MEASURING AND CORRECTING PRIOR TRANSMISSION IN CAUSAL VIDEO DIFFUSION

Anonymous authors
Paper under double-blind review

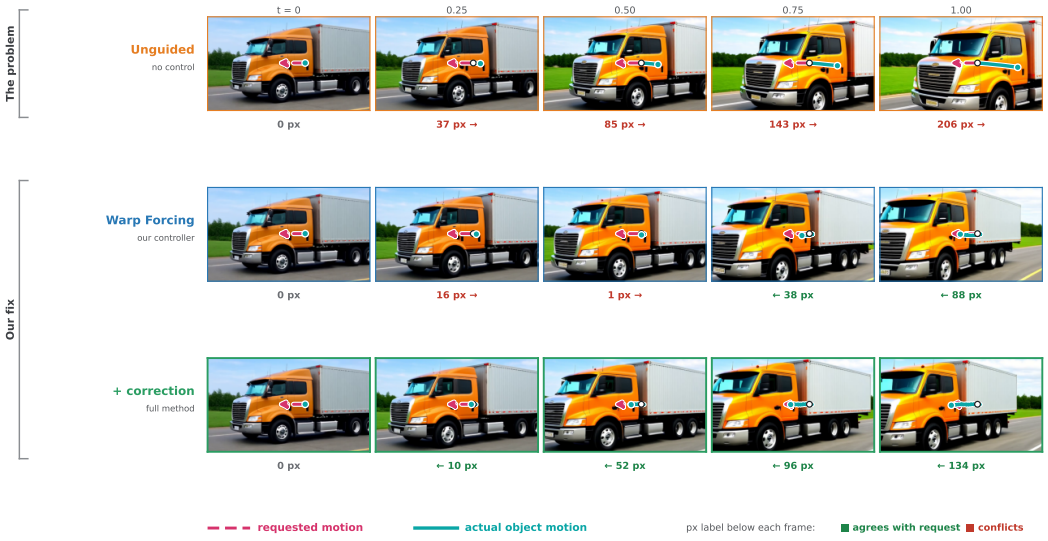


Figure 1: Controlled motion retains a component that is predictable from the matched unguided rollout, and compensating for it improves adherence. All rows use identical initial conditioning and sampler randomness for a 150-pixel leftward request. The unguided object moves 206 pixels right; Warp Forcing reverses direction but reaches only 88 pixels left, while matched-preview correction reaches 134 pixels left. Dashed paths show the request, solid paths show measured object trajectories, and values below each frame give cumulative horizontal movement.

ABSTRACT

A video that follows a requested trajectory may reflect effective control, motion that also appears in an unguided rollout from the same seed, or both. Yet evaluation based on target error alone does not reveal how much the controller improves on unguided generation or whether the remaining error reflects a predictable motion tendency. We introduce TrajectoryBench, a paired evaluation protocol that compares guided and unguided rollouts from matched conditioning and seeds, distinguishing final trajectory adherence from the improvement supplied by control. Across ten seeds on CausVid, controlled displacement is coupled to matched unguided displacement: roughly half of the unguided motion variation survives into the controlled output. We call the slope of this relationship prior transmission. The coupling persists after removing shared directional drift, survives replacement of the evaluation tracker, and appears on a second causal backbone, Causal Forcing. Predictable residual motion is not only an evaluation concern: it is a signal for improving control. We introduce Warp Forcing, a training-free trajectory controller for few-step causal video diffusion that steers objects block by block and adapts its commands to realized motion. Using the protocol's matched unguided rollout as a preview, drift compensation reduces mean path error relative to uncorrected Warp Forcing by 25% on CausVid and 38% on Causal Forcing across three seeds. A preview-free online estimator needs no extra rollout and adds no runtime: on Causal Forcing it matches the matched-preview correction, and on CausVid it re-

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.

covers about one-third of the preview-based improvement. Measuring matched unguided motion thus yields both more informative evaluation and more accurate trajectory control.

1 INTRODUCTION

Useful video control requires more than plausible motion: an object must reach the intended place at the intended time. In animation and interactive scene editing, a visually convincing clip can still be unusable if the subject misses a waypoint or continues in the wrong direction after a target update. A prompt such as "a dog runs right" leaves distance, timing, and path unspecified. Training-free trajectory controllers supply these constraints without retraining the generator, using motion objectives (Geng & Owens, 2024), attention and initial-noise interventions (Namekata et al., 2025; Qiu et al., 2026; Choo et al., 2026), or temporally correlated noise (Chang et al., 2024; Burgert et al., 2025).

Yet supplying a trajectory does not guarantee that the generated object will follow it. In our evaluation of training-free controllers on causal video diffusion, objects can move in the requested direction while traveling the wrong distance, and some interventions fail to establish reliable directional control. Such failures limit the usefulness of trajectory guidance for precise editing: users need not only plausible outputs, but predictable responses to their commands. Improving this reliability requires understanding what the controller changes and what motion persists despite its intervention.

A controlled trajectory can reflect both the intervention and motion that also appears when the same seed is generated without control. These components may support or oppose the same request. An object that already drifts toward its target can achieve low error with little improvement from control; an object that naturally moves away may remain inaccurate even after a substantial correction. Scoring only against the requested trajectory treats these outcomes as differences in adherence without revealing the improvement supplied by the controller. It also leaves an important design question unanswered: is the remaining error merely unpredictable variation, or does it follow a motion tendency that can be anticipated and compensated?

We introduce TrajectoryBench to study these questions through paired evaluation. Each guided rollout is compared with an unguided rollout generated from matched conditioning and seeds. This reference supports two complementary measurements: how much trajectory error the intervention removes, and how strongly controlled displacement remains associated with unguided motion. The first measures the benefit supplied by control; the second tests whether matched unguided motion predicts what remains in the controlled output. Together, they make evaluation more informative for controller design than target accuracy alone.

Using this protocol, we find that controlled motion is coupled to unguided motion, and call the slope of this relationship *prior transmission*. Across ten CausVid seeds, greater unguided displacement predicts greater controlled displacement along the requested axis, with the object and requested direction held fixed. The association persists after removing directional shifts shared by objects within each seed, holds under an independent tracker, and appears on Causal Forcing (CF). Comparisons across controllers show that low transmission and accurate trajectory following are distinct properties: suppressing dependence on unguided motion does not, by itself, produce useful control.

The practical value of this finding is that predictable residual motion provides a signal for correction. We test this with Warp Forcing, a training-free controller for few-step causal video diffusion that steers objects block by block and adapts its commands to realized motion. Using the protocol's matched unguided rollout as a preview, we estimate drift and compensate for it in the control command. Relative to uncorrected Warp Forcing, this reduces mean path error by 25% on CausVid and 38% on CF across three seeds. A preview-free online estimator improves control within a single rollout, matching the preview-based gain on CF and recovering about one-third of it on CausVid. Predictable residual motion is therefore not just a nuisance for evaluation but a usable correction signal.

Our contributions are:

- **Evaluation to distinguish adherence from control.** TrajectoryBench pairs guided and unguided generation to measure both error reduction and residual dependence on matched

unguided motion, providing a reference for understanding controller behavior beyond final target accuracy.

- **Evidence of predictive guided–unguided coupling.** A ten-seed CausVid analysis identifies guided–unguided dependence that persists after shared directional drift is removed, holds under an independent tracker, and appears on a second backbone. Controller comparisons show that transmission and accuracy capture different aspects of control.
- **A practical correction motivated by the analysis.** Warp Forcing uses estimated drift to improve blockwise trajectory control. Matched-preview compensation reduces path error by 25–38% on both causal backbones, and online estimation obtains this benefit without an additional rollout on CF and one-third of it on CausVid.

2 Related Work

Diffusion models dominate text- and image-conditioned video generation (Blattmann et al., 2023; Wan et al., 2025). Because whole-clip bidirectional sampling is slow, a recent line distills these models into few-step causal generators (Chen et al., 2024; Yin et al., 2025; Huang et al., 2025; Zhu et al., 2026), which incrementally stream by denoising each chunk in a handful of steps. A separate line adds trajectory control by training (Yin et al., 2023; Wang et al., 2024; Wu et al., 2024; Zhang et al., 2025; Niu et al., 2024), achieving strong adherence but needing paired data and a training run per backbone. Training-free control of autoregressive video diffusion has already been explored by DragStream (Zhou et al., 2026). We focus on reliable trajectory adherence in frozen few-step causal generators through direct clean-estimate warping, blockwise feedback, and compensation for residual motion. Alongside this controller, we introduce a paired evaluation protocol that separates the improvement supplied by guidance from motion already present in a matched unguided rollout.

