ONE ANCHOR IS NOT ENOUGH: TRAJECTORY TRACE CACHING FOR VIDEO DIFFUSION ACCELERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Video diffusion transformers spend almost all of their inference cost on repeated forward passes of a large denoiser, and training-free caching reduces that cost by reusing network outputs across denoising steps. Existing caches keep only the latest output residual and reuse it as a frozen point estimate, so they ignore that this residual rotates, decays, and oscillates along the sampling path, and they trigger reuse from open-loop heuristics that are never checked against the error the reuse actually caused. The cost is largest in continuous video editing, where successive instructions are applied to one clip, every round reruns the full sampling loop, and the accelerated output of one round becomes the input of the next, so cache error is amplified rather than merely repeated. We observe that the conditional and unconditional residuals share a low-dimensional subspace whose reduced coefficients evolve almost linearly, so a skipped step becomes advancing a small state and projecting it back rather than reconstructing a high-dimensional output. We introduce ALIGNVACE, a training-free cache that maintains an online mode basis and a stability-projected reduced generator, reconstructs a whole run of skipped steps with one matrix exponential over the entire gap, and routes skips by a risk budget built from out-of-subspace energy and reduced prediction error, calibrated online against the realized error of the guided output. Because the stored trace is compressed, the operating point we use holds *less* device memory than a single uncompressed anchor. Across four editing categories, four-round edit chains, two editing backbones, and text-to-video and text-to-image generation, ALIGNVACE reaches $2.28\times$ speedup at 35.0 dB on object editing and stays 1.5 dB above the strongest baseline at the fourth edit round, while using 0.031 GB of cache against 0.039 to 0.383 GB for prior caches. Our code is available at `https://anonymous.4open.science/r/AlignVACE-86BA/`.

1 INTRODUCTION

Diffusion transformers have become the dominant architecture for video generation and editing, and their cost is concentrated almost entirely in repeated forward passes of a large denoiser under classifier-free guidance (Peebles & Xie, 2023; Wan et al., 2025a; Kong et al., 2024; Jiang et al., 2025). Training-free feature caching attacks that cost directly by reusing a network output across neighbouring denoising steps (Liu et al., 2025a; Zhou et al., 2025; Lyu et al., 2025; Zhao et al., 2025), which requires no retraining and composes with faster solvers, quantization, and distillation. Every such cache faces the same question: given that a step was skipped, what should stand in for the output it would have produced? The prevailing answer is a single anchor: the residual observed at the most recent exact step is frozen and reused until a heuristic decides to recompute. This is a zeroth-order point estimate of a quantity that is not constant. Along the sampling path the output residual rotates in feature space, decays in magnitude, and in places oscillates, so a frozen anchor is accurate only for short gaps and degrades in a way its own trigger cannot see, because the trigger is computed from input perturbations rather than from the error the reuse caused (fig. 1).

Continuous editing makes the weakness structural rather than incidental. A user applies a chain of instructions to one clip, every round reruns the full sampling loop, and the output of round i is the input of round $i+1$, so an error introduced by caching is fed back and amplified; every baseline we evaluate loses more than 10 dB between the first and the fourth round. The chain also offers an

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

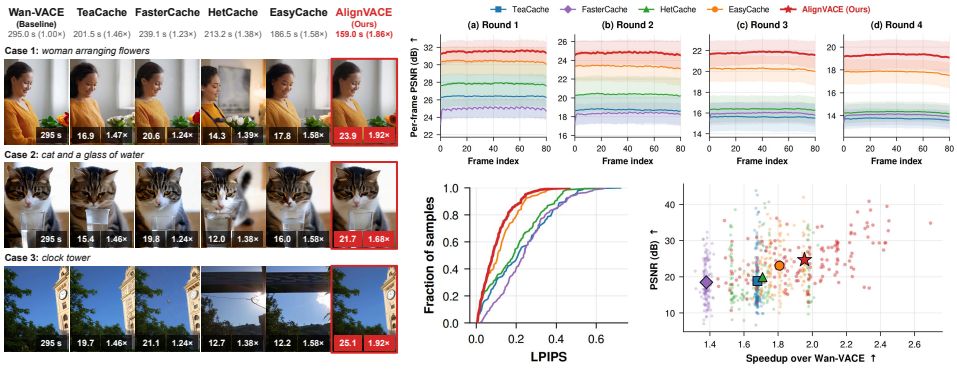

Figure 1: **Single-anchor reuse degrades beyond its trigger.** Left: editing examples with latency, speedup over unaccelerated Wan-VACE, and frame PSNR. Right: PSNR across rounds, LPIPS over clips, and speed–quality tradeoff. Fidelity drops with edit depth; ALIGNVACE's gap to the strongest baseline is present before the last round.

opportunity single-call caches cannot use: consecutive rounds edit the same clip, so dynamics fitted in one round inform the next.

We start from two measurements. Stacking the conditional and unconditional residuals over a local window gives a matrix whose singular values decay quickly, so the trajectory lies close to a k-dimensional subspace with k far below the token dimension; and inside that subspace the reduced coefficients evolve almost linearly, the regime that dynamic mode decomposition and the Koopman view are built for (Schmid, 2010; Kutz et al., 2016; Rowley et al., 2009; Tu, 2013). Together they change what a skipped step is: the cache advances a k-dimensional state and projects it back instead of reconstructing a high-dimensional output.

ALIGNVACE implements this. It maintains an orthonormal mode basis and small reduced generator online, reconstructs skipped runs with one matrix exponential over the entire gap, and decides when to skip from a risk budget built from two quantities exposed at exact steps: how much of the residual escapes stored modes, and how badly the generator predicted recent dynamics. The budget is calibrated online against the realized guided-output error, closing the loop prior caches leave open. Rank, mode compression, and history length form an explicit storage-for-compute dial, and because the basis is compressed the whole dynamic model is smaller than one uncompressed anchor: 0.031 GB against 0.375 and 0.383 GB for the two caches that store full residuals.

Our contributions are the following. We show that guided residual trajectories in video diffusion transformers are low rank and approximately linear in reduced space, and that prior caches are $\mathbf{\Omega} = \mathbf{0}$ and first-order special cases of one operator model. We build a training-free cache with a stability projection bounding extrapolation, a closed-loop calibrator tying routing to guided-output error, and cross-round mode transfer for edit chains. We evaluate on four editing categories, four-round chains, two backbones, text-to-video, and text-to-image, and report where the method does not win: it does not improve temporal warp error over the best baseline (section A.8), and its adaptive-rank and cross-round components trade fidelity for speed rather than improving both.

2 RELATED WORK

Training-free caching for diffusion. Step-axis caches reuse network outputs at fixed intervals (Zhao et al., 2025; Selvaraju et al., 2024; Ma et al., 2024) or under runtime criteria (Liu et al., 2025a; Zhou et al., 2025; Lyu et al., 2025; Kahatapitiya et al., 2025; Wimbauer et al., 2024), and spatial variants cache per token or region (Zou et al., 2024; Liu et al., 2026). Their common structure is a frozen anchor plus a trigger. ALIGNVACE differs on both: the stand-in for a skipped step is a forecast from a fitted operator, and the trigger is calibrated against the realized error of the guided output rather than against an input perturbation.

Forecasting rather than reusing. TaylorSeer (Liu et al., 2025b) predicts the feature trajectory by a Taylor expansion, which is the closest work in spirit, and confidence-gated and speculative variants

add a check on the forecast (Guan et al., 2025; Liu et al., 2025c). A scalar expansion around one point captures a local trend but not rotation or coupled decay across guidance branches, which our reduced generator represents through the complex spectrum of a $k \times k$ matrix.

Reduced-order dynamics. Dynamic mode decomposition and the Koopman operator represent nonlinear evolution by linear dynamics over chosen observables (Schmid, 2010; Kutz et al., 2016; Rowley et al., 2009; Tu, 2013; Brunton et al., 2016), with streaming variants for online use (Hemati et al., 2014; Zhang et al., 2019). We take the residual modes as observables, which makes a skip cost one $k \times k$ exponential and one projection.

Other acceleration axes and video editing. Faster solvers (Song et al., 2020; Lu et al., 2022; Karras et al., 2022), distillation (Salimans & Ho, 2022; Song et al., 2023), sparsity (Wan et al., 2025b), and quantization are orthogonal to caching and compose with it. Tuning-free and instruction-driven video editing (Ku et al., 2024; Cheng et al., 2024; Wu et al., 2023a; Geyer et al., 2024) and unified frameworks such as VACE (Jiang et al., 2025) make multi-round editing a practical workflow, which is the regime where cache error compounds and where we focus.

3 PRELIMINARIES

A video diffusion model generates a latent $\mathbf{x}_t \in \mathbb{R}^D$ through T denoising steps, where D flattens frames, space, and channels. At step t the network produces conditional and unconditional outputs, $\mathbf{y}_t^{(\mathrm{c})} = \epsilon_\theta(\mathbf{x}_t, t, \mathbf{p})$ and $\mathbf{y}_t^{(\mathrm{u})} = \epsilon_\theta(\mathbf{x}_t, t, \varnothing)$, combined by classifier-free guidance with scale γ,

$$\mathbf{g}_t = \mathbf{y}_t^{(\mathrm{u})} + \gamma\big(\mathbf{y}_t^{(\mathrm{c})} - \mathbf{y}_t^{(\mathrm{u})}\big), \qquad \mathbf{x}_{t+1} = \Psi_t(\mathbf{x}_t, \mathbf{g}_t), \tag{1}$$

with Ψ_t the solver update. A cache partitions steps into exact ones, where the network is evaluated twice, and skipped ones, where an approximation $\widehat{\mathbf{g}}_t$ is substituted. Rather than caching the raw output we cache the output residual $\mathbf{r}_t^{(b)} = \mathbf{y}_t^{(b)} - \mathbf{x}_t$ for $b \in \{\mathrm{c}, \mathrm{u}\}$, which separates the identity part of the output from the denoising correction and varies more smoothly along nearby steps, and we reconstruct a branch as $\widehat{\mathbf{y}}_t^{(b)} = \mathbf{x}_t + \widehat{\mathbf{r}}_t^{(b)}$. Stacking both branches gives $\mathbf{z}_t = [\mathbf{r}_t^{(\mathrm{c})\top}, \mathbf{r}_t^{(\mathrm{u})\top}]^\top \in \mathbb{R}^{2D}$. Two properties of this object drive the design, both measured in section A.1. Over a local window the residuals concentrate near a k-dimensional subspace, $\mathbf{z}_t \approx \mathbf{U}\mathbf{a}_t$ with $\mathbf{U} \in \mathbb{R}^{2D \times k}$ and $k \ll D$, because of redundancy across frames, spatial neighbourhoods, and the two guidance branches. And inside that subspace the coefficients evolve approximately linearly, $\mathbf{a}_{t+\delta} \approx \exp(\mathbf{\Omega}\delta)\,\mathbf{a}_t$, a finite-dimensional Koopman approximation that contains zeroth-order reuse ($\mathbf{\Omega} = \mathbf{0}$) and first-order extrapolation (a truncated exponential) as special cases.

4 METHOD

4.1 ONLINE RESIDUAL-MODE EXTRACTION

From residuals observed at recent exact steps $\tau_1 < \cdots < \tau_m$ we maintain an orthonormal basis $\mathbf{U} \in \mathbb{R}^{2D \times k}$ and a reduced generator $\mathbf{\Omega} \in \mathbb{R}^{k \times k}$ such that the reduced state $\mathbf{a}_t = \mathbf{U}^\top \mathbf{z}_t$ obeys $\dot{\mathbf{a}} \approx \mathbf{\Omega}\mathbf{a}$. Stacking branches before extraction is deliberate: conditional and unconditional residuals are strongly correlated under guidance, so a shared basis couples them and removes duplicate storage. Because exact steps are nonuniformly spaced, we fit a continuous-time generator by least squares on finite differences rather than a fixed-step operator, which lets one model serve any gap. The basis is updated by a streaming decomposition with forgetting factor ρ, so the subspace follows changing trajectory curvature without recomputation. Derivations are in section A.2. The stored object is k mode vectors, a $k \times k$ generator, and one k-dimensional coefficient per retained exact step, so remembering an additional step costs $\mathcal{O}(k)$ scalars rather than the $\mathcal{O}(D)$ of a raw anchor, and an exact step adds one $\mathcal{O}(Dk)$ projection plus $\mathcal{O}(k^3)$ algebra.

