.30 [-28.00, -20.40]	-19.80 [-24.00, -15.60]
Prediction − mapped actual	-5.90 [-8.60, -3.10]	-4.40 [-7.40, -1.40]
Prediction − shuffle	1.30 [-0.70, 3.30]	1.40 [-0.60, 3.40]

46.3 on 7B entrances. The 7B matched-content gap is positive on collisions, 5.6 points [1.5, 9.7], but uncertain on entrances and exits.

D.6 SMALL-RECEIVER EVENT OUTCOMES ON CLEVRER

The event analysis retains collision, entrance, exit, and overall outcomes for each receiver, using the native endpoints from that same family. The Qwen2.5-VL-7B mapper is actual-input fitted; the InternVL3.5-8B mapper is predicted-input fitted. Both receivers now use the same 500 CLEVRER analysis scenes, with 170 collision, 162 entrance, and 168 exit scenes; their fitting recipes differ (Appendix B.6). Tables 19 and 20 retain absolute accuracies and source-paired contrasts in both metrics. Across all 500 CLEVRER scenes, InternVL matched forecasts exceed their shuffle by 2.6 answer points [−1.1, 6.4] and observed video by 2.3 [−1.8, 6.6]; neither interval excludes zero. The collision matched-content difference remains positive at 9.4 [3.5, 15.6]. Table 18 includes the static-continuation contrasts omitted from the main event plot.

Table 18: **Event-specific forecast gains in two additional VLMs.** Answer-accuracy differences in percentage points [paired 95% scene-bootstrap interval], comparing matched predictions with each model's own baselines. Each original question retains all five choices and both forms. Qwen2.5-VL-7B uses 500 scenes and an actual-input mapper; InternVL3.5-8B uses the same 500 scenes and a predicted-input mapper. Each model has one fitted seed. Companion tables in this subsection give absolute accuracies, both-form results, and overall outcomes.

Receiver / event	N	Matched–observed	Matched–static	Matched–shuffle
Qwen2.5-VL-7B: FoReST-Latent, actual-input (T)				
Collision	170	+12.65 [+7.65, +17.65]	+6.76 [+2.94, +10.59]	+5.59 [+1.47, +9.71]
Entrance	162	+0.62 [-1.54, +2.78]	+1.54 [+0.31, +3.09]	-0.62 [-3.40, +2.16]
Exit	168	+3.27 [-1.19, +8.04]	-2.08 [-6.25, +1.79]	-1.19 [-4.17, +1.79]
InternVL3.5-8B: FoReST-Latent, predicted-input (P)				
Collision	170	+3.24 [-2.94, +9.12]	+12.35 [+6.47, +18.24]	+9.41 [+3.53, +15.59]
Entrance	162	+24.38 [+17.28, +31.48]	+22.22 [+15.42, +29.32]	+1.23 [-5.86, +8.33]
Exit	168	-19.94 [-26.79, -12.80]	-2.98 [-8.93, +2.68]	-2.98 [-9.23, +3.27]

Table 19: Small-receiver accuracies on CLEVRER (percent). O: VLM (observed video); R: static continuation; P: predicted matched; S: own shuffled. Both receivers use the same 500 scenes. Each scene has two question forms; joint accuracy requires both correct. T/P identify actual-/predicted-input fitting, respectively; both use seed 13.

Receiver/fit	Event	N	Metric	O	R	P	S
Qwen7B/T	Overall	500	form	16.00	19.50	21.60	20.30
Qwen7B/T	Overall	500	joint	10.80	14.60	15.20	13.80
Qwen7B/T	Collision	170	form	23.53	29.41	36.18	30.59
Qwen7B/T	Collision	170	joint	15.29	20.59	25.88	20.59
Qwen7B/T	Entrance	162	form	1.85	0.93	2.47	3.09
Qwen7B/T	Entrance	162	joint	0.62	0.62	1.23	1.85
Qwen7B/T	Exit	168	form	22.02	27.38	25.30	26.49
Qwen7B/T	Exit	168	joint	16.07	22.02	17.86	18.45
IV8B/P	Overall	500	form	40.20	32.10	42.50	39.90
IV8B/P	Overall	500	joint	29.80	21.60	30.60	27.80
IV8B/P	Collision	170	form	48.53	39.41	51.76	42.35
IV8B/P	Collision	170	joint	38.82	28.24	38.24	31.18
IV8B/P	Entrance	162	form	14.81	16.98	39.20	37.96
IV8B/P	Entrance	162	joint	6.79	6.79	29.63	23.46
IV8B/P	Exit	168	form	56.25	39.29	36.31	39.29
IV8B/P	Exit	168	joint	42.86	29.17	23.81	28.57

Table 20: Matched prediction effects for every CLEVRER family and overall, in percentage points with paired 95% source-bootstrap intervals. Both question forms stay attached to each scene. All comparisons use the same receiver, population, prompt, and selected fit; no cross-receiver subtraction is made. One training seed per receiver; bootstrap uncertainty conditions on that fit and the fixed donor map.