2.1 Training-free trajectory guidance

Gradient-based guidance differentiates an auxiliary motion objective through the denoiser at every guided step (Geng & Owens, 2024), imposing motion directly at the cost of a backward pass per step. Attention- and noise-based methods avoid this gradient (Jain et al., 2024; Ma et al., 2024; Ling et al., 2025; Namekata et al., 2025; Qiu et al., 2026): they influence the features attended to or the noise supplied.

Motion-Zero (Chen et al., 2025) is directly relevant training-free prior work: it combines a position-conditioned initial-noise prior, spatial attention constraints, and shifted temporal attention for bounding-box trajectory control. Warp Forcing instead intervenes on clean estimates in a few-step causal sampler, with blockwise tracking and residual-motion compensation.

Closest to our setting is DragStream (Zhou et al., 2026), which supports training-free streaming drag-based editing and trajectory-guided animation. It iteratively optimizes noisy latent codes with a feature-based objective, using distribution self-rectification and spatial-frequency selective propagation of context. Warp Forcing instead applies a mask-scoped warp to the clean estimate without iterative gradient-based guidance, re-noises it for the remaining denoising steps, reconstructs causal context from the corrected output, and replans subsequent commands from tracked motion. Its motion compensation addresses residual object displacement, distinct from DragStream's correction of latent distribution statistics. Our emphasis is on quantitatively evaluating trajectory adherence and using predictable residual motion to improve it. QWERTY (Choo et al., 2026) is concurrent work on query-warped bidirectional DiTs. We transplant both mechanisms into CausVid (Yin et al., 2025) as transfer experiments under a shared causal sampler.

Noise-based approaches also correlate noise across frames along a flow field (Chang et al., 2024), with Go-with-the-Flow (Burgert et al., 2025) fine-tuning the base model to exploit it. SDEdit (Meng et al., 2022) and drag-based editing (Shi et al., 2023) are earlier edit–denoise precedents.

2.2 Evaluating trajectory control

Trajectory-control papers commonly score the generated trajectory against the requested path and supplement it with generic video-quality measures (Yin et al., 2023; Wu et al., 2024; Zhang et al.,

2025; Namekata et al., 2025; Qiu et al., 2026). Trajectory-specific evaluation sets include Tora's 185-clip motion-control benchmark (Zhang et al., 2025) and DragStream's 204-clip REVEL benchmark (Zhou et al., 2026), which assess target adherence using TrajError and ObjMC, respectively. Their reported protocols do not pair each controlled rollout with an unguided rollout under matched conditioning and seeds to measure the improvement attributable to control. An underspecified prompt can already produce motion toward the target, contributing to the reported adherence. TrajectoryBench adds this paired comparison, measuring both the reduction in target error and the association between controlled and unguided motion; it therefore complements target-adherence benchmarks by revealing how much guidance changes motion relative to the matched unguided rollout. Video-quality measures such as VBench and DOVER (Huang et al., 2024; Wu et al., 2023) complement this paired trajectory evaluation.

3 METHOD

Our framework pairs a measurement protocol with a causal trajectory controller. TrajectoryBench measures improvement over unguided generation and the coupling between guided and unguided displacement; Warp Forcing uses the matched unguided motion to correct its command.

3.1 SETUP AND INTERFACE

The backbone generates a video block-by-block: at block b it denoises a chunk of F latent frames (3 on CausVid, 1 on CF) over a short fixed schedule $t_1 > \cdots > t_K$, conditioned on a KV cache of previously generated frames, producing at each step a denoised estimate $\hat{x}_0^{(k)}$ which is appended to the cache at the end. The user supplies a poke (a location and an object description) and a displacement Δ, or a full trajectory. We locate the object in the first generated frame with GroundingDINO (Liu et al., 2023) and SAM2 (Ravi et al., 2024), and convert the requested displacement into a per-latent-frame rate; all other motion is left to the model's prior. The final target is $c_{\text{tot}} = c_0 + \Delta$, with c_0 the object's centroid on the first generated frame in native pixel coordinates.

For piecewise trajectories, the controller replans toward the active waypoint using the time remaining to that waypoint; Appendix C.2 specifies interpolation and execution.

3.2 TRAJECTORYBENCH

TrajectoryBench has 80 items: 20 everyday-object categories, each poked in four cardinal directions by 150px, using the template "A {object} is moving." Appendix A gives the categories, split, prompts, and tracking details; Appendix G extends the protocol to diagonal and curved targets.

The two passes use the same prompt, seed, and initial conditioning. The generation code resets the seed and reconstructs the initial noise before each pass; the detector and tracker run in both the guided and unguided pass (Appendix B). The unguided rollout supplies the reference for measuring the controller's contribution.

For each clip, CoTracker3 (Karaev et al., 2025) tracks a 20×20 point grid in the initial SAM2 mask, yielding centroid trajectory $c(t)$. Our primary metric is

$$\text{trajPathErr} = \frac{1}{T}\sum_{t=0}^{T-1} \left\| \big(c(t) - c(0)\big) - q\Big(\tfrac{t}{T-1}\Big) \right\|_2,$$

the mean deviation from the requested path (in pixel coordinates). A straight request uses $q(\tau) = \tau\Delta$, for which a stationary object scores 75px at a 150px target. For each matched pair, we report the error reduction

$$\Delta E = E_{\text{base}} - E_{\text{guided}},$$

where E denotes trajPathErr. Positive ΔE measures the improvement supplied by control.

Let x and y be base and guided net displacement projected onto the requested axis. In $y = \alpha + \beta x$, $\beta = 0$ denotes no linear transmission of unguided displacement variation and $\beta = 1$ denotes unit transmission. Accuracy is measured separately by path error. We define *prior transmission* as this slope relating matched unguided and guided displacement: it measures how much of the unguided

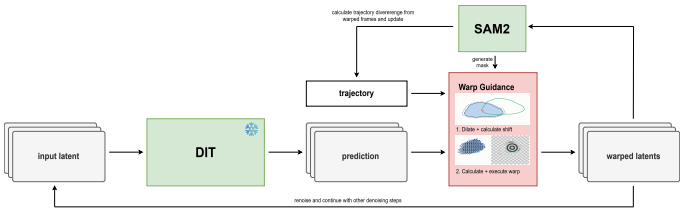


Figure 2: One guided block of Warp Forcing. The backbone denoises the block normally while SAM2 tracks the object from the last finalized frame. A shrinking-horizon replanner builds a dense flow field ϕ, which warps $\hat{x}_0$. The result is re-noised to step $k^\star{+}1$, returned to the sampler, finalized, and written to the KV cache.

motion the controlled output retains. Appendix A gives complete definitions of continuous FDA, netCos, DTW, sharpness, DOVER, and localized quality metrics.

3.3 Warp Forcing under causal constraints

Warp Forcing combines three operations: a clean-signal edit that the remaining denoising steps can reconcile, a cache update that propagates corrected content, and replanning that responds to realized motion.

Reconcilable clean-signal edit. At step $k^\star$, we warp the clean estimate with a mask-scoped flow field:

$$\tilde{x}_0^{(k^\star)}(p) = \hat{x}_0^{(k^\star)}(p - \phi(p)).$$

We construct ϕ by dilating the mask tracked on the last finalized frame, translating it to the requested destination, and feathering the destination's inner boundary into the identity field. The vacated source region receives the full displacement and is repainted during subsequent denoising (Appendix C.1). The field is resized from pixel coordinates to the spatially aligned latent grid and applied with bilinear sampling. Displacement magnitudes are divided by the VAE spatial downsampling factor of 8.

The scheduler then re-noises $\tilde{x}_0^{(k^\star)}$ to the next timestep's nominal level using its own forward rule, so editing the clean-signal estimate avoids a mismatch between edited signal and expected noise level.

