LEARNING WHAT TO RECALL: ADAPTIVE MULTI-CUE EPISODIC MEMORY FOR WORLD MODELS

Anonymous authors
Paper under double-blind review

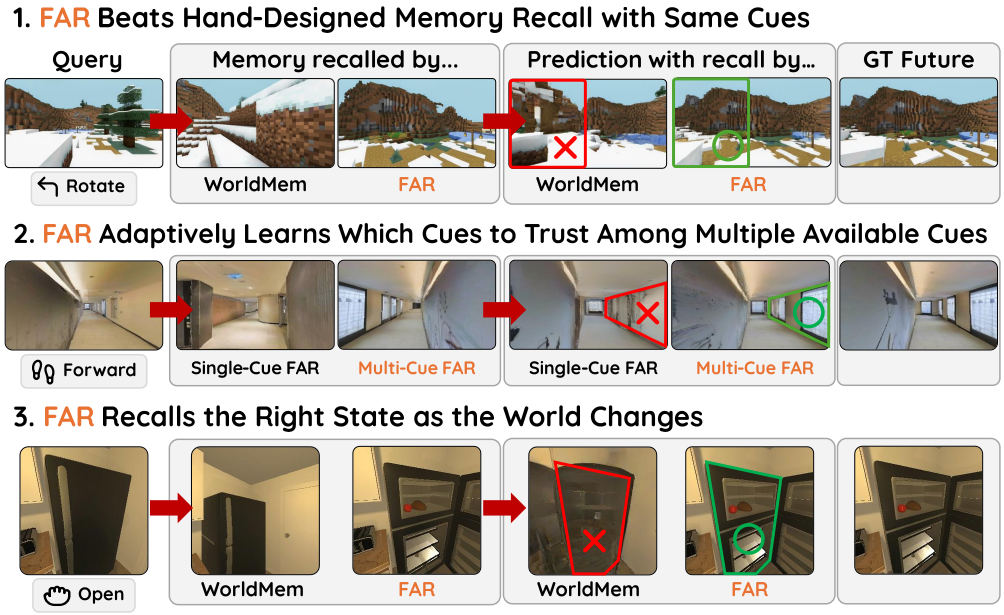


Figure 1: **The three advantages of Future-Aware Recall (FAR). Top:** FAR retrieves context based on its usefulness for predicting the future, beating hand-designed memory recall rules. In the figure, FAR is compared against WorldMem (Xiao et al., 2025). **Middle:** FAR adaptively learns query-dependent weights over multiple available retrieval cues. Here, FAR learns to rely more on audio cue to distinguish memories with occluded and clear line of sight. **Bottom:** FAR recalls memory to support state-correct future predictions. Here, FAR distinguishes between closed and open fridges, in contrast to WorldMem.

ABSTRACT

World models predict future observations from current experience and actions, yet prediction can depend on observations seen far in the past. Episodic memory preserves past observations for later recall; however, as memory accumulates, it raises a fundamental question: *which memories are useful for the current prediction, and which available retrieval cues should be trusted to find them?* This is challenging because fixed criteria based on recency, pose overlap, or visual similarity can be unreliable across environments and queries. We propose **Future-Aware Recall (FAR)**, a framework that learns episodic recall from future-aware predictive supervision and adaptive multi-cue scoring. During training, FAR measures predictive utility by the conditional log-likelihood of the realized future given recalled context, approximated by negative diffusion prediction loss, and uses it to train a retriever that remains future-blind at inference. The retriever learns cue-specific relevance and automatically determines which available retrieval cues, such as time, pose, vision, and audio, to trust for each query when selecting memories. Across three complementary settings, FAR outperforms hand-designed recall even with the same retrieval cues, automatically adapts which available cues to trust, and recalls the right history as the world changes. Together, these results establish FAR as a flexible, principled approach to episodic memory access in world models. Anonymous project page: `https://futureawarerecall.github.io/FAR-Project-Page-anon/`

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

In world modeling, the current observation provides only a partial view of the underlying environment state. Scenes and objects persist, and may continue to evolve, after leaving the field of view. To predict what will be observed upon revisit, world models must preserve information across extended interactions (Ha & Schmidhuber, 2018; Hafner et al., 2019). Memory is therefore central to long-horizon, persistent world modeling. Recent world models explore different mechanisms for maintaining such information. *Persistent-state memory* integrates observations into an evolving representation of the current world, such as a recurrent latent state (Hafner et al., 2020; 2021; 2025) or persistent 3D representation (Wu et al., 2025; Garcin et al., 2026). *Episodic memory* instead preserves individual past observations as separately accessible memories that can later be recalled (Xiao et al., 2025; Hu et al., 2026). These memory functions are complementary: persistent-state memory provides compact access to an evolving world state, while episodic memory preserves directly recoverable evidence from past experience.

Episodic memory, however, introduces a fundamental recall problem. As interaction continues, the number of stored memories grows, while only a small fraction may be useful for a particular prediction. Existing world models commonly define relevance using fixed criteria such as temporal recency (Bar et al., 2025), pose-based field-of-view overlap (Xiao et al., 2025), or visual embedding similarity (Hu et al., 2026). However, similarity under a particular cue need not reflect predictive usefulness, and the reliability of that cue can vary across situations. For example, in an elbow-shaped corridor (see Fig. 6), pose-based retrieval may favor a nearby memory across a wall, while visual appearance may be ambiguous across similar corridor segments. Spatial audio may instead better identify relevant past experience in this case, while pose or vision may be more informative elsewhere. This raises our central question:

> *How can a world model learn which past observations are useful for future prediction, and which available retrieval cues can identify them before the future is known?*

We address this with **Future-Aware Recall (FAR)**. During training, the future is observed, allowing FAR to evaluate each candidate memory by how well it helps the world model predict what actually happens. We call this *predictive utility*, and use it to supervise a retriever that must operate without access to the future at inference. To predict this relevance at recall time, FAR learns a relevance function for each available cue, such as time, pose, vision, or audio, together with query-dependent weights that determine which cues to trust. These cues are used to select memories; they need not be provided to the world model as additional generation conditions. FAR's latent-variable formulation connects naturally to retrieval-augmented language models (Lewis et al., 2020; Sachan et al., 2021), allowing us to adapt discrete retriever optimization to episodic recall for world models.

We instantiate FAR with video diffusion world models and an external episodic memory. Historical observations remain individually addressable, while the retriever selects a compact Top-K context for each prediction. Predictive utility is defined through the conditional likelihood of the realized future; for our video diffusion instantiation, we use negative diffusion prediction loss as a tractable surrogate. The recalled context then conditions the world model for future prediction.

Across three complementary environments, FAR consistently improves episodic recall and downstream prediction over fixed retrieval strategies. In LoopNav (Lian et al., 2025), FAR outperforms hand-designed relevance rules even when given the same retrieval cues, while adaptively using time, pose, and vision further improves long-horizon prediction. In SoundSpaces (Chen et al., 2020), FAR learns to rely on audio when spatial cues become ambiguous, particularly for longer trajectories and sparser memories. In changing-state AI2-THOR environments (Kolve et al., 2017), FAR recalls history consistent with the current world state and reduces errors caused by stale memories. Fig. 1 summarizes these complementary capabilities. Together, these results demonstrate FAR as a flexible, prediction-driven mechanism for episodic memory access in persistent world models.

2 BACKGROUND AND RELATED WORK

World models and episodic memory. World models predict future observations from interaction history and actions using recurrent dynamics, diffusion, or masked generative modeling (Hafner et al., 2020; 2021; 2025; Alonso et al., 2024; Bruce et al., 2024; Kim et al., 2026). Over long interactions, relevant information may lie far in the past, making repeated processing of the full history costly. External episodic memory instead preserves past observations for selective recall

using cues such as time, pose, vision, or audio. We focus on learning which memories to recall and which available cues to trust, rather than relying on fixed relevance rules for individual cues.

Internal and external recall. Generator-internal methods maintain long-range information through attention, routing, context compression, or recurrent memory (Cai et al., 2026; Yu et al., 2026; Peng et al., 2026), with recent world models also using linear attention for efficient long-horizon memory (Wang et al., 2026a; Zhu et al., 2026). Generator-external methods instead search an explicit memory and pass only a compact subset to the world model (Xiao et al., 2025; Yu et al., 2025; Chen et al., 2025; Li et al., 2026). These approaches are complementary: internal memory compactly summarizes history, while external episodic memory preserves individually addressable observations for selective recall. We focus on external recall, which can be combined with internal long-context memory mechanisms.

Learning predictive external recall. External world-model recall commonly uses fixed temporal, geometric, or embedding-based relevance, while discrete Top-K selection prevents direct gradient flow from the world-model objective. Related retrieval-augmented generation methods learn discrete retrieval from downstream likelihood; for example, EMDR^2 (Sachan et al., 2021) uses reader likelihood to supervise a document retriever. FAR applies this latent-variable principle to episodic recall from an evolving interaction history, defining relevance through future predictive utility. Its retriever learns both cue-specific relevance and query-dependent cue reliability from available cues whose usefulness can vary across queries. Section A in Section A provides a detailed comparison.

3 Our Method: Future-Aware Episodic Memory

We propose **Future-Aware Recall (FAR)**, a framework for learning predictive memory relevance in external episodic recall. At inference, FAR must select useful memories using only the episodic memory and current prediction query, since the future to be predicted is unavailable. During training, however, that future is observed. FAR exploits this additional information to identify which memories would have best supported prediction and uses these signals to train a retriever that remains future-blind at inference. FAR thereby learns both *which memories to recall* and *which retrieval cues to trust*. A high-level overview is provided in Fig. 2. We first formalize the *recall problem* underlying FAR, and then introduce its key mechanisms.

The memory recall problem. Let $\boldsymbol{o}_t$ denote the observation at step t and $\boldsymbol{a}_t$ the subsequent action, with prediction query $\mathcal{Q}_t := (\boldsymbol{o}_t, \boldsymbol{a}_t)$. The *episodic memory* contains the past observations $\mathcal{M}_{t-1} := (\boldsymbol{o}_1, \ldots, \boldsymbol{o}_{t-1})$. FAR recalls a compact context $\mathcal{C}_t \subseteq \mathcal{M}_{t-1}$ with $|\mathcal{C}_t| = K$, which the world model uses to predict the next observation: $p_\theta(\boldsymbol{o}_{t+1} \mid \mathcal{C}_t, \mathcal{Q}_t)$.

To locate useful memories, each historical observation $\boldsymbol{o}_i$ is associated with retrieval cues $\boldsymbol{z}_i$, such as time, pose, a visual representation, or an audio representation. We collect the historical cues as $\mathcal{Z}_{t-1} := (\boldsymbol{z}_1, \ldots, \boldsymbol{z}_{t-1})$ and define the current retrieval query as $\mathcal{R}_t := (\boldsymbol{z}_t, \boldsymbol{a}_t)$. These cues are used by the external retriever to determine where to recall from in episodic memory. Specifically, given $\mathcal{Z}_{t-1}$ and $\mathcal{R}_t$, the retriever assigns each candidate context $\mathcal{C}_t$ a *relevance score* $s_\phi(\mathcal{C}_t \mid \mathcal{Z}_{t-1}, \mathcal{R}_t)$, inducing the *recall distribution*

$$r_\phi(\mathcal{C}_t \mid \mathcal{Z}_{t-1}, \mathcal{R}_t) \propto \exp s_\phi(\mathcal{C}_t \mid \mathcal{Z}_{t-1}, \mathcal{R}_t). \tag{1}$$

Exact evaluation of Eq. (1) is generally intractable because it requires considering a combinatorial number of possible recalled context sets as the episodic memory $\mathcal{M}_{t-1}$ grows. For discrete Top-K recall, we therefore use the candidate-level latent-variable approximation adapted from EMDR^2 (Sachan et al., 2021), described in Section B.3.

Together, the retriever and world model define the latent-context predictive model

$$p_{\theta,\phi}(\boldsymbol{o}_{t+1} \mid \mathcal{M}_{t-1}, \mathcal{Z}_{t-1}, \mathcal{R}_t, \mathcal{Q}_t) = \textstyle\sum_{\mathcal{C}_t} p_\theta(\boldsymbol{o}_{t+1} \mid \mathcal{C}_t, \mathcal{Q}_t)\, r_\phi(\mathcal{C}_t \mid \mathcal{Z}_{t-1}, \mathcal{R}_t). \tag{2}$$

Here, r_ϕ determines which past observation is recalled, while p_θ determines how recalled context supports prediction. Importantly, retrieval cues guide the selection of $\mathcal{C}_t$; they need not be provided to the world model as generation conditions. Learning effective episodic recall thus reduces to learning the relevance function s_ϕ, and hence the recall distribution r_ϕ. We now derive predictive supervision for relevance and show how FAR adaptively uses multiple retrieval cues to estimate it.