4.2 PREDICTIVE RECONSTRUCTION BY OPERATOR ADVANCEMENT

To skip step t, let τ be the most recent exact step and $\delta = t - \tau$. We advance the reduced state and project back,

$$\widehat{\mathbf{a}}_t = \exp(\mathbf{\Omega}\,\delta)\,\mathbf{a}_\tau, \qquad \widehat{\mathbf{z}}_t = \mathbf{U}\,\widehat{\mathbf{a}}_t, \qquad \widehat{\mathbf{y}}_t^{(b)} = \mathbf{x}_t + \widehat{\mathbf{r}}_t^{(b)}, \tag{2}$$

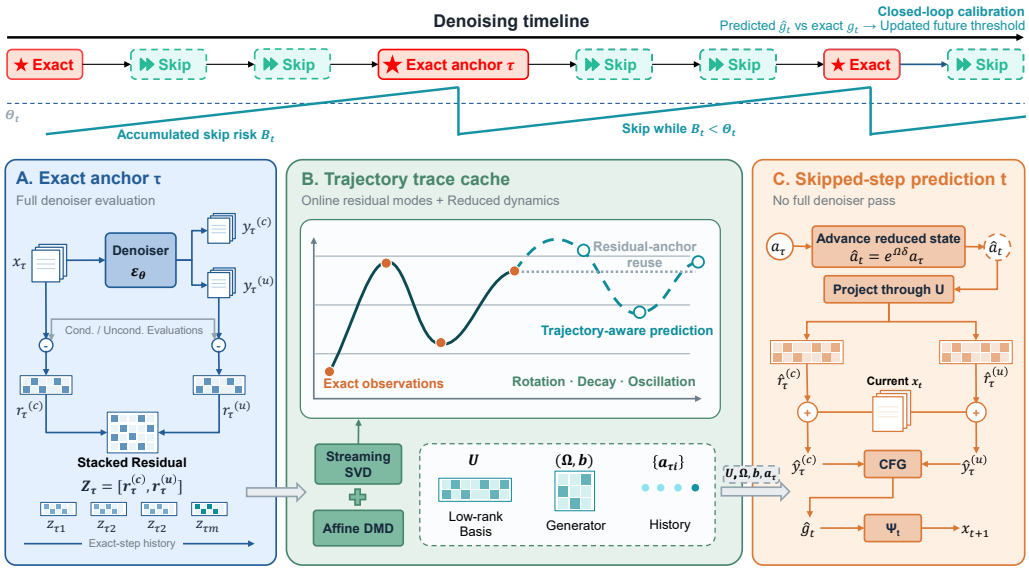

Figure 2: **Overview of ALIGNVACE.** At an exact anchor the two guidance branches are evaluated and their residuals are stacked into $\mathbf{z}_\tau$ (A). A streaming decomposition maintains a low-rank basis $\mathbf{U}$, an affine reduced generator $(\mathbf{\Omega}, \mathbf{b})$, and a short coefficient history (B). A skipped step advances the reduced state by one matrix exponential over the whole gap, projects it back through $\mathbf{U}$, and recomposes the guided output (C). Accumulated risk B_t decides when to stop skipping, with a threshold calibrated against the realized guided-output error.

where $\widehat{\mathbf{r}}_t^{(\mathrm{c})}$ and $\widehat{\mathbf{r}}_t^{(\mathrm{u})}$ are the halves of $\widehat{\mathbf{z}}_t$ and the guided output is recomposed by eq. (1). A skip therefore costs one $k \times k$ exponential and one $\mathbb{R}^{2D\times k}$ matrix-vector product in place of two forward passes.

Two properties matter. A run of skipped steps is covered by a single exponential over the whole gap rather than by repeated increments, so a fixed-step bias is not accumulated along the run, and complex eigenvalues of $\mathbf{\Omega}$ represent rotation and damped oscillation that a hold or a scalar expansion cannot. To keep long skips safe we project the eigenvalues so that their real parts do not exceed a small cap, which guarantees no mode grows during extrapolation; this projection, and an optional affine term carrying the slowly varying mean residual, are given in section A.3. The model degenerates gracefully: with $k{=}1$ and $\mathbf{\Omega} = \mathbf{0}$ it is exactly single-anchor reuse, and a single real eigenvalue recovers first-order extrapolation.

4.3 MODE-PREDICTIVE ROUTING AND CLOSED-LOOP CALIBRATION

At every exact step two signals are available for free: the out-of-subspace energy $e_\perp = \|\mathbf{z} - \mathbf{U}\mathbf{U}^\top\mathbf{z}\|_2/\|\mathbf{z}\|_2$, which measures how much of the residual escapes the stored modes, and the reduced prediction error $e_\Omega = \|\widehat{\mathbf{a}} - \mathbf{a}\|_2/\|\mathbf{a}\|_2$ between the forecast formed before the pass and the realized state. From clipped moving averages of these we define an instantaneous risk and accumulate it over the current skip run,

$$\eta_t = \phi(\delta)\left(\beta_1\,\bar{e}_\perp + \beta_2\,\bar{e}_\Omega\right), \qquad B_t = B_{t-1} + \eta_t, \qquad \Theta_t = \theta\, s_t, \tag{3}$$

with ϕ monotone in the extrapolation distance. The controller skips while $B_t < \Theta_t$, and computes exactly when the budget is exceeded, when the anchor age passes a staleness cap, or when the step lies in the warm-up and tail set. The risk is large exactly when the modes no longer span the residual or the operator has stopped predicting its own dynamics, which is when a skip is unsafe.

That risk is still a reduced-space proxy, so we calibrate it against the quantity that actually matters. When the budget triggers an exact step, we first form the predicted guided output $\widehat{\mathbf{g}}_t$ from eq. (2), then after the pass measure the realized relative error $E_t^{\mathrm{obs}} = \|\widehat{\mathbf{g}}_t - \mathbf{g}_t\|_2^2/\|\mathbf{g}_t\|_2^2$ and form the ratio $r_t = E_t^{\mathrm{obs}}/B_t$. A moving average of that ratio drives an inverse threshold scale

$s_t = \mathrm{clip}(1/\bar{r}_t, s_{\min}, s_{\max})$, so a budget that underestimates true distortion tightens the threshold and forces more exact steps, and one that overestimates relaxes it. The updates are applied only at steps triggered by the budget itself, excluding warm-up, initialization, and staleness triggers, so the calibration signal is not contaminated by steps that were exact for reasons unrelated to prediction quality (section A.4).

The same feedback controls storage. When $\bar{e}_\perp$ persistently exceeds a high watermark and the memory budget permits, a mode is appended and k grows, spending memory to regain accuracy; when it stays below a low watermark the weakest mode is dropped. The storage-for-compute exchange is therefore adaptive rather than fixed, and section 5.4 measures what that adaptivity costs and buys.

4.4 Mode memory management

The footprint is explicit and bounded,

$$\mathcal{M} = \mathrm{mem}\big(\mathcal{C}(\mathbf{U})\big) + k^2 + m\,k, \tag{4}$$

where $\mathcal{C}$ compresses each mode by spatial downsampling, by sharing structure across frames, or by a low-rank factorization, and m is the fixed length of a coefficient ring buffer. The basis dominates, which is why compressing it is what makes the whole dynamic model cheaper than one uncompressed anchor in practice. Because memory is bounded by $k_{\max}$ and the ring buffer, the footprint does not grow with T. For edit chains the basis and generator are carried into the next round under decay, so later rounds are warm started rather than refitted, and stale directions are discounted as the content changes (section A.5).

5 Experiments

Protocols. The *single-round* protocol applies one edit instruction to a source video, which is the standard setting for cache accelerators. The *continuous* protocol applies four successive edits to the same clip, where the output of round i is the input of round $i{+}1$; the initial video is generated once and shared by all methods, so it is excluded from the round count and from the reported latency. The edit chains are built from VBench prompts (Huang et al., 2024): Wan2.1 (Wan et al., 2025a) generates the initial clip from each prompt, and GLM-5.1 expands the prompt into four cumulative edit instructions, one per instruction type (style, lighting, background, and object), applied in a different order across chains. The chains are shared by all methods, so every accelerator edits the same clips under the same instructions (section A.7). Four rounds suffice for cache error, barely visible at the first round, to become the dominant failure mode.

Categories, backbones, and metrics. Following the VACE taxonomy (Jiang et al., 2025) we cover four editing categories that differ in how much of the frame is regenerated per step: object modification, replacement and removal, and background editing, both masked video to video; lighting and atmosphere editing, and stylization, both video to video. The primary backbone is VACE Wan2.1 1.3B (Jiang et al., 2025), with clips drawn from the VACE benchmark, LOVEU TGVE (Wu et al., 2023b), and DAVIS (Perazzi et al., 2016); LTX-Video (HaCohen et al., 2024) serves as a second editing backbone, and Wan2.1 (Wan et al., 2025a) serves as both a text-to-video and a text-to-image backbone. Fidelity is measured against the unaccelerated output of the same backbone under the identical seed, prompt, and round, using PSNR, SSIM, LPIPS (Zhang et al., 2018), and GMSD (Xue et al., 2013); prompt alignment uses CLIP-T (Radford et al., 2021) and temporal consistency warp error (Lai et al., 2018). Baselines are TeaCache (Liu et al., 2025a), FasterCache (Lyu et al., 2025), EasyCache (Zhou et al., 2025), and HetCache, all applied to the identical backbone and sampling configuration. HetCache is a heterogeneous anchor cache that keeps separate conditional and unconditional residuals and triggers on their disagreement; because no public implementation exists we reimplement it and report it as an internal baseline rather than as a published result. Hyperparameters are selected once on a held-out set of clips and then frozen across categories, datasets, and rounds (section A.6).

5.1 Single-round editing

In table 1 AlignVACE is fastest in all four categories, between $1.70\times$ and $2.34\times$, and its cache is the smallest in three of the four, 0.031 to 0.041 GB against 0.039 GB for the two lightest baselines

Table 1: **Single-round editing** on VACE Wan2.1 1.3B (Jiang et al., 2025) across four editing categories. Fidelity is against the unaccelerated output under the identical seed and prompt; Mem is the extra device memory held by the cache. Best and second per column within a block.

	Efficiency			Quality			
Method	Latency (s) ↓	Speedup ↑	Mem (GB) ↓	PSNR ↑	SSIM ↑	LPIPS ↓	CLIP-T ↑
(a) Object modification, replacement and removal (MV2V)							
Wan-VACE baseline	291.27	1.00×	0.000	n/a	n/a	n/a	0.2999
TeaCache	173.39	1.68×	0.375	**35.32**	0.9761	**0.0113**	**0.2914**
FasterCache	210.82	1.38×	0.039	27.66	0.8967	0.0453	0.2860
EasyCache	149.20	1.95×	0.039	31.21	0.9693	0.0190	0.2885
HetCache	169.06	1.72×	0.383	33.87	**0.9783**	0.0131	0.2885
ALIGNVACE (ours)	**127.49**	**2.28**×	**0.031**	35.00	0.9723	0.0118	0.2889
(b) Background editing (MV2V)							
Wan-VACE baseline	291.00	1.00×	0.000	n/a	n/a	n/a	0.2888
TeaCache	173.20	1.68×	0.375	17.20	0.7870	0.1893	0.2899
FasterCache	210.53	1.38×	**0.039**	21.01	0.7782	0.1506	0.2914
EasyCache	177.18	1.64×	**0.039**	23.58	0.8763	0.0879	**0.2984**
HetCache	179.95	1.62×	0.383	17.62	0.6056	0.3836	0.2989
ALIGNVACE (ours)	**171.38**	**1.70**×	0.041	**30.09**	**0.9376**	**0.0317**	0.2981
(c) Lighting and atmosphere editing (V2V)							
Wan-VACE baseline	290.37	1.00×	0.000	n/a	n/a	n/a	0.3077
TeaCache	173.35	1.68×	0.375	24.86	0.8936	0.0805	0.3068
FasterCache	210.76	1.38×	0.039	24.79	**0.9632**	0.1120	0.3090
EasyCache	163.68	1.77×	0.039	**30.90**	0.9566	**0.0217**	0.3096
HetCache	190.38	1.53×	0.383	28.27	0.9249	0.0531	0.3068
ALIGNVACE (ours)	**131.82**	**2.20**×	**0.031**	30.82	0.9098	0.0376	**0.3098**
(d) Stylization (V2V)							
Wan-VACE baseline	290.68	1.00×	0.000	n/a	n/a	n/a	0.2564
TeaCache	173.48	1.68×	0.375	28.65	0.9438	0.0440	0.2571
FasterCache	211.32	1.38×	0.039	25.71	0.8118	0.0835	0.2559
EasyCache	162.27	1.79×	0.039	30.76	**0.9624**	**0.0201**	0.2557
HetCache	146.84	1.98×	0.383	30.66	0.9556	0.0293	**0.2573**
ALIGNVACE (ours)	**124.14**	**2.34**×	**0.031**	**30.99**	0.9204	0.0319	0.2572

and 0.375 to 0.383 GB for the two that store full residual anchors. Its PSNR is best or second best in every category; the two places worth reading closely are background editing, where it is 6.5 dB above the best baseline, and lighting, where it is within 0.1 dB of EasyCache while running about 24% faster.