Cache consistency and replanning. Each completed block updates both the context used by the generator and the position estimate used by the controller.

After final denoising, we restore the pre-block cache snapshot and regenerate its entries from the corrected content. Later blocks therefore condition on the emitted frames. During generation, finalized latents are VAE-decoded to RGB for SAM2 updates, providing the last finalized frame for replanning; tracking and decoding dominate the measured loop overhead (Table 8). Online SAM2 tracking (Ravi et al., 2024) supplies the current centroid c_b, from which we replan toward the target:

$$\text{rate}_b = \frac{c_{\text{tot}} - c_b}{B},$$

where B counts the remaining latent frames, including the current block. For piecewise requests, c_{tot} denotes the active segment endpoint $c_0 + \Delta_j$, and B counts the latent frames remaining to that endpoint. Within the block, frame $f \in \{0, \dots, F-1\}$ receives displacement $d_f = (f+1)\,\text{rate}_b$, with an independently constructed field ϕ_f. Recomputing this ramp after each block distributes the remaining position error over the remaining trajectory. We call this shrinking-horizon controller WF-r and use it throughout the main experiments.

3.4 Motion-based command correction

We compensate for unguided motion by subtracting an estimated drift rate from the replanned command:

$$\text{rate}_b = \frac{c_{\text{tot}} - c_b}{B} - g\,\delta_{\text{prior}}.$$

The matched-preview estimate uses an unguided rollout with the same prompt and seed:

$$\delta_{\text{prior}} = \frac{c^{\text{base}}(T_{\text{pix}} - 1) - c^{\text{base}}(0)}{T_{\text{lat}} - 1}.$$

Here T_{pix} and T_{lat} count pixel and latent frames. This converts net pixel displacement into a 2D rate per latent frame. The estimate is fixed throughout the guided rollout; feedback continues to update the target residual after each block. Offline CoTracker3 supplies the unguided preview centroids c^{base} and hence δ_{prior} before guided generation; SAM2 supplies the online feedback centroid c_b. Preview centroids use a fixed query-point set.

We also evaluate two preview-free estimates. The amortized estimate averages unguided drift over disjoint tune-half object categories for the same backbone and seed. The online estimate updates an exponential moving average of realized minus commanded block motion, with weight $\lambda = 0.5$, initialized from the uncorrected block 0 (Appendix C.5; on CF this initialization is effectively zero). Both operate without an additional rollout at generation time. We select the correction gain on the tune half: $g = 1.0$ for CausVid and $g = 1.5$ for CF.

4 Results

We first test prior transmission across ten CausVid seeds (§4.1), then evaluate correction across three seeds and compare drift estimators (§4.2). We next examine accuracy, quality, and computational cost (§4.3); scope and limitations follow in §5.

Unless noted, intervals are 95% percentile bootstraps (≥ 2000 draws) and paired tests are two-sided Wilcoxon tests. All four directions of an object share one base clip; regressions therefore cluster by object and, when pooled, by seed. Paired tests average directions within each of ten objects.

4.1 Guided displacement is coupled to matched unguided displacement

Prior transmission, as defined in §3.2, measures how controlled displacement varies with matched unguided displacement. We estimate it across seeds within each category-direction item, holding the requested trajectory fixed.

Primary estimate: 10-seed item fixed effects. We use cross-seed variation within each category-direction item:

$$y_{i,s} - \bar{y}_i \;=\; \beta\,(x_{i,s} - \bar{x}_i) + \epsilon_{i,s}$$

over 20 objects, four directions, and ten seeds: 800 observations from 200 base rollouts. The slope is 0.523, with two-way object$\times$seed 95% CI $[0.25, 0.80]$.

Adding seed-by-direction fixed effects gives $\beta = 0.525$ (two-way clustered 95% CI $[0.26, 0.79]$): the relationship persists after removing directional shifts shared by objects within each seed. All five dependence-aware inference schemes for the primary estimate exclude zero and one; leave-one-seed-out estimates range from 0.43 to 0.58 (Appendix H.3).

Seed matching. Reassigning each item's base rollouts to different seeds gives a mean slope of -0.059, with randomization interval $[-0.143, +0.037]$ across 2000 draws. The matched estimate lies above this interval. The small negative mean follows from the centered derangement design (Appendix H.3).

Tracker independence. Swapping CoTracker3 for an independent Lucas-Kanade tracker leaves the held-out pooled slope unchanged (0.665 vs. 0.671) and keeps the three-seed fixed-effects slope clearly positive (0.452, sign-flip $p<0.0001$). The pooled estimate is insensitive to any single object (Appendix H.3).

4.2 Correction and drift estimators

Matched-preview correction reduces path error by 25% on CausVid and 38% on CF across three seeds (Table 1). Mean errors fall from 47.4 to 35.7px and from 150.0 to 93.1px, respectively; both object-clustered intervals on the paired reduction exclude zero.

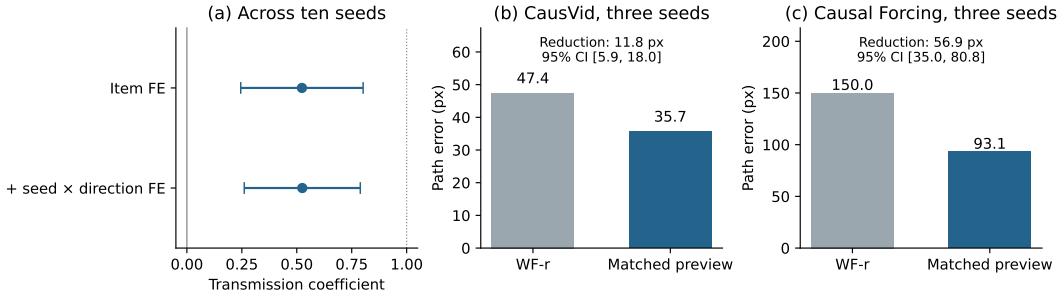


Figure 3: Measurement and correction. (a) The ten-seed transmission coefficient remains stable after adding seed-by-direction fixed effects; bars show two-way object×seed clustered 95% intervals. (b,c) Matched-preview correction lowers mean path error on both backbones across seeds 42, 7, and 123. Intervals describe the paired error reduction, clustering all directions and seeds by object. The correction panels use the campaign in Table 1; controller comparisons appear in Table 20.

backbone	WF-r	matched preview	reduction	relative reduction
CausVid	47.4	35.7	11.8 [5.9, 18.0]	25%
CF	150.0	93.1	56.9 [35.0, 80.8]	38%

Table 1: Matched-preview result pooled over three seeds (120 direction-items, 10 objects, 3 seeds per backbone). Brackets are 95% object-clustered intervals on the paired error reduction, computed before rounding. The seed-42 subset is the gain sweep in Appendix C.5 and in Table 2. Table 5 gives the per-seed means.

Response to correction strength. The seed-42 subset of the correction campaign (Appendix C.5) shows how correction changes accuracy and transmission together. At the selected gains, CausVid path error falls from 53.5 to 34.5px and its slope from 0.67 to 0.32; CF error falls from 158.8 to 91.0px and its slope from 0.56 to 0.23. On CF, the fraction exceeding the stationary-object error falls from 75% to 30%.

Value of the drift estimate. Larger unguided drift predicts greater benefit from correction: regressing each CausVid item's improvement at $g = 1$ on the magnitude of its measured unguided drift gives 0.224px additional improvement per pixel of drift ($r = 0.774$; object-cluster bootstrap 95% CI $[0.154, 0.340]$). A magnitude-matched derangement control subtracts another object's drift vector. It reaches 40.3px: better than uncorrected WF-r by 13.2px (object-bootstrap 95% CI $[0.0, 27.7]$, exact sign-flip $p = 0.119$), but worse than the correctly matched correction by 5.8px ($[0.8, 12.5]$, $p = 0.057$). Part of the benefit is generic drift rejection, and matching the drift to the item adds the rest.