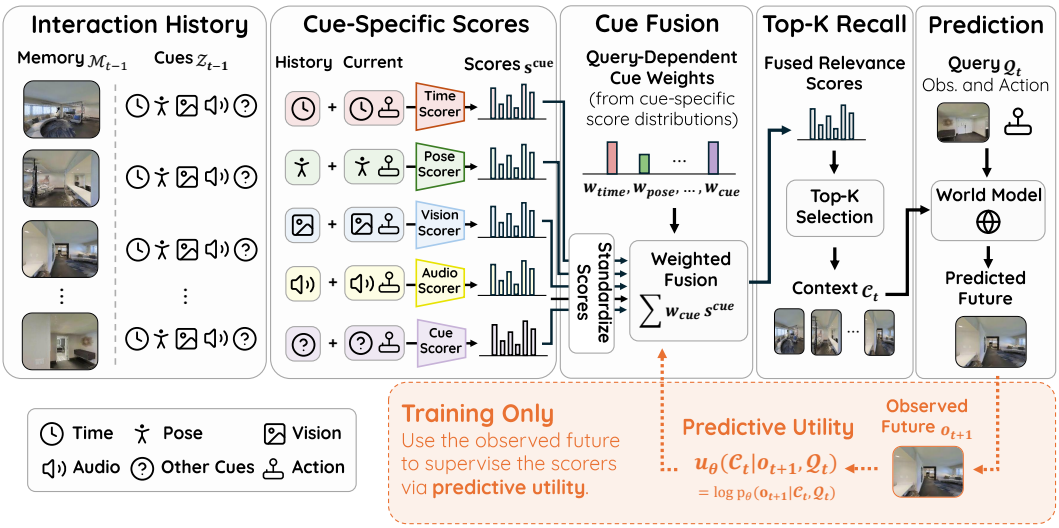


Figure 2: **Overview of Future-Aware Recall (FAR).** FAR scores memories using general retrieval cues $\mathcal{Z}_{t-1}$, adaptively fuses cue-specific relevance, and recalls a compact Top-K context $\mathcal{C}_t$ from episodic memory $\mathcal{M}_{t-1}$ for world-model prediction. During training, the observed future $\boldsymbol{o}_{t+1}$ provides predictive utility to supervise the retriever, while recall remains future-blind at inference.

3.1 Learning Which Memories to Recall

The recall model above is governed by the future-blind relevance score $s_\phi(\mathcal{C}_t \mid \mathcal{Z}_{t-1}, \mathcal{R}_t)$. The key question is how this score should be learned so that highly ranked contexts are actually useful for prediction. Our principle is predictive: *a recalled context is useful when it provides information about the future beyond what is already available in the current prediction query.*

Predictive utility. Let p_{data} denote the environment data-generating distribution over interaction histories, retrieval cues, and future observations. For a recalled context $\mathcal{C}_t \sim r_\phi(\cdot \mid \mathcal{Z}_{t-1}, \mathcal{R}_t)$, its predictive information about the next observation is naturally measured by $I_\phi(\boldsymbol{o}_{t+1}; \mathcal{C}_t \mid \mathcal{Q}_t)$, where the subscript emphasizes that $\mathcal{C}_t$ is selected by the retriever r_ϕ. I_ϕ measures how much the recalled context reduces uncertainty about the future beyond what is already known from $\mathcal{Q}_t$. The following proposition connects this information-theoretic notion of relevance to the world model p_θ.

Proposition 1 (Predictive relevance bound)**.** *For any predictive distribution* $p_\theta(\boldsymbol{o}_{t+1} \mid \mathcal{C}_t, \mathcal{Q}_t)$,

$$I_\phi(\boldsymbol{o}_{t+1}; \mathcal{C}_t \mid \mathcal{Q}_t) \geq \mathbb{E}[\log p_\theta(\boldsymbol{o}_{t+1} \mid \mathcal{C}_t, \mathcal{Q}_t) - \log p_{\text{data}}(\boldsymbol{o}_{t+1} \mid \mathcal{Q}_t)]\,, \tag{3}$$

where the expectation is over training interactions from p_{data} *and* $\mathcal{C}_t \sim r_\phi(\cdot \mid \mathcal{Z}_{t-1}, \mathcal{R}_t)$.

See the derivation in Section B.1. For a fixed training pair $(\boldsymbol{o}_{t+1}, \mathcal{Q}_t)$, the second term in Eq. (3) depends only on p_{data}, and is therefore identical across recalled contexts and independent of the trainable parameters. The remaining context-dependent term hence motivates the *predictive utility*:

$$u_\theta(\mathcal{C}_t \mid \boldsymbol{o}_{t+1}, \mathcal{Q}_t) \coloneqq \log p_\theta(\boldsymbol{o}_{t+1} \mid \mathcal{C}_t, \mathcal{Q}_t). \tag{4}$$

A context has high predictive utility when conditioning on it makes the realized future more likely under the world model. Thus, memory relevance depends on the prediction being made: the same past experience may be highly useful for one query and largely irrelevant for another.

Future-aware predictive supervision. Predictive utility depends on the realized future $\boldsymbol{o}_{t+1}$, which is not available when memories must be recalled at inference time. During training, however, $\boldsymbol{o}_{t+1}$ is observed and reveals which recalled contexts would have best supported the prediction. FAR uses this additional information to construct a future-aware posterior over contexts. Combining the recall distribution in Eq. (1) with the predictive utility in Eq. (4), Bayes' rule gives

$$q_{\theta,\phi}(\mathcal{C}_t \mid \boldsymbol{o}_{t+1}, \mathcal{M}_{t-1}, \mathcal{Z}_{t-1}, \mathcal{R}_t, \mathcal{Q}_t) \propto \exp[s_\phi(\mathcal{C}_t \mid \mathcal{Z}_{t-1}, \mathcal{R}_t) + u_\theta(\mathcal{C}_t \mid \boldsymbol{o}_{t+1}, \mathcal{Q}_t)]\,. \tag{5}$$

This posterior combines two signals: s_ϕ captures how relevant a context appears from information available before the future is observed, while u_θ provides predictive credit from the realized future. Accordingly, $q_{\theta,\phi}$ serves as a future-aware teacher for the future-blind recall distribution. As shown in Section B.2, this posterior provides the maximum-likelihood training target for recall: it favors contexts that better explain the observed future while retaining the relevance already inferred by the future-blind retriever. We therefore train r_ϕ to match this fixed target:

$$\mathcal{L}_{\mathrm{Ret}}(\phi) \coloneqq \mathbb{E}[\mathcal{D}_{\mathrm{KL}}(\mathrm{sg}[q_{\theta,\phi}] \,\|\, r_\phi)]\,, \tag{6}$$

where $\mathrm{sg}[\cdot]$ is the stop-gradient, treating $q_{\theta,\phi}$ as fixed during the retriever update. This transfers predictive credit from the observed future into a retriever that operates without the future at inference.

3.2 Adaptively Learning Which Cues to Trust

The previous subsection specifies what the context relevance score s_ϕ should learn through future-aware predictive credit. At inference, however, this credit is unavailable, so s_ϕ must estimate relevance from the retrieval cues contained in $\mathcal{Z}_{t-1}$ and $\mathcal{R}_t$. FAR is agnostic to the choice of cues: any signal associated with a historical observation that helps locate useful memories can be used for recall. In our experiments, we consider time, pose, visual representations, and audio representations, with the available subset depending on the environment. Because the reliability of these cues can vary across queries, FAR learns both cue-specific relevance and how strongly each available cue should contribute to the final context score.

Cue-specific relevance. Let $\boldsymbol{z}_i^m$ denote the component of $\boldsymbol{z}_i$ corresponding to retrieval cue m. We define the corresponding cue history and retrieval query as $\mathcal{Z}_{t-1}^m \coloneqq (\boldsymbol{z}_1^m, \ldots, \boldsymbol{z}_{t-1}^m)$ and $\mathcal{R}_t^m \coloneqq (\boldsymbol{z}_t^m, \boldsymbol{a}_t)$. For each historical observation $\boldsymbol{o}_i$, cue m produces an observation-level relevance score

$$s_{\phi,i}^m \coloneqq s_\phi^m\big(\boldsymbol{z}_i^m; \mathcal{Z}_{t-1}^m, \mathcal{R}_t^m\big)\,. \tag{7}$$

This score estimates how relevant the corresponding memory appears when viewed through cue m alone. Because different cues may produce scores on different numerical scales, we standardize each cue's scores across the episodic memory and denote the resulting scores by $\widetilde{s}_{\phi,i}^m$.

Adaptive cue fusion. FAR combines the cue-specific scores using query-dependent weights,

$$s_{\phi,i} = \textstyle\sum_m \lambda_{\phi,t}^m \widetilde{s}_{\phi,i}^m, \qquad \sum_m \lambda_{\phi,t}^m = 1, \tag{8}$$

where $\lambda_{\phi,t}^m$ is produced by a learned gate from the cue-specific retrieval signals available for the current query. The fused observation-level scores define the context relevance score in Eq. (1) as

$$s_\phi(\mathcal{C}_t \mid \mathcal{Z}_{t-1}, \mathcal{R}_t) \coloneqq \textstyle\sum_{\boldsymbol{o}_i \in \mathcal{C}_t} s_{\phi,i}. \tag{9}$$

Together, Eqs. (8) and (9) parameterize the recall distribution $r_\phi(\mathcal{C}_t \mid \mathcal{Z}_{t-1}, \mathcal{R}_t)$ used by the future-aware posterior in Eq. (5). For fixed-size recall with $|\mathcal{C}_t| = K$, the maximum-score context is obtained by selecting the K observations with the largest fused relevance scores,

$$\mathcal{C}_{t,\mathrm{TopK}} = \arg\max_{\mathcal{C}_t \subseteq \mathcal{M}_{t-1}} s_\phi(\mathcal{C}_t \mid \mathcal{Z}_{t-1}, \mathcal{R}_t) \quad \text{so that} \quad |\mathcal{C}_t| = K. \tag{10}$$

3.3 Video Diffusion World Model Instantiation

We now instantiate FAR with a video diffusion world model, and we describe the central details here. A precise description of implementation is provided in Section C.1.

Cacheable retrieval encoders. For high-dimensional cues such as vision and audio, cue-specific encoders map historical cues to compact keys and the current cue and action to a query embedding. We contrastively pretrain and freeze the memory-side encoders so that historical keys can be computed once and cached, while lightweight query-side adapters and cue-fusion components remain trainable through FAR.

Diffusion-based predictive utility. Because the exact likelihood in Eq. (4) is expensive for diffusion models, we approximate predictive utility by the negative diffusion prediction loss (Ho et al., 2020; Kingma et al., 2021; Song et al., 2021b; Lai et al., 2025) of the realized future conditioned on each candidate memory, averaged over four diffusion timesteps. Memories that yield lower prediction loss receive greater future-aware credit.

Temporally structured recall. To avoid spending multiple Top-K slots on redundant nearby frames, we partition memory into temporal chunks, retain the highest-scoring observation from each chunk, and recall the K highest-scoring representatives.

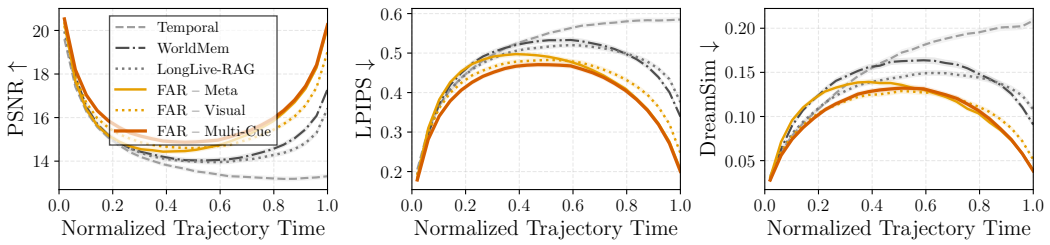


Figure 3: **LoopNav rollout quality.** We report frame-wise PSNR↑, LPIPS↓, and DreamSim↓ between generated and ground-truth return trajectories as a function of normalized trajectory time. Curves show the mean across trajectories, with shaded regions indicating 95% bootstrap confidence intervals. Errors are highest near the middle of the return phase, where relevant observations are typically farthest in time and memory is most sparse.

