CAST: RECONSTRUCTION-COUPLED ACCELERATION OF INTERACTIVE WORLD MODELS

Anonymous authors
Paper under double-blind review



Figure 1: **Action-conditioned rollouts.** Native (top) and our method CAST (bottom) on Matrix-Game 3.0 (left) and HY-World 1.5 (right). CAST maintains visual fidelity close to Native while achieving $2.15\times$ and $3.48\times$ speedups, respectively (Figure 2); W/S denote forward/backward movement.

ABSTRACT

Interactive world models must respond quickly to controls while preserving scene consistency. Existing acceleration methods can miss heterogeneous control responses and spatial transport when recovering skipped features. We observe that interaction-induced feature changes correlate with approximation error, while low-frequency interpolation errors are phase-sensitive and show more predictable phase progression. These findings motivate CAST, a reconstruction-coupled inference framework. CAST selects anchors by interaction sensitivity and cross-layer coverage, reconstructs skipped residuals with frequency- and confidence-aware Phase-Aware Reconstruction (PAR), and coordinates historical KV routing according to downstream reconstruction responsibility. On Matrix-Game 3.0 and HY-World 1.5, CAST achieves $2.15\times$ and $3.48\times$ speedups, respectively, while maintaining visual quality close to Native (Figure 1). It also attains the highest VBench scores among compared methods and leads non-native baselines on seven and six of thirteen WorldMark dimensions, demonstrating a balance of generation speed, visual quality, and interactive responsiveness under real-time control.

1 INTRODUCTION

Interactive world models turn visual generation into a feedback loop: users act and the model generates the next observation (Bruce et al., 2024; Valevski et al., 2025). Recent systems combine streaming generation, camera and action conditioning, and visual history (Wang et al., 2026; Sun et al., 2025). Their quality depends on responsive controls and consistent revisits, beyond plausible frames alone (Xu et al., 2026). Inference latency remains a practical obstacle to interactive use (Lu et al., 2026), making efficiency part of the modeling problem.

Efficient sampling algorithms reduce the number of denoising-network evaluations required for generation (Song et al., 2021; Lu et al., 2022). Distillation and consistency models further enable few-step generation (Salimans & Ho, 2022; Song et al., 2023), including autoregressive video synthesis (Yin et al., 2025; Huang et al., 2025). Yet each network evaluation can remain costly even with few denoising steps (Tang et al., 2026). CAST targets this remaining cost by reducing computation within each evaluation. Training-free methods exploit redundancy by caching features across denoising steps (Ma et al., 2024b; Liu et al., 2025a; Lyu et al., 2025), computing selected frames and interpolating their neighbors (Tang et al., 2026), or reducing historical attention (Lu et al., 2026; Li et al., 2026). These strategies offer complementary savings, although few-step inference limits cross-step reuse (Tang et al., 2026). In our setting, the approximations are connected: computed frames supply the endpoints for reconstructing skipped updates, so errors in their historical attention can also propagate beyond anchor frames themselves through reconstruction (Section 3.3).

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

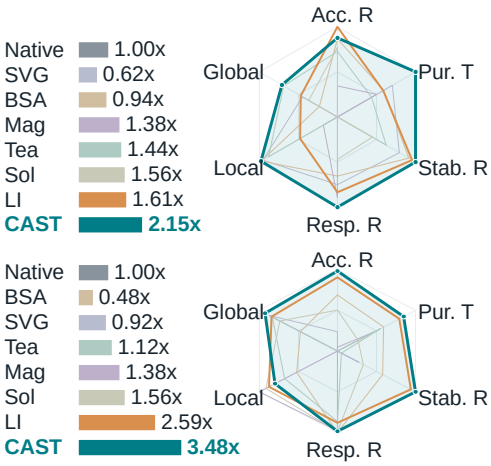

Figure 2: **Speed and interactive quality.** Speedup over Native (left) and selected WorldMark dimensions (right), min–max normalized per backbone across all methods including Native; colors match the speedup bars.

This dependency suggests organizing acceleration around the recovery of skipped states. Prior work shows that motion content and token sensitivity can guide nonuniform computation (Kahatapitiya et al., 2025; Zou et al., 2025). We ask whether control responses provide a useful signal for selecting frame anchors. Figure 3 shows that stronger camera- and action-induced feature responses are associated with larger next-block skipping errors. Because these responses arise in the conditioning path, they can guide allocation without first evaluating the dense residual we seek to avoid. Selection must also maintain coverage: interleaving frames across layers is important in frame-sparse inference so that exact computation will not be allocated to several fixed frames (Tang et al., 2026). We combine sensitivity with a refresh debt to balance responsive updates and regular coverage.

In interactive world models, moving the camera or controlling an object shifts scene content across spatial locations (Wang et al., 2026). Recovering skipped updates between anchors therefore requires accounting for where content moves, beyond how feature values change. Direct interpolation blends endpoint residuals at fixed coordinates, potentially mixing different scene content rather than tracking its displacement. Phase transport offers a way to align these residuals before interpolation: locally coherent motion can be represented through phase shifts (Meyer et al., 2015). Our diagnostics in Figure 4 support this view: oracle phase correction substantially reduces a low-frequency interpolation-error peak, and phase progression is more concentrated at low frequencies. These findings motivate phase-aware reconstruction, with transport strength adapted to frequency and local phase confidence to limit corrections where coherent motion is poorly supported.

Reconstruction changes what an anchor needs from history. Prior analyses show that local approximation errors can have unequal downstream effects (Zou et al., 2025). Historical routing in Light Interaction selects blocks by query–key relevance (Lu et al., 2026). In our setting, an anchor also supplies residuals for skipped frames, so local relevance alone does not account for this dependency. If neighboring anchors omit useful evidence together, their errors may jointly affect the reconstructed interval. We therefore coordinate their historical supports under fixed per-anchor budgets, aiming to preserve complementary evidence for both the anchors and the states they reconstruct.

Building on these observations, we introduce CAST, a reconstruction-coupled inference framework for interactive world models. *Control-Aware Frame Selection* balances interaction sensitivity with cross-layer coverage. *Phase-Aware Reconstruction* recovers skipped residuals through frequency- and confidence-aware phase transport. Finally, *Reconstruction-Coupled Sparse Attention* refines historical KV supports according to both local omission effects and the recovery of neighboring states. Together, these components connect where computation is retained, how omitted updates are recovered, and which history supports that recovery. Our contributions are as follows:

- **Control-Aware Frame Selection.** We select anchors using camera- and action-induced feature responses. Cross-layer coverage promotes regular refreshes, balancing interaction-sensitive updates with the need to maintain representations throughout the current chunk and across the entire active action horizon of each generated sequence.
- **Phase-Aware Reconstruction.** We recover skipped residuals by aligning endpoint spectra before interpolation or one-sided extrapolation. Transport strength adapts to frequency and local phase confidence, limiting corrections when evidence for coherent motion remains uncertain across the full interactive control sequence.
- **Reconstruction-Coupled Sparse Attention.** We coordinate neighboring anchors' historical KV supports under fixed budgets according to their local omission effects and shared responsibility for reconstructing skipped states across adjacent intervals.
- **Evaluation on interactive world models.** As shown in Figure 2, on Matrix-Game 3.0 and HY-World 1.5, CAST achieves $2.15\times$ and $3.48\times$ acceleration, respectively, and leads non-native methods on seven and six of thirteen WorldMark dimensions.

2 Related Work

2.1 Interactive Video World Models

Dreamer and IRIS learn behaviors through imagined rollouts (Hafner et al., 2020; Micheli et al., 2023). DIAMOND introduces diffusion-based world modeling (Alonso et al., 2024), GameNGen simulates games (Valevski et al., 2025), and Genie learns interactive environments through latent actions (Bruce et al., 2024). Video diffusion and Stable Video Diffusion provide generative foundations (Ho et al., 2022; Blattmann et al., 2023); CameraCtrl and MotionCtrl add explicit camera and motion control (He et al., 2025; Wang et al., 2024), enabling directed changes to viewpoint and scene motion.

CausVid and Self Forcing enable few-step autoregressive generation (Yin et al., 2025; Huang et al., 2025). We target interactive chunk-autoregressive models, including Matrix-Game 3.0 and WorldPlay (Wang et al., 2026; Sun et al., 2025), where processing frames, controls, and history makes per-evaluation latency critical for responsive online control.

2.2 Efficient Inference for Video Diffusion Models

Caching reuses intermediate computation across denoising steps. DeepCache and Block Caching reuse features or block outputs (Ma et al., 2024b; Wimbauer et al., 2024); Learning-to-Cache learns layer schedules (Ma et al., 2024a). TeaCache and MagCache adapt reuse to input changes, residual magnitudes, or block dynamics (Liu et al., 2025a; Ma et al., 2025). QuantCache combines hierarchical latent and layer caching with importance-guided quantization and pruning for video DiTs (Wu et al., 2025). AdaCache adapts schedules to video content (Kahatapitiya et al., 2025); TaylorSeer forecasts features instead of directly reusing them (Liu et al., 2025b).

ToCa selects tokens by sensitivity and redundancy (Zou et al., 2025), while DuCa alternates aggressive and conservative caching (Zou et al., 2026). These methods establish the value of selective computation, but few-step models offer limited cross-step reuse. Recent few-step DiT methods also interleave frame subsets within an evaluation (Tang et al., 2026).

Light Interaction combines context management, output reuse, and historical sparsity (Lu et al., 2026); Sol Engine integrates caching, sparsity, pruning, quantization, and kernels (Li et al., 2026). CAST coordinates retained frames and historical supports through their shared reconstruction responsibility.

2.3 Sparse Attention

Longformer and BigBird use structured sparsity (Beltagy et al., 2020; Zaheer et al., 2020); Reformer and Routing Transformer route by content (Kitaev et al., 2020; Roy et al., 2021). FlashAttention and FlashAttention-2 optimize exact attention (Dao et al., 2022; Dao, 2024). For video, Sliding Tile Attention exploits local windows (Zhang et al., 2025), LongCat-Video uses 3D block sparsity (Meituan LongCat Team et al., 2025), and Sparse VideoGen and VMoBA exploit spatial–temporal structure (Xi et al., 2025; Wu et al., 2026). Building on pooled historical routing (Lu et al., 2026), CAST considers both local omission errors and their propagation to reconstructed frames. This dependency makes retained historical evidence important beyond the computed anchors themselves, linking sparse attention decisions to the accuracy of skipped-frame reconstruction.

3 Preliminaries and Motivation

Our setting combines diffusion denoising (Ho et al., 2020), latent representations (Rombach et al., 2022), and transformer blocks (Peebles & Xie, 2023). We study one evaluation of a latent chunk conditioned on controls and visual history. Let R_i^ℓ denote an eligible visual residual update for frame i within transformer block ℓ, and $\widehat{R}_i^\ell$ its approximation. CAST is applied at eligible visual attention and FFN residual sites, while conditioning and text updates retain native dense execution. Within each network evaluation, retained anchors are computed directly and skipped frames receive reconstructed residual updates. Historical sparsity changes the evidence available to those anchors, linking frame selection, residual recovery, and KV routing within each transformer block.

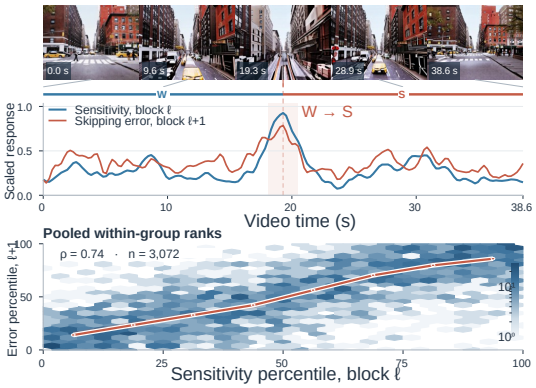

Figure 3: **Control sensitivity and skipping error.** Top: native rollout. Middle: independently scaled sensitivity at block ℓ and error at $\ell + 1$. Bottom: pooled percentiles (Spearman $\rho = 0.74$, $n = 3{,}072$), with counts and binned medians.