Preview-free estimators. In the seed-42 comparison, a drift vector fitted on disjoint tune categories reduces CausVid error from 53.5 to 47.4px and CF error from 158.8 to 110.1px. Online estimation instead reaches 47.0px on CausVid and 92.8px on CF, without a preview or tune-half fitting. It recovers 34% of the matched-preview gain on CausVid. At CF's tune-selected gain of 1.5, the online estimator reduces CF error from 158.8 to 92.8px, matching the matched-preview result ($p = 0.0020$, object-clustered; Appendix C.5). This is the most practically relevant result: the online estimator needs neither an additional rollout, extra runtime, nor the per-seed refitting the amortized vector requires.

4.3 Accuracy, quality, and computational cost

We assess Warp Forcing through changes to its own control loop and paired unguided references. In the held-out seed-42 CausVid campaign, WF-r reaches 53.5px path error versus 91.7px unguided, with netCos 0.970 and no items below 0.5. These runs also supply the seed-42 uncorrected baseline for correction.

	trajPathErr ↓		FDA ↑	netCos ↑	netCos<0.5 ↓	Sharp. ↑	DOVER ↑	Runtime
	guided	unguided	guided	guided	guided	guided	full (tech.)	(×WF-r)
SG-I2V transplant, 5 it	65.9 ±21.5	91.7	0.461	0.428 ±.657	40%	224.1	0.406 (0.354)	–
FreeTraj transplant	114.9 ±83.8	91.7	0.388	0.145 ±.723	63%	666.8	0.693 (0.604)	–
DragStream transplant, 5 it	87.2 ±34.0	91.7	0.329	0.238 ±.716	65%	270.0	0.686 (0.678)	–
Warp Forcing (WF-r), CausVid	53.5 ±27.7	91.7	0.648	0.970 ±.057	0%	1072.0	0.682 (0.604)	1.00×
Warp Forcing (WF-r), CF	158.8 ±114.6	203.6	0.631	0.737 ±.507	18%	424.3	0.595 (0.490)	1.00×
QWERTY transplant, CausVid	72.4 ±41.1	91.7	0.449	0.277 ±.692	53%	1196.9	0.689 (0.591)	–

Table 2: Causal-transfer diagnostics, all methods run under the same causal sampler with matched conditioning and seeds (n=40, seed 42, 150px, test10). DOVER gives full (technical) scores. Transplant details in §H.1.

	trajPathErr ↓	netCos ↑	DOVER ↑	Runtime (×WF-r)
FreeTraj / VideoCrafter2	23.0 ±13.1	0.997 ±.005	0.388 (0.304)	9.2×
SG-I2V / SVD	29.0 ±11.4	0.999 ±.001	0.423 (0.359)	11.0×

Table 3: Native bidirectional references on their own backbones, resolutions, and conditioning. DOVER gives full (technical) scores.

Table 2 shows how existing mechanisms behave when moved into a causal sampler: none of the transplants reaches reliable directional control (netCos at most 0.43), whereas WF-r reaches 0.970 on CausVid. Table 3 gives the native bidirectional references; Appendix H gives the transfer details.

Evidence for the control-loop design. Reconstructing causal context from the emitted frames matters in the held-out CausVid campaign: writing the cache from pre-warp predictions instead raises path error from 53.5 to 74.8px and lowers netCos from 0.970 to 0.516. In a separate full-80-item structural campaign, we isolate replanning by comparing WF-r with WF-v, which uses the same video-memory SAM2 tracker without replanning. Replanning improves CausVid path error from 74.4 to 61.5px and netCos from 0.636 to 0.953. On CF, it reduces path error from 207.8 to 181.2px and improves netCos from 0.569 to 0.705. Feedback therefore improves both directional response and path accuracy on both backbones. The open-loop WF-b, which also uses the per-frame image tracker, still has lower path error on CF (166.4px) at a much lower netCos (0.317).

The injection-step sweep on tune10 further shows that the edit timing must balance adherence and quality: the selected steps maximize netCos, while later steps can yield lower path error at lower technical DOVER score (DOVER-t; Table 9). Table 4 records the split and seed of each campaign.

Accuracy and transmission. The distinction also appears within a controller: increasing the raw-latent surrogate's budget from 5 to 25 iterations reduces its transmission slope from 0.73 to 0.18 and path error from 78.9 to 40.6px (Table 20). The latter is lower than uncorrected WF-r's 53.5px. For WF-r itself, the correction gain minimizing error differs from the gain driving transmission toward zero (Appendix C.5).

Quality and cost. Matched-preview correction adds an unguided rollout, increasing measured CausVid runtime from 152.3s to 268.0s on A40 (Table 7); the preview-free estimators avoid that rollout. Correction leaves the CausVid background essentially intact but costs background fidelity on CF: at a 40px foreground-exclusion margin, CausVid background LPIPS changes from 0.114 to 0.118, while CF changes from 0.258 to 0.308 and background PSNR falls from 17.98 to 16.87dB. CF shows worse background-preservation scores at every tested margin (Appendix F).

5 Scope and Limitations

The paired evaluation exposes two complementary aspects of trajectory control: how much error a controller removes and how strongly its output follows variation in matched unguided motion. Their separation across mechanisms explains why a controller can improve one without optimizing the other. For Warp Forcing, explicit drift compensation improves adherence while retaining blockwise generation. This suggests a design direction for streaming controllers: adapt disturbance estimates and correction strength as motion unfolds.

Control and perception limits. The flow field is a rigid translation of a single tracked centroid, so a user can only specify the object's center trajectory; rotation, scale change, and deformation arise only insofar as the denoiser invents them while reconciling the displacement. We leave an affine or per-part field for future work.

TrajectoryBench focuses on image-plane translations. Extending the protocol to 3D motion would require handling depth, occlusion, and perspective changes beyond the current translation field. We leave that to future work.

Warp Forcing inherits its perception stack's failure modes: a mislocalized poke or a lost track produces a flow field aimed at the wrong region, propagating localization error into the control command. Because the flow support comes from the last finalized frame, fast drift between frames can also leave part of the object outside the warped region (Appendix C.1). Each warp must be small enough for the remaining denoising steps to repaint the vacated region, bounding target speed relative to chunk length.

Interpreting prior transmission. Prior transmission is a predictive statistic. Guided and unguided rollouts share the prompt, seed, and initial noise, so the coupling is consistent with an intrinsic generator motion prior but also with shared initial noise, a seed-dependent scene or camera state, category-specific motion susceptibility, an unobserved seed-level variable, a controller that responds differently to different latent states, or tracker effects. Seed-by-direction fixed effects absorb only additive shared shifts. The block-commitment interventions that would localize the source of the coupling have wide intervals (Appendix H.3). Tracker replacement covers the held-out pooled fit and the three-seed fixed-effects subset; on the latter, measurement error from the coarser Lucas-Kanade tracker attenuates the estimated magnitude.

Scope of the evidence. The transmission analysis uses ten CausVid seeds and the matched-preview correction three seeds per backbone; the preview-free estimator comparison, the derangement control, and the control-loop design studies use a single seed each, with the splits and configurations in Table 4. These single-seed results do not establish multi-seed robustness of the preview-free estimators or their equivalence to matched-preview correction.

This is an evaluation-and-correction study; we do not claim that Warp Forcing outperforms existing trajectory controllers. The transplants in Table 2 use released defaults with chunk-local adaptations and no dedicated causal tuning, so they characterize the tested adaptations. Native bidirectional references achieve lower absolute path errors than Warp Forcing (Table 3).

6 Conclusion

TrajectoryBench shows that WF-r's controlled displacement retains roughly half of the matched unguided displacement. Warp Forcing turns this signal into a command correction that reduces path error by 25% on CausVid and 38% on CF. Online estimation obtains the same gain on CF, and one-third of it on CausVid, within a single rollout. These results motivate controllers that estimate and compensate for motion predictable from the unguided rollout throughout generation.

AI use statement

AI assistance was used to implement experiments and analysis code, to critique draft hypotheses and experimental designs, to help interpret results, to search for related work, and to edit prose for readability.

All research decisions were made by the authors. We did not use language-model assistance to fabricate, estimate, or impute any experimental observation or mathematical claim; the generated videos evaluated in the paper were produced by the video models under study. Every quantity in this paper is computed by code we ran over generated video and stored per-item outputs. We reviewed all AI-assisted work: LLM-generated code was read and checked against tests, and at least two authors independently verified the logic behind AI-assisted interpretations before they entered the paper. We take full responsibility for the final content, including all text, claims and artifacts produced with AI assistance.