4 Experiments

We present experiments across three complementary world-modeling settings: *LoopNav* (Lian et al., 2025), *SoundSpaces* (Chen et al., 2020), and *AI2-THOR* (Kolve et al., 2017). LoopNav evaluates consistency in a static Minecraft environment using metadata and visual retrieval cues, while SoundSpaces evaluates realistic indoor navigation where metadata and spatial audio provide complementary signals under partial observability. AI2-THOR evaluates interactive household environments with manipulable objects, where interactions change the world state and memories may become stale. Each trajectory consists of an *exploration phase* and a *return phase*. The world model generates the return-phase video using exploration-phase observations as its memory pool for context retrieval. Fig. 4 shows exploration and return paths on SoundSpaces.

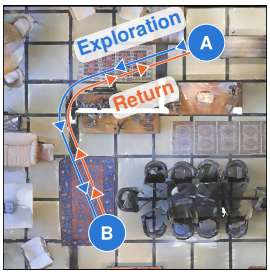


Figure 4: **Illustration of a data trajectory.**

4.1 LoopNav – FAR Outperforms Hand-Designed Recall

We compare against three retrieval baselines: the temporal baseline (Bar et al., 2025), which uses the most recent observations; WorldMem (Xiao et al., 2025), which retrieves frames with high field-of-view (FOV) overlap with the goal; and LongLive-RAG (Hu et al., 2026), which retrieves by reconstructive embedding similarity. We evaluate three variants of our method using metadata cues (time and pose), visual cues, or their adaptive fusion.

As shown in Fig. 3, the temporal baseline performs worst, while WorldMem and LongLive-RAG benefit from geometry- and appearance-based retrieval. At loop closure, our learned visual retriever outperforms LongLive-RAG (17% lower DreamSim), and our learned metadata retriever outperforms WorldMem (19% lower DreamSim), despite using the same respective cue types. Fig. 1 illustrates why: WorldMem retrieves a frame with high FOV overlap but a heavily occluded view of the goal, whereas our learned metadata retriever selects a frame with a clearer predictive view. This highlights a key limitation of hand-crafted relevance rules: cue similarity does not necessarily imply predictive utility. Finally, the *Multi-Cue* variant in Fig. 3, which fuses metadata and visual cues, shows that FAR can exploit complementary retrieval signals. Its rollout error closely follows the better of the metadata-only and visual-only variants across the horizon, suggesting that adaptive fusion emphasizes whichever cue is more informative for the current prediction.

4.2 SoundSpaces – FAR Adaptively Learns Which Cues to Trust

We next evaluate FAR on SoundSpaces, where we construct an indoor navigation dataset augmented with spatial audio. We generate two corpora that differ in the density of available episodic memories: one performs periodic 360° scans during exploration, while the other scans only at the exploration endpoints. The former thus provides substantially denser historical coverage from which the world model can retrieve. We stratify evaluation by return-phase length, i.e., the physical distance traversed during the trajectory over which rollout quality is measured. Longer return trajectories are generally more challenging, as they require prediction through more rooms, corridors, and viewpoint changes.

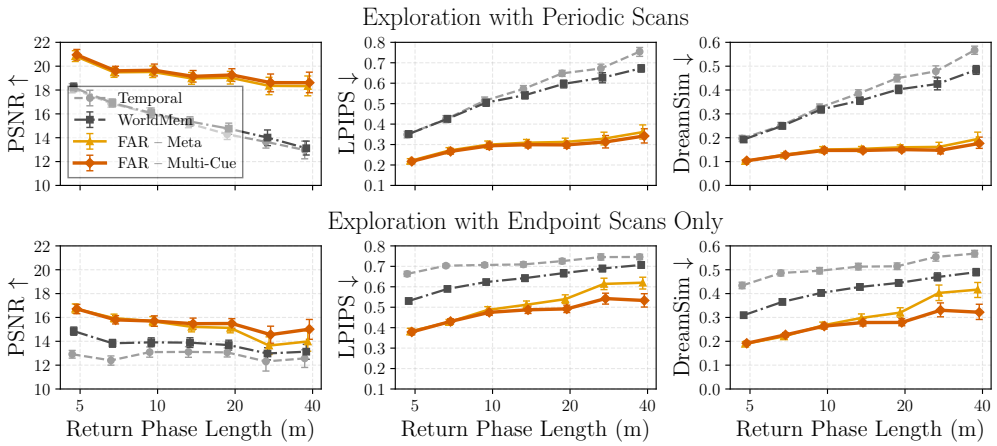


Figure 5: **SoundSpaces rollout quality.** We report PSNR, LPIPS, and DreamSim, where the error is averaged over predicted frames along return trajectory. Thus, the x-axis denotes the length of the trajectory over which the rollout is evaluated, *not the rollout horizon*. **Top:** results on the corpus where the agent performs periodic 360° scans during exploration. **Bottom:** results on the corpus where the agent scans at the endpoints, i.e., at the beginning and end of exploration.

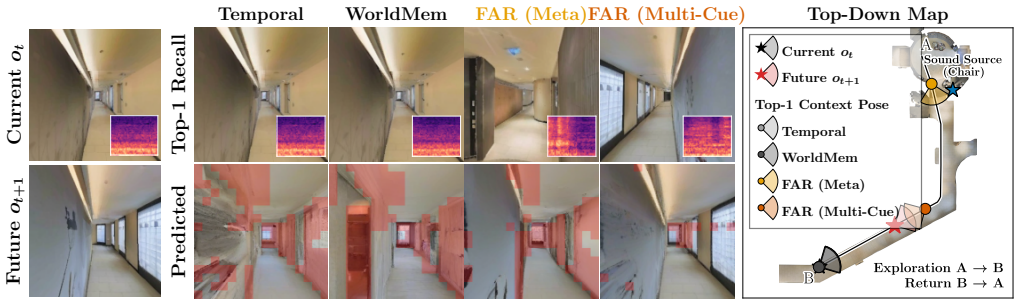


Figure 6: **Retrieval comparison on SoundSpaces.** Given the observation and goal view, we visualize the top-1 context retrieved by each method and the generated goal frame, with regions of large error overlaid in **red**. The spectrograms show the audio cue associated with each state. The map shows the exploration and return trajectories with the memories selected by each method.

Fig. 5 shows that FAR consistently outperforms the temporal baseline and WorldMem across both corpora. With periodic scans, metadata alone provides a strong retrieval signal, and Multi-Cue fusion of metadata and audio yields modest gains. Under the sparser endpoint-scan setting, however, the advantage of Multi-Cue grows with return-phase length, particularly in LPIPS and DreamSim. This suggests that metadata often suffices when relevant visual memories are densely available, whereas audio provides a valuable complementary cue for long trajectories or sparse histories.

Fig. 6 illustrates how audio can complement metadata when geometric relevance alone is insufficient. The agent is traversing a corridor and must predict a view farther down. Temporal retrieval selects the most recent observation, which looks in the correct direction but provides limited information about the distant portion of the corridor. WorldMem selects the same context because it has high FOV overlap with the goal pose, and consequently suffers from the limitation. FAR with metadata retrieves an observation taken farther along the trajectory and oriented toward the goal, but the view is obstructed by the corridor geometry. Incorporating audio cue instead retrieves a context with a clearer view of the region around the goal, leading to a more faithful prediction. The inset spectrograms show the audio associated with each retrieved state, highlighting how spatial audio can provide a complementary retrieval cue when pose and visual geometry alone are ambiguous.

4.3 AI2-THOR – FAR Recalls the Right State as the World Changes

We next evaluate FAR in AI2-THOR environments where interactions change the state of the world. During exploration, the agent randomly moves objects between surfaces and containers, producing multiple observations of the same location that may correspond to conflicting world states. During the return phase, the agent revisits these locations and must generate observations consistent with

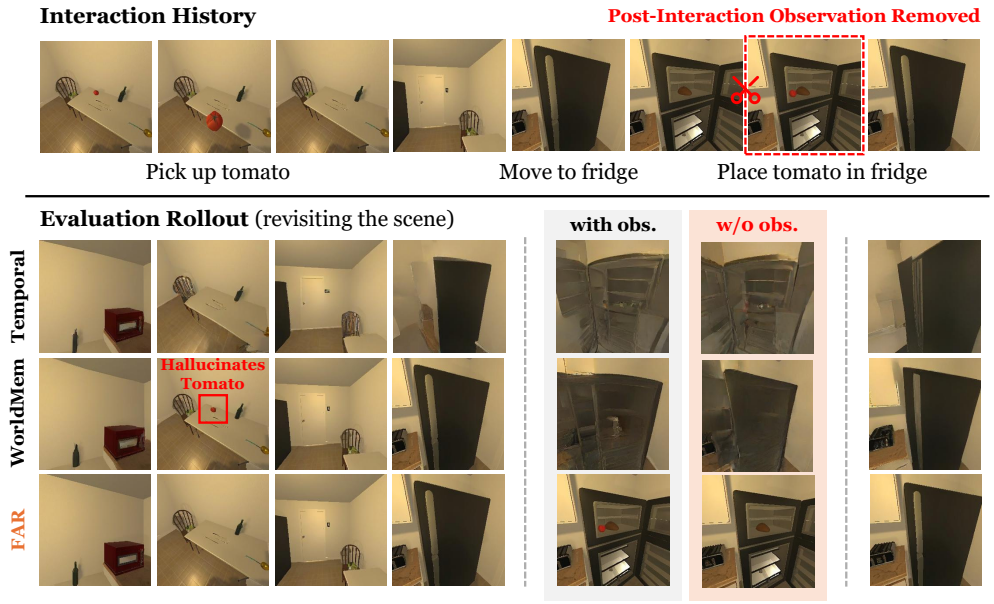


Figure 7: **History-dependent counterfactual prediction in AI2-THOR.** We compare two trajectories with the same current closed-fridge observation and opening action but different histories: one history contains the event of placing a tomato inside the fridge, while the other does not.

the state established by the preceding interactions. This setting is challenging for retrieval based primarily on recency or geometry, since a spatially well-matched memory may depict a stale state.

State rendering accuracy. We evaluate whether the generated rollout reflects the current world state rather than a stale state from earlier in the trajectory. Specifically, we report the fraction of changed states satisfying $d(\boldsymbol{o}_{\mathrm{GT}}, \boldsymbol{o}_{\mathrm{Pred}}) < d(\boldsymbol{o}_{\mathrm{GT}}, \boldsymbol{o}_{\mathrm{Stale}})$, where $\boldsymbol{o}_{\mathrm{GT}}$ is the ground-truth frame, $\boldsymbol{o}_{\mathrm{Pred}}$ is the predicted rollout frame, $\boldsymbol{o}_{\mathrm{Stale}}$ is a pose-matched observation depicting the pre-interaction state, and d denotes LPIPS on a crop around the changed region.

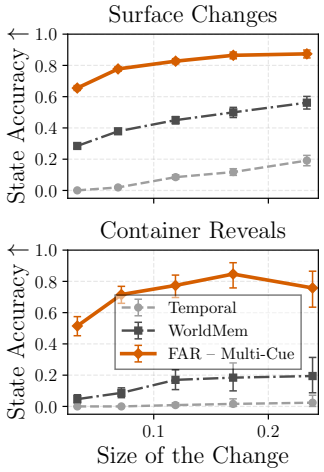


Figure 8: **State accuracy under object manipulation.**

Fig. 8 stratifies accuracy by state-change magnitude, $d(\boldsymbol{o}_{\mathrm{GT}}, \boldsymbol{o}_{\mathrm{Stale}})$. We consider two cases: *surface changes*, where the updated state is directly visible upon revisit, and *container reveals*, where it becomes observable only after opening a container. Multi-Cue FAR, fusing metadata and vision, substantially outperforms temporal and geometry-based recall across change magnitudes in both settings. The advantage is especially pronounced for container reveals, where the current observation contains little information about hidden contents and accurate prediction therefore depends strongly on recalling the relevant interaction history. Performance decreases for all methods when current and stale states differ only locally, consistent with the limited sensitivity of the diffusion prediction objective to small visual changes.

History-dependent counterfactual prediction. Fig. 7 shows a paired counterfactual example in which the current observation and action are held fixed, while the preceding interaction history differs. In one history, the agent places a tomato inside the refrigerator; in the other, it does not. When the agent later observes the same closed refrigerator and executes the same opening action, FAR generates different futures consistent with the corresponding histories, rendering the tomato inside the refrigerator only when it was previously placed there. This shows that the predicted future is not determined by the current observation alone, but depends on recalling the relevant past interaction. In contrast, Temporal and WorldMem fail to preserve the interaction-dependent state, with WorldMem additionally hallucinating the tomato at its stale table location as a result of prioritizing stale contexts with high FOV overlap over up-to-date contexts.