3.1 Interaction Sensitivity Predicts Frame Approximation Error

Let $\mathcal{K}_\ell \subseteq \{\text{cam}, \text{act}\}$ contain the available control branches, with feature updates $\Delta^\ell_{i,k}$. For $\mathcal{K}_\ell \neq \varnothing$, define control sensitivity and relative approximation error by

$$d^\ell_i = \frac{1}{|\mathcal{K}_\ell|} \sum_{k \in \mathcal{K}_\ell} \text{Norm}(\Delta^\ell_{i,k}), \qquad e^\ell_i = \frac{\left\| R^\ell_i - \widehat{R}^\ell_i \right\|_2}{\left\| R^\ell_i \right\|_2 + \epsilon}. \tag{1}$$

Here Norm rescales update magnitudes across the current chunk. Sensitivity is available from the conditioning path; error requires a dense reference and is used only for diagnosis.

Figure 3 compares d^ℓ_i with $e^{\ell+1}_i$, matching the scheduler's one-block prediction lag. Both curves rise near the W-to-S transition; pooled percentiles show positive Spearman correlation ($\rho = 0.74$). Because the temporal curves are independently rescaled, their proximity indicates similar temporal variation, not equal magnitudes. The spread in the percentile plot also shows that sensitivity does not fully determine error. This motivates sensitivity-based allocation balanced by cross-layer coverage.

3.2 Spectral Error and Frequency-Dependent Phase Transport

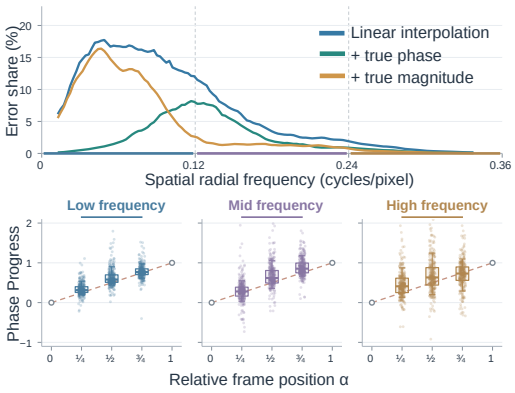

Figure 4: **Spectral error and phase progression.** Top: interpolation error with oracle phase/magnitude corrections. Bottom: phase progress across frequency bands; dashed lines denote uniform progression.

For a skipped frame with $a < t < b$, linear interpolation uses $R^{\text{lin}}_t = (1-\alpha)R_a + \alpha R_b$, where $\alpha = (t-a)/(b-a)$. It mixes fixed coordinates even when content moves. Figure 4 separates alignment and amplitude errors using oracle corrections that replace either phase or magnitude with its reference value while retaining the other interpolated component (Appendix C.2).

The upper plot shows a low-frequency error peak largely removed by phase correction, while magnitude correction leaves substantial error. This supports alignment correction in that band. The advantage varies across frequencies, and error share alone does not establish difficulty relative to target-band energy at each frequency.

The lower panels show tighter phase progression at low frequencies and broader distributions at high frequencies. Coherent translation gives linear unwrapped phase at every frequency (Meyer et al., 2015); the observed spread reflects reliability, not an intrinsic low-frequency rule. Thus PAR prioritizes alignment-sensitive components while tempering transport with local confidence.

3.3 Reconstruction Dependency Implies Coupled Attention

Once two anchors reconstruct an interval, attention approximation at either endpoint can affect every skipped state. For $\widehat{R}_t = \mathcal{P}_t(R_a, R_b)$, anchor perturbations induce

$$\Delta^{\text{rec}}_t = \mathcal{P}_t(R_a + \delta R_a, R_b + \delta R_b) - \mathcal{P}_t(R_a, R_b). \tag{2}$$

The interval error $\sum_{t \in (a,b)} \|\Delta^{\text{rec}}_t\|^2$ depends on both endpoints and the reconstruction operator. For linear interpolation, $\Delta^{\text{rec}}_t = (1-\alpha)\delta R_a + \alpha \delta R_b$: the squared error includes a cross term, so endpoint errors can reinforce or offset each other. Choosing historical KV supports independently therefore need not minimize interval error, even when each anchor is locally well approximated. This motivates coordinating supports under fixed per-anchor budgets to protect both anchors and reconstructed states. Section 5.3 tests this dependency at matched frame and historical KV budgets.

4 Method

4.1 Overview and Problem Formulation

Write the block input as $X^\ell = [X^{\text{mem},\ell}, X^{\text{cur},\ell}]$, with N_{cur} current latent frames. Memory frames retain native computation or caching; only current residuals are skipped. At block ℓ, the anchor

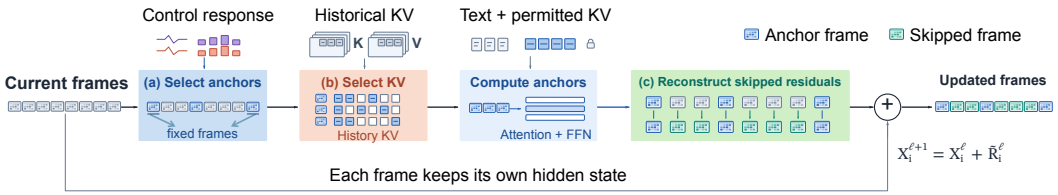

Figure 5: **CAST pipeline.** Select anchors and historical KV blocks, compute anchor residuals, and reconstruct skipped updates while retaining each frame's own skip connection.

set $\mathcal{A}_\ell = \{a_0 < \cdots < a_J\}$ contains $K_f = J + 1$ frames, and adjacent anchors define intervals $\mathcal{I}_{a,b} = \{t : a < t < b\}$. The anchor set is selected once per transformer block and shared by its eligible visual attention and FFN residual sites. Suppressing the residual-site index, each eligible update is

$$X_i^{\ell+1} = X_i^{\ell} + \widetilde{R}_i^{\ell}, \qquad \widetilde{R}_i^{\ell} = \begin{cases} R_i^{\ell,\mathrm{sp}}, & i \in \mathcal{A}_\ell, \\ \mathcal{P}_i(R_a^{\ell,\mathrm{sp}}, R_b^{\ell,\mathrm{sp}}), & i \in \mathcal{I}_{a,b}, \\ \mathcal{P}_i^{\mathrm{ext}}(R_{a_0}^{\ell,\mathrm{sp}}, R_{a_1}^{\ell,\mathrm{sp}}), & i < a_0, \\ \mathcal{P}_i^{\mathrm{ext}}(R_{a_{J-1}}^{\ell,\mathrm{sp}}, R_{a_J}^{\ell,\mathrm{sp}}), & i > a_J, \end{cases} \tag{3}$$

Here $\mathcal{P}_i^{\mathrm{ext}}$ uses the same spectral transport as $\mathcal{P}_i$ with $\alpha < 0$ on the left and $\alpha > 1$ on the right. The two anchors nearest each boundary are used: (a_0, a_1) for $i < a_0$ and (a_{J-1}, a_J) for $i > a_J$. This construction requires $K_f \geq 2$. $R_i^{\ell,\mathrm{sp}}$ denotes the directly computed anchor update at the eligible residual site. Each skipped frame retains its own input X_i^{ℓ}; reconstruction supplies only its residual update. Figure 5 summarizes the pipeline. Execution routes masks before evaluating anchors and reconstructing skipped updates; the routing surrogate needs no dense anchor outputs.

4.2 Control-Aware Frame Selection

Interaction sensitivity. The conditioning path supplies d_i^{ℓ} for all current frames (Eq. (1)). We use these scores to select anchors at the next block, so selection does not require the residuals it aims to skip. As illustrated in Figure 6, frames with stronger control responses receive higher priority, while coverage debt prevents repeatedly favoring the same subset. Scoring costs are included in the reported end-to-end chunk latency.

Coverage debt. Following the coverage motivation of frame interleaving (Tang et al., 2026), let h_i^{ℓ} count blocks since refresh. Define the refresh age and its normalized coverage debt:

$$h_i^{\ell+1} = \begin{cases} 0, & i \in \mathcal{A}_\ell, \\ h_i^{\ell} + 1, & i \notin \mathcal{A}_\ell, \end{cases} \tag{4}$$

$$H_{\mathrm{cov}} = \left\lceil \frac{N_{\mathrm{cur}}}{K_f} \right\rceil, \qquad D_i^{\ell+1} = \frac{h_i^{\ell+1}}{H_{\mathrm{cov}}}.$$

$$S_i^{\ell+1} = \begin{cases} d_i^{\ell} + \lambda_{\mathrm{cov}} D_i^{\ell+1}, & \text{fresh response available,} \\ \lambda_{\mathrm{cov}} D_i^{\ell+1}, & \text{otherwise.} \end{cases} \tag{5}$$

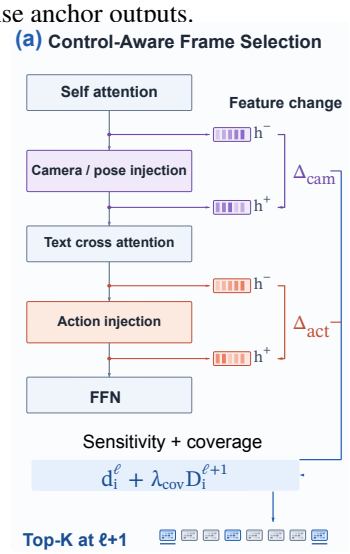

Figure 6: Control responses and coverage debt guide selection.

We set $\lambda_{\mathrm{cov}} = 0.50$; H_{cov} scales age but does not guarantee refresh. Deterministic Top-K_f selects K_f anchors across all N_{cur} current frames: $\mathcal{A}_{\ell+1} = \mathrm{TopK}_i(S_i^{\ell+1}, K_f)$. Neither endpoint is fixed. Without a fresh conditioning response, we omit sensitivity rather than reuse a stale score. Each denoising evaluation starts with a dense block and resets ages (Appendix A.1).

4.3 Phase-Aware Reconstruction

Local phase transport. Following phase-based interpolation and extrapolation (Meyer et al., 2015), partition residuals into tiles p and suppress the block index. For anchors $a < b$ and target frame $t \notin \{a, b\}$, let $\alpha = (t-a)/(b-a)$ and $F_{i,p,c}(\omega) = \mathrm{RFFT2}(R_{i,p,c})(\omega)$ for channel c. Interior targets have $0 < \alpha < 1$; boundary targets have $\alpha < 0$ or $\alpha > 1$. Figure 7 illustrates the phase transport and spectral blending used to reconstruct these skipped residuals.
Define the pooled cross-spectrum and magnitude product:

$$Z_p(\omega) = \sum_c F_{b,p,c}(\omega)\overline{F_{a,p,c}(\omega)}, \quad W_p(\omega) = \sum_c |F_{b,p,c}(\omega)|\,|F_{a,p,c}(\omega)|. \tag{6}$$

The pooled cross-spectrum favors shared displacement over weak-channel evidence, but it neither estimates optical flow nor resolves phase unwrapping.

On reliable bins, set $\theta_p = \operatorname{Arg} Z_p$ and $U_p = e^{\mathrm{i}\theta_p}$; otherwise $U_p = 1$. Fractional powers use the fixed branch $U_p^{\beta} = e^{\mathrm{i}\beta\theta_p}$.

Frequency-confidence affinity. Let Ω_+ be the reliable non-DC spectral bins, with multiplicity weights κ_ω for the real FFT. A tile-level consensus score across the reliable spectral bins is

$$c_p = \frac{\sum_{\omega\in\Omega_+} \kappa_\omega |Z_p(\omega)|}{\sum_{\omega\in\Omega_+} \kappa_\omega W_p(\omega) + \epsilon} \in [0,1]. \quad (7)$$

Agreement across channel phases raises c_p, whereas conflicting evidence reduces it. With normalized radial frequency $\nu_\omega \in [0,1]$, define

$$\tau(\nu) = \tau_{\text{low}} + (\tau_{\text{high}} - \tau_{\text{low}})\nu^q,$$
$$g_{p,\omega} = \sigma\left(\frac{c_p - \tau(\nu_\omega)}{T}\right), \quad (8)$$

where $q, T > 0$ and $\tau_{\text{high}} \geq \tau_{\text{low}}$. Thus high frequencies require stronger transport evidence. Frequency supplies a prior and confidence estimates reliability, without target access.