Ethics statement

This work improves the controllability of pretrained video generators without increasing their underlying visual-generation capacity. More precise trajectory control supports benign applications such as interactive animation and video editing, but it also lowers the effort required to produce a video depicting a specific, targeted event, which can be misused. The method does not address provenance, watermarking, or synthetic-content detection, and deployments should retain the safeguards and licensing restrictions of the underlying generators it is applied to.

Reproducibility statement

TrajectoryBench's design is specified in Appendix A: the 20 categories, the 4 cardinal directions and displacement magnitude, the prompt template, the tune/test partition, and the tracking and metric definitions. Because every item is generated from a template prompt and a fixed seed, the benchmark requires no external image dataset. Exact reproduction additionally needs the seeds, detector and tracker versions and thresholds, and the estimation procedure for the amortized constant; these are in the released code/ appendix. Implementation details for the controller (injection step, flow construction, feathering, tracker configuration) are in Appendix C. We release the benchmark generation scripts, generation pipelines for both backbones and the metric implementations, as an anonymized repository linked in the supplementary material.

References

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. Stable video diffusion: Scaling latent video diffusion models to large datasets. *CoRR*, abs/2311.15127, 2023. URL https://doi.org/10.48550/arXiv.2311.15127.

Ryan Burgert, Yuancheng Xu, Wenqi Xian, Oliver Pilarski, Pascal Clausen, Mingming He, Li Ma, Yitong Deng, Lingxiao Li, Mohsen Mousavi, Michael Ryoo, Paul Debevec, and Ning Yu. Go-with-the-flow: Motion-controllable video diffusion models using real-time warped noise. In *CVPR*, 2025.

Pascal Chang, Jingwei Tang, Markus Gross, and Vinicius C. Azevedo. How i warped your noise: a temporally-correlated noise prior for diffusion models. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/abs/2504.03072. Oral.

Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=yDo1ynArjj.

Changgu Chen, Junwei Shu, Gaoqi He, Changbo Wang, and Yang Li. Motion-zero: A zero-shot trajectory control framework of moving object for diffusion-based video generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 2016–2024, 2025. doi: 10.1609/aaai.v39i2.32198.

Kyobin Choo, Youngmin Kim, Hyunkyung Han, Geunrip Park, Chanyoung Kim, Sunyoung Jung, and Seong Jae Hwang. Qwerty: Training-free motion control via query-warped video diffusion transformers, 2026. URL https://arxiv.org/abs/2607.01869.

Daniel Geng and Andrew Owens. Motion guidance: Diffusion-based image editing with differentiable motion estimators. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 17451–17472, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/4b1d9a1fbf7b2a93bea08e18792fe436-Paper-Conference.pdf.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=mSiN7i0BYH.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Yash Jain, Anshul Nasery, Vibhav Vineet, and Harkirat Behl. Peekaboo: Interactive video generation via masked-diffusion, 2024. URL https://arxiv.org/abs/2312.07509.

Nikita Karaev, Iurii Makarov, Jianyuan Wang, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. CoTracker3: Simpler and better point tracking by pseudo-labelling real videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

Pengyang Ling, Jiazi Bu, Pan Zhang, Xiaoyi Dong, Yuhang Zang, Tong Wu, Huaian Chen, Jiaqi Wang, and Yi Jin. Motionclone: Training-free motion cloning for controllable video generation. In *International Conference on Learning Representations (ICLR)*, 2025.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023.

Wan-Duo Kurt Ma, J. P. Lewis, and W. Bastiaan Kleijn. Trailblazer: Trajectory control for diffusion-based video generation. In *SIGGRAPH Asia Conference Papers*, 2024.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022.

Koichi Namekata, Sherwin Bahmani, Ziyi Wu, Yash Kant, Igor Gilitschenski, and David B. Lindell. Sg-i2v: Self-guided trajectory control in image-to-video generation, 2025. URL https://arxiv.org/abs/2411.04989.

Muyao Niu, Xiaodong Cun, Xintao Wang, Yong Zhang, Ying Shan, and Yinqiang Zheng. Mofa-video: Controllable image animation via generative motion field adaptions in frozen image-to-video diffusion model. *arXiv preprint arXiv:2405.20222*, 2024.

Haonan Qiu, Zhaoxi Chen, Zhouxia Wang, Yingqing He, Menghan Xia, and Ziwei Liu. Freetraj: Tuning-free trajectory control via noise guided video diffusion. *Int. J. Comput. Vision*, 134(4), March 2026. ISSN 0920-5691. doi: 10.1007/s11263-026-02732-3. URL https://doi.org/10.1007/s11263-026-02732-3.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. URL https://arxiv.org/abs/2408.00714.

Yujun Shi, Chuhui Xue, Jiachun Pan, Wenqing Zhang, Vincent YF Tan, and Song Bai. Dragdiffusion: Harnessing diffusion models for interactive point-based image editing. *arXiv preprint arXiv:2306.14435*, 2023.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Zhouxia Wang, Ziyang Yuan, Xintao Wang, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. Motionctrl: A unified and flexible motion controller for video generation, 2024. URL https://arxiv.org/abs/2312.03641.

Haoning Wu, Erli Zhang, Liang Liao, Chaofeng Chen, Jingwen Hou, Annan Wang, Wenxiu Sun, Qiong Yan, and Weisi Lin. Exploring video quality assessment on user generated contents from aesthetic and technical perspectives. In *International Conference on Computer Vision (ICCV)*, 2023.

Weijia Wu, Zhuang Li, Yuchao Gu, Rui Zhao, Yefei He, David Junhao Zhang, Mike Zheng Shou, Yan Li, Tingting Gao, and Di Zhang. Draganything: Motion control for anything using entity representation. In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29 – October 4, 2024, Proceedings, Part XXII*, pp. 331–348, Berlin, Heidelberg, 2024. Springer-Verlag. ISBN 978-3-031-72669-9.

Shengming Yin, Chenfei Wu, Jian Liang, Jie Shi, Houqiang Li, Gong Ming, and Nan Duan. Dragnuwa: Fine-grained control in video generation by integrating text, image, and trajectory, 2023. URL https://arxiv.org/abs/2308.08089.

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, 2025. URL https://arxiv.org/abs/2412.07772.

Zhenghao Zhang, Junchao Liao, Menghao Li, Zuozhuo Dai, Bingxue Qiu, Siyu Zhu, Long Qin, and Weizhi Wang. Tora: Trajectory-oriented diffusion transformer for video generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 2063–2073, 2025.

Junbao Zhou, Yuan Zhou, Kesen Zhao, Qingshan Xu, Beier Zhu, Richang Hong, and Hanwang Zhang. Streaming drag-oriented interactive video manipulation: Drag anything, anytime!, 2026. URL https://arxiv.org/abs/2510.03550.

Hongzhou Zhu, Min Zhao, Guande He, Hang Su, Chongxuan Li, and Jun Zhu. Causal forcing: Autoregressive diffusion distillation done right for high-quality real-time interactive video generation. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=BYInOck3gr.

A TrajectoryBench details

A.1 Category and direction grid

TrajectoryBench pairs 20 object categories with 4 cardinal poke directions each, for 80 items total, generated once per backbone under evaluation:

- Categories (20): car, dog, person, bird, boat, train, bicycle, horse, ball, airplane, bus, motorcycle, cat, truck, robot, fish, leaf, cloud, skateboard, kite
- Diagonal/curved-target subset (10 categories): airplane, ball, bicycle, bird, boat, car, dog, horse, person, train
- Directions (4): right $(+150, 0)$, down $(0, +150)$, up $(0, -150)$, left $(-150, 0)$
- Magnitude: total displacement in pixels (default: 150px)

The main sweep is exactly these 4 cardinal directions, giving $20 \times 4 = 80$ items. Diagonal and quarter-circle targets are a separate extension built on a 10-category subset and reported in Appendix G.

The list mixes rigid, high-inertia movers (car, bus, train, truck, motorcycle, bicycle, airplane, boat), deformable/articulated movers (dog, cat, horse, bird, fish, person), and small or unusually-shaped objects (ball, robot, leaf, cloud, kite, skateboard). We selected categories for coverage across different object types.