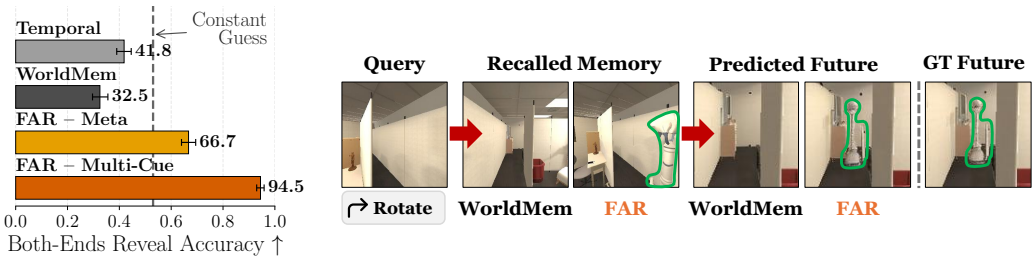


Figure 9: **Off-scene dynamics prediction. Left:** trajectory-level accuracy for predicting the future location of the independently moving agent A_2. **Right:** qualitative example showing that FAR recalls a motion-informative episode containing A_2, and predicts the correct future state with A_2. WorldMem retrieves a spatially relevant but dynamically uninformative memory, failing to render A_2. All A_2 appearances are outlined in **green** for visual aid.

Off-scene dynamics prediction. We test whether FAR can predict off-scene dynamics in a two-agent corridor setting. A_1 is the observer, while A_2 patrols back and forth and is only intermittently visible through a doorway. The task is to predict A_2's location when A_1 exits the room and looks down the corridor. To capture motion, each recalled context consists of two observations separated by 1.5 seconds. We compare FAR using only time and pose metadata with a *Multi-Cue* variant that fuses metadata with an A_2 observation cue. As shown in Fig. 9, Temporal and WorldMem achieve 41.9% and 32.4% accuracy, while FAR with metadata alone reaches 67.1%, indicating that future-aware training can learn when informative crossings are likely to occur. Multi-Cue further improves accuracy to 94.6% by more precisely identifying motion-informative episodes. Qualitatively, WorldMem retrieves a geometrically relevant but dynamically uninformative memory, whereas Multi-Cue FAR recalls an episode containing A_2 and predicts its future off-scene location correctly.

4.4 Ablation Study

Table 1 analyzes each component of FAR. A frozen pretrained vision encoder provides a strong retrieval signal, substantially outperforming baselines across three rollout metrics. Adapting the query representation with MLP further improves performance from 16.25 to 16.48 PSNR, with gains in LPIPS and DreamSim, indicating that visual similarity benefits from task-specific adaptation toward predictive relevance. Incorporating metadata cues yields a larger improvement: fixed fusion with $\lambda = 0.5$ increases PSNR to 16.91. Finally, learning the fusion weights achieves the best performance across all metrics, showing that metadata and visual cues are complementary and adaptively weighting their reliability is preferable to uniform fusion.

Retrieval	PSNR↑	LPIPS↓	DreamSim↓
Temporal (NWM)	$13.229_{\pm 0.114}$	$0.582_{\pm 0.007}$	$0.200_{\pm 0.004}$
LongLive-RAG	$14.847_{\pm 0.121}$	$0.463_{\pm 0.006}$	$0.133_{\pm 0.003}$
WorldMem	$15.372_{\pm 0.112}$	$0.448_{\pm 0.006}$	$0.130_{\pm 0.003}$
FAR with Vision Cue			
Enc. Pretrained	$16.248_{\pm 0.112}$	$0.390_{\pm 0.006}$	$0.100_{\pm 0.003}$
+ MLP Adapter	$16.480_{\pm 0.108}$	$0.374_{\pm 0.006}$	$0.091_{\pm 0.002}$
+ Meta, $\lambda = 0.5$	$16.912_{\pm 0.117}$	$0.353_{\pm 0.005}$	$0.088_{\pm 0.002}$
+ Meta, λ learned	$\mathbf{17.267_{\pm 0.116}}$	$\mathbf{0.337_{\pm 0.005}}$	$\mathbf{0.082_{\pm 0.002}}$

Table 1: **Ablation on LoopNav.** We report average loop closure errors for various configurations.

5 Limitations and Conclusion

We introduced FAR, which learns which episodic memories support prediction and which retrieval cues to trust. FAR uses the observed future for predictive credit during training, while remaining future-blind at inference. By learning cue-specific relevance and query-dependent cue weights, FAR improves long-horizon prediction across navigation, multi-cue, and changing-state settings, supporting predictive utility as a principled basis for episodic memory recall in world models.

Our work focuses on recall from an external episodic memory, leaving memory writing, compression, forgetting, and higher-order interactions among recalled memories to future work. FAR also requires informative retrieval cues and additional training-time computation for predictive-utility evaluation. Our diffusion-loss utility may underweight semantically important local changes; alternative task-aware utility functions are a promising direction. Finally, our experiments use controlled simulations, motivating evaluation in richer real-world settings and integration with learned memory formation and persistent-state representations.

AI USE STATEMENT

Generative AI tools were used during the preparation of this work to support language editing, such as correcting grammar, refining phrasing, and enhancing the clarity and readability of text originally written by the authors. We also used AI-based tools to assist with finding related works, code inspection and debugging, as well as to explore initial designs for figures. All AI-assisted outputs and modifications were carefully reviewed by the authors, who assume full responsibility for the final content of this work.

ETHICS STATEMENT

Our experiments use established simulation environments and existing datasets, and we do not collect new human-subject data. FAR is intended as a general mechanism for selective access to an agent's past observations, but persistent episodic memory could raise privacy concerns if deployed in real-world systems that record people or sensitive environments. Such deployments should therefore consider appropriate consent, data-retention and deletion policies, access control, and safeguards against unintended use of stored observations. Our experiments are limited to controlled Minecraft, Matterport3D, and AI2-THOR environments and do not evaluate FAR for surveillance, identification, or other applications involving sensitive personal information.

REPRODUCIBILITY STATEMENT

We provide the mathematical formulation of FAR and its training objective in Sections 3.1 and 3.2, with derivations and the candidate-level approximation given in the supplementary material. Section C.1 specifies the video diffusion world-model and retrieval implementation, while Sections C.2 to C.4 document the datasets, preprocessing, model and optimization hyperparameters, training protocols, and evaluation procedures used for each experimental setting. We report dataset splits and evaluation operating points explicitly, including differences between training and inference context and memory-pool sizes, and use shared evaluation seeds across compared methods. The anonymous project page provides additional qualitative results and demonstrations.

REFERENCES

Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for World Modeling: Visual Details Matter in Atari. In *NeurIPS*, 2024.

Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation World Models. In *CVPR*, 2025.

Jake Bruce et al. Genie: Generative Interactive Environments. In *ICML*, 2024.

Shengqu Cai, Ceyuan Yang, Lvmin Zhang, Yuwei Guo, Junfei Xiao, Ziyan Yang, Yinghao Xu, Zhenheng Yang, Alan Yuille, Leonidas Guibas, Maneesh Agrawala, Lu Jiang, and Gordon Wetzstein. Mixture of Contexts for Long Video Generation. In *ICLR*, 2026.

Ishaan Preetam Chandratreya, David Charatan, Basile Van Hoorick, Sergey Zakharov, Vitor Guizilini, Phillip Isola, and Vincent Sitzmann. MilliVid: Hierarchical Latents for Long-Range Consistency in Video Generation. *arXiv preprint arXiv:2606.09056*, 2026.

Changan Chen, Unnat Jain, Carl Schissler, Sebastia Vicenc Amengual Gari, Ziad Al-Halah, Vamsi Krishna Ithapu, Philip Robinson, and Kristen Grauman. SoundSpaces: Audio-Visual Navigation in 3D Environments. In *ECCV*, 2020.

Taiye Chen, Xun Hu, Zihan Ding, and Chi Jin. VRAG: Learning World Models for Interactive Video Generation. In *NeurIPS*, 2025.

Decart, Julian Quevedo, Quinn McIntyre, Spruce Campbell, Xinlei Chen, and Robert Wachen. Oasis: A Universe in a Transformer, 2024. URL https://oasis-model.github.io/.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*, 2021.

Samuel Garcin, Thomas Walker, Steven McDonagh, Tim Pearce, Hakan Bilen, Tianyu He, Kaixin Wang, and Jiang Bian. Beyond Pixel Histories: World Models with Persistent 3D State. In *ICML*, 2026.

David Ha and Jürgen Schmidhuber. Recurrent World Models Facilitate Policy Evolution. In *NeurIPS*, 2018.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning Latent Dynamics for Planning from Pixels. In *ICML*, 2019.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to Control: Learning Behaviors by Latent Imagination. In *ICLR*, 2020.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering Atari with Discrete World Models. In *ICLR*, 2021.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640:647–653, 2025.

Jinkun Hao, Mingda Jia, Ruiyan Wang, Xihui Liu, Ran Yi, Lizhuang Ma, Jiangmiao Pang, and Xudong Xu. EgoSim: Egocentric World Simulator for Embodied Interaction Generation. In *ECCV*, 2026.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models. *NeurIPS*, 2020.

Qixin Hu, Shuai Yang, Wei Huang, Song Han, and Yukang Chen. LongLive-RAG: A General Retrieval-Augmented Framework for Long Video Generation. *arXiv preprint arXiv:2606.02553*, 2026.

Dongwon Kim, Gawon Seo, Jinsung Lee, Minsu Cho, and Suha Kwak. Planning in 8 Tokens: A Compact Discrete Tokenizer for Latent World Model. In *CVPR*, 2026.

Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational Diffusion Models. *NeurIPS*, 2021.

Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Matt Deitke, Kiana Ehsani, Daniel Gordon, Yuke Zhu, Aniruddha Kembhavi, Abhinav Gupta, and Ali Farhadi. AI2-THOR: An Interactive 3D Environment for Visual AI. *arXiv preprint arXiv:1712.05474*, 2017.

Chieh-Hsin Lai, Yang Song, Dongjun Kim, Yuki Mitsufuji, and Stefano Ermon. The Principles of Diffusion Models. *arXiv preprint arXiv:2510.21890*, 2025.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *NeurIPS*, 2020.

Jia Li, Han Yan, Yihang Chen, Siqi Li, Xibin Song, Yifu Wang, Jianfei Cai, Tien-Tsin Wong, and Pan Ji. I3DM: Implicit 3D-aware Memory Retrieval and Injection for Consistent Video Scene Generation. *arXiv preprint arXiv:2603.23413*, 2026.

Kewei Lian, Shaofei Cai, Yitao Liang, and Anji Liu. LoopNav: Benchmarking Spatial Consistency in World Models. *arXiv preprint arXiv:2505.22976*, 2025.

William Peebles and Saining Xie. Scalable Diffusion Models with Transformers. In *ICCV*, 2023.

Zhan Peng, Jie Ma, Huiqiang Sun, Chong Gao, Zhijie Xue, Zhiyu Pan, Zhiguo Cao, Jun Liang, and Jing Li. Compression and Retrieval: Implicit Memory Retrieval for Video World Models. *arXiv preprint arXiv:2606.23105*, 2026.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis. In *ICLR*, 2024.

Devendra Singh Sachan, Siva Reddy, William Hamilton, Chris Dyer, and Dani Yogatama. End-to-End Training of Multi-Document Reader and Retriever for Open-Domain Question Answering. In *NeurIPS*, 2021.

Junyoung Seo, Hyunwook Choi, Minkyung Kwon, Jinhyeok Choi, Siyoon Jin, Gayoung Lee, Junho Kim, JoungBin Lee, Geonmo Gu, Dongyoon Han, Sangdoo Yun, Seungryong Kim, and Jin-Hwa Kim. Grounding World Simulation Models in a Real-World Metropolis. In *ECCV*, 2026.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising Diffusion Implicit Models. In *ICLR*, 2021a.

Yang Song, Conor Durkan, Iain Murray, and Stefano Ermon. Maximum Likelihood Training of Score-based Diffusion Models. In *NeurIPS*, 2021b.

Fan Wang, Zhiyuan Chen, Yuxuan Zhong, Sunjian Zheng, Pengtao Shao, Bo Yu, Shaoshan Liu, Jianan Wang, Ning Ding, Yang Cao, and Yu Kang. Context and Diversity Matter: The Emergence of In-Context Learning in World Models. In *ICLR*, 2026a.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual Geometry Grounded Transformer. In *CVPR*, 2025.

Zile Wang et al. Matrix-Game 3.0: Real-Time and Streaming Interactive World Model with Long-Horizon Memory. *arXiv preprint arXiv:2604.08995*, 2026b.

Tong Wu, Shuai Yang, Ryan Po, Yinghao Xu, Ziwei Liu, Dahua Lin, and Gordon Wetzstein. Video World Models with Long-term Spatial Memory. In *NeurIPS*, 2025.