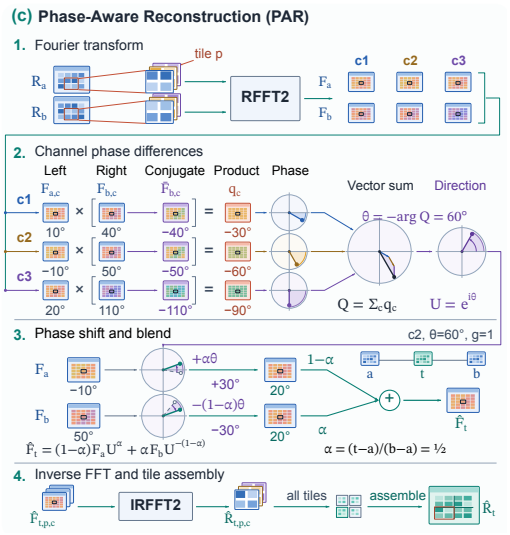

Figure 7: Phase-aware reconstruction with $Q = \overline{Z}$, where Z is defined in Eq. (6).

Unified reconstruction. PAR aligns endpoint spectra before interpolation or one-sided extrapolation, applying the following rule to each channel and spatial tile:

$$\widehat{F}_{t,p,c}(\omega) = (1-\alpha)F_{a,p,c}(\omega)U_p(\omega)^{g_{p,\omega}\alpha} + \alpha F_{b,p,c}(\omega)U_p(\omega)^{-g_{p,\omega}(1-\alpha)}. \quad (9)$$

An inverse real FFT recovers $\widehat{R}_{t,p,c}$, and the tiles are assembled into $\widehat{R}_t = \mathcal{P}_t(R_a, R_b)$. Setting $g = 0$ gives direct value interpolation or extrapolation; setting $g = 1$ gives full endpoint phase transport. The anchor endpoints are reproduced exactly. Numerical conventions are given in Appendix A.2.

4.4 Reconstruction-Coupled Sparse Attention

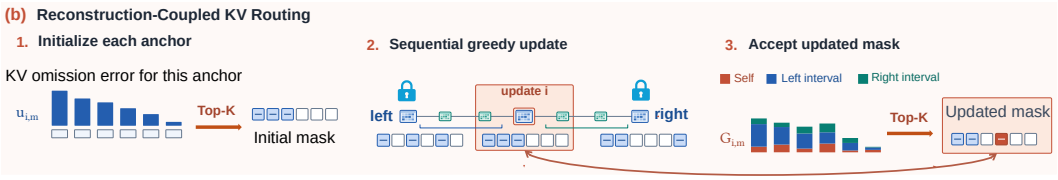

Figure 8: Initialize each anchor's mask, then refine it using local and neighboring reconstruction gains while preserving a fixed historical KV budget at every anchor.

Local KV-omission importance. As in pooled historical routing (Lu et al., 2026), divide each head's historical KV into M equal-size blocks. Pooled queries and keys/values give

$$s_{i,m} = \frac{\bar{q}_i^{\top}\bar{k}_m}{\sqrt{d_h}}, \qquad \pi_{i,m} = \operatorname{softmax}_m(s_{i,m}), \qquad \bar{o}_i = \sum_m \pi_{i,m}\bar{v}_m. \quad (10)$$

Pooled one-block omission scores initialize the masks in Figure 8:

$$\delta_{i,m} = \frac{\pi_{i,m}}{1-\pi_{i,m}+\epsilon}(\bar{o}_i - \bar{v}_m), \qquad u_{i,m} = \|P\delta_{i,m}\|_2^2. \quad (11)$$

We use $P = W_O^{(h)}$, the corresponding head slice of the native attention output projection. This inexpensive proxy excludes current/text interactions and later nonlinearities, so it estimates pooled-output deviation rather than the complete nonlinear block error after all residual updates.

Pairwise reconstruction risk. Let $x_{i,m} \in \{0,1\}$ indicate whether anchor i retains block m. For interval (a,b), define

$$A_{a,b} = \sum_{t\in\mathcal{I}_{a,b}} (1-\alpha_t)^2, \quad B_{a,b} = \sum_{t\in\mathcal{I}_{a,b}} \alpha_t^2, \quad C_{a,b} = \sum_{t\in\mathcal{I}_{a,b}} \alpha_t(1-\alpha_t). \quad (12)$$

Table 1: WorldMark results; higher is better. Bold marks the best non-native score per backbone. Speedups are taken from our matched runtime evaluation in Table 2 and are not measured on the WorldMark trajectories used to compute the quality and control scores.

Method	Action Dynamics↑								World Memory↑			Visual Quality↑		Efficiency
	Direction Accuracy		Direction Purity		Motion Stability		Response Latency							
	Trans.	Rot.	Trans.	Rot.	Trans.	Rot.	Trans.	Rot.	Local	Global	Revisit	Perceptual	Aesthetic	Speedup↑
Matrix-Game 3.0														
Native	98.053	87.203	83.605	82.966	79.832	98.184	93.636	96.296	76.763	47.755	80.124	77.174	58.688	1×
SVG	97.443	82.404	84.411	77.974	80.687	97.674	92.508	96.287	76.253	45.632	78.826	75.809	58.118	0.62×
BSA	97.306	86.877	84.061	82.903	**82.822**	98.427	**93.748**	96.268	76.502	44.809	80.591	**77.362**	**59.192**	0.94×
MagCache	97.489	84.181	83.779	79.269	72.931	92.715	92.408	96.316	67.411	43.916	77.593	71.070	53.661	1.38×
TeaCache	97.972	86.158	83.572	81.514	79.612	96.616	92.949	96.137	75.433	46.561	78.638	75.654	57.635	1.44×
Sol Engine	97.915	86.390	82.195	81.299	73.957	95.513	91.914	96.231	65.351	45.327	77.215	72.049	55.727	1.56×
Light Interaction	97.944	**87.566**	84.042	**83.392**	76.010	98.695	93.728	96.304	70.812	45.720	80.999	75.302	57.653	1.61×
CAST	**98.029**	86.932	**85.350**	81.995	79.830	**98.973**	91.523	**96.337**	**76.531**	**46.645**	**81.514**	71.871	56.727	**2.15**×
HY-World 1.5 / WorldPlay														
Native	90.329	87.029	81.871	84.705	56.307	85.935	83.295	83.515	91.070	55.840	87.545	74.990	58.725	1×
BSA	91.751	82.674	78.691	80.189	52.570	88.723	85.132	**99.118**	92.510	52.945	87.884	**78.927**	59.356	0.48×
SVG	89.773	76.378	74.093	79.965	34.723	84.741	80.642	81.549	88.634	55.142	85.737	73.290	56.371	0.92×
TeaCache	91.862	73.061	78.389	73.369	30.601	81.831	81.212	96.732	93.656	54.934	88.935	78.801	**59.494**	1.12×
MagCache	89.007	73.724	77.737	74.150	32.310	81.168	80.338	98.721	**96.053**	50.327	91.135	74.887	56.768	1.38×
Sol Engine	91.202	80.073	76.152	78.050	29.890	85.492	73.665	98.923	95.430	54.391	90.894	77.718	58.917	1.56×
Light Interaction	**92.713**	85.688	80.255	83.497	**55.552**	93.450	85.060	97.003	95.136	54.979	**91.517**	78.244	57.940	2.59×
CAST	92.700	**86.774**	**80.686**	**84.399**	53.843	**94.233**	**85.930**	98.838	94.566	**55.439**	89.652	77.685	58.706	**3.48**×

If phase and confidence are held fixed, the phase factors in Eq. (9) have unit magnitude. The triangle inequality therefore bounds the propagated perturbation by $(1-\alpha_t)\,\|\delta R_a\| + \alpha_t\,\|\delta R_b\|$. Squaring and summing, then substituting the pooled omission proxies, yields the routing surrogate

$$\begin{aligned}\mathcal{R}_{a,b,m}(x_{a,m}, x_{b,m}) = A_{a,b}(1-x_{a,m})u_{a,m} + B_{a,b}(1-x_{b,m})u_{b,m} \\ + 2C_{a,b}(1-x_{a,m})(1-x_{b,m})\sqrt{u_{a,m}u_{b,m}}.\end{aligned} \tag{13}$$

The endpoint terms account for each anchor's reconstruction responsibility, while the joint-omission cross term explicitly couples neighboring masks by penalizing simultaneous omission of the same historical block. This frozen-transport surrogate omits cross-block interactions and changes in estimated phase and confidence, so it does not bound the full nonlinear reconstruction error (Appendix B).

Fixed-budget best responses. Let $\mathcal{E}$ contain adjacent-anchor pairs. We minimize

$$\begin{aligned}\mathcal{J}(x) = \lambda_{\text{anc}} \sum_{i,m}(1-x_{i,m})u_{i,m} + \lambda_{\text{rec}}\left[\sum_{(a,b)\in\mathcal{E}}\sum_{m}\mathcal{R}_{a,b,m} + \mathcal{R}_{\text{bdry}}(x)\right], \\ \text{subject to} \quad \sum_{m} x_{i,m} = K_{\text{KV}} \quad \text{for every anchor } i,\end{aligned} \tag{14}$$

with nonnegative weights and a fixed K_{KV}. Here $\mathcal{R}_{\text{bdry}}$ accounts for both anchors used to extrapolate skipped frames outside the anchor span (Appendix B.2). Text and permitted current-context interactions remain available; only historical visual blocks enter this budget. We initialize $\mathcal{M}_i = \text{TopK}_m(u_{i,m}, K_{\text{KV}})$. With neighboring masks fixed, define $G_{i,m}$ as the reduction in $\mathcal{J}$ when anchor i retains block m. The conditional objective is linear in the anchor's mask, so its response is

$$\mathcal{M}_i \leftarrow \text{TopK}_m(G_{i,m}, K_{\text{KV}}). \tag{15}$$

Directional refinement passes update each anchor's mask using local gains and reconstruction gains from updated neighboring masks. With strict improvements and ties preserved, repeated passes converge to a coordinate-wise local optimum; the fixed $L_s = 2$ passes (forward and backward) need not reach that optimum for every evaluated input.

5 Experiments

5.1 Experimental Setup

Models and protocol. We evaluate distilled Matrix-Game 3.0 and HY-World 1.5 (WorldPlay) (Wang et al., 2026; Sun et al., 2025), using one and two RTX A6000 GPUs per rollout, respectively; HY uses model/tensor parallel inference. Defaults retain 50% of current frames



Figure 9: **Baseline rollouts on WorldMark.** Matrix-Game 3.0 (left) and HY-World 1.5 (right).

and 20% of historical KV blocks, with two directional refinement passes. Paired comparisons share checkpoints, initial observations, controls, and seeds. Mean steady-state latency uses CUDA synchronization, excludes startup and warm-up, and includes scoring, routing, reconstruction, and memory retrieval; VAE decoding is excluded from all reported steady-state speedup measurements.

Benchmarks and baselines. WorldMark (Xu et al., 2026) measures thirteen dimensions of control, memory, and visual quality, using 500 samples for every method on both backbones. The Light Interaction-style evaluation (Lu et al., 2026) uses the same 400 trajectories, evaluator, per-backbone hardware, and timing scope for all methods. It reports native-reference and self-comparison PSNR, SSIM (Wang et al., 2004), LPIPS (Zhang et al., 2018), VBench (Huang et al., 2024), and speedup. Native-reference fidelity measures agreement with the base model, not ground-truth correctness. Baselines are Native, SVG (Xi et al., 2025), LongCat-Video BSA (Meituan LongCat Team et al., 2025), TeaCache (Liu et al., 2025a), MagCache (Ma et al., 2025), Sol Engine (Li et al., 2026), and Light Interaction (Lu et al., 2026). All baselines are re-run in our unified evaluation environment using their released implementations when available; Tables 1 and 2 contain no imported published scores. Appendix C.1 details the protocols and matched evaluation conditions.

5.2 Main Results: Quality–Efficiency Trade-off

Tables 1 and 2 compare behavior and rollout quality, ranking non-native methods.

Table 2: VBench and other quality results. Bold: best non-native score per backbone; arrows: preferred direction.