The category structure explains where single-anchor reuse breaks. Where a large fraction of the frame changes coherently, as in stylization and object editing, every method retains above 25 dB. Background editing is hardest, because the mask boundary separates a regenerated region from a preserved one and the residual trajectory is less smooth. Three of the four baselines fall below 22 dB there and HetCache reaches 0.3836 LPIPS, while an operator that can represent rotation and decay stays at 30.09 dB.

The memory column carries the observation the title is about: the two full-residual caches hold more than ten times the 0.031 GB of our entire dynamic model, because a compressed rank-k basis plus a $k \times k$ generator is smaller than one uncompressed $2D$-dimensional snapshot. Two caveats belong with the table. ALIGNVACE is not best on SSIM in any category, consistent with a predictor that recovers the residual in an energy-weighted sense rather than preserving local structural statistics exactly. And CLIP-T spans at most 0.014 and does not order methods consistently; it only confirms that no accelerator weakens the edit.

5.2 CONTINUOUS EDITING

Table 2 reports the protocol the paper is built around. ALIGNVACE leads at every round on both metrics, by 1.97 dB at the first round and 1.49 dB at the fourth, while also being the fastest method in the table. Per-round values separate two failure modes that a single endpoint conflates: a method that

Table 2: **Continuous editing** on VACE Wan2.1 1.3B (Jiang et al., 2025) with four successive rounds. Each chain starts from a clip generated by Wan2.1 (Wan et al., 2025a) from a VBench prompt (Huang et al., 2024), and its four edit instructions are expanded from that prompt by GLM-5.1 (section A.7). Fidelity at round i is against the unaccelerated chain at round i; speedup is averaged over rounds. Drop is the PSNR decline from the first to the fourth round and must be read together with the fourth-round columns.

		PSNR per round ↑				LPIPS per round ↓				End of chain	
Method	Speedup ↑	R1	R2	R3	R4	R1	R2	R3	R4	Drop (dB)	LPIPS@4 ↓
Wan-VACE baseline	1.00×	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a
TeaCache	1.68×	26.45	18.60	15.64	13.73	0.087	0.181	0.286	0.353	12.72	0.353
FasterCache	1.38×	24.77	18.32	15.98	14.06	0.098	0.221	0.305	0.385	**10.71**	0.385
EasyCache	1.80×	29.79	23.29	20.30	17.81	0.038	0.099	0.153	0.207	11.98	0.207
HetCache	1.69×	27.52	20.29	16.47	14.31	0.071	0.149	0.245	0.318	13.21	0.318
ALIGNVACE (ours)	**1.92×**	**31.76**	**24.76**	**21.79**	**19.30**	**0.028**	**0.075**	**0.123**	**0.181**	12.46	**0.181**

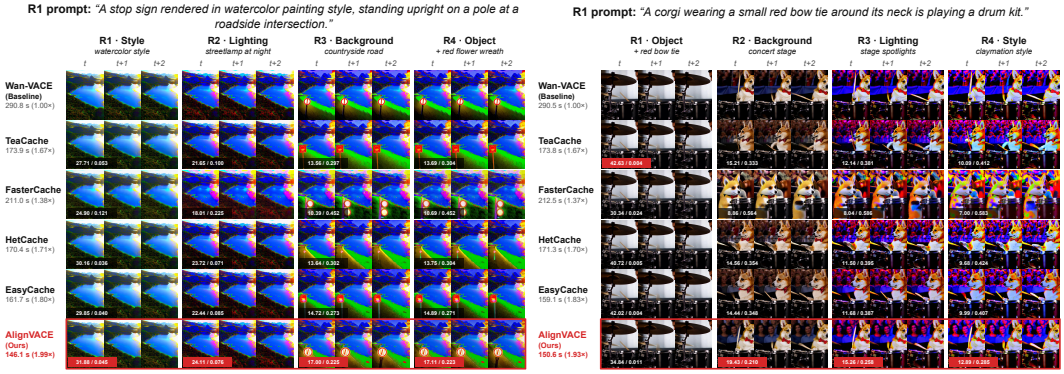

Figure 3: **Two four-round editing chains.** Each row is a method and each block is one round, showing three consecutive frames; the numbers inside a frame are PSNR and LPIPS against the unaccelerated chain at that round. Differences concentrate at mask boundaries, on identity-bearing regions, and in high-frequency texture, where a frozen anchor least tracks the residual; ten more chains are in section A.10.

is uniformly slightly worse, and one that is competitive early and collapses once its own outputs are recycled.

The drop column is the one most easily misread. FasterCache has the smallest decline, 10.71 dB, and is nonetheless the worst method at the fourth round on both PSNR and LPIPS, because it starts 7 dB lower. Our own decline, 12.46 dB, is larger than EasyCache's 11.98 dB. We therefore do not claim that ALIGNVACE slows error accumulation; the measurements support that it starts higher and stays higher at every round, and that its fourth-round LPIPS is 13% below the strongest baseline.

In fig. 3 the object inserted in an early round is the first thing the anchor caches lose once later instructions are applied, and the first round is nearly indistinguishable across methods, which is why a single-round evaluation does not characterize an accelerator for this workload.

5.3 SKIP BEHAVIOUR AND PER-CLIP WIN RATES

Figure 4 separates what the aggregate tables average over. Win rates exceed parity on every metric against three baselines, reaching 97 to 100 against FasterCache and 86 to 88 on PSNR against TeaCache and HetCache; the one cell below parity is SSIM against EasyCache at 43%, the per-clip form of the SSIM caveat above. Against the strongest baseline the PSNR and speed win rates are 65% and 68%, a consistent tendency rather than a few large wins. The middle panel shows the mechanism behind the speed column: ALIGNVACE skips 54% of the steps against 44% for the closest anchor cache while remaining more faithful, because exact steps are spent where the residual leaves the stored modes rather than at a fixed rate.

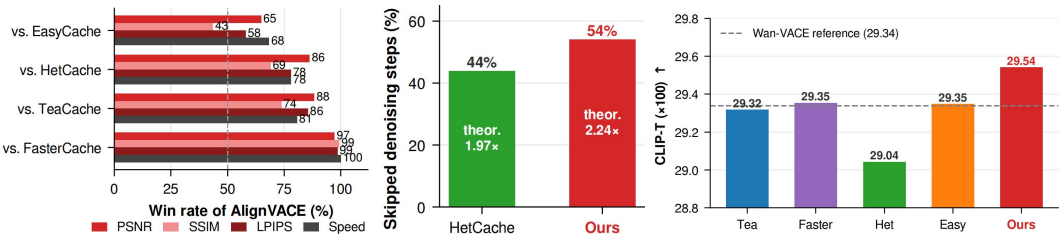

Figure 4: **Aggregate behaviour of AlignVACE.** Left: per-clip win rate against each baseline, 50 is parity. Middle: fraction of steps skipped; annotated speedups count network passes only and exceed the realized speedups of table 1, which include projection, warm-up and tail steps. Right: CLIP-T ($\times 100$) against the reference at 29.34.

Table 3: **Component ablations** under the four-round protocol, all other settings fixed. PSNR and LPIPS are averaged over the four rounds; drift is the increase in LPIPS from the first to the fourth round.

Variant	Mem (GB) ↓	Speedup ↑	PSNR ↑	LPIPS ↓	LPIPS drift ↓
Without predictive increment	0.0654	1.91×	24.34	0.1257	0.2220
Without adaptive rank	**0.0503**	1.80×	**24.86**	**0.0977**	**0.1693**
Without cross-round transfer	0.0525	**2.08**×	24.18	0.1088	0.1881
Full (ours)	0.0657	1.92×	24.31	0.1057	0.1836

5.4 Ablations

Table 3 is not uniformly won by the full configuration. The predictive increment is the component that earns its place on quality. Replacing the operator forecast with a frozen coefficient inside the learned subspace leaves PSNR essentially unchanged, 24.34 against 24.31, but raises LPIPS by 19% and the four-round drift by 21% at the same speed and memory. A flat PSNR with clearly worse perceptual and drift profiles is what a variant reconstructing the correct energy in the wrong directions would produce, and it is the sharpest evidence that modeling the dynamics, not only the subspace, is what matters.

Adaptive rank and cross-round transfer are speed mechanisms, not quality mechanisms. Fixing the rank gives the best fidelity in the table, 24.86 dB with the lowest LPIPS and drift, at 0.12× less speedup and 0.015 GB less memory; disabling cross-round transfer gives the highest speedup, 2.08×, with 0.13 dB lower PSNR and worse drift. The full configuration converts roughly half a decibel into 0.12× of speed; a deployment that values fidelity should fix the rank. The higher speedup without transfer is counterintuitive; the most plausible explanation is that a carried basis is initially mismatched to the new instruction and triggers exact steps a freshly fitted basis would not, a mechanism we regard as not yet isolated.

5.5 Generalization across backbones and modalities

The residual argument of section 3 depends neither on one denoiser nor on the temporal axis, so the predictor should carry over to other backbones, to generation, and to images.

A second editing backbone. On LTX-Video (table 4a) AlignVACE is again fastest, 1.80× against 1.71×, with 1.84 dB higher PSNR, 11% lower LPIPS, and 7% less drift than the strongest baseline. The baseline ordering matches Wan-VACE, with EasyCache clearly second and the other three between 14.2 and 14.9 dB, which suggests that the gap comes from the reuse model rather than from one denoiser. Every method drifts substantially here, so this backbone is harder for caching in general; the forecast reduces that difficulty but does not remove it.

Text-to-video generation. On Wan2.1 (table 5) AlignVACE is fastest at 1.86× and best on all four reference metrics, 2.78 dB and 0.049 LPIPS ahead of the best baseline. We do not lean on VBench: all accelerators but FasterCache score within 0.4 points of the baseline, TeaCache above it, although reference PSNRs span more than 10 dB. A reference-free score measures whether the output is a

Table 4: **Transfer to a second editing backbone and to images.** (a) LTX-Video VACE (HaCohen et al., 2024), inpainting; Drift@4 is the LPIPS increase from round 1 to 4 under the protocol of table 2. (b) Wan2.1 1.3B (Wan et al., 2025a), text to image, latency per image; ImageReward (Xu et al., 2023) is reference free.

Method	Latency (s)↓	Speedup↑	PSNR↑	LPIPS↓	Drift@4↓ / ImageReward↑
(a) LTX-Video VACE, inpainting (MV2V); last column Drift@4↓					
LTX-Video VACE baseline	298.99	1.00×	n/a	n/a	0.000
TeaCache	190.28	1.57×	14.21	0.3331	0.4884
FasterCache	211.86	1.41×	14.90	0.3839	0.5489
EasyCache	174.64	1.71×	19.90	0.1958	0.3368
HetCache	194.04	1.54×	14.68	0.2978	0.4308
ALIGNVACE (ours)	**166.45**	**1.80×**	**21.73**	**0.1736**	**0.3120**
(b) Wan2.1 1.3B, text to image; last column ImageReward↑					
Wan2.1 baseline	3.630	1.00×	n/a	n/a	−0.0053
TeaCache	2.488	1.46×	20.00	0.1980	−0.0570
FasterCache	2.951	1.23×	20.55	0.2176	−0.0799
EasyCache	2.302	1.58×	19.41	0.2240	−0.0047
HetCache	2.632	1.38×	13.48	0.4756	−0.0407
ALIGNVACE (ours)	**1.963**	**1.85×**	**23.91**	**0.1163**	**0.0129**

Table 5: **Text-to-video generation** on Wan2.1 1.3B (Wan et al., 2025a) with VBench prompts (Huang et al., 2024), 81 frames at 832×480. For VBench the baseline is a reference, not an upper bound; per-clip results in section A.9.