Zeqi Xiao, Yushi Lan, Yifan Zhou, Wenqi Ouyang, Shuai Yang, Yanhong Zeng, and Xingang Pan. WorldMem: Long-term Consistent World Simulation with Memory. In *NeurIPS*, 2025.

Jiwen Yu, Jianhong Bai, Yiran Qin, Quande Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Xihui Liu. Context as Memory: Scene-Consistent Interactive Long Video Generation with Memory Retrieval. In *SIGGRAPH Asia*, 2025.

Jiwen Yu, Jianxiong Gao, Jianhong Bai, Yiran Qin, Kaiyi Huang, Quande Liu, Xintao Wang, Pengfei Wan, Kun Gai, and Xihui Liu. MemLearner: Learning to Query Context Memory for Video World Models. In *ECCV*, 2026.

Lvmin Zhang, Shengqu Cai, Muyang Li, Gordon Wetzstein, and Maneesh Agrawala. Frame Context Packing and Drift Prevention in Next-Frame-Prediction Video Diffusion Models. In *NeurIPS*, 2025.

Jinjing Zhao, Fangyun Wei, Zhening Liu, Hongyang Zhang, Chang Xu, and Yan Lu. Spatia: Video Generation with Updatable Spatial Memory. In *CVPR*, 2026.

Haoyi Zhu, Haozhe Liu, Yuyang Zhao, Tian Ye, Junsong Chen, Jincheng Yu, Tong He, Song Han, and Enze Xie. SANA-WM: Efficient minute-scale world modeling with hybrid linear diffusion transformer. *arXiv preprint arXiv:2605.15178*, 2026.

Supplementary Material

A Additional Discussion of Related Work

Method	World Model	External Retriever	Prediction Supervised	Adaptive Cue Fusion	Cue Options
FramePack (Zhang et al., 2025)	✗	✓	✗	✗	Time, Vision
LongLive-RAG (Hu et al., 2026)	✗	✓	✗	✗	Vision
WorldMem (Xiao et al., 2025)	✓	✓	✗	✗	Time, Pose
Context-as-Memory (Yu et al., 2025)	✓	✓	✗	✗	Pose
VRAG (Chen et al., 2025)	✓	✓	✗	✗	Pose
SPMem (Wu et al., 2025)	✓	✓	✗	✗	Pose
I3DM (Li et al., 2026)	✓	✓	✗	✗	Pose, Vision
Matrix-Game 3.0 (Wang et al., 2026b)	✓	✓	✗	✗	Pose
Mixture-of-Contexts (Cai et al., 2026)	✗	✗	✓	✗	Vision, Text
MemLearner (Yu et al., 2026)	✓	✗	✓	✗	Pose, Vision
Compression-and-Retrieval (Peng et al., 2026)	✓	✗	✓	✗	Pose, Vision
Ours	✓	✓	✓	✓	Time, Pose, Vision, Audio

Table 2: **Conceptual comparison of episodic memory recall for video and world models. World Model** indicates whether the method is designed for action-conditioned future prediction; **External Retriever** whether memory selection is performed outside the generator; **Prediction Supervised** whether retrieval is optimized using the downstream prediction objective; **Adaptive Cue Fusion** whether the retriever dynamically weights multiple retrieval cues for each query rather than using a fixed cue or predefined combination; and **Cue Options** lists the information sources used to determine memory relevance in the respective papers.

Episodic memory retrieval for world models. Recent video and world models augment limited context windows with external memories that preserve historical frames or latent representations for later retrieval (Xiao et al., 2025; Yu et al., 2025; Wu et al., 2025; Li et al., 2026; Hu et al., 2026). As summarized in Section A, existing external retrievers typically rely on fixed notions of relevance, including temporal proximity, pose or field-of-view overlap, geometry-aware similarity, or learned visual embeddings. These approaches demonstrate that selectively revisiting non-local observations can improve long-horizon consistency, but the retrieval rule itself is generally not optimized using the downstream world-model prediction objective. FAR instead learns memory relevance directly from predictive utility and further learns query-dependent fusion over heterogeneous temporal, spatial, visual, and auditory cues. Thus, FAR learns both *which memories are predictively useful* and *which cues should be trusted to find them*.

Internal long-context memory. A complementary line retains long-range history within the generator and learns how to route or compress it during prediction. Mixture of Contexts sparsely routes queries to informative history chunks, MemLearner queries context using the video generator itself, and CaR combines attention-based memory retrieval with context compression (Cai et al., 2026; Yu et al., 2026; Peng et al., 2026). Recent world models also use recurrent or hybrid linear attention to propagate information over long sequences, including L2World and SANA-WM (Wang et al., 2026a; Zhu et al., 2026). These internal mechanisms and external episodic recall are complementary: the former efficiently summarize or route long histories within the model, while the latter preserves individually addressable observations that can be selectively revisited.

Persistent-state representations. Rather than retrieving individual observations, another family of approaches maintains an explicit or latent state that is updated as the environment evolves. PERSIST evolves a latent 3D scene representation, Spatia maintains an updatable 3D point-cloud memory, and EgoSim continuously updates a 3D scene state across embodied interactions (Garcin et al., 2026; Zhao et al., 2026; Hao et al., 2026). Persistent state provides a compact, viewpoint-independent representation of geometry and world changes, whereas episodic memory preserves past observations as individually recoverable evidence. The two are complementary: persistent state summarizes the model's current belief about the world, while episodic recall can recover specific past evidence when that state is incomplete or uncertain.

Learned retrieval and retrieval-augmented generation. Retrieval-augmented generation (RAG) treats retrieved documents as latent context and learns retrieval jointly with downstream generation (Lewis et al., 2020; Sachan et al., 2021). In particular, EMDR^2 uses the reader likelihood of the observed target to provide posterior supervision for discrete document retrieval (Sachan et al., 2021). FAR draws on this principle but applies it to an agent's own episodic history: the realized future provides predictive credit for candidate memories, and this supervision trains a retriever that operates without access to the future at inference. Unlike document RAG, FAR additionally learns to combine heterogeneous temporal, spatial, visual, and auditory cues to locate useful memories.

Long-horizon world models. A broader line of work targets long-horizon consistency by changing how context or state is represented rather than explicitly learning episodic recall. FramePack compresses frame histories according to importance and introduces anti-drifting mechanisms, while MilliVid uses hierarchical latents and coarse-to-fine rollout to preserve long-range structure efficiently (Zhang et al., 2025; Chandratreya et al., 2026). Seoul World Model instead repeatedly regrounds generation using nearby real-world observations to stabilize long trajectories (Seo et al., 2026). These approaches address complementary challenges of context scaling, drift, and external grounding; FAR focuses specifically on learning which observations from the agent's own history are useful for the current prediction.

B Additional Theoretical Details

In this section, we provide more theoretical and practical details about FAR. First, we provide the proof of Proposition 1 in Section B.1. We summarize the overall pseudo code as Algorithm 1.

B.1 Derivation of Predictive Relevance Bound

Proof. Consider the joint distribution induced by the environment data distribution and the retriever,

$$p_\phi(\boldsymbol{o}_{t+1}, \mathcal{M}_{t-1}, \mathcal{Z}_{t-1}, \mathcal{R}_t, \mathcal{Q}_t, \mathcal{C}_t) \coloneqq p_{\text{data}}(\boldsymbol{o}_{t+1}, \mathcal{M}_{t-1}, \mathcal{Z}_{t-1}, \mathcal{R}_t, \mathcal{Q}_t)\, r_\phi(\mathcal{C}_t \mid \mathcal{Z}_{t-1}, \mathcal{R}_t).$$

The subscript ϕ reflects that the recalled context $\mathcal{C}_t$ is selected by the retriever r_ϕ.

Introducing the retriever does not change the marginal distribution of the future and prediction query. Because $r_\phi(\cdot \mid \mathcal{Z}_{t-1}, \mathcal{R}_t)$ is normalized,

$$\begin{aligned} p_\phi(\boldsymbol{o}_{t+1}, \mathcal{Q}_t) &= \int p_{\text{data}}(\boldsymbol{o}_{t+1}, \mathcal{M}_{t-1}, \mathcal{Z}_{t-1}, \mathcal{R}_t, \mathcal{Q}_t) \sum_{\mathcal{C}_t} r_\phi(\mathcal{C}_t \mid \mathcal{Z}_{t-1}, \mathcal{R}_t)\, \mathrm{d}\mathcal{M}_{t-1}\, \mathrm{d}\mathcal{Z}_{t-1}\, \mathrm{d}\mathcal{R}_t \\ &= p_{\text{data}}(\boldsymbol{o}_{t+1}, \mathcal{Q}_t). \end{aligned}$$

Hence,

$$p_\phi(\boldsymbol{o}_{t+1} \mid \mathcal{Q}_t) = p_{\text{data}}(\boldsymbol{o}_{t+1} \mid \mathcal{Q}_t).$$

By the definition of conditional mutual information,

$$I_\phi(\boldsymbol{o}_{t+1}; \mathcal{C}_t \mid \mathcal{Q}_t) = \mathbb{E}_{p_\phi}\left[\log \frac{p_\phi(\boldsymbol{o}_{t+1} \mid \mathcal{C}_t, \mathcal{Q}_t)}{p_\phi(\boldsymbol{o}_{t+1} \mid \mathcal{Q}_t)}\right] = \mathbb{E}_{p_\phi}\left[\log p_\phi(\boldsymbol{o}_{t+1} \mid \mathcal{C}_t, \mathcal{Q}_t) - \log p_{\text{data}}(\boldsymbol{o}_{t+1} \mid \mathcal{Q}_t)\right].$$

Introducing any predictive distribution $p_\theta(\boldsymbol{o}_{t+1} \mid \mathcal{C}_t, \mathcal{Q}_t)$ and adding and subtracting its log-density gives

$$\begin{aligned} I_\phi(\boldsymbol{o}_{t+1}; \mathcal{C}_t \mid \mathcal{Q}_t) = \mathbb{E}_{p_\phi}&\left[\log p_\theta(\boldsymbol{o}_{t+1} \mid \mathcal{C}_t, \mathcal{Q}_t) - \log p_{\text{data}}(\boldsymbol{o}_{t+1} \mid \mathcal{Q}_t)\right] \\ &+ \mathbb{E}_{p_\phi(\mathcal{C}_t, \mathcal{Q}_t)}\left[\mathcal{D}_{\text{KL}}(p_\phi(\boldsymbol{o}_{t+1} \mid \mathcal{C}_t, \mathcal{Q}_t) \,\|\, p_\theta(\boldsymbol{o}_{t+1} \mid \mathcal{C}_t, \mathcal{Q}_t))\right]. \end{aligned}$$

The second term is nonnegative by the nonnegativity of KL divergence. Therefore,

$$I_\phi(\boldsymbol{o}_{t+1}; \mathcal{C}_t \mid \mathcal{Q}_t) \geq \mathbb{E}_{\substack{(\boldsymbol{o}_{t+1}, \mathcal{M}_{t-1}, \mathcal{Z}_{t-1}, \mathcal{R}_t, \mathcal{Q}_t) \sim p_{\text{data}} \\ \mathcal{C}_t \sim r_\phi(\cdot \mid \mathcal{Z}_{t-1}, \mathcal{R}_t)}}\left[\log p_\theta(\boldsymbol{o}_{t+1} \mid \mathcal{C}_t, \mathcal{Q}_t) - \log p_{\text{data}}(\boldsymbol{o}_{t+1} \mid \mathcal{Q}_t)\right],$$

which is exactly Eq. (3).

Moreover, the decomposition above shows that the gap in the bound is

$$\mathbb{E}_{p_\phi(\mathcal{C}_t, \mathcal{Q}_t)}\left[\mathcal{D}_{\text{KL}}(p_\phi(\boldsymbol{o}_{t+1} \mid \mathcal{C}_t, \mathcal{Q}_t) \,\|\, p_\theta(\boldsymbol{o}_{t+1} \mid \mathcal{C}_t, \mathcal{Q}_t))\right].$$

Thus, the bound is tight when the predictive world model matches the conditional distribution of the future induced by the data distribution and retriever, almost surely over $(\mathcal{C}_t, \mathcal{Q}_t)$. □

B.2 Latent-Context Maximum-Likelihood View of the Retriever Update

We show that the future-aware posterior and retriever objective in Eqs. (5) and (6) follow directly from maximum likelihood in the latent-context predictive model Eq. (2). This also clarifies why FAR matches a posterior that combines predictive utility with the current recall distribution, rather than a target based on predictive utility alone.