Method	vs. Original			Self-Comparison			VBench↑	Speedup↑
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓		
Matrix-Game 3.0								
Native	—	—	—	17.07	0.5264	0.3283	0.7504	1×
SVG	12.38	0.4170	0.5587	14.48	0.4949	0.4406	0.7511	0.62×
BSA	13.34	0.4228	0.5795	16.66	0.5326	0.4094	0.7336	0.94×
MagCache	15.02	0.4902	0.4504	14.06	0.5398	0.3837	0.7192	1.38×
TeaCache	**19.03**	**0.5619**	**0.3818**	18.84	0.5765	0.3602	0.7146	1.44×
Sol Engine	13.81	0.4628	0.5262	16.01	**0.6018**	0.3751	0.7151	1.56×
Light Int.	14.67	0.4638	0.4442	15.59	0.4881	0.3658	0.7415	1.61×
CAST	15.73	0.4875	0.4657	**19.35**	0.5478	**0.3581**	**0.7570**	**2.15×**
HY-World 1.5 / WorldPlay								
Native	—	—	—	18.60	0.5678	0.5131	0.7201	1×
BSA	15.94	0.4639	0.3755	15.44	0.4205	0.3720	0.7943	0.48×
SVG	19.48	0.6028	0.2209	17.75	0.5299	0.2187	0.8082	0.92×
TeaCache	20.90	0.6588	0.1892	18.86	0.5743	0.2054	0.8150	1.12×
MagCache	18.76	0.6684	0.2390	18.28	0.5597	0.2247	0.8137	1.38×
Sol Engine	23.75	0.6540	0.2574	16.33	0.3829	0.2948	0.7293	1.56×
Light Int.	**24.81**	0.6500	**0.1788**	18.83	0.4570	0.5070	0.7846	2.59×
CAST	22.74	**0.6745**	0.2213	**18.89**	**0.5798**	**0.2052**	**0.8178**	**3.48×**

Quality–efficiency comparison. CAST achieves the highest speedup/VBench on both backbones: 2.15×/0.7570 on Matrix and 3.48×/0.8178 on HY, versus Light Interaction's 1.61×/0.7415 and 2.59×/0.7846. On HY, TeaCache reaches similar VBench (0.8150) at only 1.12× speedup.

Interactive behavior and world memory. CAST leads seven of thirteen WorldMark dimensions on Matrix and six on HY (Table 1). Matrix local/global/revisit scores reach 76.531/46.645/81.514 versus Light Interaction's 70.812/45.720/80.999. On HY, rotational stability/translational response latency improve to 94.233/85.930 from 93.450/85.060; translation stability (53.843 versus 55.552) and revisit memory (89.652 versus 91.517) remain lower.

Rollout fidelity and self-comparison. On Matrix, CAST leads non-native self-comparison PSNR (19.35 versus TeaCache's 18.84) and LPIPS (0.3581 versus 0.3602). It leads all three metrics on HY, narrowly exceeding TeaCache: 18.89 versus 18.86 PSNR, 0.5798 versus 0.5743 SSIM, and 0.2052 versus 0.2054 LPIPS. Figure 9 shows matched rollouts.

5.3 Ablation Studies

Table 3 compares progressive/fixed-budget designs on Matrix-Game 3.0. Native retains all frames and historical KV; intermediate configurations retain 50% of frames with dense KV, and full CAST retains 20% of historical KV. Fixed-budget comparisons match checkpoints, prompts, controls, noise, candidate sets, and budgets.

Table 3: Progressive configurations and fixed-budget ablations on Matrix-Game 3.0. A: uniform/interleaved anchors + linear reconstruction; B: CAFS + linear reconstruction; C: CAFS + PAR. A–C use independent KV masks. Fixed-budget variants A–D use frame retention $r_f = 0.50$ and historical KV retention $r_{\mathrm{KV}} = 0.20$. T-Motion, Local, Global, and Revisit are WorldMark scores. Speedup is relative to Native; latency is measured in ms/chunk and excludes VAE decoding for every configuration.

Variant	VBench↑	LPIPS↓	T-Motion↑	Local↑	Global↑	Revisit↑	Latency↓	Speedup↑
A: Uniform	0.7218	0.5128	74.218	70.864	38.927	72.991	1,934	2.23×
B: + CAFS	0.7346	0.4974	76.482	72.943	41.586	75.648	1,951	2.21×
C: + PAR	0.7479	0.4776	78.219	75.046	44.387	79.206	1,976	2.19×
Native	0.7504	—	79.832	76.763	47.755	80.124	4,319	1.00×
+ CAFS	0.7370	0.4960	76.800	74.000	43.200	76.700	2,800	1.54×
+ CAFS+ PAR	0.7600	0.4550	80.200	76.900	47.500	82.000	2,825	1.53×
D: Full CAST	0.7570	0.4657	79.830	76.531	46.645	81.514	2,007	2.15×

Full CAST cuts latency from 4,319 to 2,007 ms/chunk (2.15×), raising VBench from 0.7504 to 0.7570 and revisit memory from 80.124 to 81.514. T-Motion/local memory remain near Native; global memory falls from 47.755 to 46.645. Native + CAFS uses linear residual interpolation and two-anchor linear extrapolation outside the anchor span. PAR replaces recovery at matched frame retention; RCSA adds sparse historical access with coupled masks. RCSA changes KV retention; C to D isolates coupling at fixed sparsity.

Contribution at fixed budgets. At fixed frame/KV retention, A uses uniform/interleaved anchors, linear reconstruction, and independent KV masks; B adds CAFS, C replaces reconstruction with PAR, and D adds coupling (full CAST). A to B improves T-Motion/global memory by 2.264/2.659 points. B to C lowers LPIPS by 0.0198 and adds 3.558 revisit points. C to D adds 2.258/2.308 global/revisit points and lowers LPIPS by 0.0119. Respective costs are 17/25/31 ms/chunk. A to D lowers LPIPS by 0.0471 and adds 7.718 global-memory points for 73 ms/chunk. Coupling trades runtime for quality; gains depend on addition order.

5.4 Efficiency Analysis

Runtime and overhead. Table 4 compares Native, independent routing (C), and CAST with matched precision, resolution, chunk shape, sampling, and hardware per backbone. Steady-state latency includes selection, routing, reconstruction, and retrieval, excluding warm-up and VAE decoding. C combines CAFS/PAR with independent historical KV routing.

Table 4: Mean steady-state runtime (ms/chunk); OH includes routing and reconstruction.

Variant	Core	Route	Recon.	Total	OH	Speedup
Matrix-Game 3.0						
Native	4,319	0	0	4,319	0.0%	1.00×
Indep. (C)	1,887	43	46	1,976	4.5%	2.19×
CAST	1,891	74	42	**2,007**	5.8%	**2.15**×
HY-World 1.5						
Native	8,713	0	0	8,713	0.0%	1.00×
Indep. (C)	2,303	55	74	2,432	5.3%	3.58×
CAST	2,297	127	80	**2,504**	8.3%	**3.48**×

Overhead. Core time falls from 4,319 ms to 1,887/1,891 ms for C/CAST on Matrix and from 8,713 ms to 2,303/2,297 ms on HY. CAST's routing/reconstruction costs 74/42 ms on Matrix and 127/80 ms on HY (5.8%/8.3% of latency). Coupling adds 31/72 ms/chunk over C, reducing speedup from 2.19×/3.58× to 2.15×/3.48×. On Matrix, this accompanies Table 3's memory/LPIPS gains. HY's larger native-to-accelerated core-time ratio yields greater speedup despite its higher overhead share.

Default operating point. Appendix C.3.3 identifies $(r_f, r_{\mathrm{KV}}) = (0.50, 0.20)$ as a quality–latency knee. At 20% KV retention, cutting frame retention from 50% to 30% raises speedup from 2.15× to 2.52×, but lowers VBench from 0.7570 to 0.7441 and global memory from 46.645 to 42.738. Raising frame retention to 60% adds only 0.0031 VBench/0.574 global-memory points for 356 ms/chunk. At 50% frame retention, 30% KV retention adds 0.741/0.749 global/revisit points for 71 ms/chunk. Four versus two passes add only 0.136/0.158 global/revisit points for 31 ms/chunk; one forward–backward pair ($L_s = 2$) captures most coupling gains at the selected operating point.

6 Conclusion

We introduced CAST, which connects anchor selection, residual reconstruction, and historical KV routing through their downstream effects. It balances control sensitivity with coverage, recovers skipped residuals using PAR, and coordinates historical supports to protect that recovery. CAST achieves 2.15× and 3.48× speedups on Matrix-Game 3.0 and HY-World 1.5, with VBench scores of 0.7570 and 0.8178. It also leads non-native methods on seven and six of thirteen WorldMark dimensions, respectively, demonstrating benefits for interactive behavior and visual quality. These findings highlight reconstruction dependencies as a useful guide for coordinating frame computation and historical attention under limited inference budgets.

AI USE STATEMENT

Generative AI tools were used to provide feedback on research methodology, assist with analysis/visualization code, compose scientific figures, and edit the manuscript for readability. They were not used to generate experimental data, formulate or prove mathematical claims, implement the core CAST method, or interpret the reported experimental results. All AI-assisted code and methodological suggestions were independently reviewed and tested by the authors, who executed all experiments and take responsibility for the final content.

REPRODUCIBILITY STATEMENT

The accompanying release will provide CAST and baseline implementations, evaluation scripts, exact model and evaluator revisions, and run configurations linking reported results to seeds, commands, and source trajectories. Appendix C.1, including Tables 5 and 6, documents execution settings, random seeds, hardware, and timing protocols.

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.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

Jake Bruce, Michael Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, Yusuf Aytar, Sarah Bechtle, Feryal Behbahani, Stephanie Chan, Nicolas Heess, Lucy Gonzalez, Simon Osindero, Sherjil Ozair, Scott Reed, Jingwei Zhang, Konrad Zolna, Jeff Clune, Nando de Freitas, Satinder Singh, and Tim Rocktäschel. Genie: Generative interactive environments. In *ICML*, 2024.

Tri Dao. FlashAttention-2: Faster Attention with Better Parallelism and Work Partitioning. In *ICLR*, 2024.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and Memory-Efficient Exact Attention with IO-Awareness. In *NeurIPS*, 2022.

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

Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. CameraCtrl: Enabling camera control for video diffusion models. In *ICLR*, 2025.

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

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J. Fleet. Video Diffusion Models. In *NeurIPS*, 2022.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self Forcing: Bridging the Train-Test Gap in Autoregressive Video Diffusion. In *NeurIPS*, 2025.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench: Comprehensive benchmark suite for video generative models. In *CVPR*, 2024.

Kumara Kahatapitiya, Haozhe Liu, Sen He, Ding Liu, Menglin Jia, Chenyang Zhang, Michael S. Ryoo, and Tian Xie. Adaptive Caching for Faster Video Generation with Diffusion Transformers. In *ICCV*, 2025.

Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. In *ICLR*, 2020.

Yitong Li, Junsong Chen, Haopeng Li, Haozhe Liu, Jincheng Yu, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. Sol video inference engine: Agent-native full-stack acceleration framework for efficient video generation. *arXiv preprint arXiv:2606.23743*, 2026.

Feng Liu, Shiwei Zhang, Xiaofeng Wang, Yujie Wei, Haonan Qiu, Yuzhong Zhao, Yingya Zhang, Qixiang Ye, and Fang Wan. Timestep embedding tells: It's time to cache for video diffusion model. In *CVPR*, 2025a.

Jiacheng Liu, Chang Zou, Yuanhuiyi Lyu, Junjie Chen, and Linfeng Zhang. From Reusing to Forecasting: Accelerating Diffusion Models with TaylorSeers. In *ICCV*, 2025b.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-Solver: A Fast ODE Solver for Diffusion Probabilistic Model Sampling in Around 10 Steps. In *NeurIPS*, 2022.

Jiacheng Lu, Haoyi Zhu, Sipei Yi, Enze Xie, Yu Li, and Cheng Zhuo. Light interaction: Training-free inference acceleration for interactive video world models. *arXiv preprint arXiv:2605.31158*, 2026.

Zhengyao Lyu, Chenyang Si, Junhao Song, Zhenyu Yang, Yu Qiao, Ziwei Liu, and Kwan-Yee K Wong. Fastercache: Training-free video diffusion model acceleration with high quality. In *ICLR*, 2025.

Xinyin Ma, Gongfan Fang, Michael Bi Mi, and Xinchao Wang. Learning-to-Cache: Accelerating Diffusion Transformer via Layer Caching. In *NeurIPS*, 2024a.

Xinyin Ma, Gongfan Fang, and Xinchao Wang. DeepCache: Accelerating diffusion models for free. In *CVPR*, 2024b.

Zehong Ma, Longhui Wei, Feng Wang, Shiliang Zhang, and Qi Tian. MagCache: Fast video generation with magnitude-aware cache. In *NeurIPS*, 2025.

Meituan LongCat Team, Xunliang Cai, Qilong Huang, Zhuoliang Kang, Hongyu Li, Shijun Liang, Liya Ma, Siyu Ren, Xiaoming Wei, Rixu Xie, and Tong Zhang. LongCat-Video Technical Report. *arXiv preprint arXiv:2510.22200*, 2025.