Method	Latency (s)↓	Speedup↑	PSNR↑	SSIM↑	LPIPS↓	GMSD↓	VBench (%)
Wan2.1 baseline (T=50)	295.00	1.00×	n/a	n/a	n/a	n/a	98.25
TeaCache	201.46	1.46×	20.06	0.7456	0.1966	0.1770	98.41
FasterCache	239.13	1.23×	21.29	0.7338	0.1630	0.1581	92.12
EasyCache	186.49	1.58×	19.48	0.7247	0.2226	0.1854	98.18
HetCache	213.20	1.38×	13.52	0.5506	0.4742	0.2497	97.87
ALIGNVACE (ours)	**158.99**	**1.86×**	**24.07**	**0.8118**	**0.1136**	**0.1201**	98.15

plausible video, not whether it is the video the unaccelerated model would have produced; section A.9 shows anchor caches producing well-formed videos of a different scene.

Text-to-image generation. On Wan2.1 text to image (table 4b) ALIGNVACE cuts latency from 3.63 to 1.96 s, 1.85× against 1.58× for the fastest baseline, while being 3.35 dB above the best baseline PSNR and 41% below its LPIPS. Its ImageReward is the highest, slightly above the reference, which we attribute to the mild denoising of a smooth forecast rather than to better prompt fidelity. Without a temporal axis the basis captures only spatial and cross-branch redundancy, so the reduced dynamics, not frame redundancy alone, are what the predictor exploits.

6 CONCLUSION AND OUTLOOK

A skipped denoising step does not have to be filled with a frozen anchor. Guided residual trajectories in video diffusion transformers are low rank and approximately linear in reduced coordinates, which turns a skip into the advance of a small state, lets one matrix exponential cover a whole skip run, and supplies signals that close the routing loop against realized guided-output error. Because the trace is compressed, the dynamic model is smaller than the anchor it replaces, and the ordering holds across two editing backbones, continuous edit chains, text-to-video, and text-to-image generation, including settings where errors compound across rounds. Three directions follow. First, a temporally structured basis may improve warp error by making reconstruction errors coherent across frames. Second, composition with solvers, quantization, sparsity, and distillation should test whether gains add or whether approximation errors interact. Third, the calibrated risk signal could schedule rank, compression, guidance, and solver order, turning a cache into a controller over the inference budget rather than a fixed reuse rule. More broadly, trajectory-aware caching suggests replacing isolated feature reuse with local dynamics modeling of the sampler.

AI USE STATEMENT

Large language models were used in two distinct roles. As part of the experimental method, GLM-5.1, accessed through its API, expands VBench prompts into the four-round edit instructions of the continuous protocol (section 5, section A.7); its outputs define evaluation inputs shared by all methods. Separately, large language models were used during manuscript preparation for editing the writing, condensing the text to the page limit, and formatting LaTeX tables. All technical content, results, and conclusions were produced and verified by the authors.

ETHICS STATEMENT

All authors have read and adhere to the ICLR Code of Ethics. This work proposes a training-free inference accelerator that approximates the output of an existing backbone and adds no new generative capability. It involves no human subjects, and all experiments use publicly released models and public benchmarks. We acknowledge that cheaper video editing also lowers the cost of misuse, such as producing deceptive manipulated video, and that the method inherits any biases of the backbones it accelerates.

REPRODUCIBILITY STATEMENT

Anonymous source code is available at the link in the abstract. The method is specified in section 4, with component details in sections A.2 to A.5, and the protocols, datasets, metrics, baselines, and hyperparameter selection are described in section 5 and section A.6, and the construction of the edit chains in section A.7. All fidelity numbers are measured against the unaccelerated output of the same backbone under an identical seed, prompt, and round, and per-clip results are given in sections A.9 and A.10.

REFERENCES

Steven L Brunton, Bingni W Brunton, Joshua L Proctor, and J Nathan Kutz. Koopman invariant subspaces and finite linear representations of nonlinear dynamical systems for control. *PloS one*, 11(2):e0150171, 2016.

Jiaxin Cheng, Tianjun Xiao, and Tong He. Consistent video-to-video transfer using synthetic dataset. In *International Conference on Learning Representations*, volume 2024, pp. 16867–16879, 2024.

Michal Geyer, Omer Bar Tal, Shai Bagon, and Tali Dekel. Tokenflow: Consistent diffusion features for consistent video editing. In *International Conference on Learning Representations*, volume 2024, pp. 1608–1620, 2024.

Xiaoliu Guan, Lielin Jiang, Hanqi Chen, Xu Zhang, Jiaxing Yan, Guanzhong Wang, Yi Liu, Zetao Zhang, and Yu Wu. Forecasting when to forecast: Accelerating diffusion models with confidence-gated taylor. *Knowledge-Based Systems*, pp. 114635, 2025.

Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, et al. Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024.

Maziar S Hemati, Matthew O Williams, and Clarence W Rowley. Dynamic mode decomposition for large and streaming datasets. *Physics of Fluids*, 26(11), 2014.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21807–21818. IEEE, 2024.

Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. Vace: All-in-one video creation and editing. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17191–17202. IEEE, 2025.

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 *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15240–15252. IEEE, 2025.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Max Ku, Cong Wei, Weiming Ren, Harry Yang, and Wenhu Chen. Anyv2v: A tuning-free framework for any video-to-video editing tasks. *arXiv preprint arXiv:2403.14468*, 2024.

J Nathan Kutz, Steven L Brunton, Bingni W Brunton, and Joshua L Proctor. *Dynamic mode decomposition: data-driven modeling of complex systems*. SIAM, 2016.

Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, and Ming-Hsuan Yang. Learning blind video temporal consistency. In *European conference on computer vision*, pp. 179–195. Springer, 2018.

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 *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7353–7363. IEEE, 2025a.

Jiacheng Liu, Chang Zou, Yuanhuiyi Lyu, Junjie Chen, and Linfeng Zhang. From reusing to forecasting: Accelerating diffusion models with taylorseers. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15853–15863. IEEE, 2025b.

Jiacheng Liu, Chang Zou, Yuanhuiyi Lyu, Fei Ren, Shaobo Wang, Kaixin Li, and Linfeng Zhang. Speca: Accelerating diffusion transformers with speculative feature caching. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 10024–10033, 2025c.

Ziming Liu, Yifan Yang, Chengruidong Zhang, Yiqi Zhang, Lili Qiu, Yang You, and Yuqing Yang. Region-adaptive sampling for diffusion transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2346–2356, 2026.

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. *Advances in neural information processing systems*, 35:5775–5787, 2022.

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 *International Conference on Learning Representations*, volume 2025, pp. 33132–33156, 2025.

Xinyin Ma, Gongfan Fang, and Xinchao Wang. Deepcache: Accelerating diffusion models for free. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15762–15772. IEEE, 2024.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4172–4182. IEEE, 2023.

Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 724–732, 2016.

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

Clarence W Rowley, Igor Mezić, Shervin Bagheri, Philipp Schlatter, and Dan S Henningson. Spectral analysis of nonlinear flows. *Journal of fluid mechanics*, 641:115–127, 2009.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022.

Peter J Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of fluid mechanics*, 656:5–28, 2010.

Pratheba Selvaraju, Tianyu Ding, Tianyi Chen, Ilya Zharkov, and Luming Liang. Fora: Fast-forward caching in diffusion transformer acceleration. *arXiv preprint arXiv:2407.01425*, 2024.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023.

Jonathan H Tu. *Dynamic mode decomposition: Theory and applications*. Princeton University, 2013.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025a.

Zhang Wan, Yu Li, Haochen Li, Juan Cao, and Sheng Tang. Accelerating video diffusion transformers with spatial-temporal sparsity. 2025b.

Felix Wimbauer, Bichen Wu, Edgar Schoenfeld, Xiaoliang Dai, Ji Hou, Zijian He, Artsiom Sanakoyeu, Peizhao Zhang, Sam Tsai, Jonas Kohler, et al. Cache me if you can: Accelerating diffusion models through block caching. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6211–6220. IEEE, 2024.

Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7589–7599. IEEE, 2023a.

Jay Zhangjie Wu, Xiuyu Li, Difei Gao, Zhen Dong, Jinbin Bai, Aishani Singh, Xiaoyu Xiang, Youzeng Li, Zuwei Huang, Yuanxi Sun, et al. Cvpr 2023 text guided video editing competition. *arXiv preprint arXiv:2310.16003*, 2023b.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023.

Wufeng Xue, Lei Zhang, Xuanqin Mou, and Alan C Bovik. Gradient magnitude similarity deviation: A highly efficient perceptual image quality index. *IEEE transactions on image processing*, 23(2): 684–695, 2013.

Hao Zhang, Clarence W Rowley, Eric A Deem, and Louis N Cattafesta. Online dynamic mode decomposition for time-varying systems. *SIAM Journal on Applied Dynamical Systems*, 18(3): 1586–1609, 2019.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 586–595. IEEE, 2018.

Xuanlei Zhao, Xiaolong Jin, Kai Wang, and Yang You. Real-time video generation with pyramid attention broadcast. In *International Conference on Learning Representations*, volume 2025, pp. 3296–3319, 2025.

Xin Zhou, Dingkang Liang, Kaijin Chen, Tianrui Feng, Xiwu Chen, Hongkai Lin, Yikang Ding, Feiyang Tan, Hengshuang Zhao, and Xiang Bai. Less is enough: Training-free video diffusion acceleration via runtime-adaptive caching. *arXiv preprint arXiv:2507.02860*, 2025.

Chang Zou, Xuyang Liu, Ting Liu, Siteng Huang, and Linfeng Zhang. Accelerating diffusion transformers with token-wise feature caching. *arXiv preprint arXiv:2410.05317*, 2024.

A Appendix

A.1 Extended preliminaries

Residual view. Caching the full output $\mathbf{y}_t^{(b)}$ is unstable because it varies with both the latent input and the timestep. Caching the residual $\mathbf{r}_t^{(b)} = \mathbf{y}_t^{(b)} - \mathbf{x}_t$ isolates the denoising correction, and since $\mathbf{x}_t$ is always available in the sampling loop, only the slowly varying part has to be predicted. Zeroth-order reuse holds it constant, $\widehat{\mathbf{r}}_t^{(b)} = \mathbf{r}_\tau^{(b)}$, which is accurate when the residual is near constant between exact steps and degrades when the trajectory rotates, decays, or oscillates, even if the residual norm changes slowly. That distinction, direction against magnitude, is why a trigger based on the norm cannot detect the failure it causes.

Low-dimensional structure. Over a window of m exact steps, collect $\mathbf{Z} = [\mathbf{z}_{t_1}, \dots, \mathbf{z}_{t_m}] \in \mathbb{R}^{2D \times m}$ and take a truncated decomposition $\mathbf{Z} \approx \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$. The singular values decay quickly, so a small k captures most of the energy and $\mathbf{z}_t \approx \mathbf{U}\mathbf{a}_t$ with $\mathbf{a}_t = \mathbf{U}^\top \mathbf{z}_t$. The caching target changes from storing snapshots in $\mathbb{R}^{2D}$ to storing a shared basis plus compact coefficients.

Koopman approximation. Over a local window the reduced coefficients satisfy $\mathbf{a}_{t+\delta} \approx \mathbf{A}_\delta \mathbf{a}_t$ with $\mathbf{A}_\delta = \exp(\mathbf{\Omega}\delta)$, a finite-dimensional Koopman approximation realized by dynamic mode decomposition, where nonlinear high-dimensional evolution is represented by linear dynamics over chosen observables, here the residual modes (Schmid, 2010; Kutz et al., 2016; Rowley et al., 2009; Tu, 2013). Given windowed coefficients the reduced operator is the least-squares fit

$$\widehat{\mathbf{A}} = \arg\min_{\mathbf{A}} \sum_{i=1}^{m-1} \left\| \mathbf{a}_{t_{i+1}} - \mathbf{A}\, \mathbf{a}_{t_i} \right\|_2^2, \tag{5}$$

whose eigendecomposition gives magnitudes encoding decay or growth and phases encoding rotation, which is precisely what a single anchor, with unit magnitude and no phase, cannot represent.