For readability, within this subsection we abbreviate

$$r_\phi(\mathcal{C}_t) \coloneqq r_\phi(\mathcal{C}_t \mid \mathcal{Z}_{t-1}, \mathcal{R}_t), \quad \text{and} \quad u_\theta(\mathcal{C}_t) \coloneqq u_\theta(\mathcal{C}_t \mid \boldsymbol{o}_{t+1}, \mathcal{Q}_t),$$

and let

$$\mathfrak{C}_t \coloneqq \{\mathcal{C}_t \subseteq \mathcal{M}_{t-1} : |\mathcal{C}_t| = K\}$$

denote the set of all valid recalled context sets, i.e., all subsets of the episodic memory containing exactly K historical observations. For a fixed training example, Eq. (2) gives

$$p_{\theta,\phi}(\boldsymbol{o}_{t+1} \mid \cdots) = \sum_{\mathcal{C}_t \in \mathfrak{C}_t} r_\phi(\mathcal{C}_t)\, p_\theta(\boldsymbol{o}_{t+1} \mid \mathcal{C}_t, \mathcal{Q}_t) = \sum_{\mathcal{C}_t \in \mathfrak{C}_t} r_\phi(\mathcal{C}_t)\, \exp u_\theta(\mathcal{C}_t).$$

The corresponding posterior over recalled contexts is

$$q_{\theta,\phi}(\mathcal{C}_t) \coloneqq \frac{r_\phi(\mathcal{C}_t) \exp u_\theta(\mathcal{C}_t)}{\sum_{\mathcal{C}'_t \in \mathfrak{C}_t} r_\phi(\mathcal{C}'_t) \exp u_\theta(\mathcal{C}'_t)}.$$

For any distribution q over $\mathfrak{C}_t$, a direct rearrangement gives the standard latent-variable decomposition

$$\log p_{\theta,\phi}(\boldsymbol{o}_{t+1} \mid \cdots) = \mathbb{E}_{\mathcal{C}_t \sim q}\left[u_\theta(\mathcal{C}_t)\right] - \mathcal{D}_{\mathrm{KL}}(q \,\|\, r_\phi) + \mathcal{D}_{\mathrm{KL}}(q \,\|\, q_{\theta,\phi})\,. \tag{11}$$

Since the final term is nonnegative, the bound is maximized at $q = q_{\theta,\phi}$, or equivalently,

$$q_{\theta,\phi} = \arg\max_q \; \left\{\mathbb{E}_{\mathcal{C}_t \sim q}\left[u_\theta(\mathcal{C}_t)\right] - \mathcal{D}_{\mathrm{KL}}(q \,\|\, r_\phi)\right\}.$$

Thus, the future-aware posterior increases expected predictive utility while remaining anchored to the relevance inferred by the future-blind retriever.

Because $r_\phi(\mathcal{C}_t) \propto \exp s_\phi(\mathcal{C}_t \mid \mathcal{Z}_{t-1}, \mathcal{R}_t)$, the optimizer has the form

$$q_{\theta,\phi}(\mathcal{C}_t) \propto \exp[s_\phi(\mathcal{C}_t \mid \mathcal{Z}_{t-1}, \mathcal{R}_t) + u_\theta(\mathcal{C}_t \mid \boldsymbol{o}_{t+1}, \mathcal{Q}_t)]\,,$$

which recovers the future-aware posterior in Eq. (5). The realized future therefore updates the relevance already inferred from the available retrieval cues, rather than replacing it.

This also explains why we do not construct the target from predictive utility alone. Normalizing only predictive utility would give

$$\bar{q}_\theta(\mathcal{C}_t) \propto \exp u_\theta(\mathcal{C}_t).$$

For finite $\mathfrak{C}_t$, this distribution solves

$$\bar{q}_\theta = \arg\max_q \; \{\mathbb{E}_q[u_\theta] + \mathcal{H}(q)\} = \arg\max_q \; \{\mathbb{E}_q[u_\theta] - \mathcal{D}_{\mathrm{KL}}(q \,\|\, \mathrm{Unif}(\mathfrak{C}_t))\}\,,$$

up to the constant $\log |\mathfrak{C}_t|$. Hence, a utility-only target implicitly replaces the query-dependent recall distribution r_ϕ with a uniform prior over contexts, discarding the relevance already inferred from the current query and retrieval cues. For example, if all contexts have equal predictive utility, then $q_{\theta,\phi} = r_\phi$, so the realized future provides no evidence for changing the current recall preference. In contrast, $\bar{q}_\theta$ becomes uniform and would alter the retriever despite receiving no differential predictive evidence. More generally, $q_{\theta,\phi} \propto r_\phi \exp u_\theta$ is the posterior implied by the latent-context model, whereas $\bar{q}_\theta$ corresponds to a different model with a uniform latent-context prior.

The same variational formulation determines how the retriever should be updated. Let ϕ^- denote the current retriever parameters, and first compute q_{θ,ϕ^-} using the current model. Holding this posterior fixed, maximizing Eq. (11) with respect to ϕ reduces to

$$\max_\phi \; \mathbb{E}_{\mathcal{C}_t \sim q_{\theta,\phi^-}}\left[\log r_\phi(\mathcal{C}_t)\right],$$

or equivalently,

$$\min_{\phi} \mathcal{D}_{\mathrm{KL}}\big(q_{\theta,\phi^-} \,\big\|\, r_\phi\big)\,.$$

Thus, the update alternates between using the realized future to compute posterior credit and fitting the future-blind retriever to this fixed target. In the notation of the main text, this is implemented by

$$\mathcal{L}_{\mathrm{Ret}}(\phi) \coloneqq \mathbb{E}\left[\mathcal{D}_{\mathrm{KL}}(\mathrm{sg}[q_{\theta,\phi}] \,\|\, r_\phi)\right],$$

which is exactly Eq. (6). The stop-gradient prevents gradients from flowing through the posterior during the retriever update, implementing the fixed-target step of this alternating optimization rather than introducing an additional heuristic objective.

With the exact posterior, the same update also recovers the retriever-side marginal-likelihood gradient. At the parameters $\phi = \phi^-$ used to construct the posterior,

$$\nabla_\phi \log p_{\theta,\phi}(\boldsymbol{o}_{t+1} \mid \cdots)\big|_{\phi=\phi^-} = \mathbb{E}_{\mathcal{C}_t \sim q_{\theta,\phi^-}}\left[\nabla_\phi \log r_\phi(\mathcal{C}_t)\right]_{\phi=\phi^-} = -\nabla_\phi \mathcal{D}_{\mathrm{KL}}\big(q_{\theta,\phi^-} \,\big\|\, r_\phi\big)\Big|_{\phi=\phi^-}. \tag{12}$$

Hence, for the exact posterior, posterior matching gives exactly the negative marginal-likelihood gradient with respect to the retriever.

The derivation above applies to the exact posterior over recalled context sets. Under discrete Top-K recall, enumerating these sets is combinatorial. The candidate-level objective in Section B.3 therefore applies the same posterior-credit principle to a finite candidate pool using singleton predictive utilities. Because this restricts both the context space and the utility evaluation, the exact gradient identity in Eq. (12) need not hold globally for the practical approximation.

B.3 Retriever Posterior Approximation

The future-aware posterior in Eq. (5) is defined over all admissible recalled context sets. Under discrete Top-K recall, exact evaluation is therefore combinatorial. We adapt the latent-variable retriever training strategy of EMDR^2 (Sachan et al., 2021) to obtain a tractable candidate-level approximation over a finite pool of historical observations.

For each retriever update, let $\mathcal{A}_t \subseteq \mathcal{M}_{t-1}$ denote the candidate pool. The observation-level relevance scores from Eq. (8) induce the future-blind candidate distribution

$$r_\phi^{\mathcal{A}_t}(\boldsymbol{o}_i \mid \mathcal{Z}_{t-1}, \mathcal{R}_t) \coloneqq \mathrm{softmax}_{\boldsymbol{o}_i \in \mathcal{A}_t}\, s_{\phi,i}. \tag{13}$$

This distribution represents the retriever's relative preference among the candidates using only information available at inference time.

To assign predictive credit, we evaluate how well each candidate individually supports prediction of the observed future. The singleton predictive utility of candidate $\boldsymbol{o}_i$ is

$$u_{\theta,i} \coloneqq u_\theta(\{\boldsymbol{o}_i\} \mid \boldsymbol{o}_{t+1}, \mathcal{Q}_t) = \log p_\theta(\boldsymbol{o}_{t+1} \mid \{\boldsymbol{o}_i\}, \mathcal{Q}_t)\,. \tag{14}$$

Applying the same predictive reweighting principle as Eq. (5) gives the future-aware candidate posterior

$$q_{\theta,\phi}^{\mathcal{A}_t}(\boldsymbol{o}_i \mid \boldsymbol{o}_{t+1}, \mathcal{Z}_{t-1}, \mathcal{R}_t, \mathcal{Q}_t) \coloneqq \mathrm{softmax}_{\boldsymbol{o}_i \in \mathcal{A}_t}\left[s_{\phi,i} + u_{\theta,i}\right]. \tag{15}$$

Thus, $r_\phi^{\mathcal{A}_t}$ captures which candidates appear relevant before the future is known, while $q_{\theta,\phi}^{\mathcal{A}_t}$ reweights them according to their predictive utility for the realized future.

We train the retriever by matching its future-blind candidate distribution to this future-aware target:

$$\widehat{\mathcal{L}}_{\mathrm{Ret}}(\phi) \coloneqq \mathbb{E}\left[\mathcal{D}_{\mathrm{KL}}\left(\mathrm{sg}\left[q_{\theta,\phi}^{\mathcal{A}_t}\right] \,\middle\|\, r_\phi^{\mathcal{A}_t}\right)\right], \tag{16}$$

where $\mathrm{sg}[\cdot]$ denotes stop-gradient and the expectation is over training examples and candidate-pool construction. For a fixed candidate pool with exact singleton likelihoods, this update yields the same retriever gradient as the corresponding candidate-level latent-variable objective of EMDR^2 (Sachan

et al., 2021). FAR differs in how this candidate score is parameterized. Through Eq. (8), the posterior supervision acts on both cue-specific relevance and query-dependent cue weights. For a fixed training example and candidate pool,

$$\begin{aligned}\nabla_\phi \mathcal{D}_{\mathrm{KL}}\left(\mathrm{sg}[q_{\theta,\phi}^{\mathcal{A}_t}] \,\middle\|\, r_\phi^{\mathcal{A}_t}\right) &= \mathbb{E}_{\boldsymbol{o}_i \sim r_\phi^{\mathcal{A}_t}}[\nabla_\phi s_{\phi,i}] - \mathbb{E}_{\boldsymbol{o}_i \sim q_{\theta,\phi}^{\mathcal{A}_t}}[\nabla_\phi s_{\phi,i}] \\ &= \sum_m \lambda_{\phi,t}^m \left(\mathbb{E}_{r_\phi^{\mathcal{A}_t}}\left[\nabla_\phi \widetilde{s}_{\phi,i}^m\right] - \mathbb{E}_{q_{\theta,\phi}^{\mathcal{A}_t}}\left[\nabla_\phi \widetilde{s}_{\phi,i}^m\right]\right) \\ &\quad + \sum_m \left(\mathbb{E}_{r_\phi^{\mathcal{A}_t}}\left[\widetilde{s}_{\phi,i}^m\right] - \mathbb{E}_{q_{\theta,\phi}^{\mathcal{A}_t}}\left[\widetilde{s}_{\phi,i}^m\right]\right) \nabla_\phi \lambda_{\phi,t}^m,\end{aligned}$$

where $q_{\theta,\phi}^{\mathcal{A}_t}$ is treated with stop-gradient. The decomposition exposes two learning pathways: one through the cue-specific relevance scores and another through the query-dependent cue weights. Thus, future-aware predictive credit shapes both how memories are scored under each cue and how strongly each cue influences recall for the current query. Thus, a cue is reinforced when it favors memories that receive greater future-aware predictive credit. With a single cue, the cue-reliability term vanishes, recovering the EMDR^2-style candidate update; with multiple cues, the same predictive signal also learns which cues to trust for the current recall query.

This approximation affects only retrieval credit assignment. The world model is still trained on the complete recalled context and can therefore reason jointly over multiple memories. Candidate-level training does not explicitly score higher-order selection effects, such as memories that become informative only when recalled together, while such interactions remain represented by the set-level model in Eqs. (2) and (5). For our video instantiation, singleton likelihood is replaced by the diffusion-based utility estimator below, and the candidate pools are constructed from the temporally structured recall policy described next.

C Additional Implementation Details

C.1 Implementation Details of FAR for the Video Diffusion Instantiation

This section provides the architectural and optimization details of the video instantiation described in Section 3.3. We specify the diffusion objective, retrieval-score parameterization, adaptive cue fusion, predictive-utility estimator, and training procedure.