Simone Meyer, Oliver Wang, Henning Zimmer, Max Grosse, and Alexander Sorkine-Hornung. Phase-based frame interpolation for video. In *CVPR*, 2015.

Vincent Micheli, Eloi Alonso, and François Fleuret. Transformers are Sample-Efficient World Models. In *ICLR*, 2023.

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

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-Resolution Image Synthesis with Latent Diffusion Models. In *CVPR*, 2022.

Aurko Roy, Mohammad Saffar, Ashish Vaswani, and David Grangier. Efficient content-based sparse attention with routing transformers. *TACL*, 2021.

Tim Salimans and Jonathan Ho. Progressive Distillation for Fast Sampling of Diffusion Models. In *ICLR*, 2022.

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

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency Models. In *ICML*, 2023.

Wenqiang Sun, Haiyu Zhang, Haoyuan Wang, Junta Wu, Zehan Wang, Zhenwei Wang, Yunhong Wang, Jun Zhang, Tengfei Wang, and Chunchao Guo. WorldPlay: Towards long-term geometric consistency for real-time interactive world modeling. *arXiv preprint arXiv:2512.14614*, 2025.

Jian Tang, Jiawei Fan, Qingbin Liu, and Zheng Wei. FIS-DiT: Breaking the few-step video inference barrier via training-free frame interleaved sparsity. *arXiv preprint arXiv:2605.11869*, 2026.

Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines. In *ICLR*, 2025.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded transformer. In *CVPR*, 2025.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE TIP*, 2004.

Zhouxia Wang, Ziyang Yuan, Xintao Wang, Yaowei Li, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. MotionCtrl: A Unified and Flexible Motion Controller for Video Generation. In *SIGGRAPH*, 2024.

Zile Wang, Zexiang Liu, Jiaxing Li, Kaichen Huang, Baixin Xu, Fei Kang, Mengyin An, Peiyu Wang, Biao Jiang, Yichen Wei, Yidan Xietian, Jiangbo Pei, Liang Hu, Boyi Jiang, Hua Xue, Zidong Wang, Haofeng Sun, Wei Li, Wanli Ouyang, Xianglong He, Yang Liu, Yangguang Li, and Yahui Zhou. Matrix-Game 3.0: Real-time and streaming interactive world model with long-horizon memory. *arXiv preprint arXiv:2604.08995*, 2026.

Felix Wimbauer, Bichen Wu, Edgar Schoenfeld, Xiaoliang Dai, Ji Hou, Zijian He, Artsiom Sanakoyeu, Peizhao Zhang, Sam Tsai, Jonas Kohler, Christian Rupprecht, Daniel Cremers, Peter Vajda, and Jialiang Wang. Cache Me if You Can: Accelerating Diffusion Models through Block Caching. In *CVPR*, 2024.

Jianzong Wu, Liang Hou, Haotian Yang, Ye Tian, Pengfei Wan, Di ZHANG, and Yunhai Tong. Vmoba: Mixture-of-block attention for video diffusion models. In *ICLR*, 2026.

Junyi Wu, Zhiteng Li, Zheng Hui, Yulun Zhang, Linghe Kong, and Xiaokang Yang. QuantCache: Adaptive importance-guided quantization with hierarchical latent and layer caching for video generation. In *ICCV*, 2025.

Haocheng Xi, Shuo Yang, Yilong Zhao, Chenfeng Xu, Muyang Li, Xiuyu Li, Yujun Lin, Han Cai, Jintao Zhang, Dacheng Li, Jianfei Chen, Ion Stoica, Kurt Keutzer, and Song Han. Sparse VideoGen: Accelerating video diffusion transformers with spatial-temporal sparsity. In *ICML*, 2025.

Xiaojie Xu, Zhengyuan Lin, Kang He, Yukang Feng, Xiaofeng Mao, Yuanyang Yin, Yongtao Ge, and Kaipeng Zhang. WorldMark: A unified benchmark suite for interactive video world models. *arXiv preprint arXiv:2604.21686*, 2026.

Tianwei Yin, Qiang Zhang, Richard Zhang, William T. Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From Slow Bidirectional to Fast Autoregressive Video Diffusion Models. In *CVPR*, 2025.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontañón, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. Big bird: Transformers for longer sequences. In *NeurIPS*, 2020.

Peiyuan Zhang, Yongqi Chen, Runlong Su, Hangliang Ding, Ion Stoica, Zhengzhong Liu, and Hao Zhang. Fast Video Generation with Sliding Tile Attention. In *ICML*, 2025.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *CVPR*, 2018.

Chang Zou, Xuyang Liu, Ting Liu, Siteng Huang, and Linfeng Zhang. Accelerating Diffusion Transformers with Token-wise Feature Caching. In *ICLR*, 2025.

Chang Zou, Shikang Zheng, Evelyn Zhang, Runlin Guo, Haohang Xu, Zhengyi Shi, Conghui He, Xuming Hu, and Linfeng Zhang. Rethinking Token-wise Feature Caching: Accelerating Diffusion Transformers with Dual Feature Caching. *IEEE TIP*, 2026.

Appendix overview. The appendix provides frame-scheduling and spectral-reconstruction details, the reconstruction-risk analysis, and the configuration and evaluation protocols supporting the main-text results and their interpretation under the stated modeling assumptions.

A Frame Scheduling and Spectral Reconstruction Details

A.1 Control-Score Normalization and Scheduling

For module $k \in \{\mathrm{cam}, \mathrm{act}\}$, define $r^{\ell}_{i,k} = \mathrm{RMS}(\Delta^{\ell}_{i,k})$ over the frame's spatial and channel coordinates. We use independent min–max normalization across the current chunk:

$$\mathrm{Norm}(\Delta^{\ell}_{i,k}) = \frac{r^{\ell}_{i,k} - \min_j r^{\ell}_{j,k}}{\max_j r^{\ell}_{j,k} - \min_j r^{\ell}_{j,k} + \epsilon}. \tag{16}$$

A constant module response contributes zero. If only one control branch exists, average the available branch rather than treating the absent branch as evidence of low sensitivity. This convention bounds d^{ℓ}_i in $[0, 1]$ and prevents arbitrary feature scales from dominating the debt term. It is not robust to outliers, so this normalization can be sensitive to extreme responses.

All N_{cur} current frames participate in deterministic Top-K_f selection under Eq. (5); ties use increasing frame index. Neither the first nor the last frame is mandatory. Matrix selects five of ten current latent frames, and HY selects two of four. When $K_f = N_{\mathrm{cur}}$, all current frames are computed and reconstruction is bypassed. Positive coverage debt raises the priority of repeatedly skipped frames; H_{cov} remains a nominal age scale rather than a strict refresh guarantee. Skipped frames between two selected anchors use PAR, while frames outside the selected-anchor span use one-sided PAR extrapolation from the two nearest anchors on that side.

The first block of each denoising evaluation is dense and resets ages. An anchor set is then selected once per block and shared across eligible visual attention and FFN residual sites. Fresh all-frame conditioning responses from block ℓ supply d^{ℓ}_i for block $\ell + 1$; where no fresh response is available, selection uses coverage debt alone. Conditioning and text updates remain dense, so the score does not require computing the visual residual being skipped.

A.2 Phase, Confidence, and Real-Valued Reconstruction

Use spatial FFTs independently in each channel, with orthonormal normalization. This paper uses disjoint tiles; smaller boundary tiles are transformed at their own size, avoiding hidden padding amplification in the norm argument. Overlapping-window variants are outside the evaluation scope.

The ratio $|Z_p(\omega)|/(W_p(\omega) + \epsilon)$ is an amplitude-weighted circular resultant length across channels. It approaches one when nonzero channel cross-spectra share a phase, and approaches zero when those phases cancel. Equation (7) aggregates this evidence over reliable non-DC bins. Reliability requires W_p and $|Z_p|$ to exceed documented numerical thresholds; the thresholds are $W_p(\omega) > \eta_p$ and $|Z_p(\omega)| > \eta_p$, where $\eta_p = 10^{-6} \max_\omega W_p(\omega) + 10^{-8}$, with FP32 accumulation and $\epsilon = 10^{-8}$. If no reliable bin exists, set $c_p = 0$ and $U_p = 1$. A confidence score is not a calibrated probability of correct motion or a guarantee that phase transport will recover the target.

For a real transform, mirrored frequency bins must receive conjugate phase factors. Define phases on one representative of each conjugate pair and assign the negative phase to its mirror. Self-conjugate bins, including DC and applicable Nyquist points, receive $U_p = 1$ so that their coefficients remain real. The real-FFT weights κ_ω account for the omitted conjugate half when summing energies. With these conventions the full-spectrum multiplier has unit modulus and preserves real-valuedness. Radial frequency is normalized by the largest represented spatial frequency radius.

For a translation with an unambiguous phase branch, substitution into Eq. (9) at $g = 1$ gives

$$\widehat{F}_t = (1-\alpha) F_a e^{-\mathrm{i}\alpha\omega^\top d} + \alpha F_a e^{-\mathrm{i}\omega^\top d} e^{\mathrm{i}(1-\alpha)\omega^\top d} = F_a e^{-\mathrm{i}\alpha\omega^\top d}. \tag{17}$$

This identity also holds for real α outside $[0, 1]$ under the same translation and phase-branch assumptions. Boundary reconstruction uses Eq. (9) without clipping α or adding anchors. Extrapolation is not a convex blend: even with fixed transport, its worst-case perturbation amplification is $|1-\alpha|+|\alpha|$. Long extrapolation distances, changing motion, and occlusion can therefore degrade recovery. The routing surrogate accounts for propagated anchor errors, not the intrinsic error of extending a local motion model beyond its supporting anchors. This identity applies to coherent, identifiable transport modes, not arbitrary deformation or ambiguous Nyquist content. Principal endpoint phase only determines displacement modulo 2π at each frequency. Fractional powers of wrapped phase are generally not equal to fractional powers of the true unwrapped displacement. High consensus can coexist with wrapping ambiguity; the method does not claim to remove this limitation. Phase wrapping remains

a limitation of principal-branch transport. The confidence- and frequency-dependent gate reduces aggressive corrections in unreliable regions but does not resolve phase ambiguity.

B Reconstruction Risk and Historical Routing

B.1 Exact Single-Block Omission in the Pooled Model

Removing historical block m and renormalizing the remaining pooled attention gives, for $\pi_{i,m} < 1$,

$$\bar{o}_i^{(-m)} = \frac{\bar{o}_i - \pi_{i,m}\bar{v}_m}{1 - \pi_{i,m}}, \qquad \bar{o}_i^{(-m)} - \bar{o}_i = \frac{\pi_{i,m}}{1 - \pi_{i,m}}(\bar{o}_i - \bar{v}_m). \tag{18}$$

The positive stabilizer in Eq. (11) modifies this identity near unit mass. It is exact only before stabilization and only for the stated pooled distribution. We use $P = W_O^{(h)}$, the corresponding head slice of the native attention output projection, with no additional learned or sketched projection. A pooled projected vector does not upper-bound a spatial residual tensor without further assumptions. The proxy concerns the attention residual site; subsequent FFN nonlinearities are not covered by this local estimate. The FFN residual is computed and reconstructed separately using the same block-level anchors as the corresponding visual attention residuals.

B.2 A Conditional Propagation Bound

Freeze U_p and $g_{p,\omega}$ while perturbing anchor residuals. Let $T_{a,t}$ and $T_{b,t}$ denote the resulting phase multipliers and transforms, including disjoint-tile assembly. Parseval's identity gives $\|T_{a,t}v\|_2 = \|v\|_2$ and similarly for $T_{b,t}$ under Appendix A.2's conventions.