Breakdown by object type. The category list was chosen to span rigid, articulated and small or irregular movers, so we report performance by that grouping on the held-out half (12/12/16 items).

	base	WF-r	+FF	netCos (+FF)
CausVid				
rigid (motorcycle, train, truck)	97.9	68.4	35.3	0.974
articulated (fish, horse, person)	84.8	44.2	33.4	0.986
small (kite, leaf, robot, skateboard)	92.3	49.3	34.7	0.982
CF				
rigid	280.7	242.7	162.2	0.650
articulated	179.2	105.2	55.5	0.828
small	164.2	136.1	64.2	0.904

Rigid, high-inertia movers are the hardest group on both backbones and also have the largest unguided drift, creating a larger disturbance to reject. Feedforward rejection nearly removes the between-group spread on CausVid ($68.4/44.2/49.3 \rightarrow 35.3/33.4/34.7$). On CF, the correction helps every group but does not equalize them; rigid movers remain far above the stationary-object error.

Tune/test split and campaign provenance. The 20 categories form two disjoint 10-category halves (40 items each). Hyperparameter selection uses tune10 (airplane, ball, bicycle, bird, boat, bus, car, cat, cloud, dog); the main baseline comparison uses test10 (fish, horse, kite, leaf, motorcycle, person, robot, skateboard, train, truck). Table 4 records the split and seed for the baseline, cross-controller dependence, timestep, and structural-ablation campaigns. Comparisons within each campaign use its corresponding generation and evaluation configuration.

reported in	backbone	split	seed	items	trajPathErr
Tables 2, 20 (WF-r)	CausVid	test	42	40	53.5
Table 9 (injection sweep, $k^\star$=0)	CausVid	tune	42	40	61.9
Table 12 (structural ablation)	CausVid	full	42	80	61.5
Tables 2, 20 (WF-r)	CF	test	42	40	158.8
Table 9 (injection sweep, $k^\star$=2)	CF	tune	42	40	157.5
Table 12 (structural ablation)	CF	full	42	80	181.2

Table 4: WF-r configurations across experimental campaigns. The other causal-controller rows in Table 20 also use test10, seed 42, and 40 items per configuration. SG-I2V/SVD uses CF-generated initial images. Injection-sweep unguided errors are 79.8px on CausVid and 177.1px on CF.

backbone	seed	WF-r	matched preview
CausVid	42	53.5	34.5
	7	49.3	41.7
	123	39.5	30.9
	pooled	47.4	35.7
CF	42	158.8	91.0
	7	185.3	109.3
	123	105.8	79.0
	pooled	150.0	93.1

Table 5: Correction campaign accounting: mean path error on the same test10 categories, four directions, and 150px requests, with $g = 1$ on CausVid and $g = 1.5$ on CF. Each seed contributes 40 paired items. The seed-42 rows also supply the gain sweep; pooled values average all 120 items.

The 20-object slope-analysis set. TrajectoryBench is the 80-item, 20-category benchmark specified above. The slope estimators use a different set, which we call the 20-*object slope-analysis set*. Because the slope-on-prior estimates otherwise rest on only the held-out half's 10 objects, we generated 10 additional categories on CausVid, extending the held-out half to 20 objects for this

experiment	rows	unguided rollouts	dependence units
Held-out accuracy (Tables 2, 19)	40 items	10	10 objects, 1 seed
Primary slope estimate (§4.1)	800 items	200	20 objects, 10 seeds
Feedforward, pooled (§4.2)	120 items	30	10 objects, 3 seeds
Structural ablation (Appendix D.1)	80 items	20	20 objects, 1 seed

Table 6: Effective sample sizes. "Rows" counts scored observations, where the four directions of an object share one unguided rollout, so the dependence units govern every interval. The 20-object slope-analysis set behind the primary estimate is an expanded set with 10 additional objects.

estimator: helicopter, tractor, scooter (rigid), elephant, penguin, squirrel, kangaroo (articulated), umbrella, drone, and surfboard (small/irregular). We first generated this set at the 3 seeds used elsewhere (42/7/123), then added 7 seeds (17, 99, 314, 555, 777, 2718, 31415) under the same protocol, for 10 seeds total.

Prompt template Each category o is expanded into a single fixed prompt, "A o is moving." (e.g. "A robot is moving."), with no per-item prompt engineering. The prompt is intentionally underspecified, so the base clip in every guided/base pair shows whatever motion the backbone produces for that category and seed without control. The object type is known both during generation and evaluation.

A.2 Tracking and metric computation

For every clip we detect the poked object on the shared first frame with GroundingDINO (text-prompted by the category name) and SAM2, seed a 20×20 grid of query points inside the mask, and track all points jointly with CoTracker3 (offline mode) (Karaev et al., 2025). The same first-frame mask and query grid are shared by the guided and unguided clip in every pair, and by every method compared against a given pair, so no method sees a more favorable initial mask than another. Tracking runs on frames downscaled so the longer side is 512px, to bound GPU memory on long clips; tracks are rescaled to native resolution before any error is computed, so all reported pixel quantities are native-resolution regardless of tracking resolution. The per-frame centroid $c(t)$ of CoTracker3 is the mean of all tracked points, which keeps the trajectory from jittering as points enter or leave visibility.

Warp Forcing uses SAM2 for flow construction and online closed-loop tracking, while CoTracker3 supplies evaluation trajectories.

For continuous FDA, let $u = \Delta/\|\Delta\|_2$ and $v_t = c(t+1) - c(t)$. We report

$$\mathrm{FDA} = \frac{1}{T-1} \sum_{t=0}^{T-2} \frac{1 + \langle v_t, u \rangle / \|v_t\|_2}{2}.$$

netCos is the unshifted cosine between $c(T-1) - c(0)$ and Δ. For DTW, both the tracked cumulative displacement and the time-indexed piecewise-linear target path are sampled at T points. We use unconstrained dynamic time warping with Euclidean local cost and horizontal, vertical, and diagonal transitions, divided by the selected warping-path length; the result is in native pixels. Thus trajPathErr evaluates geometry and timing against the intended curve, whereas DTW permits local timing changes against the same target path.

Sharpness is the mean over frames of grayscale Laplacian variance. DOVER and its technical component are computed per generated clip with the public DOVER implementation and then averaged over items; no guided/base normalization is applied. For outside-mask preservation, foreground is independently detected and SAM2-video-tracked in guided base clips. We exclude the per-frame union of their foreground masks, dilated with square radii of 20, 40, 80, and 120 native pixels, and sample every eighth frame. PSNR is computed on retained pixels. LPIPS features are computed on unmasked images; at each feature resolution, a bin is retained only if it contains no excluded image pixels. Retained bins can still have receptive fields that overlap excluded pixels (Appendix F). Learned channel-weighted squared feature differences are normalized by retained weight separately at each layer and summed. We average valid sampled frames within each item and then average

Method	Runtime (s) ↓ base	guided	Overhead +s	×
Warp Forcing (CausVid, WF-r)	**115.7**	**152.3**	**36.5**	**1.32**
Warp Forcing (CF, WF-r)	119.3	159.5	40.2	1.34
+ matched-preview feedforward (CausVid)	**115.7**	268.0	152.3	2.32
FreeTraj	491.1	1401.6	910.6	2.85
SG-I2V	600.3	1676.5	1076.1	2.79

Table 7: End-to-end runtime per 81-frame clip (5.0s from first to last frame at 16fps; 5.0625s encoded duration; 1 A40 GPU). For the matched-preview feedforward row, "guided" is total wall time including its own required unguided preview pass.

component	denoise	warp	track+decode	cache restore	final decode	other
WF-r / CausVid, s	2.31	0.84	9.01	0.78	2.34	3.50
share of 18.78s total	12%	4%	48%	4%	12%	19%

Table 8: WF-r guided-pass component breakdown (81 frames). Every mode runs one unmeasured warm-up item and then three measured clips; timing only includes sampling. Tracking and decoding account for the largest share of this sampling profile.

items. Empty-mask frames are excluded; retained fractions and margin sensitivity are reported in Table 16.