Storage for compute. One exact guided step costs C_{net}, dominated by the network. A rank-k skipped step costs $\mathcal{O}(k^3)$ for the operator advance plus $\mathcal{O}(Dk)$ for the projection, negligible for $k \ll D$, and the price is memory $\mathcal{M} = \mathcal{O}(Dk) + \mathcal{O}(k^2) + \mathcal{O}(mk)$ for basis, operator, and history. The objective, given a budget $\mathcal{M}_{\max}$ and a distortion tolerance ε, is to minimize the number of exact evaluations subject to $\mathcal{M} \le \mathcal{M}_{\max}$ and an accumulated distortion below ε. Because $\mathbf{g}_t$ is unavailable on skipped steps, a practical policy must estimate this risk from exact-step quantities and correct itself online, which is the closed loop of section 4.3.

Prior caches as special cases. Zeroth-order anchor reuse is $\mathbf{A}_\delta = \mathbf{I}$, which freezes the coefficient and ignores rotation and decay. First-order extrapolation is the truncation $\exp(\mathbf{\Omega}\delta) \approx \mathbf{I} + \mathbf{\Omega}\delta$, a single local trend with no curvature or oscillation. Keeping the full exponential represents decay, bounded growth after eigenvalue clipping, rotation, and coupled evolution of the two guidance branches in one compact state.

A.2 Online mode extraction, details

We form lagged matrices $\mathbf{Z} = [\mathbf{z}_{\tau_1}, \dots, \mathbf{z}_{\tau_{m-1}}]$ and $\mathbf{Z}' = [\mathbf{z}_{\tau_2}, \dots, \mathbf{z}_{\tau_m}]$, take a thin decomposition truncated to rank k for the basis $\mathbf{U}$, and recover the reduced one-step operator by the standard projection $\widetilde{\mathbf{A}} = \mathbf{U}^\top \mathbf{Z}' \mathbf{V} \mathbf{\Sigma}^{-1}$. Because exact steps are not uniformly spaced we work with a continuous-time generator instead. With reduced states $\mathbf{a}_{\tau_j} = \mathbf{U}^\top \mathbf{z}_{\tau_j}$ and finite differences $\dot{\mathbf{a}}_{\tau_j} = (\mathbf{a}_{\tau_{j+1}} - \mathbf{a}_{\tau_j})/(\tau_{j+1} - \tau_j)$ we fit

$$\mathbf{\Omega} = \dot{\mathbf{A}}\, \mathbf{A}^+, \qquad \mathbf{A} = [\mathbf{a}_{\tau_1}, \dots, \mathbf{a}_{\tau_{m-1}}], \quad \dot{\mathbf{A}} = [\dot{\mathbf{a}}_{\tau_1}, \dots, \dot{\mathbf{a}}_{\tau_{m-1}}], \tag{6}$$

with $\mathbf{A}^+$ the pseudoinverse, so prediction over any gap δ is the exact exponential. To bound extrapolation we project the eigenvalues, $\mathbf{\Omega} \leftarrow \mathbf{W} \operatorname{diag}(\mu_i')\, \mathbf{W}^{-1}$ with $\mu_i' = \min(\operatorname{Re}\mu_i, \mu_{\max}) + \mathrm{i}\operatorname{Im}\mu_i$, which guarantees no mode blows up during a long skip run. Online, the basis is updated by an incremental thin decomposition with forgetting factor $\rho \in (0, 1]$ and $\mathbf{\Omega}$ is refit by eq. (6) over the retained window, at $\mathcal{O}(Dk)$ plus $\mathcal{O}(k^3)$ per exact step.

A.3 RECONSTRUCTION, DETAILS

We optionally separate the slowly varying mean of the residual with an affine offset in the reduced space, $\widehat{\mathbf{z}}_t = \mathbf{U}(\exp(\mathbf{\Omega}\delta)\mathbf{a}_\tau + \mathbf{b})$, which helps when a constant component of the residual is not aligned with the dynamic modes. The exponential is computed once per skip on the $k \times k$ generator and reused for both branches, since the basis is shared. Reconstructing a run of δ skipped steps uses one exponential rather than δ repeated applications, so a fixed-step bias is not accumulated along the run.

A.4 ROUTING AND CALIBRATION, DETAILS

The moving averages are $\bar{e}_{\bullet,t} = \lambda \bar{e}_{\bullet,t-1} + (1-\lambda)\,\mathrm{clip}(e_{\bullet,t}, 0, e_{\max})$ for $\bullet \in \{\perp, \Omega\}$, and the amplifier is linear in the gap relative to the mean exact spacing $\bar{s}$, namely $\phi(\delta) = 1 + \delta/\bar{s}$. The budget resets at every exact step and the staleness cap upper bounds any skip run, so no trajectory is extrapolated indefinitely. Calibration uses the guided-level error $E_t^{\mathrm{obs}} = \|\widehat{\mathbf{g}}_t - \mathbf{g}_t\|_2^2 / \|\mathbf{g}_t\|_2^2$ and the ratio $r_t = E_t^{\mathrm{obs}}/B_t$, with

$$\bar{r}_t = \lambda \bar{r}_{t-1} + (1-\lambda)\,\mathrm{clip}(r_t, r_{\min}, r_{\max}), \qquad s_t = \mathrm{clip}\Big(\tfrac{1}{\bar{r}_t},\, s_{\min},\, s_{\max}\Big), \qquad \Theta_t = \theta\, s_t. \quad (7)$$

The adaptive rank rule appends a mode when $\bar{e}_\perp$ exceeds a high watermark and the budget permits, initializing it from the leading discarded singular direction, and drops the smallest singular direction when $\bar{e}_\perp$ falls below a low watermark, with k clipped to $[1, k_{\max}]$ and projected onto the feasible set of eq. (4).

A.5 MEMORY MANAGEMENT, DETAILS

The compression operator has three interchangeable modes: spatial downsampling stores each mode at reduced resolution and upsamples on read, frame sharing stores one spatial pattern broadcast along the temporal axis, and low-rank factorization writes each mode as a product of two thin factors over the token and channel axes. Reconstruction is applied to $\mathcal{C}(\mathbf{U})$, and because the calibrator measures realized error including the compression bias, the compression rate acts as an additional dial whose induced error is absorbed online. The coefficient history is a ring buffer of fixed length, so memory is bounded independently of T. For edit chains the compressed basis and generator are carried into the next round under decay, and an optional cross-request bank keyed by a pooled prompt embedding persists bases across prompts under least-recently-used eviction.

A.6 EXPERIMENTAL SETUP, DETAILS

All latency numbers within a table are measured on the same device with the same precision and the same solver, and fidelity is always measured against the unaccelerated output of the same backbone under the identical seed, prompt, and round. Baselines use their official implementations where available and are otherwise reimplemented faithfully. The controller exposes the initial rank k, the maximum rank $k_{\max}$, the compression rate of $\mathcal{C}$, the buffer length m, the forgetting factor ρ, the base threshold θ, the staleness cap A, and the warm-up and tail set, all selected once on a held-out set of clips with two seeds each and then frozen across categories, datasets, and rounds, so no per-task or per-round tuning is used.

The ablation of table 3 reports a different memory range, 0.050 to 0.066 GB, than the single-round table, 0.031 to 0.041 GB, because the ablation runs the four-round protocol in which the carried basis and the coefficient history of the previous round are resident. The two ranges are therefore not comparable across tables, and only within-table comparisons should be made. The LTX-Video, Wan2.1 text-to-image, and Wan2.1 text-to-video results of tables 4 and 5 use the same device and precision within each table; latencies are comparable within a block of a table but not across backbones or modalities.

A.7 EDIT-CHAIN CONSTRUCTION

Each chain starts from a VBench prompt (Huang et al., 2024), from which Wan2.1 (Wan et al., 2025a) generates the initial clip once. GLM-5.1, called through its API, then expands the prompt into four

cumulative edit instructions, one for each instruction type: style, lighting, background, and object. The same four types are applied in a different order across chains. Each round receives the output of the preceding round together with its instruction; the round-1 prompt is printed above each chain in fig. 3 and section A.10. This expansion is part of benchmark construction: its outputs define the evaluation inputs and are identical for every method, backbone, and round.

A.8 PROMPT ALIGNMENT AND TEMPORAL STABILITY

Table 6: **Prompt alignment and temporal stability** at the fourth editing round. Warp error measures temporal consistency and is the one metric on which ALIGNVACE does not lead.

Method	CLIP-T ↑	Warp error ↓	GMSD ↓	SSIM ↑
Wan-VACE baseline	0.2954	0.00951	n/a	n/a
TeaCache	0.2974	0.00929	0.2104	0.6281
FasterCache	0.2963	0.01597	0.2114	0.4941
EasyCache	0.2980	**0.00889**	0.1671	**0.7495**
HetCache	0.2938	0.00948	0.2041	0.6533
ALIGNVACE (ours)	**0.3001**	0.00964	**0.1395**	0.7461

At the end of the chain ALIGNVACE has the best prompt alignment and the best gradient-magnitude similarity, and is second on SSIM behind EasyCache by 0.0034. Its warp error, 0.00964, is the highest among the accelerators and slightly above the unaccelerated baseline at 0.00951. We read this as a real limitation rather than noise: the predictor reconstructs each step from a shared spatial basis whose temporal structure is only implicit, so residual errors are not guaranteed to be temporally coherent even when per-frame fidelity is high. A temporally structured basis is the natural remedy and we leave it to future work.

A.9 SINGLE-ROUND GENERATION, PER-CLIP RESULTS

Figures 5 to 9 give the per-clip form of table 5 on twenty prompts. ALIGNVACE is the most faithful method on every clip shown, by margins between 1.2 and 12.9 dB over the best baseline on that clip, and the per-clip speedup varies from $1.63\times$ to $2.06\times$ around the $1.86\times$ average of table 5, because the router spends exact steps where a clip needs them rather than at a fixed rate. Two patterns recur. The large PSNR gaps are not uniform blur but scene substitution: on several clips an anchor cache produces a well-formed video of a slightly different scene, which is why the reference metrics and VBench disagree in section 5.5. And the cases where all methods are close, such as the object-centred clips with a static background, are the ones where a frozen anchor is a good approximation, which is the regime our model contains as its $\mathbf{\Omega} = \mathbf{0}$ special case.

A.10 FOUR-ROUND EDITING CHAINS

Figures 10 to 19 give the per-clip form of table 2 on ten chains, each with the same four instruction types applied in a different order. Three observations hold across them. Differences concentrate at mask boundaries, on identity-bearing objects such as the red beret, the ribbon, and the red sail, and in high-frequency texture, which are the places where a frozen anchor is least able to track the residual during a long skip run. The first round is close across methods on several chains and is occasionally won by a baseline, so an accelerator for this workload cannot be characterized by a single-round comparison. And the per-clip speedup of ALIGNVACE varies from $1.74\times$ to $2.02\times$ around the $1.92\times$ average of table 2, with EasyCache faster on one chain and equal on another, so the speed advantage is a tendency rather than a uniform property.