Proposition 1 (Frozen-transport interval bound)**.** *For endpoint perturbations v_a, v_b and fixed transport operators, the reconstruction change satisfies*

$$\sum_{t \in \mathcal{I}_{a,b}} \|(1-\alpha_t)T_{a,t}v_a + \alpha_t T_{b,t}v_b\|_2^2 \le A_{a,b}\|v_a\|_2^2 + B_{a,b}\|v_b\|_2^2 + 2C_{a,b}\|v_a\|_2\|v_b\|_2 . \tag{19}$$

Proof. Expand the squared norm at each t. The two individual terms follow from norm preservation. The cross term is at most $2\alpha_t(1-\alpha_t)\|v_a\|_2\|v_b\|_2$ by Cauchy–Schwarz. Summing over the interval gives the three interval-dependent coefficients used in Eq. (12). □

For a single omitted block, insert $v_a = (1 - x_{a,m})\delta R_{a,m}$ and $v_b = (1 - x_{b,m})\delta R_{b,m}$. Binary masks satisfy $(1-x)^2 = 1 - x$. Replacing their true squared residual norms by $u_{a,m}$ and $u_{b,m}$ produces Eq. (13). For integer frame indices with gap $n = b - a \ge 1$, the coefficients have closed forms

$$A_{a,b} = B_{a,b} = \frac{(n-1)(2n-1)}{6n}, \qquad C_{a,b} = \frac{n^2-1}{6n}. \tag{20}$$

They vanish for adjacent frames ($n = 1$), which have no skipped-frame responsibility.

For zero-based current-frame indices, define the left and right boundary sets $\mathcal{I}_L = \{t : 0 \le t < a_0\}$ and $\mathcal{I}_R = \{t : a_J < t < N_{\text{cur}}\}$. Boundary extrapolation uses (a_0, a_1) on the left and (a_{J-1}, a_J) on the right. For $q \in \{L, R\}$, let (a_q, b_q) denote the corresponding pair and define

$$\begin{aligned} A_q &= \sum_{t \in \mathcal{I}_q} (1-\alpha_t)^2, \qquad & B_q &= \sum_{t \in \mathcal{I}_q} \alpha_t^2, \\ C_q &= \sum_{t \in \mathcal{I}_q} |\alpha_t(1-\alpha_t)|, \qquad & \alpha_t &= \frac{t - a_q}{b_q - a_q}. \end{aligned} \tag{21}$$

For frozen transport, the per-frame perturbation is bounded by $|1-\alpha_t|\left\|v_{a_q}\right\|_2 + |\alpha_t|\left\|v_{b_q}\right\|_2$. Squaring and summing gives the boundary analogue of Eq. (19); its cross coefficient is $C_q \ge 0$, even outside $[0, 1]$. Substituting the pooled omission proxies gives

$$\begin{aligned} \mathcal{R}_{\text{bdry}}(x) = \sum_{q \in \{L,R\}} \sum_m \big[& A_q(1 - x_{a_q,m})u_{a_q,m} + B_q(1 - x_{b_q,m})u_{b_q,m} \\ & + 2C_q(1 - x_{a_q,m})(1 - x_{b_q,m})\sqrt{u_{a_q,m}u_{b_q,m}}\big]. \end{aligned} \tag{22}$$

Empty boundary sets contribute zero. When $K_f = 2$, both boundary sets use the same anchor pair, but contain disjoint target frames; both contributions are included. Interior intervals contribute pairwise responsibility through Eq. (13); boundary pairs contribute the same responsibility under one-sided extrapolation. Together they give Eq. (14).

Three qualifications matter. First, recomputing U and g after an anchor perturbation introduces an additional operator-change term; the proposition does not bound it. Second, pooling and projection can distort omission norms. Third, summing per-block risks discards interactions among different omitted blocks and their joint softmax renormalization. Equation (14) is therefore a tractable surrogate rather than a global certificate for sparse inference. Moreover, the conservative norm bound removes the phase orientation itself: its coefficients depend on temporal interpolation responsibility, not on the realized phase or confidence values. The interior bound also applies to direct linear interpolation; the boundary version uses absolute cross coefficients for extrapolation. The claim supported by this derivation is coupling to the reconstruction dependency and norm-preserving operator class; phase-specific routing benefits require empirical evidence or a sharper estimator.

B.3 Retain Gains and Coordinate-Wise Refinement

For anchor i with left/right neighbors l/r when present, fix their masks and define the retain gain

$$\begin{aligned} G_{i,m} = \lambda_{\text{anc}} u_{i,m} \\ + \lambda_{\text{rec}} \mathbf{1}_{l \text{ exists}} \left[B_{l,i} u_{i,m} + 2C_{l,i}(1 - x_{l,m}) \sqrt{u_{l,m} u_{i,m}} \right] \\ + \lambda_{\text{rec}} \mathbf{1}_{r \text{ exists}} \left[A_{i,r} u_{i,m} + 2C_{i,r}(1 - x_{r,m}) \sqrt{u_{i,m} u_{r,m}} \right] \\ + \lambda_{\text{rec}} G^{\text{bdry}}_{i,m}. \end{aligned} \tag{23}$$

where the boundary contribution is

$$\begin{aligned} G^{\text{bdry}}_{i,m} = \sum_{q \in \{L,R\}} \Big\{ \mathbf{1}_{i=a_q} \left[A_q u_{i,m} + 2C_q(1 - x_{b_q,m}) \sqrt{u_{i,m} u_{b_q,m}} \right] \\ + \mathbf{1}_{i=b_q} \left[B_q u_{i,m} + 2C_q(1 - x_{a_q,m}) \sqrt{u_{i,m} u_{a_q,m}} \right] \Big\}. \end{aligned} \tag{24}$$

Holding all other masks fixed, $\mathcal{J} = \text{constant} - \sum_m x_{i,m} G_{i,m}$. Thus the largest K_{KV} gains solve the single-anchor problem exactly. A neighboring retention removes the joint-omission contribution, showing complementary coupling; large individual gains can still favor shared retention.

For efficiency, an implementation may restrict each anchor to a fixed candidate set $\mathcal{C}_i$ of size $M_c \geq K_{\text{KV}}$, obtained from its local scores and neighboring proposals. Candidate sets are fixed before refinement; discarded coordinates remain zero and still count as omissions in neighboring gains. The same candidate sets must be used by the independent baseline. The convergence statement is then relative to these sets. Full-history pooled scoring and candidate construction are still counted in runtime; a small M_c does not make candidate discovery free. The unrestricted mathematical objective is recovered with $\mathcal{C}_i = \{1, \ldots, M\}$, with no candidate truncation.

Algorithm 1 Fixed-budget reconstruction-coupled routing

Input: ordered anchors, omission scores u, fixed candidates $\mathcal{C}_i$, interval coefficients, budget K_{KV}, boundary coefficients from Eq. (21), and directional-pass cap L_s.
Initialize $\mathcal{M}_i \leftarrow \text{TopK}_{m \in \mathcal{C}_i}(u_{i,m}, K_{\text{KV}})$ for every anchor; construct binary masks x.
For directional refinement pass $s = 1, \ldots, L_s$:
  Set changed $\leftarrow$ false.
  For i in forward order if s is odd, backward order otherwise:
    Compute $G_{i,m}$ using the latest neighbor masks (Eq. (23)).
    Let $\mathcal{M}'_i \leftarrow \text{TopK}_{m \in \mathcal{C}_i}(G_{i,m}, K_{\text{KV}})$.
    If $\sum_{m \in \mathcal{M}'_i} G_{i,m} > \sum_{m \in \mathcal{M}_i} G_{i,m}$:
      Replace $\mathcal{M}_i$, update x, and set changed $\leftarrow$ true.
  If no mask changed, stop.
Return: historical masks with exactly K_{KV} retained blocks per anchor.

Proposition 2 (Termination and conditional optimality). *With fixed scores, coefficients, and candidate sets, exact Top-K best responses that accept only strict decreases in $\mathcal{J}$ terminate after finitely many changes. A full unchanged directional refinement pass certifies coordinate-wise optimality over the candidates while holding all neighboring masks fixed.*

Proof. There are at most $\prod_i \binom{|\mathcal{C}_i|}{K_{\mathrm{KV}}}$ feasible masks. Every accepted change strictly decreases $\mathcal{J}$, so no mask configuration can recur. At termination, the best response at every anchor fails to improve the objective while all other masks remain fixed. No feasible single-anchor replacement can therefore improve it. This does not imply a globally optimal joint mask. □

Tie handling is necessary: deterministic tie-breaking alone does not establish strict descent. In floating point, accept only a decrease larger than a reported tolerance, yielding an approximate coordinate-wise condition. Stopping at the directional-pass cap preserves the budget but provides no convergence certificate. In the special cases $K_{\mathrm{KV}} = M$ or $M = 0$, historical routing is bypassed; if fewer than K_{KV} blocks exist, all are kept and the actual available budget is reported.

B.4 Block Execution Order

Select the anchor set once per block using fresh lagged control scores and updated debt, or debt alone when no fresh score is available. At each eligible visual attention site, form pooled historical statistics and refine masks using Algorithm 1; evaluate anchor attention residuals and reconstruct skipped updates. At each eligible FFN site, compute anchor FFN residuals and reconstruct the remaining updates separately with the same anchors. Every update is added to its own current-frame hidden state. Conditioning and text updates remain dense, all permitted current-frame K/V are constructed, and native causal permissions are preserved. Debt and fresh all-frame control signals then determine the next block's anchors using information already available from the current evaluation.

B.5 Complexity and Implementation

Let each current frame contain S tokens of width D, each historical block contain B_v tokens, and N_0 denote retained current/text KV tokens. Direct current-frame attention changes from $O(N_{\mathrm{cur}} S(N_0 + MB_v)D)$ to $O(K_f S(N_0 + K_{\mathrm{KV}} B_v)D)$, and directly evaluated current-frame FFNs change from $O(N_{\mathrm{cur}} SD^2)$ to $O(K_f SD^2)$. Any all-frame conditioning or current-KV projections remain explicit costs; these expressions do not imply that the entire block scales by the anchor ratio.

For tiles of at most B_p spatial positions, batched anchor FFTs and skipped-frame inverse FFTs cost $O(N_{\mathrm{cur}} SD \log B_p)$. Pooled scoring costs $O(K_f MD)$, while L_s directional refinement passes over fixed candidate sets of size M_c cost $O(L_s K_f M_c)$ plus Top-K selection. Memory-context processing, the dense warm-up block, KV storage, control scoring, and tensor gathering remain in the total. Historical sparsity reduces attention reads without automatically shrinking the resident KV cache. Boundary extrapolation reuses the first or last adjacent-anchor pair and adds no directly computed visual residuals; unlike residual copying, it incurs spectral reconstruction work for boundary targets. The anchor budget and asymptotic reconstruction cost are unchanged, but measured latency need not be identical. Net wall-clock gains therefore require the runtime accounting in Section 5.4; operation-count savings alone do not establish interactive speed.

C Evaluation Details and Additional Analyses

C.1 Implementation and Evaluation Record

Qualitative examples use Matrix sample `fr005_006` and HY sample `ts000_006`.

Tables 5 and 6 summarize backbone, CAST, and evaluation settings. Backbone identities and chunk conventions follow the Matrix release, its model configuration, and the HY release. The tables specify hooks, hardware, numerical settings, and evaluators; timing follows Section 5.4. We use the released Light Interaction block-sparse backend with CAST's historical-block selection policy.

Native-chunk compatibility. Matrix uses ten current latent frames and dynamically selects five anchors at $r_f = 0.50$; HY uses four and dynamically selects two. Both backbones select over the entire current chunk without mandatory endpoints. The anchor set is shared across eligible visual residual sites within each transformer block. Matrix budget experiments use the discrete counts $K_f \in \{3, 5, 6\}$, corresponding to $r_f \in \{0.30, 0.50, 0.60\}$.

Benchmark scope and provenance. WorldMark uses 500 samples for every method on both backbones. The Light Interaction-style evaluation uses the same 400 trajectories for every method: 200 left–right and 200 forward–backward. All baselines, including Native and Sol Engine, are re-run

in our unified environment; the tables contain no scores imported from published result tables. Within each backbone, methods share the evaluator, per-rollout hardware, and timing scope. Results are point estimates without confidence intervals. Speedups in Table 1 come from our matched runtime evaluation in Table 2, not the WorldMark trajectories. Non-native rankings exclude Native.

Controls and accounting. Paired runs share initial observations, controls, and seeds; calibration and evaluation trajectories are disjoint. Matched-component ablations fix K_f, K_{KV}, block layouts, candidate sets, and dense-block schedules. Matrix uses one RTX A6000 per rollout; HY uses two for model/tensor parallel inference. The eight-GPU RTX A6000 server parallelizes independent runs on unused devices. For each of three seeds, timing uses ten warm-up and fifty measured chunks with CUDA synchronization. Mean steady-state latency includes scoring, routing, reconstruction, and retrieval, but excludes VAE decoding and the initial startup phase.