FDA's per-frame angle θ_t is ill-defined when a frame's centroid increment is near zero. We resolve that by assigning a frame whose flow magnitude falls at or below 10^{-6}px a $\cos\theta_t = 0$. Under the continuous score this contributes the neutral value 0.5 to that frame's term; under the binary directional-agreement criterion ($\cos\theta_t > 0.5$) it counts as non-agreement, since a near-stationary frame did not move in the requested direction. This convention is applied uniformly across all methods and both backbones, which means FDA on clips with many near-stationary frames is biased toward 0.5. netCos uses the same 10^{-6}px floor on the net (whole-clip) displacement magnitude, assigning 0 in the rare case a clip's start and end centroid coincide to within that tolerance.

B Pairing diagnostic

The zero-flow sham runs the complete control stack of WF-r with an identically zero flow field and gives slope 1.00 with r=1.0 for the measured displacement statistic over 40 held-out items. This checks that the control stack preserves the measured displacement when no edit is applied. The paired generation code resets the seed and reconstructs the initial noise before each pass.

C Implementation details

Both backbones generate at 480×832px with a Wan2.1 VAE of spatial downsampling factor 8, giving a 60×104 latent grid; temporal downsampling is $4\times$ for both. CausVid denoises each chunk in K=3 steps, $\{1000, 757, 522\}$ at indices $k \in \{0, 1, 2\}$; CF uses K=4 steps, $\{1000, 750, 500, 250\}$ at indices $k \in \{0, 1, 2, 3\}$. The block size is F=3 latent frames on CausVid and F=1 on CF, so CF commits every latent frame while CausVid commits three at a time. All results use 81 total pixel frames, i.e. T_{lat}=21 latent frames.

The injection step is $k^\star$=0 (t=1000) on CausVid and $k^\star$=2 on CF selected per backbone. Block 0 is never warped (no prior chunk to correct against). The dilation margin is 5px (square structuring element) and the feather radius r_f defaults to 5px on both backbones, both in native pixels and below the 8px latent cell size (Appendix C.1).

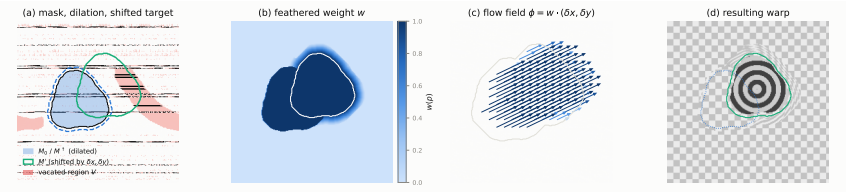


Figure 4: From poke to warped prediction. Given a mask M_0 (left) and a per-frame displacement $d_f = (\delta x, \delta y)$, M_0 is dilated (square structuring element) to a source region and translated by d_f to a destination footprint; the field ϕ (center and right; color encodes direction, opacity encodes magnitude) holds full displacement through the vacated source region and the destination's interior, and is feathered down only near the destination's inner boundary.

C.1 Exact flow-field construction

Let $M_0 \subset \mathbb{Z}^2$ be the binary object mask on the frame a given per-frame displacement $d_f \in \mathbb{R}^2$ (§3.3) is planned from, m the dilation margin, and r_f the feather radius. Define the source support $S = M_0 \oplus B_m$ (M_0 dilated by the square structuring element B_m of side $2m{+}1$, i.e. a Chebyshev ball of radius m), the destination footprint $D = S + d_f$ (S translated by d_f), and the vacated region $V = S \setminus D$. The backward flow field used in $\tilde{x}_0(p) = \hat{x}_0(p - \phi(p))$ is

$$\phi(p) = \begin{cases} d_f, & p \in V \\ w(p)\, d_f, & p \in D \\ 0, & \text{otherwise,} \end{cases} \qquad w(p) = \min\Big(1,\ \tfrac{\operatorname{dist}(p, \partial D)}{r_f}\Big)$$

where $\operatorname{dist}(p, \partial D)$ is the (unsigned) distance from p to the boundary of D. This holds ϕ at the full displacement d_f throughout the vacated region V and the interior of D beyond r_f pixels from its edge, and ramps it linearly to 0 over the feather band inside D, so the field joins the identity flow continuously along portions of ∂D adjacent to the region outside $S \cup D$. It is not continuous everywhere: V itself is not feathered, so ϕ steps from d_f to 0 across ∂S (the outer edge of the vacated region). There is also a discontinuity at the internal interface between V and D, where the full displacement in V meets a destination-side weight tending to zero. Feathering only the destination footprint is deliberate, as the destination is where new content is being pulled in, while the vacated region is left for the subsequent denoising steps to repaint outright. The field is constructed at native pixel resolution and then bilinearly resized to the latent grid without anti-aliasing, so each latent cell takes the average of the two native pixels nearest its center. With $m{=}r_f{=}5$px, both the dilation margin and the feather band are narrower than one 8px latent cell. The feather therefore survives only as intermediate values in at most one latent row or column along the destination edge, and at latent resolution ϕ is close to a hard-edged translation of the dilated mask. In our qualitative observations, the resulting discontinuities did not produce visible object tearing or reduce object quality. The most plausible mitigating factor is subsequent denoising, which can reconcile local inconsistencies and repaint the vacated region; the sub-cell dilation margin places the support boundary only marginally outside the tracked mask.

The support S is taken from the last finalized frame, not from the frame being warped. $\hat{x}_0^{(k^\star)}$ for the current block is generated after that frame, so its object has already moved by any unguided drift since then. On CausVid, one mask also serves all $F = 3$ frames of a block, so the lag grows with f. On CF, the unguided rollout moves $+360$px over 20 latent-frame intervals on average (Appendix I), about 18px per latent frame, well beyond the 5px margin m. The leading part of the object can therefore fall outside S and is not displaced by ϕ, which may contribute to CF's weaker adherence. We did not evaluate predicting the support forward with the estimated drift.

C.2 Controller pseudocode

For an arbitrary trajectory, the user supplies time-indexed cumulative-displacement waypoints $w_j = (\tau_j, \Delta_j)$ with $\tau_j \in (0, 1]$. We prepend $w_0 = (0, 0)$ and linearly interpolate between adjacent waypoints to define the cumulative-displacement path $q(\tau)$. At block b, the controller selects the segment containing the elapsed fraction, targets that segment's endpoint $c_0 + \Delta_j$, and divides

the residual by the latent frames remaining until τ_j; a straight request uses a single segment. The quarter-circle experiments use a piecewise-linear sampling of the arc, providing intermediate targets throughout the trajectory.

For each block b of a guided rollout:

1. Read the current tracked centroid c_b from the online SAM2 tracker on the last finalized frame (block 0: skip warping entirely, run the backbone unmodified).
2. Compute $\text{rate}_b = (c_{\text{tot}} - c_b)/B$ (§3.3); if feedforward rejection is enabled, subtract $g\,\delta_{\text{prior}}$ (§4.2).
3. For each latent frame $f \in \{0, \ldots, F-1\}$ in the block, compute $d_f = (f+1) \cdot \text{rate}_b$ and build ϕ_f (Appendix C.1).
4. Denoise the block's noisy latent with the backbone's own sampler up to step $k^\star$, producing $\hat{x}_0^{(k^\star)}$.
5. Bilinearly resize ϕ_f to the latent grid and divide both displacement components by 8, giving ϕ_f^{lat}. Warp: $\tilde{x}_0^{(k^\star)}(p) = \hat{x}_0^{(k^\star)}(p - \phi_f^{\text{lat}}(p))$ for each frame f in the block.
6. Re-noise $\tilde{x}_0^{(k^\star)}$ to the scheduler's nominal level for step $k^\star{+}1$ and continue the unmodified sampler to $k{=}K$.
7. Finalize the block: restore the KV cache to its pre-block snapshot and re-run the generator once at $t{=}0$ on the warped, finalized frames, writing cache entries from the corrected content.
8. Update the online tracker on the newly finalized frame and proceed to block $b{+}1$.