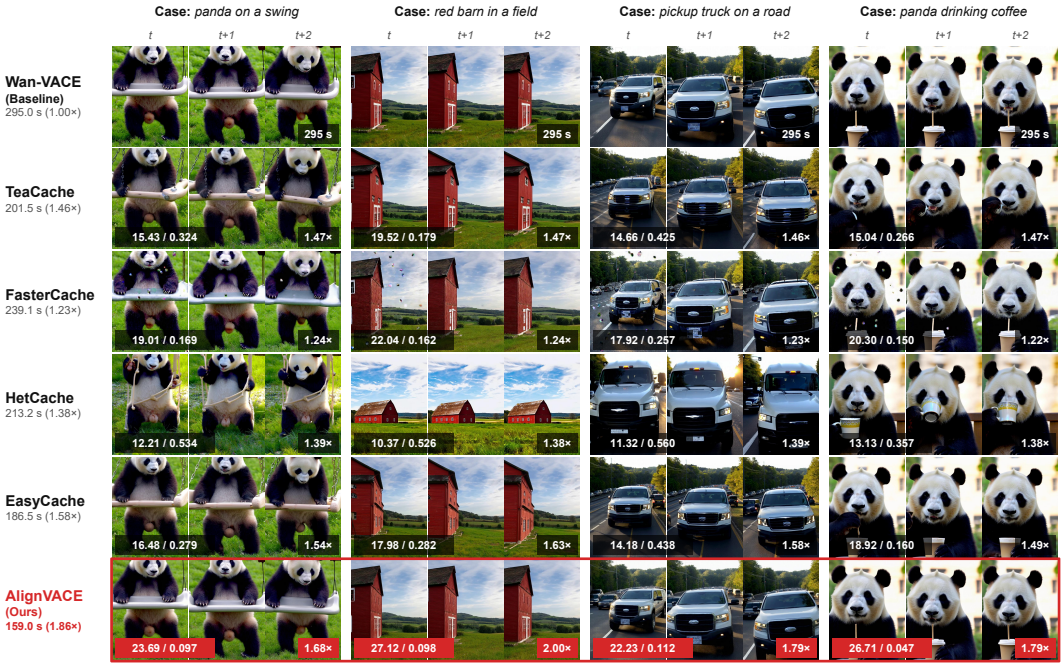

Figure 5: **Single-round generation, clips 1 to 4.** Three consecutive frames per clip for the unaccelerated Wan2.1 backbone and the five accelerators, on *panda on a swing*, *red barn in a field*, *pickup truck on a road*, and *panda drinking coffee*. The pair inside each block is PSNR and LPIPS against the unaccelerated output of the same prompt and seed, and the number at the right is the speedup measured on that clip. The per-method latency at the left of each row is the average over the clip set of table 5.

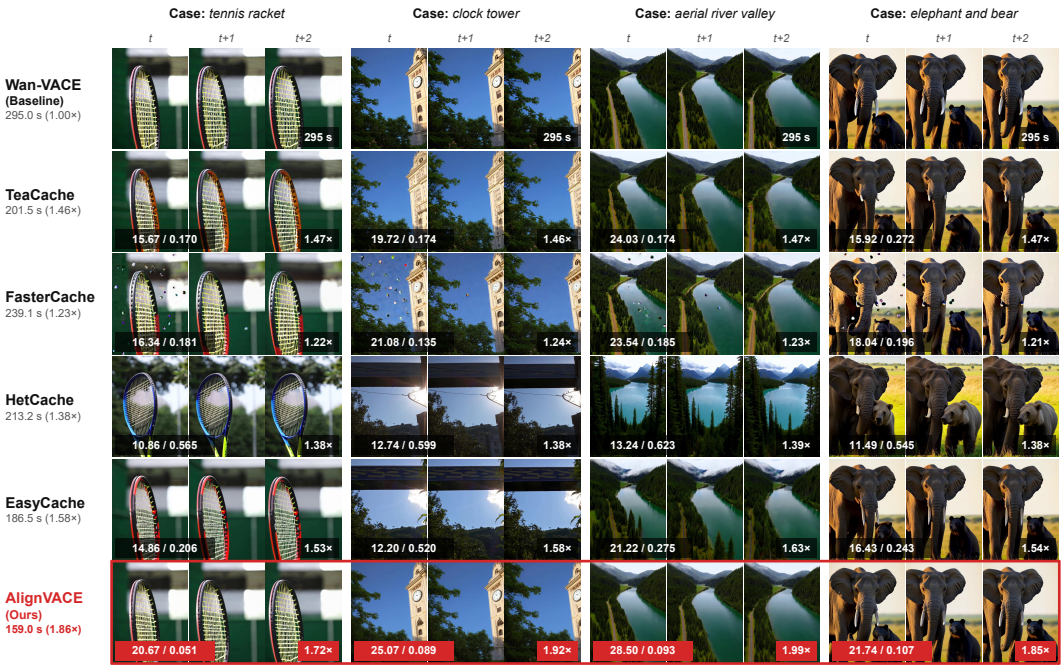

Figure 6: **Single-round generation, clips 5 to 8.** Panel layout as in fig. 5, on *tennis racket*, *clock tower*, *aerial river valley*, and *elephant and bear*. The clock tower and river valley clips show the failure that reference metrics summarize: HetCache and EasyCache replace the sky and the valley walls with a different scene while keeping a plausible video, which a reference-free score would not penalize.

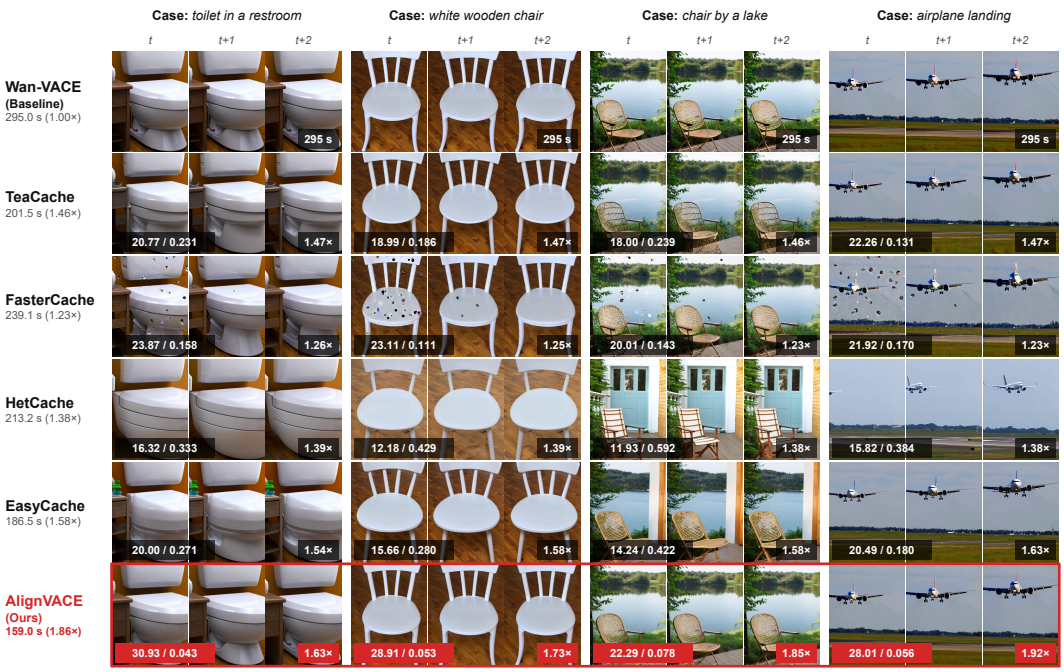

Figure 7: **Single-round generation, clips 9 to 12.** Panel layout as in fig. 5, on *toilet in a restroom*, *white wooden chair*, *chair by a lake*, and *airplane landing*. FasterCache shows the speckle artefacts that account for its low LPIPS relative to its PSNR.

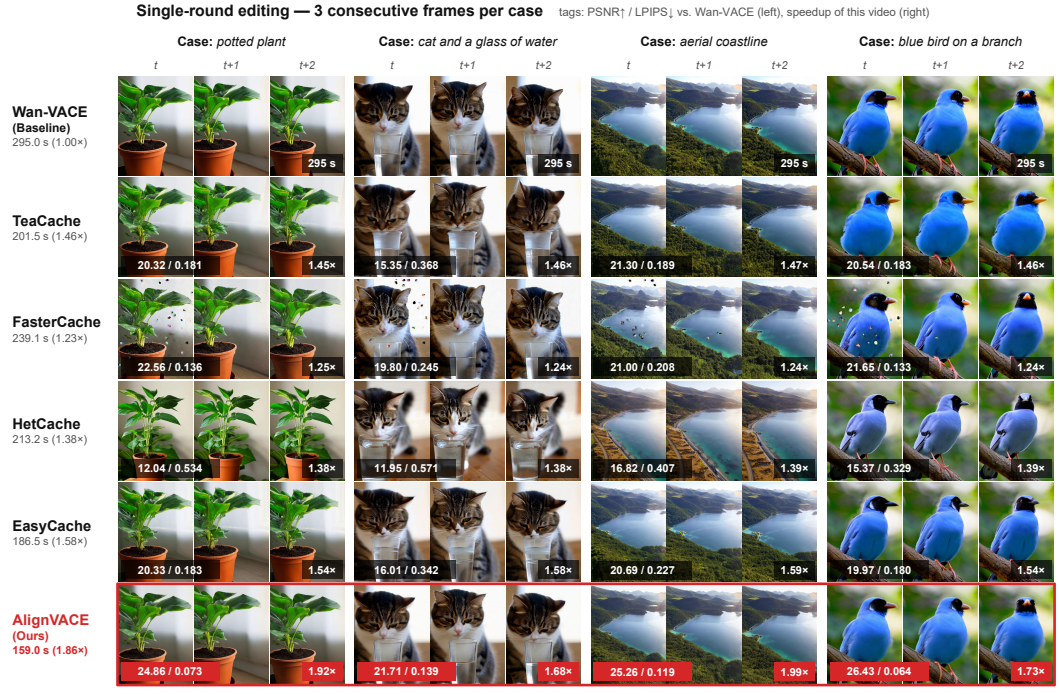

Figure 8: **Single-round generation, clips 13 to 16.** Panel layout as in fig. 5, on *potted plant*, *cat and a glass of water*, *aerial coastline*, and *blue bird on a branch*.

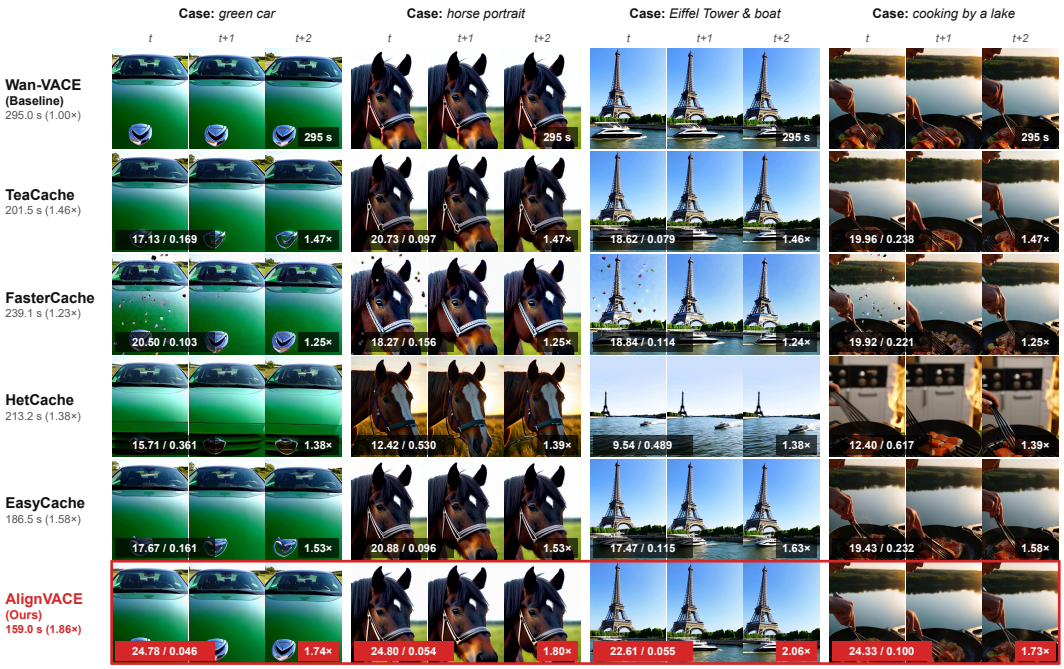

Figure 9: **Single-round generation, clips 17 to 20.** Panel layout as in fig. 5, on *green car*, *horse portrait*, *Eiffel Tower and boat*, and *cooking by a lake*.