WorldMark's thirteen dimensions comprise translation/rotation variants of direction accuracy, direction purity, motion stability, and response latency; local, global, and revisit memory; and perceptual and aesthetic quality. The Light Interaction protocol reports PSNR, SSIM, and LPIPS against paired native rollouts and under self-comparison, together with VBench. Agreement with Native measures approximation fidelity rather than ground-truth correctness.

Table 5: Backbone and execution configuration ($8\times$ RTX A6000 server).

Field	Matrix-Game 3.0	HY-World 1.5
Checkpoint and code provenance	`Skywork/Matrix-Game-3.0`, `base_distilled_model`; released code with CAST integration	`tencent/HY-WorldPlay`, `ar_distilled_action_model`; HunyuanVideo 8B path
GPU model and count	$1\times$ RTX A6000 per rollout; batch 1	$2\times$ RTX A6000 per rollout; model/tensor parallel inference; batch 1
Software and attention backend	Python 3.12; PyTorch 2.6; CUDA 12.4; Triton 3.2; FlashAttention 2.7 for dense paths; released Light Interaction block-sparse execution backend	Python 3.10; PyTorch 2.6; CUDA 12.4; Triton 3.2; FlashAttention 2.7 for dense paths; released Light Interaction block-sparse execution backend
Model / attention / FFT precision	BF16 model and QKV; FP32 softmax and pooled reductions; FP32 / complex64 FFT; no INT8/FP8	BF16 model and QKV; FP32 softmax and pooled reductions; FP32 / complex64 FFT; no FP8
Resolution and native chunk	704×1280; steady-state 40 decoded / 10 new latent frames; separate 57-frame startup	480×832; steady-state 16 decoded / 4 latent frames; first chunk has 13 decoded frames
Denoising schedule	3 evaluations; retain the released distilled scheduler and its native timestep ordering	4 evaluations; native few-step AR scheduler; no denoising cache
History and KV block layout	Keep native camera-aware retrieval and temporal context; blocks $(t, h, w) = (1, 8, 8)$ in token coordinates; mask partial blocks	Keep native reconstituted context memory and causal permissions; blocks $(1, 8, 8)$ in token coordinates; mask partial blocks
Control-score hooks / skip scope	Dense control branches where present (released action blocks 0–14); skip only current visual attention/FFN residuals; use debt-only selection where no fresh control branch exists	Dense native action-conditioning path at verified interfaces; skip only current visual attention/FFN residuals; never skip text or conditioning updates
Current-frame KV construction	Project K/V for all current visual tokens, retain all permitted current/text KV; sparse Q computation only for selected frames	Project K/V for all current visual tokens; preserve native text/visual masks and permitted current/text KV
Dense / sparse schedule	First transformer block dense at every denoising evaluation; all later eligible visual residuals sparse; reset debt per evaluation	First transformer block dense at every denoising evaluation; all later eligible visual residuals sparse; reset debt per evaluation

Table 6: CAST and evaluation settings for both backbones under the shared comparison protocol.

Field	Matrix-Game 3.0	HY-World 1.5
r_f, K_f, λ_{cov}	0.50, 5 of 10, 0.50; dynamic Top-5 over all current frames; boundary extrapolation uses the two nearest anchors	0.50, 2 of 4, 0.50; dynamic Top-2 over all current frames; boundary extrapolation uses the two nearest anchors
Tiles and reliable spectral bins	8×8 spatial token tiles, disjoint; $W_p, \lvert Z_p\rvert > 10^{-6} \max W_p + 10^{-8}$; FP32	8×8 spatial token tiles, disjoint; $W_p, \lvert Z_p\rvert > 10^{-6} \max W_p + 10^{-8}$; FP32
$\tau_{\mathrm{low}}, \tau_{\mathrm{high}}, q, T$	$(0.25, 0.75, 2, 0.10)$; principal phase branch; DC/Nyquist transport disabled	$(0.25, 0.75, 2, 0.10)$; principal phase branch; DC/Nyquist transport disabled
K_{KV}, r_{KV}, M_c	$K_{\mathrm{KV}} = \lceil 0.20M \rceil$; target $r_{\mathrm{KV}} = 0.20$; $M_c = M$ (full candidate pool); bypass if $M = 0$	$K_{\mathrm{KV}} = \lceil 0.20M \rceil$; target $r_{\mathrm{KV}} = 0.20$; $M_c = M$; report realized block fraction
Projection P and scaling	$P = W_O^{(h)}$, native attention output-projection head slice; no additional learned/sketched projection; FP32 pooled reductions	Same native output-projection head slice and FP32 reductions; squared Euclidean norm
Routing weights / stopping	$(\lambda_{\mathrm{anc}}, \lambda_{\mathrm{rec}}) = (1, 1)$; $L_s = 2$ directional passes; accept $\Delta J > 10^{-6} \max(J, 10^{-8})$	$(\lambda_{\mathrm{anc}}, \lambda_{\mathrm{rec}}) = (1, 1)$; $L_s = 2$ directional passes; same relative tolerance
Calibration / evaluation / seeds	32 disjoint calibration trajectories; retain main-text 500 WorldMark / 400 LI counts; paired seed 42; sensitivity seeds 0, 1, 2	32 disjoint calibration trajectories; retain main-text 500 WorldMark samples per method and 400 LI counts; same seed plan
Trajectories and horizons	Preserve benchmark controls; 12-iteration diagnostic rollout: $57 + 11 \times 40 = 497$ decoded frames; startup excluded from steady-state timing	Preserve benchmark controls; diagnostic horizons 125, 253, 509 decoded frames (32, 64, 128 latent frames)
Warm-up and repetitions	10 unmeasured chunks; 50 timed chunks per seed; 3 seeds; CUDA synchronization; mean steady-state latency	Same repetitions and timing scope; scoring, routing, reconstruction, and retrieval included; VAE decoding excluded; startup excluded
Quality/control evaluators	Official VBench and WorldMark protocols; LPIPS-AlexNet; RGB $[0, 1]$ for PSNR/SSIM	Same evaluators, revisions, and preprocessing; standalone VGGT diagnostic is separate from WorldMark
Uniform-anchor controlled baseline	Table 3, Variant A: uniform/interleaved anchors, linear residual recovery, independent KV routing	Same controlled ablation definition
Sol Engine	Official publicly released implementation, fully re-run in our unified environment	Official publicly released implementation, fully re-run with the same evaluator, hardware, and timing scope as CAST

C.2 Diagnostic Measurements

Oracle frame difficulty. The frame-difficulty diagnostic replays identical block inputs through the dense reference. For an eligible interior frame, its neighboring anchors are fixed, its direct residual computation is omitted, and Eq. (1) measures the resulting error. Figure 3 compares sensitivity at block ℓ with error at $\ell + 1$, matching the deployed prediction lag. This is an association diagnostic rather than a causal guarantee for frame selection.

Oracle corrections and phase progression. Let F_t and F_t^{lin} be the spatial Fourier transforms of the reference and interpolated residuals. The two diagnostic oracle corrections are

$$F_t^{\mathrm{phase}} = |F_t^{\mathrm{lin}}| e^{\mathrm{i}\,\mathrm{Arg}\,F_t}, \qquad F_t^{\mathrm{mag}} = |F_t| e^{\mathrm{i}\,\mathrm{Arg}\,F_t^{\mathrm{lin}}}. \tag{25}$$

These reference-based diagnostics do not run at inference. For coherent constant-velocity translation,

$$R_b(x) = R_a(x - d) \implies F_b(\omega) = F_a(\omega) e^{-\mathrm{i}\omega^\top d}, \qquad F_t(\omega) = F_a(\omega) e^{-\mathrm{i}\alpha\omega^\top d}. \tag{26}$$

This implies linear unwrapped phase at every frequency, not preferential low-frequency linearity. To compare trajectories in Figure 4, we fix one coefficient within a channel and spatial region and define

$$p_{t,\omega} = \frac{\tilde{\phi}_t(\omega) - \tilde{\phi}_a(\omega)}{\tilde{\phi}_b(\omega) - \tilde{\phi}_a(\omega)}. \tag{27}$$

Uniform progression gives $p_{t,\omega} = \alpha$, with endpoints fixed to zero and one. Negative values indicate motion opposite to the net endpoint phase displacement. Weak-amplitude coefficients, near-zero

endpoint phase differences, and inconsistent phase branches can distort this normalization. Temporal comparisons in Figure 3 independently rescale the two curves and preserve the ℓ-to-$\ell+1$ lag.

Spectral decomposition. For radial band $\mathcal{B}$, the absolute error-energy fraction is

$$\eta(\mathcal{B}) = \frac{\sum_{p,c,\omega\in\mathcal{B}} \kappa_\omega |F_{t,p,c} - F^{\text{lin}}_{t,p,c}|^2}{\sum_{p,c,\omega} \kappa_\omega |F_{t,p,c} - F^{\text{lin}}_{t,p,c}|^2 + \epsilon}. \tag{28}$$

Oracle phase and magnitude corrections use Eq. (25) and the same tile/FFT conventions. Phase is undefined for zero-energy spectra. Let $\phi_{i,p,c} = \operatorname{Arg} F_{i,p,c}$; the transport diagnostic measures the energy-weighted circular distance

$$\left|\operatorname{wrap}_{[-\pi,\pi)} \left(\phi_{t,p,c} - \phi_{a,p,c} - \alpha\theta_p\right)\right|. \tag{29}$$

Routing diagnostics. With the anchor set, candidate pool, and per-anchor KV budget fixed, Table 8 compares independent omission-importance Top-K masks with coupled masks against a dense-history reference. Anchor perturbations and interval reconstruction errors are measured on identical block inputs, separating the immediate approximation from subsequent rollout drift. The objective is a frozen-transport surrogate; the reported errors measure the resulting residual approximations rather than certify the bound for the complete nonlinear inference process.

Generation quality and controls. The quality score Q uses the official VBench quality aggregation (Huang et al., 2024). Paired RGB PSNR uses a declared intensity range and the same initial observation, control sequence, and random seed. It measures closeness to the dense generator, which can itself be wrong. Fixed-input residual diagnostics isolate approximation effects from subsequent context drift accumulated over later chunks in free-running rollouts.

The camera-control diagnostic compares inferred and commanded relative rotations in a common coordinate system. The rotation error is the geodesic angle $\arccos(\text{clip}((\text{tr}(R_{\text{cmd}}^\top R_{\text{est}}) - 1)/2, -1, 1))$. Translation-direction error is reported only for commands with nonzero movement. The complementary camera-pose diagnostic uses VGGT-1B (Wang et al., 2025). Its OpenCV world-to-camera matrices are converted to camera-to-world poses, commanded poses are transformed to the same axes, and both trajectories are expressed relative to the first camera. Relative rotations and unit translation directions are evaluated in degrees, without fitting a similarity transform to each generated rollout. Translation-direction scoring requires finite poses, a proper orthonormal rotation (tolerance 10^{-3}), and nonzero translation norm ($> 10^{-6}$ in normalized coordinates). The valid-pose rates are 94.6% (Matrix) and 96.2% (HY). This diagnostic complements the official WorldMark evaluation. Visual quality alone does not establish discrete action consistency. Revisit consistency assesses accumulated drift. These protocols assess control adherence and accumulated drift alongside the benchmark quality metrics.

Latency measurement. We report mean steady-state chunk latency over fifty timed chunks per seed and three seeds, following ten unmeasured warm-up chunks. CUDA synchronization brackets the measured execution, including retrieval, scoring, routing, and reconstruction while excluding VAE decoding; startup is excluded. Speedup divides matched Native latency by method latency at identical resolution, chunk size, and per-backbone hardware. Generation throughput is decoded frames per measured time and does not by itself equal the interaction response rate. Quality comparisons remain point estimates without confidence intervals for the reported evaluation samples.

C.3 Additional Ablations and Qualitative Comparisons

C.3.1 Controlled Design Alternatives

The main paper retains the progressive component table; the following one-factor comparisons separate the choice of selector and reconstruction operator under the same end-to-end evaluation.

Controlled frame-selection and reconstruction ablations. Following the standard one-factor-at-a-time organization used in diffusion-acceleration studies, Table 7 changes one design choice while holding the rest of the pipeline fixed. In the first panel, all selectors use PAR and coupled routing; in the second, all reconstruction operators use control-aware anchors and coupled routing. Each row is evaluated end-to-end with the same metrics, rather than mixing selector-only and reconstruction-only diagnostics with different evaluation targets in one set of columns.