Video diffusion world model and objective. We parameterize p_θ with a diffusion transformer (DiT) (Peebles & Xie, 2023). Retrieved context-frame tokens are processed jointly with noisy target-frame tokens, using alternating intra-frame and global self-attention following VGGT (Wang et al., 2025). The world model predicts the next observation from the current observation and action together with the recalled context frames. Time and pose metadata associated with recalled frames are treated as part of their context representation.

Below, we make the diffusion objective explicit because it serves both to train the world model and, through its standard connection to likelihood-based diffusion training, to provide a tractable surrogate for predictive utility. Let $\boldsymbol{x}_0$ denote the clean diffusion representation of the target observation $\boldsymbol{o}_{t+1}$. At diffusion time τ, we sample $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ and construct

$$\boldsymbol{x}_\tau = \alpha_\tau \boldsymbol{x}_0 + \sigma_\tau \boldsymbol{\epsilon},$$

where α_τ and σ_τ define the noise schedule. Given recalled context $\mathcal{C}_t$ and prediction query $\mathcal{Q}_t$, the DiT outputs $\boldsymbol{f}_\theta(\boldsymbol{x}_\tau, \tau; \mathcal{C}_t, \mathcal{Q}_t)$. Let $\boldsymbol{y}_\tau$ denote the corresponding training target under the chosen diffusion parameterization, e.g., $\boldsymbol{y}_\tau = \boldsymbol{\epsilon}$ for noise prediction. The per-perturbation loss is

$$\ell_{\mathrm{diff}}(\theta; \boldsymbol{o}_{t+1}, \mathcal{C}_t, \mathcal{Q}_t, \tau, \boldsymbol{\epsilon}) = w(\tau) \left\| \boldsymbol{f}_\theta(\boldsymbol{x}_\tau, \tau; \mathcal{C}_t, \mathcal{Q}_t) - \boldsymbol{y}_\tau \right\|_2^2,$$

where $w(\tau)$ is the weighting used by the diffusion objective.

The world model is trained on the complete recalled context $\mathcal{C}_{t,\mathrm{ChunkTopK}}$:

$$\mathcal{L}_{\mathrm{WM}}(\theta) := \mathbb{E}\left[\ell_{\mathrm{diff}}(\theta; \boldsymbol{o}_{t+1}, \mathcal{C}_{t,\mathrm{ChunkTopK}}, \mathcal{Q}_t, \tau, \boldsymbol{\epsilon})\right], \tag{17}$$

where the expectation is over training examples and diffusion perturbations.

Retrieval cues affect which observations enter $\mathcal{C}_{t,\mathrm{ChunkTopK}}$. In particular, visual and audio retrieval representations are used for memory selection and are not passed to the world model as additional generation conditions. This allows the retriever to exploit additional cues without changing the generator's conditioning interface.

Cue-specific retrieval scores and cacheable representations. For the metadata cue, consisting of time and pose, we use a lightweight scoring network:

$$s_{\phi,i}^{\mathrm{meta}} = \tanh\big(\mathrm{MLP}_{\phi}\left([\boldsymbol{z}_i^{\mathrm{meta}}, \boldsymbol{z}_t^{\mathrm{meta}}, \boldsymbol{a}_t]\right)\big).$$

For a high-dimensional cue $m \in \{\mathrm{vision}, \mathrm{audio}\}$, we encode each historical cue as a memory key and the current cue as an action-conditioned query:

$$\begin{aligned} \boldsymbol{k}_i^m &\coloneqq \mathrm{enc}_m(\boldsymbol{z}_i^m, \varnothing), & \boldsymbol{h}_t^m &\coloneqq \mathrm{enc}_m(\boldsymbol{z}_t^m, \boldsymbol{a}_t), \\ \boldsymbol{q}_t^m &\coloneqq \mathrm{MLP}_{\phi,m}(\boldsymbol{h}_t^m), & s_{\phi,i}^m &\coloneqq \mathrm{CosSim}(\boldsymbol{k}_i^m, \boldsymbol{q}_t^m), \end{aligned}$$

where $\varnothing$ denotes a null action. We use a ViT (Dosovitskiy et al., 2021), inject the action or null token through AdaLN conditioning (Peebles & Xie, 2023), and use a learnable `[CLS]` token as the retrieval representation. Audio follows the same construction with an audio-specific encoder.

To make historical keys cacheable, we first pretrain the high-dimensional encoders with a predictive contrastive objective. The action-conditioned embedding $\boldsymbol{h}_t^m$ is trained to match the subsequent key $\boldsymbol{k}_{t+1}^m = \mathrm{enc}_m(\boldsymbol{z}_{t+1}^m, \varnothing)$. Given $N-1$ negative keys $\{\boldsymbol{k}_j^{m,-}\}_{j=1}^{N-1}$, we minimize

$$\mathcal{L}_{\mathrm{NCE}}^m = -\mathbb{E}\left[\log \frac{\exp\big(\mathrm{CosSim}(\boldsymbol{h}_t^m, \boldsymbol{k}_{t+1}^m)/\tau_{\mathrm{NCE}}\big)}{\exp\big(\mathrm{CosSim}(\boldsymbol{h}_t^m, \boldsymbol{k}_{t+1}^m)/\tau_{\mathrm{NCE}}\big) + \sum_{j=1}^{N-1} \exp\big(\mathrm{CosSim}(\boldsymbol{h}_t^m, \boldsymbol{k}_j^{m,-})/\tau_{\mathrm{NCE}}\big)}\right],$$

where τ_{NCE} is the contrastive temperature. After pretraining, the encoder is frozen and historical keys are computed once and cached. The lightweight query adapter $\mathrm{MLP}_{\phi,m}$ remains trainable through FAR.

Adaptive cue weighting. Because different cues may produce scores on different scales, we standardize each cue over the current episodic memory:

$$\widetilde{s}_{\phi,i}^m = \frac{s_{\phi,i}^m - \mu_t^m}{\sigma_t^m + \delta},$$

where μ_t^m and σ_t^m are the mean and standard deviation of $\{s_{\phi,i}^m\}_{\boldsymbol{o}_i \in \mathcal{M}_{t-1}}$, and $\delta > 0$ is a small numerical constant.

The query-dependent weights in Eq. (8) are produced from summary statistics of each cue:

$$\lambda_{\phi,t}^m \propto \exp\left(\boldsymbol{w}_\phi^\top \begin{bmatrix} \max_{\boldsymbol{o}_i \in \mathcal{M}_{t-1}} s_{\phi,i}^m \\ \mu_t^m \\ \sigma_t^m \\ \boldsymbol{e}_m \\ \Delta t \end{bmatrix}\right), \qquad \sum_m \lambda_{\phi,t}^m = 1,$$

where $\boldsymbol{e}_m$ is a one-hot cue-type indicator and Δt is the simulation stride. The normalization is taken over the cues available in the current environment. Because these score statistics depend on the current query, FAR can vary the relative importance of the available cues from one retrieval query to another.

Diffusion-based predictive utility. The exact singleton utility $u_{\theta,i} = \log p_\theta(\boldsymbol{o}_{t+1} \mid \{\boldsymbol{o}_i\}, \mathcal{Q}_t)$ in Eq. (14) requires the conditional log-likelihood, which is expensive to evaluate for a diffusion model. Diffusion models are instead trained with denoising objectives that arise from, or are closely related to, variational likelihood training (Ho et al., 2020; Kingma et al., 2021; Song et al., 2021b; Lai et al., 2025). We therefore use the negative diffusion prediction loss as a tractable surrogate for comparing the predictive utility of candidate memories. For candidate memory $\boldsymbol{o}_i$, we evaluate the realized future using the singleton context $\{\boldsymbol{o}_i\}$:

$$\widehat{u}_{\theta,i} \coloneqq -\frac{1}{S}\sum_{s=1}^{S} \ell_{\mathrm{diff}}(\theta; \boldsymbol{o}_{t+1}, \{\boldsymbol{o}_i\}, \mathcal{Q}_t, \tau_s, \boldsymbol{\epsilon}_s). \tag{18}$$

Algorithm 1 Training Future-Aware Recall with a Video Diffusion World Model

Require: retrieval size K, chunk size n_{chunk}, retriever interval N_{lazy}, utility samples S

for training step $n = 1, 2, \ldots$ **do**
  Sample $(\boldsymbol{o}_{t+1}, \mathcal{M}_{t-1}, \mathcal{Z}_{t-1}, \mathcal{R}_t, \mathcal{Q}_t)$
  Compute and fuse cue-specific scores using Eq. (8)
  Select $\mathcal{C}_{t,\text{ChunkTopK}}$ using Eq. (19)
  Sample $(\tau, \boldsymbol{\epsilon})$ and update θ using $\ell_{\text{diff}}(\theta; \boldsymbol{o}_{t+1}, \mathcal{C}_{t,\text{ChunkTopK}}, \mathcal{Q}_t, \tau, \boldsymbol{\epsilon})$
  if $n \bmod N_{\text{lazy}} = 0$ **then**
    Set $\mathcal{A}_t^{\text{global}} \leftarrow \mathcal{C}_{t,\text{ChunkTopK}}$
    Sample a represented chunk $\mathcal{B}_{t-1,j}$ and set $\mathcal{A}_t^{\text{local}} \leftarrow \mathcal{B}_{t-1,j}$
    Sample shared perturbations $\{(\tau_s, \boldsymbol{\epsilon}_s)\}_{s=1}^{S}$
    for $\mathcal{A}_t \in \{\mathcal{A}_t^{\text{global}}, \mathcal{A}_t^{\text{local}}\}$ **do**
      Evaluate $\widehat{u}_{\theta,i}$ for $\boldsymbol{o}_i \in \mathcal{A}_t$ using Eq. (18)
      Construct Eq. (15) using $\widehat{u}_{\theta,i}$
      Compute the corresponding loss in Eq. (16)
    end for
    Update ϕ with the sum of the global and local losses
  end if
end for

We use $S = 4$. A memory receives greater predictive credit when conditioning on it yields lower diffusion prediction loss for the future that actually occurred.

All candidates in the same training example are evaluated using the same perturbations $\{(\tau_s, \boldsymbol{\epsilon}_s)\}_{s=1}^{S}$. This common randomness reduces variation caused by diffusion sampling and makes candidate comparisons more directly reflect the recalled memory. We substitute $\widehat{u}_{\theta,i}$ for $u_{\theta,i}$ in Eq. (15), with the resulting posterior treated with stop-gradient during the retriever update.

Singleton contexts are used for retrieval credit assignment, while the world model is trained on the complete recalled context $\mathcal{C}_{t,\text{ChunkTopK}}$ through Eq. (17).

Temporally structured recall and candidate pools. Nearby video frames are often redundant, so unconstrained Top-K selection may allocate several context slots to the same short temporal region. We partition $\mathcal{M}_{t-1}$ into non-overlapping chunks $\{\mathcal{B}_{t-1,j}\}_j$ of n_{chunk} consecutive observations and allow at most one recalled observation from each chunk:

$$\mathcal{C}_{t,\text{ChunkTopK}} \coloneqq \arg\max_{\mathcal{C}_t \subseteq \mathcal{M}_{t-1}} \sum_{\boldsymbol{o}_i \in \mathcal{C}_t} s_{\phi,i} \quad \text{s.t.} \quad |\mathcal{C}_t| = K, \qquad |\mathcal{C}_t \cap \mathcal{B}_{t-1,j}| \le 1 \ \ \forall j. \tag{19}$$

Equivalently, we retain the highest-scoring observation from each chunk and then select the K highest-scoring representatives.

For retriever training, we use a global candidate pool containing the selected representatives and a local pool containing one of their corresponding chunks:

$$\mathcal{A}_t^{\text{global}} = \mathcal{C}_{t,\text{ChunkTopK}}, \qquad \mathcal{A}_t^{\text{local}} = \mathcal{B}_{t-1,j}.$$

The global pool assigns predictive credit across temporal regions, while the local pool distinguishes observations within one selected region. For each pool, we compute Eq. (18) and apply the future-aware candidate target in Eq. (15) with the loss in Eq. (16).

Joint training and inference. The world model is updated at every optimization step using Eq. (17). Candidate-wise utility evaluation requires additional DiT forward passes, so we update the retriever once every N_{lazy} world-model steps. At each retriever update, the global and local candidate pools are evaluated using shared diffusion perturbations, and ϕ is updated with the sum of their candidate-distillation losses.

At inference, FAR requires neither the future observation nor predictive-utility evaluation. Historical retrieval keys remain cached; FAR computes the available cue-specific scores, adaptively fuses them, and recalls $\mathcal{C}_{t,\text{ChunkTopK}}$ for world-model prediction.