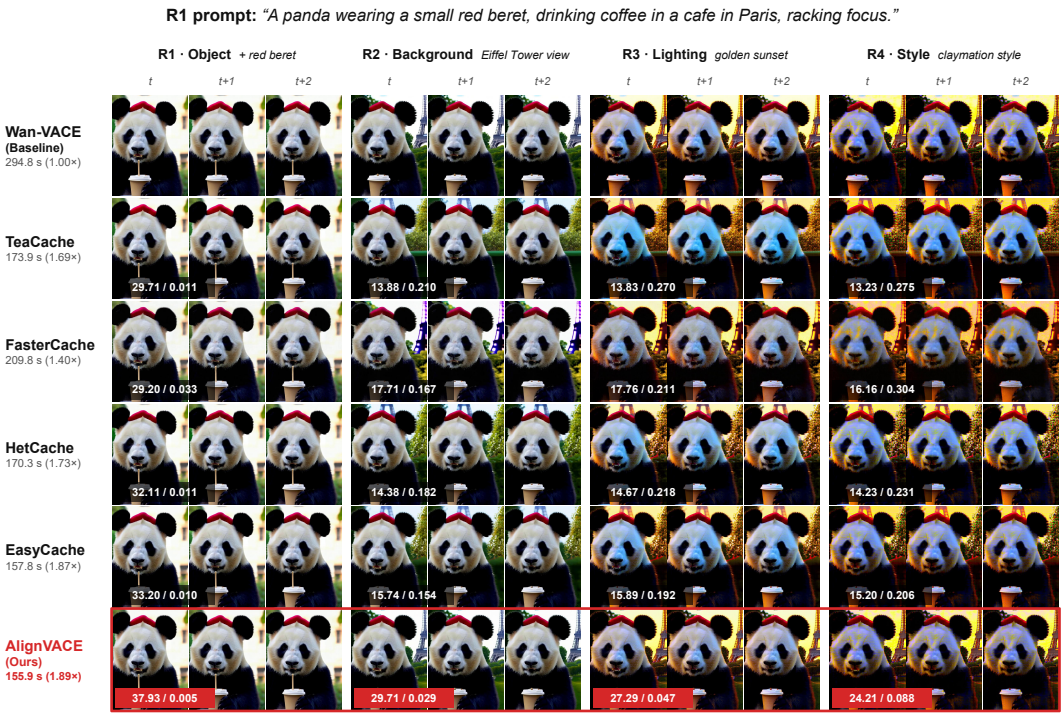

Figure 10: **Chain 1: panda in a Paris cafe.** Four successive edits applied to one clip, three consecutive frames per round; the round-1 prompt is printed above and each column adds one instruction. The pair inside each block is PSNR and LPIPS against the unaccelerated chain at that round, and the latency at the left is the average over the four rounds. ALIGNVACE holds 37.93 to 24.21 dB across the chain while every baseline is between 13.2 and 17.8 dB from the second round on, which is the per-clip form of the gap in table 2.

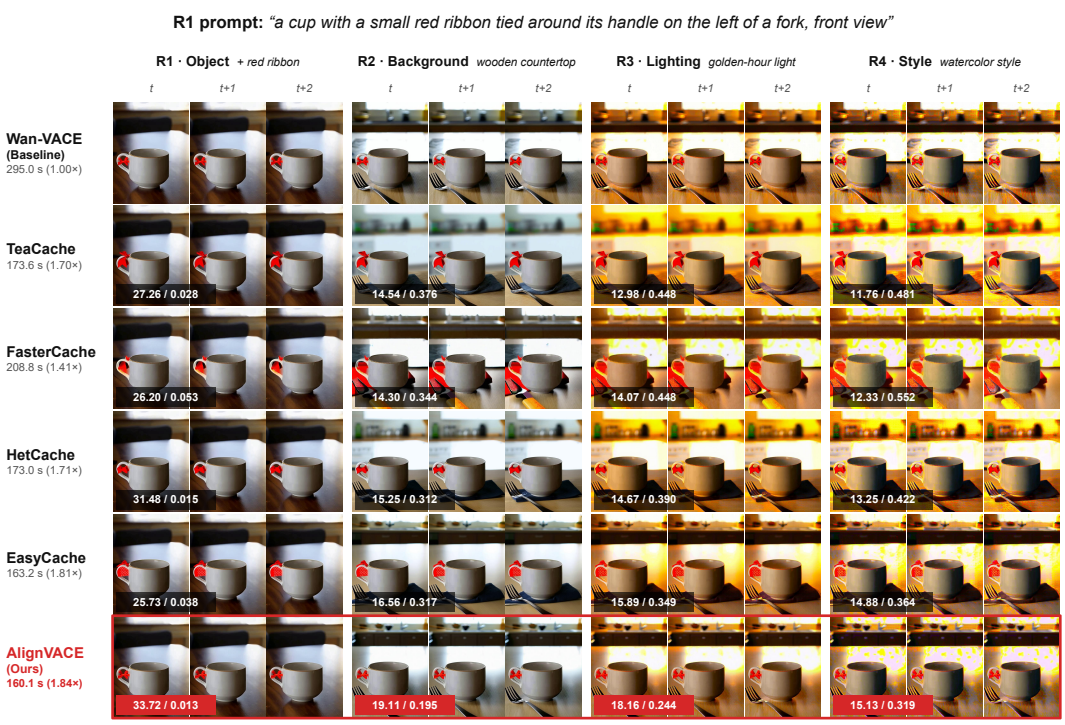

Figure 11: **Chain 2: cup with a red ribbon.** Panel layout as in fig. 10. All methods are close at round 1, where the edit is a small object insertion, and separate once the background is replaced at round 2; by round 4 the ribbon and the fork geometry survive only under ALIGNVACE, at 15.13 dB against 11.76 to 14.88 dB.

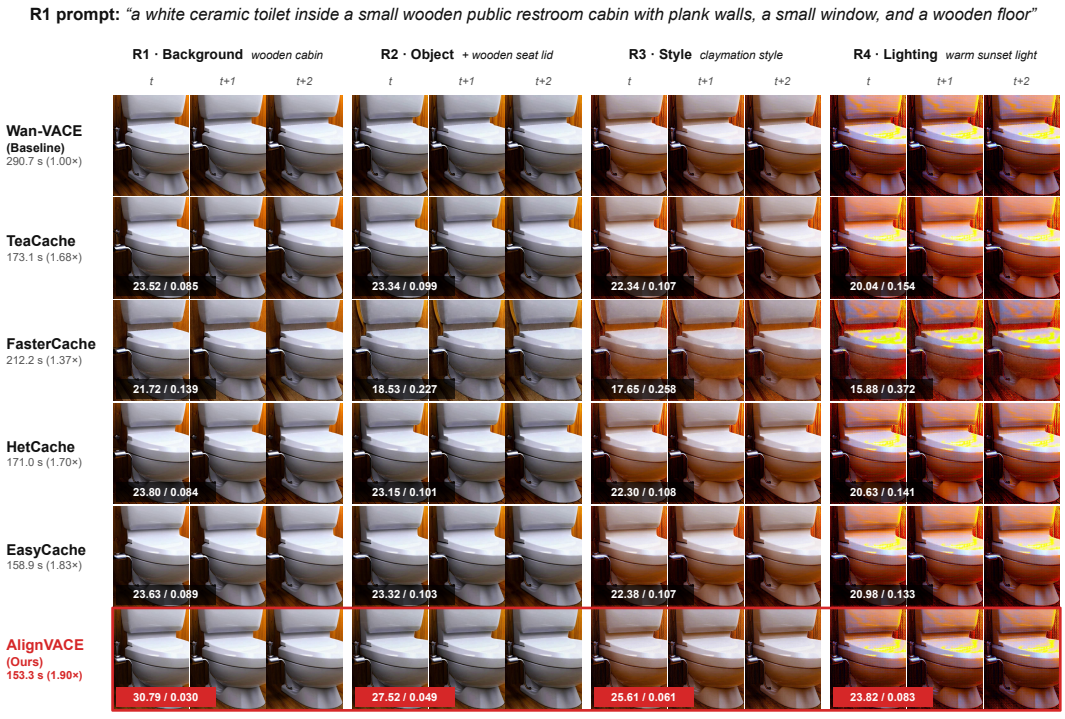

Figure 12: **Chain 3: toilet in a wooden cabin.** Panel layout as in fig. 10. This is the easiest chain in the set, with three baselines still above 20 dB at round 4; ALIGNVACE leads at every round, 30.79 to 23.82 dB, and FasterCache is the only method that degrades sharply.

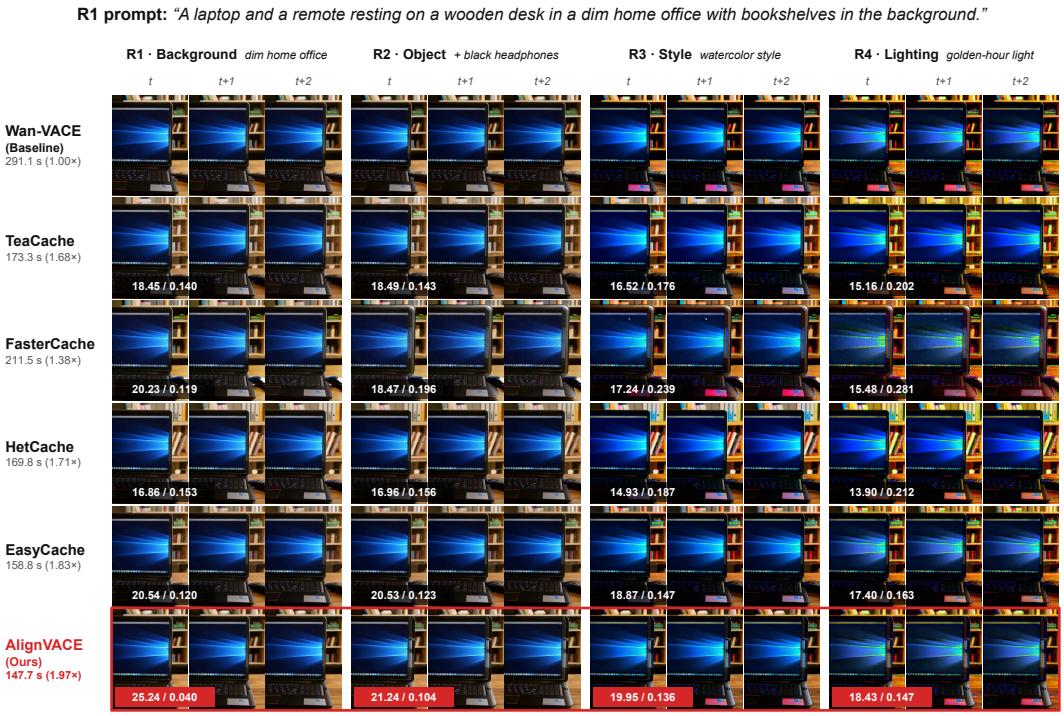

Figure 13: **Chain 4: laptop on a desk.** Panel layout as in fig. 10. The margin over EasyCache is largest at round 1, 4.70 dB, and narrows to 1.03 dB at round 4, so the lead is not uniform across chains and is smallest where the scene is dominated by flat regions.



Figure 14: **Chain 5: horse on sand dunes.** Panel layout as in fig. 10. This chain carries the largest speedup of the set, 2.02×, together with the highest round-1 fidelity, 36.19 dB, which is the combination the routing rule is meant to produce when the residual stays inside the stored modes.

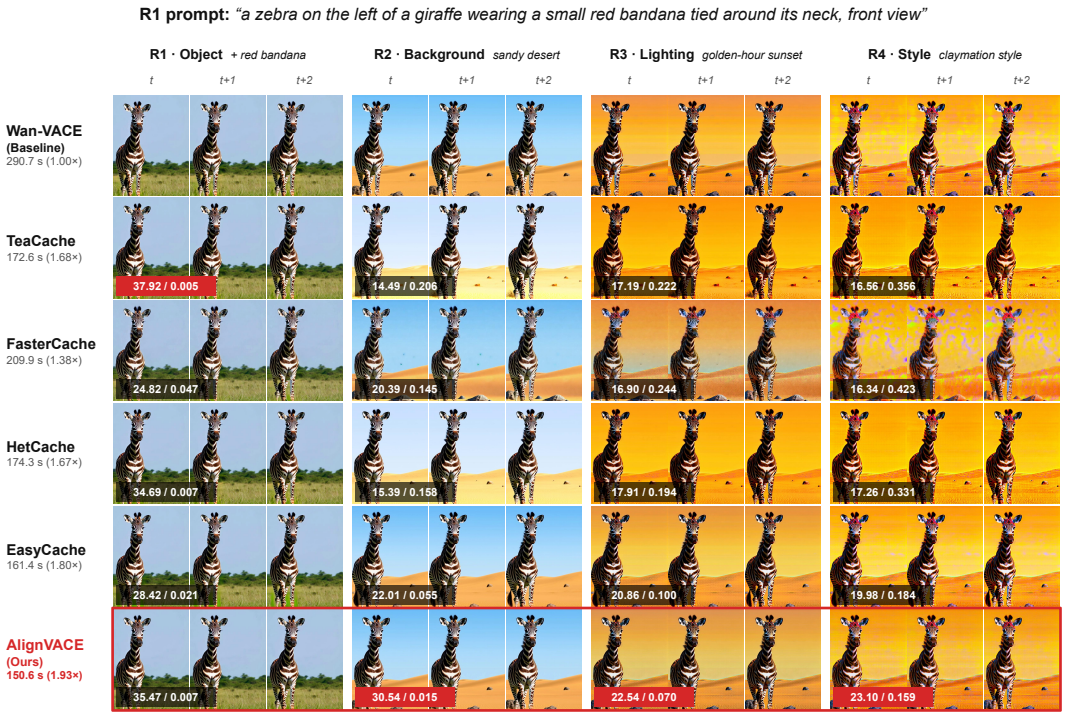

Figure 15: **Chain 6: zebra and giraffe.** Panel layout as in fig. 10. TeaCache is the most faithful method at round 1 here, 37.92 against our 35.47 dB, and falls to 16.56 dB by round 4 while ALIGNVACE holds 23.10 dB: a first-round win does not predict chain behaviour.