Table 7: Controlled frame-selection and reconstruction ablations on Matrix-Game 3.0. Every row uses $r_f = 0.50$, $r_{\mathrm{KV}} = 0.20$, and the same end-to-end evaluation. T-Motion, Local, Global, and Revisit are representative WorldMark scores; all are higher-is-better.

			WorldMark↑			
Variant	VBench↑	LPIPS↓	T-Motion	Local	Global	Revisit
Frame selection; PAR and coupled routing fixed						
Uniform/interleaved	0.7437	0.4884	76.914	73.478	42.806	77.138
Latent-motion magnitude	0.7506	0.4769	79.312	75.087	44.578	79.341
Control sensitivity	**0.7570**	**0.4657**	**79.830**	**76.531**	**46.645**	**81.514**
Reconstruction; control-aware selection and coupled routing fixed						
Linear interpolation	0.7392	0.4948	77.103	73.337	42.164	76.782
Phase transport without confidence	0.7495	0.4781	78.864	75.008	44.612	79.203
Full PAR	**0.7570**	**0.4657**	**79.830**	**76.531**	**46.645**	**81.514**

With reconstruction and routing fixed, control sensitivity improves VBench by 0.0133 over uniform/interleaved selection, reduces LPIPS by 0.0227, and raises global and revisit memory by 3.839 and 4.376 points. Latent-motion magnitude is consistently intermediate, suggesting that generic motion is useful but does not fully capture action-conditioned frame importance. With selection and routing fixed, full PAR improves VBench by 0.0178 over linear interpolation, reduces LPIPS by 0.0291, and gains 4.481 global-memory and 4.732 revisit-memory points. Phase transport without confidence recovers part of this gap, while the remaining gain from frequency-dependent confidence is visible across both visual and memory metrics. Residual-domain measurements are reported separately in Appendix C.2; these tables use uniform end-to-end metrics for direct comparison.

C.3.2 Fixed-Input Coupling Diagnostics

Does reconstruction coupling add value? We compare C and D on identical block inputs with the same control-aware anchors, historical candidate pool, and per-anchor KV cardinality. Independent routing retains the Top-K blocks under the local omission score $u_{i,m}$; coupled routing starts from those masks and applies one forward and one backward directional refinement pass using the reconstruction-coupled objective in Eq. (14). This changes which historical blocks are retained without changing the 20% historical KV retention budget at any anchor.

Table 8: Fixed-input coupling ablation on Matrix-Game 3.0. Anchor and interval errors are normalized residual errors under identical anchors and KV budgets. Local, Global, and Revisit are WorldMark memory scores measured on the paired rollouts for both routing variants.

KV routing	Anchor error↓	Interval error↓	VBench↑	Local↑	Global↑	Revisit↑
Independent (C)	0.0831	0.1127	0.7479	75.046	44.387	79.206
Coupled, $L_s = 2$ (D)	**0.0774**	**0.1013**	**0.7570**	**76.531**	**46.645**	**81.514**

Coupling reduces anchor-local error by 6.9% and interval reconstruction error by 10.1%. The larger reduction between anchors is accompanied by gains of 1.485, 2.258, and 2.308 points in local, global, and revisit memory. This pattern supports the reconstruction-risk objective: coupled routing selects historical evidence that is useful both to each endpoint and to the frames reconstructed between adjacent endpoints. C and D share the same budget, ruling out additional KV capacity as the cause.

C.3.3 Budget and Refinement Sensitivity

Sensitivity to computation budgets. We vary the retained-frame ratio $r_f = K_f/N_{\mathrm{cur}}$ and retained-history ratio $r_{\mathrm{KV}} = K_{\mathrm{KV}}/M$ one at a time on Matrix-Game 3.0, while fixing $L_s = 2$. Table 9 reports both visual quality and the WorldMark global and revisit memory scores most directly affected by current-frame and historical-context budgets in these comparisons.

Table 9: Budget sensitivity on Matrix-Game 3.0 with two directional refinement passes. Ratios are retained fractions; Global and Revisit are WorldMark memory scores, and latency is milliseconds per chunk. All settings use identical prompts, actions, and paired random seeds.

				WorldMark↑			
r_f	r_{KV}	VBench↑	LPIPS↓	Global	Revisit	Latency↓	Speedup↑
0.30	0.20	0.7441	0.4936	42.738	77.246	1,717	2.52×
0.50	0.10	0.7512	0.4788	42.911	77.926	1,909	2.26×
0.50	**0.20**	**0.7570**	**0.4657**	**46.645**	**81.514**	**2,007**	**2.15×**
0.50	0.30	0.7584	0.4608	47.386	82.263	2,078	2.08×
0.60	0.20	0.7601	0.4529	47.219	82.487	2,363	1.83×

Reducing r_f from 0.50 to 0.30 saves 290 ms but lowers VBench by 0.0129 and reduces global and revisit memory by 3.907 and 4.268 points. Raising r_f to 0.60 provides only 0.0031 additional VBench and less than one point of memory improvement, while reducing speedup from 2.15× to 1.83×. Historical context shows a similar saturation pattern. Retaining 10% rather than 20% of historical KV blocks loses 3.734 global-memory and 3.588 revisit-memory points, whereas increasing the ratio from 20% to 30% adds 71 ms for gains of only 0.741 and 0.749 points. We therefore use $(r_f, r_{\mathrm{KV}}) = (0.50, 0.20)$ as the default operating point.

Number of directional refinement passes. We finally vary L_s at $(r_f, r_{\mathrm{KV}}) = (0.50, 0.20)$. Here, $L_s = 0$ uses the independent per-anchor Top-K initialization, $L_s = 1$ applies one forward directional refinement pass, $L_s = 2$ applies one forward followed by one backward directional refinement pass, and $L_s = 4$ repeats this pair twice. Each pass visits all anchors once, alternating forward and backward directions across the temporally ordered anchor sequence.

Table 10: Directional refinement sensitivity on Matrix-Game 3.0 at $r_f = 0.50$ and $r_{\mathrm{KV}} = 0.20$. Global and Revisit are WorldMark memory scores; interval NRMSE is measured against fully computed residuals, and latency is milliseconds per chunk.

			WorldMark↑				
L_s	VBench↑	LPIPS↓	Global	Revisit	Interval NRMSE↓	Latency↓	Speedup↑
0	0.7479	0.4776	44.387	79.206	0.1127	1,976	2.19×
1	0.7549	0.4689	45.916	80.721	0.1058	1,991	2.17×
2	**0.7570**	**0.4657**	**46.645**	**81.514**	**0.1013**	**2,007**	**2.15×**
4	0.7574	0.4649	46.781	81.672	0.1008	2,038	2.12×

The first two directional refinement passes recover most of the quality and memory lost by independent routing. Moving from $L_s = 0$ to $L_s = 2$ raises VBench by 0.0091, improves global and revisit memory by 2.258 and 2.308 points, and reduces interval NRMSE by 10.1%, for 31 ms of additional latency. Increasing L_s from 2 to 4 yields only 0.0004 VBench, 0.136 global-memory points, 0.158 revisit-memory points, and a 0.5% relative interval-error improvement while reducing speedup to 2.12×. This saturation supports the fixed two-pass schedule used in the main experiments.

C.3.4 Additional Qualitative Comparisons

Figures 10–14 extend Figure 9 with five additional samples per backbone. Each panel compares Native, SVG, TeaCache, Light Interaction (LI), and CAST at matched frame indices under the same control sequence. These selected rollouts illustrate scene evolution and action reversals; aggregate comparisons are reported in the main experiments.

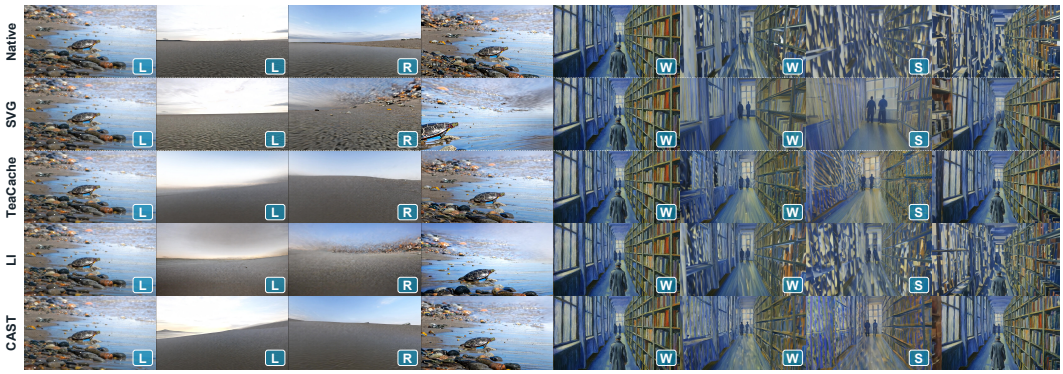

Figure 10: WorldMark rollouts: Matrix-Game 3.0, `fr012_007` (L/R, left), and HY-World 1.5, `ts018_006` (W/S, right). All methods use matched frame indices. Columns show the initial view, the midpoint and end of the first action, and the end of the second action, including its reversal of the initial command.

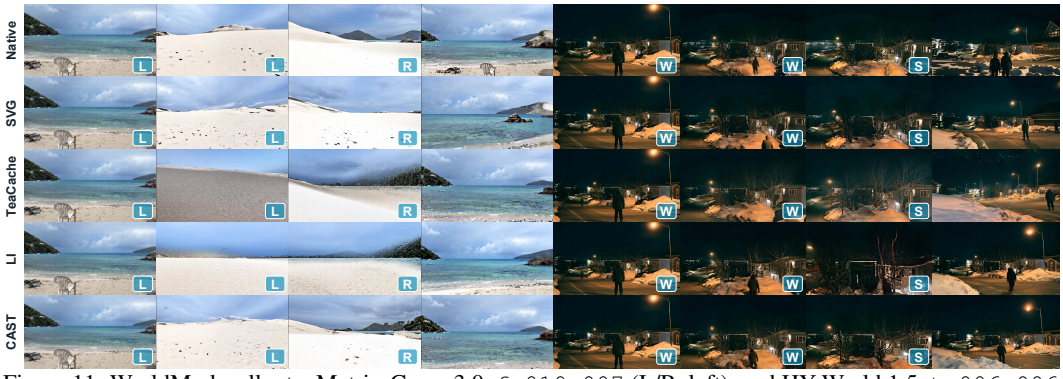

Figure 11: WorldMark rollouts: Matrix-Game 3.0, `fr019_007` (L/R, left), and HY-World 1.5, `tr006_006` (W/S, right). Methods and frame selection follow Figure 10.



Figure 12: WorldMark rollouts: Matrix-Game 3.0, `fr007_006` (W/S, left), and HY-World 1.5, `fr018_007` (L/R, right). Methods and frame selection follow Figure 10.

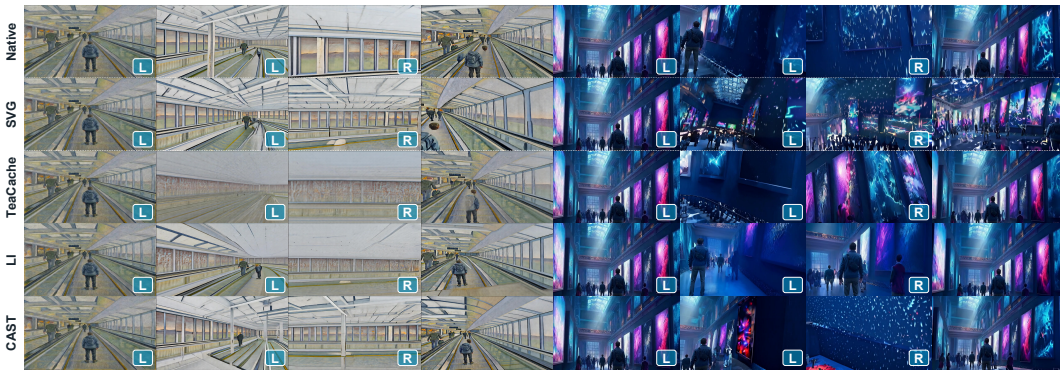

Figure 13: WorldMark rollouts: Matrix-Game 3.0, `ts020_007` (L/R, left), and HY-World 1.5, `ts024_007` (L/R, right). Methods and frame selection follow Figure 10.



Figure 14: WorldMark rollouts: Matrix-Game 3.0, `fr003_010` (S/L, left), and HY-World 1.5, `fr012_007` (L/R, right). Methods and frame selection follow Figure 10.