	LoopNav	SoundSpaces v1	SoundSpaces v2	AI2-THOR	AI2-THOR-dyn
Environment	Minecraft	Matterport3D	Matterport3D	iTHOR	iTHOR
Episodes	19,200	14,425	14,425	8,125	1,202
Scenes	120	81	81	119	7
Train/test split	episode-level	scene-disjoint	scene-disjoint	scene-disjoint	episode-level
Train/test scenes	120/120	57/24	57/24	95/24	7/7
Frames	12.62M	9.28M	10.95M	9.23M	1.85M
FPS	20	10	10	10	10
Resolution	360×640	256×256	256×256	256×256	256×256
Additional signal	–	binaural audio	binaural audio	interactions	second-agent state

Table 3: **Dataset statistics.** SoundSpaces and AI2-THOR use scene-disjoint evaluation, whereas LoopNav and the reported AI2-THOR-dyn experiments use episode-level splits with the same scenes appearing in training and test.

C.2 Dataset Details

Table 3 summarizes the corpora used in our experiments. All datasets follow an exploration-or-outbound phase followed by a return or reveal phase, but differ in what information must be recovered from memory.

LoopNav. We use the released LoopNav dataset (Lian et al., 2025), containing 19,200 loop-navigation episodes across 120 procedurally generated Minecraft village scenes. Episodes follow ABA or ABCA routes over four range tiers, and the frame at which the released goal metadata turns toward home defines the boundary between the outbound and return legs. We use the outbound leg as episodic memory and evaluate prediction over the return leg. Our 80/20 split is performed over episodes without a scene constraint, so all 120 scenes appear in both training and test and the benchmark measures generalization to unseen trajectories within known worlds.

SoundSpaces. We construct two paired SoundSpaces corpora (Chen et al., 2020) from Matterport3D environments, each containing 14,425 episodes over 81 eligible scenes with a scene-disjoint split of 57 training and 24 test scenes. An agent traverses an ABA or ABCA loop while two to four static sound sources play continuously, with spatialized binaural audio rendered at 16 kHz throughout the trajectory. The outbound portion serves as memory and the final return leg is a freshly planned shortest path back to the starting location. SoundSpaces v1 performs 360° scans only at loop endpoints, whereas v2 adds deterministic turnaround and mid-path scans triggered by region changes, elevation changes, or long intervals without a scan. The two variants use the same episode manifest, seeds, waypoints, sound sources, clips, and gains, so matched v1/v2 comparisons isolate the effect of denser visual coverage. No scans are inserted into the return leg.

AI2-THOR. We construct 8,125 single-room iTHOR episodes (Kolve et al., 2017) over 119 scenes, using 95 scenes for training and 24 held-out scenes for evaluation. During exploration, the agent visits object stations, opens containers, and moves objects between surfaces and containers before revisiting the same stations during the return phase. Station inspections use reproducible canonical viewpoints, creating same-pose observations that can depict different world states before and after an interaction. Moved objects are hidden while carried, so the updated state must be inferred from the interaction history rather than visually tracked in transit.

AI2-THOR-dyn. We additionally construct 1,202 corridor episodes over seven iTHOR rooms, including 1,002 episodes with a moving second agent and 200 frame-aligned static controls. The observer watches a corridor through a constrained aperture while the second agent repeatedly crosses between its two ends, then traverses a door and looks toward both ends. The observation post is verified to reveal the crossing agent at the aperture while hiding both corridor ends, and quality control rejects episodes containing visibility leaks outside the intended crossing windows. The second agent alternates ends on each crossing, so its final location depends on the observed crossing history rather than the current view alone. The reported experiments follow the seed-based split used by our training manifest, with 800 acted episodes for training and 202 acted episodes. All seven

Setting	Value
DiT depth / hidden / heads	$12/768/12$
Patch size	2
Training context size	4 (current $+3$ recalled)
Diffusion steps	1000
Noise schedule	Linear
Prediction target	ϵ-prediction
Variance	Learned
Optimizer	AdamW
Generator learning rate	10^{-4}
Retriever learning rate	10^{-4}
AdamW ϵ / weight decay	$10^{-6}/0$
Warm-up	5k steps, $0.2\times \rightarrow 1\times$
Gradient clipping	1.0
EMA decay	0.9999
Utility noise levels	4
Training precision	fp32

Table 4: **Shared model and optimization settings.** These settings are used across all corpora unless stated otherwise.

	LoopNav	SoundSpaces v1	SoundSpaces v2	AI2-THOR	AI2-THOR-dyn
Training pool size	200	200	200	400	400
Prediction horizon (frames)	128	128	128	32	$32/135^{\dagger}$
Goals per observation	4	4	4	4	–
Retrieval chunk size	10	10	10	10	30
FAR retrieval cues	Meta+Vision	Meta+Audio	Meta+Audio	Meta+Vision	Meta / Meta+Agent
Metadata MLP layers	2	2	2	3	3
Retriever update interval	20	20	20	20	7
Event-anchored sampling	–	–	–	0.30	0.35
Change-mask weight	–	–	–	0.3	5.0
FAR effective batch size	128	64	64	64	64

Table 5: **Dataset-specific training settings.** The FAR cue row gives the signals used for retrieval; visual and audio retrieval representations are not passed to the world model as generation conditions. †AI2-THOR-dyn uses a 135-frame horizon for reveal-anchored pairs and 32 frames otherwise.

rooms therefore appear in both splits, so these results measure generalization to unseen trajectories within known rooms rather than held-out-room generalization.

C.3 Training Details

Latent preprocessing. All corpora are trained in latent space with a frozen tokenizer, and frame latents are precomputed once before world-model training. LoopNav uses the ViTVAE tokenizer from OASIS (Decart et al., 2024), producing $16 \times 18 \times 32$ latents from 360×640 RGB frames with latent scale 0.07843137255. SoundSpaces and both AI2-THOR corpora use the SDXL VAE (Podell et al., 2024) with the fp16 fix, producing $4 \times 32 \times 32$ latents from 256×256 RGB frames with latent scale 0.13025.

Optimization and sampling. The learning rate is linearly warmed from $0.2\times$ to its target value over the first $5{,}000$ steps and held constant thereafter. We clip the global gradient norm to 1.0, maintain an EMA with decay 0.9999, and evaluate the EMA model every $2{,}500$ steps. FAR uses an effective batch size of 128 on LoopNav and 64 on the remaining corpora. The Temporal baseline uses effective batch sizes of 112 on LoopNav and 63 on SoundSpaces, while the remaining baseline arms match the corresponding FAR batch size.

Training pairs and memory pools. The dataset-specific settings are summarized in Table 5. For LoopNav, training queries are sampled from the return leg while the retrievable pool is restricted to the outbound leg. SoundSpaces uses the corresponding outbound portion as memory for prediction

	LoopNav	SoundSpaces v1	SoundSpaces v2	AI2-THOR	AI2-THOR-dyn
Evaluation checkpoint	700k	700k	700k	700k	100k
Evaluation protocol	rollout	rollout	rollout	rollout	both-ends probe
Stride (frames / seconds)	$4/0.2$	$16/1.6$	$16/1.6$	$4/0.4$	–
Context size	12	12	12	12	4
Retrieval pool size	1000	1000	1000	1000	–
Retrieval pool phase	outbound	outbound	outbound	exploration	pre-reveal history
Retrieval chunk size	10	10	10	10	30
Sampler	DDIM-20	DDIM-20	DDIM-20	DDIM-20	DDIM-20

Table 6: **Dataset-specific inference settings.** The four rollout benchmarks expand the context from 4 frames during training to 12 frames at evaluation and query up to 1000 real memory frames.

on the return trajectory. AI2-THOR restricts the retrievable pool to the pre-evaluation exploration phase. With probability 0.30, an AI2-THOR training example is replaced by an event-anchored pair centered on an evaluation-phase open/close event, providing direct supervision near changed-state reveals. AI2-THOR-dyn uses reveal-anchored sampling with probability 0.35; anchored examples place the current frame during the door transit before the reveal and use the longer 135-frame prediction range.

Retriever initialization and updates. The high-dimensional FAR retrieval encoders are initialized by contrastive pretraining before joint world-model training, and cached historical keys are exported from the same checkpoint used to initialize the corresponding query tower. The QueryViT has six transformer layers, hidden dimension 384, 12 attention heads, and key dimension 256. Pretraining uses eight positive and 24 negative keys per query. The LoopNav, SoundSpaces, and AI2-THOR retrievers are pretrained for 400k steps with learning rate 2×10^{-4}, while AI2-THOR-dyn uses a 12.5k-step object-centric fine-tuning stage with learning rate 5×10^{-5}. For SoundSpaces, positive pairs are selected by spatial proximity without a heading constraint so that a return-phase query can match an outbound observation seen from the opposite heading, and negatives are drawn from the same episode to avoid an episode-identity shortcut. During joint training, the world model is updated at every optimization step from the context selected by the retriever. Future-aware retriever updates use four stratified diffusion noise levels with shared perturbations across candidates and combine the global and within-chunk retrieval losses with equal weight. LoopNav, both SoundSpaces variants, and AI2-THOR update the retriever every 20 world-model steps, while AI2-THOR-dyn uses an interval of 7 steps and applies the retriever update only to reveal-anchored examples.

Change-focused supervision in AI2-THOR. Because object interactions often affect only a small image region, AI2-THOR reweights the diffusion loss using a per-goal change mask. The mask is normalized to preserve unit mean loss weight and uses coefficient 0.3 with a maximum weighted-area fraction of 0.15. The same mask weighting is applied to every AI2-THOR retrieval arm so that comparisons remain matched. AI2-THOR-dyn instead uses the second-agent silhouette as the loss mask with coefficient 5.0 and the same maximum fraction, and applies this weighting to both the diffusion reader loss and FAR's predictive-utility reward.

Second-agent supervision in AI2-THOR-dyn. Each historical frame carries the second agent's visibility and egocentric floor position together with the same quantities 1.5 seconds earlier, allowing a recalled slot to encode local motion. An attention-pooling predictor over the recalled slots predicts second-agent presence and position using binary cross-entropy for presence and a masked position loss with weight 0.05. This auxiliary loss enters the training objective with weight 1.0, and the same agent-state term is included in the candidate reward used for retriever training. During training, the diffusion model receives the ground-truth goal agent state through teacher forcing, while inference uses the predictor output as described below.

C.4 Inference Details

Autoregressive rollout protocol. For LoopNav, SoundSpaces, and AI2-THOR, we evaluate with replace-tail autoregressive rollouts along the ground-truth camera path using the settings in Table 6.

At each step, the world model generates the next latent and the generated latent replaces only the current tail frame for the following step. The retrievable episodic memory remains composed entirely of real observations from the outbound or exploration phase, so rollout errors do not contaminate the stored memory. Camera poses are re-centered on the new current frame at every step using the same coordinate convention as in training. The retrieval module is queried at every generation step. Every method on a given clip uses the same diffusion-noise seed and target trajectory.

Generation conditioning. At each rollout step, the generator receives the current frame, recalled context frames, per-frame pose and trajectory-time conditioning, and the target action represented as a local pose displacement together with the generation stride. For AI2-THOR, the generator additionally receives the interaction labels occurring between the current and target states together with the absolute door fraction at the target. These interaction labels are supplied by the evaluation trajectory rather than generated by the model because camera motion alone does not specify object interactions. Visual and audio retrieval representations are used only to rank episodic memory and are not passed to the generator as additional conditioning variables. Thus, SoundSpaces uses binaural audio to select context while generation remains conditioned on the selected visual frames together with pose, time, and action information.

Evaluation operating point. All reported autoregressive rollouts use EMA weights and DDIM (Song et al., 2021a) with 20 sampling steps. Relative to training, the rollout protocol increases the context size from 4 to 12 and the retrieval pool from 200 or 400 frames to as many as 1000 frames, uniformly across compared methods within each corpus. WorldMem retains a 200-frame pool cap on LoopNav and SoundSpaces to match its training configuration, while AI2-THOR uses the full evaluation pool. The reported batteries contain 3,796 LoopNav clips, 1,167 clips for each SoundSpaces variant, and 1,309 AI2-THOR clips.

AI2-THOR-dyn inference. AI2-THOR-dyn is evaluated with a one-step both-ends prediction probe rather than an autoregressive rollout. For each memory-dependent test trajectory, the two possible reveal ends are generated separately from the same current observation using the same diffusion-noise seed, so the generations differ only in the goal and retrieved memory. The world model uses the agent-state predictor's own binarized output at inference without teacher forcing. Real memory frames retain their ground-truth second-agent detections, while generated states are threaded forward using the predictor output together with the corresponding state from 1.5 seconds earlier. A three-layer latent-space detector determines whether the generated frame contains the second agent, with at least 10 firing latent cells counted as a positive rendering. On real test frames, this detector has 0.906 recall and 0.070 false-positive rate at the chosen threshold.