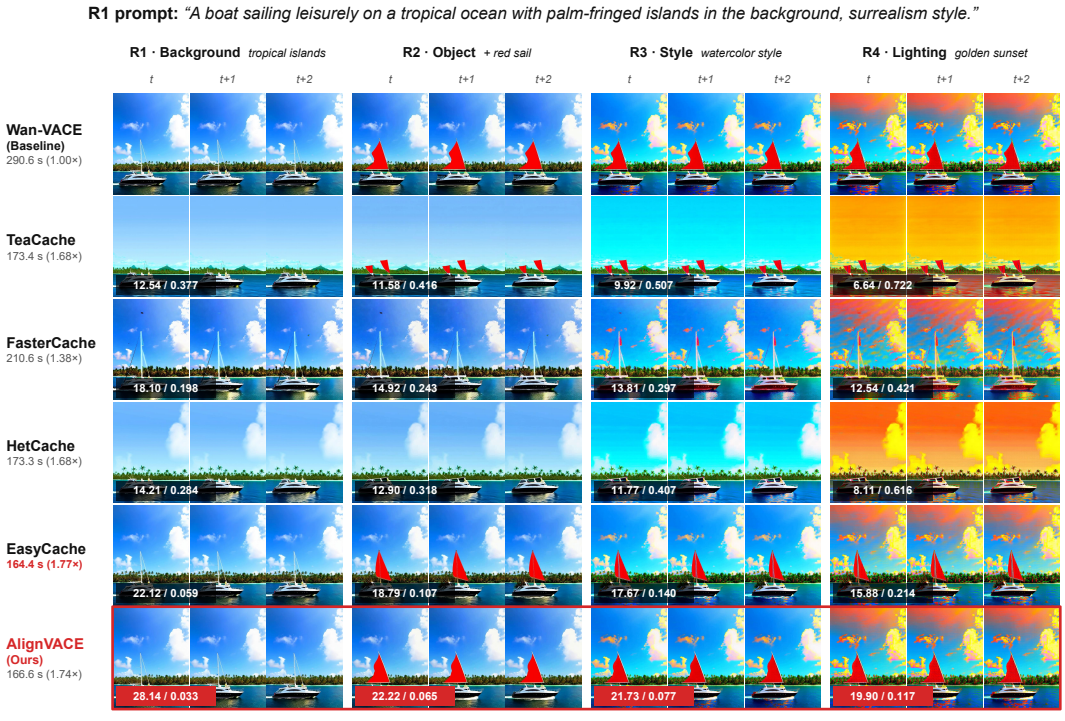

Figure 16: **Chain 7: boat on a tropical ocean.** Panel layout as in fig. 10. EasyCache is slightly faster on this clip, 1.77× against 1.74×, and is 3 to 6 dB less faithful at every round; the red sail inserted at round 2 is lost by TeaCache and HetCache and preserved by ALIGNVACE.

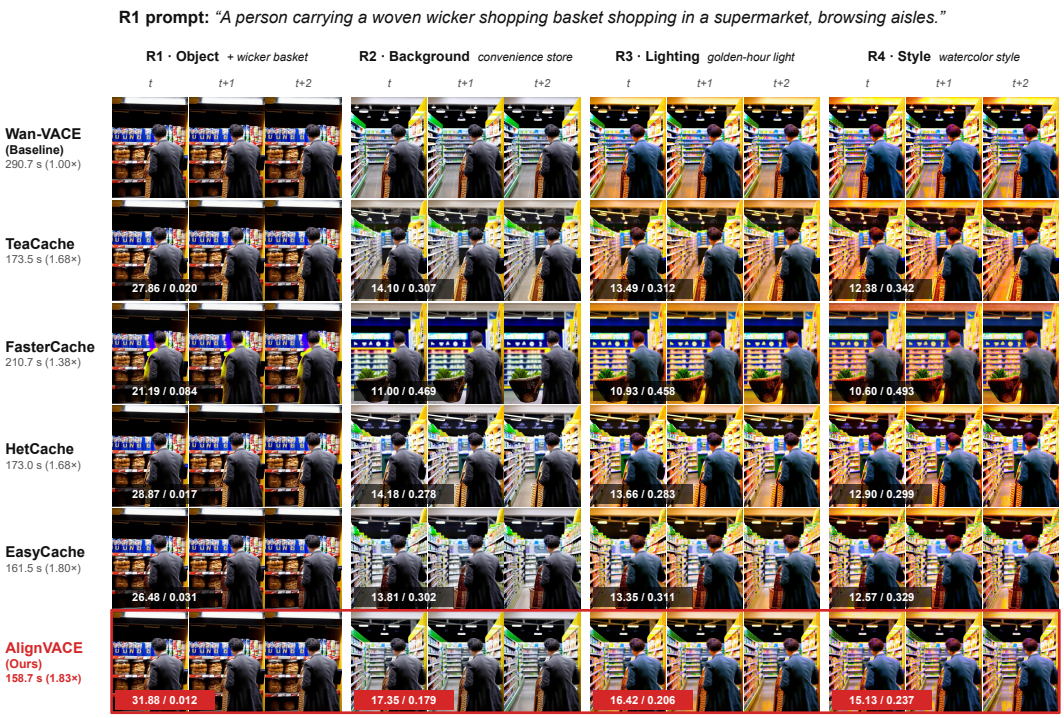

Figure 17: **Chain 8: shopper with a wicker basket.** Panel layout as in fig. 10. This is the hardest chain for every method, since the background replacement at round 2 regenerates most of the frame; all methods fall below 18 dB from round 2 on, and ALIGNVACE retains the largest relative margin in LPIPS.

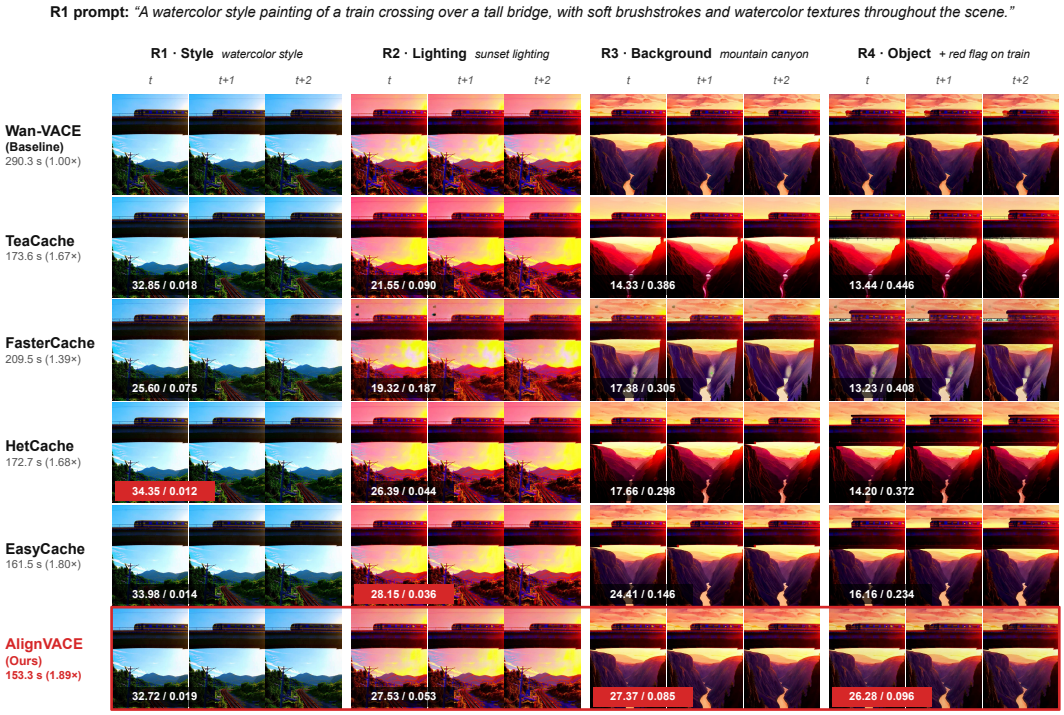

Figure 18: **Chain 9: watercolour train on a bridge.** Panel layout as in fig. 10. ALIGNVACE does not lead in the first two rounds here, where HetCache is best at round 1 (34.35 against 32.72 dB) and EasyCache at round 2 (28.15 against 27.53 dB), and leads by 2.96 and 10.12 dB at rounds 3 and 4 once the background and object edits compound.

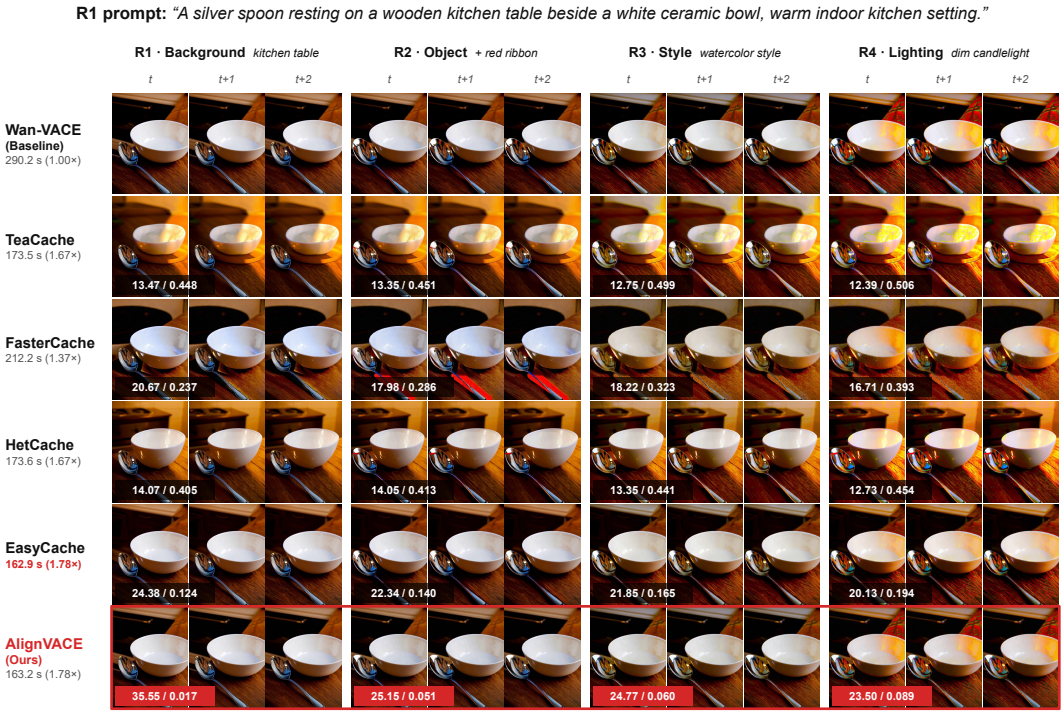

Figure 19: **Chain 10: spoon and bowl on a kitchen table.** Panel layout as in fig. 10. EasyCache matches our speedup on this clip, $1.78\times$ for both, and is 2.8 to 11.2 dB less faithful; TeaCache and HetCache lose the warm indoor lighting of the source at round 1 and never recover it.