Receiver/fit	Event	N	Metric	P–O	P–R	P–S
Qwen7B/T	Overall	500	form	+5.60 [+3.10, +8.00]	+2.10 [+0.10, +4.10]	+1.30 [-0.70, +3.30]
Qwen7B/T	Overall	500	joint	+4.40 [+1.80, +7.00]	+0.60 [-1.40, +2.60]	+1.40 [-0.60, +3.40]
Qwen7B/T	Collision	170	form	+12.65 [+7.65, +17.65]	+6.76 [+2.94, +10.59]	+5.59 [+1.47, +9.71]
Qwen7B/T	Collision	170	joint	+10.59 [+5.29, +15.88]	+5.29 [+1.18, +9.41]	+5.29 [+0.59, +10.00]
Qwen7B/T	Entrance	162	form	+0.62 [-1.54, +2.78]	+1.54 [+0.31, +3.09]	-0.62 [-3.40, +2.16]
Qwen7B/T	Entrance	162	joint	+0.62 [-1.23, +2.47]	+0.62 [+0.00, +1.85]	-0.62 [-3.09, +1.23]
Qwen7B/T	Exit	168	form	+3.27 [-1.19, +8.04]	-2.08 [-6.25, +1.79]	-1.19 [-4.17, +1.79]
Qwen7B/T	Exit	168	joint	+1.79 [-2.98, +6.55]	-4.17 [-8.33, +0.00]	-0.60 [-3.57, +2.38]
IV8B/P	Overall	500	form	+2.30 [-1.80, +6.60]	+10.40 [+6.80, +14.10]	+2.60 [-1.10, +6.40]
IV8B/P	Overall	500	joint	+0.80 [-3.60, +5.40]	+9.00 [+5.20, +13.00]	+2.80 [-1.40, +7.00]
IV8B/P	Collision	170	form	+3.24 [-2.94, +9.12]	+12.35 [+6.47, +18.24]	+9.41 [+3.53, +15.59]
IV8B/P	Collision	170	joint	-0.59 [-7.65, +5.88]	+10.00 [+4.12, +16.47]	+7.06 [+0.00, +14.12]
IV8B/P	Entrance	162	form	+24.38 [+17.28, +31.48]	+22.22 [+15.42, +29.32]	+1.23 [-5.86, +8.33]
IV8B/P	Entrance	162	joint	+22.84 [+15.43, +30.25]	+22.84 [+15.43, +30.25]	+6.17 [-1.85, +14.20]
IV8B/P	Exit	168	form	-19.94 [-26.79, -12.80]	-2.98 [-8.93, +2.68]	-2.98 [-9.23, +3.27]
IV8B/P	Exit	168	joint	-19.05 [-26.79, -10.71]	-5.36 [-11.90, +1.19]	-4.76 [-11.31, +1.79]

D.7 ComPhy references

Removing all four reference videos also removes their descriptive sentence, preserving target, question, candidate semantics, mapper, and donor map. The interaction subtracts the method-minus-persistence gain without references from that with references on the same 200 groups. Every stored interaction interval includes zero. For FoReST-Latent seed 13, effects are 1.00 option points $[-2.75, 4.51]$ and 0.50 all-options points $[-6.00, 7.00]$; FoReST-Direct gives 0.25 $[-3.50, 4.00]$ and -0.50 $[-7.01, 6.50]$. This does not establish equivalence of reference settings. Without references, FoReST-Latent minus FoReST-Direct is 2.25 option points [0.00, 4.51] and 3.00 all-options points $[-1.50, 7.50]$.

E Additional Related-Work Context

Representation-learning foundations. VideoMAE reconstructs masked video pixels (Tong et al., 2022), while I-JEPA predicts representations of hidden image regions and V-JEPA extends feature prediction to video (Assran et al., 2023; Bardes et al., 2024). V-JEPA 2 scales predictive video learning for understanding and anticipation, with an action-conditioned extension for planning (Assran et al., 2025). These methods provide the representation-learning context for FoReST; our experiments keep the released world model fixed. CLIP aligns image and text representations (Radford et al., 2021), and LLaVA develops visual instruction tuning (Liu et al., 2023). V-JEPA 2 also aligns its encoder with a language model through staged multimodal training (Assran et al., 2025). FoReST instead retains a complete pretrained receiver and learns an additive visual update without text or QA supervision.

Direction and target of predictive transfer. ThinkJEPA uses VLM semantic guidance to improve JEPA-style world modeling (Zhang et al., 2026), reversing FoReST's direction of transfer from a world model into a VLM. VL-JEPA predicts target-text embeddings and selectively decodes text (Chen et al., 2025); these targets differ from temporal future-video states. Neither the presence of a JEPA architecture nor the use of both model families implies the same interface. FoReST's two alternatives either translate a pretrained latent forecast or learn a future residual from world-model observations, while preserving native visual grounding.

Executable worlds and learned simulation. PhysMind reconstructs geometry and fits scene-specific dynamics to build a reusable, question-agnostic executable world. It answers through trajectory and contact tools (Yang et al., 2026). FoReST shares the possibility of reusing an observation-derived representation across questions, but updates visual tokens rather than reconstructing an executable simulator. SlotFormer and Causal-JEPA use learned object trajectories with a trained QA reasoner (Wu et al., 2023; Nam et al., 2026); unsupervised dynamics learning does not make the complete QA pipeline label-free. These approaches and the original CLEVRER and ComPhy systems belong to a simulation-to-reasoning lineage, with task definitions and supervision distinct from our frozen-receiver evaluation.

Forecast access and forecast use. The tool-use difficulties studied by Qian et al. (2026) motivate distinguishing predictor availability from effective downstream use. Their agentic tasks use direct environment simulation, while spatial VQA uses generated visuals; these protocols neither test FoReST's latent interface nor establish its mechanism. Similarly, robotics policy steering, action-plan refinement, and spatial-image QA provide relevant interface precedents without yielding directly comparable CLEVRER-Next scores. Our route, content, fitting-input, and addition/replacement controls evaluate the proposed interface under its own fixed protocols.