C.3 Injection timestep

$k^\star$ (t)	trajPathErr ↓	netCos ↑	FDA ↑	DOVER-t ↑	Sharp. ↑
CausVid (unguided: trajPathErr 79.8)					
0 (1000)	61.9	**0.964**	**0.765**	**0.561**	**1082.9**
1 (757)	58.3	0.957	0.756	0.556	1052.9
2 (522)	**54.3**	0.953	0.754	0.482	930.8
CF (unguided: trajPathErr 177.1)					
0 (1000)	167.3	0.587	0.711	**0.496**	634.6
1 (750)	151.8	0.854	0.754	0.490	610.5
2 (500)	157.5	**0.920**	**0.786**	0.482	570.3
3 (250)	**134.4**	0.849	0.736	0.415	**661.6**

Table 9: Injection-step sweep (WF-r, 150px, tune half). Bold rows mark the default per backbone; unguided values come from the same paired runs and are constant across $k^\star$. CausVid's three-step schedule has no $k^\star{=}3$.

We select the setting with the highest netCos: $k^\star{=}0$ on CausVid, which is also best on DOVER-t and sharpness, and $k^\star{=}2$ on CF, which stays within 0.015 DOVER-t of the sweep's best. The different orderings across backbones suggest that $k^\star$ must be set per backbone and may depend on the schedules' different terminal timesteps.

On CausVid, warping at $k^\star = 0$ gives the best qualitative results, although its quantitative scores are similar to those at $k^\star \in \{1, 2\}$. On CF, $k^\star{=}2$ has the highest netCos on the tune half.

C.4 Displacement strength

Increasing requested displacement improves directional control up to 200px on CausVid, after which it saturates or slightly declines while sharpness falls monotonically; CF improves more gradually across the whole range.

s	trajPathErr ↓	FDA ↑	netCos ↑	Sharp. ↑	DOVER ↑
Warp Forcing, CausVid (WF-r)					
50	50	0.644	0.697	1161.7	0.672
100	55	0.723	0.873	1105.0	0.674
200	67	0.777	0.967	976.1	0.673
300	82	0.767	0.949	855.6	0.669
Warp Forcing, CF (WF-r)					
50	165	0.561	0.548	491.6	0.615
100	170	0.607	0.679	489.6	0.613
200	174	0.665	0.721	468.4	0.609
300	173	0.676	0.767	454.7	0.601
SG-I2V					
50	27	0.634	0.963	226.0	0.395
100	42	0.827	0.998	238.8	0.418
200	31	0.943	0.999	237.1	0.430
300	34	0.976	0.999	231.4	0.444

Table 10: Displacement strength sweep, tune half. Sharpness and DOVER measure perceptual quality independently of the trajectory.

gain	trajPathErr ↓	mean error reduction	netCos ↑	slope on prior	% worse than stationary
CausVid					
0.0 (WF-r)	53.5	—	0.970 ±.06	+0.67 [.52, .92]	15%
0.5	42.4	+11.0 [+4.3, +18.6], p=0.0059	0.980 ±.04	+0.51 [.35, .73]	8%
1.0	34.5	+19.0 [+8.4, +31.7], p=0.0020	**0.981** ±.04	+0.32 [.13, .59]	**2%**
1.5	**34.1**	**+19.4** [+9.0, +32.1], p=0.0020	0.977 ±.04	+0.16 [.04, .43]	**2%**
2.0	37.7	+15.8 [+7.3, +25.6], p=0.0039	0.976 ±.04	+0.04 [-.06, .31]	5%
CF					
0.0 (WF-r)	158.8	—	0.737 ±.50	+0.56 [.14, 1.01]	75%
0.5	113.1	+45.7 [+17.6, +77.2], p=0.0059	0.786 ±.44	+0.36 [.08, .72]	52%
1.0	100.7	+58.1 [+29.5, +89.8], p=0.0020	0.790 ±.46	+0.30 [.06, .60]	35%
1.5	**91.0**	**+67.8** [+38.4, +101.9], p=0.0020	**0.805** ±.47	+0.23 [-.05, .61]	**30%**
2.0	107.8	+51.0 [+22.3, +83.2], p=0.0098	0.802 ±.45	+0.40 [.00, .90]	40%
2.5	116.8	+42.0 [+7.2, +77.1], p=0.0527	0.657 ±.64	+0.26 [.00, .45]	55%

Table 11: Feedforward gain sweep (n=40, one seed). Reductions, CIs, and p values compare with $g = 0$ and cluster by object. CausVid $g = 1/1.5$ tie ($p = .77$); CF favors $g = 1.5$.

C.5 Feedforward gain

At gains $0, 0.5, 1.0, 1.5, 2.0$ for generated matched previews (40 held-out items per gain, seed-42), trajPathErr is 53.5, 42.4, 34.5, 34.1, and 37.7px, respectively; netCos is 0.970, 0.980, 0.981, 0.977, and 0.976. The previously selected $g = 1$ reduces error by 19.0px (35.5%), with $g = 1.5$ numerically close.

Online correction on CF. On 40 held-out items at seed 42, with $\lambda = 0.5$ and $g = 1.5$, online correction reduces trajPathErr from 158.8 to 92.8px (41.6%) and raises netCos from 0.737 to 0.807. The paired reduction is 65.9px (95% object-bootstrap CI $[32.52, 105.79]$; exact two-sided object sign-flip $p = 0.0020$). We average each object's four direction-level differences before resampling 10 objects (20,000 draws, analysis seed 42). All ten object means and $35/40$ individual items improve; $14/40$ items (35%) remain worse than stationary, and $4/40$ have netCos below 0.5. Reference base videos match byte-for-byte. The mean is close to the reported matched-preview result (91.0px). The online estimator uses the same gain as CF's matched-preview correction (1.5).

Table 11 and Figure 5 report the full gain curve on the held-out half. We select gains using only the tune half: CausVid trajPathErr is 49.8, 40.6, 43.3, and 59.8px at g=0.5, 1.0, 1.5, 2.0, respectively, and is minimized at g=1.0; CF is minimized at g=1.5. CF's higher selected gain is consistent with its larger unguided displacement.

Figure 5 plots path error and transmission against gain. CausVid path error reaches its minimum near $g = 1.5$, while transmission approaches zero near $g = 2.0$. Gain selection uses trajectory accuracy on the tune half.

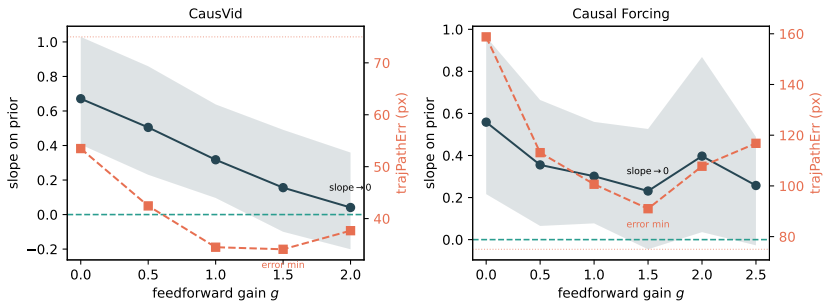


Figure 5: Feedforward gain against path error (right axis) and the slope on the prior (left axis), with bootstrap CIs on the slope and the stationary-object error marked. The error minimum and the slope-zero crossing do not coincide.

The error minimum and the slope-zero crossing do not coincide: path error is minimized near $g{\approx}1.5$, while transmission reaches zero only near $g{\approx}2.0$. We select the gain empirically on the tune half to account for the controller's response.

How to avoid the feedforward preview. The disturbance is persistent: switching the warp off partway through a guided rollout produces drift agreeing with the never-warped base pass over the same frames (cosine 0.56 on CausVid, 0.80 on CF). A single per-backbone mean drift vector already accounts for 33%/70% of the matched-preview correction vector on CausVid/CF. Regenerating every held-out clip with that tune-half-fitted constant, at each backbone's own reported gain, gives 53.5 $\rightarrow$ 47.4px on CausVid and 158.8 $\rightarrow$ 110.1px on CF, i.e. 32% and 72% of the matched-preview improvement (19.0px and 67.8px respectively). A purely online estimator (a causal, per-block exponential moving average of each block's innovation) matches the amortized constant on CausVid (47.0px vs. 47.4px) with no preview and no tune-half fitting, at the cost of a start-up delay before its estimate reflects measured drift (see below). Under the protocol's primary inference unit both are clearly weaker than the matched-preview correction: object-clustered mean reductions of $+6.1$px (95% CI $[+1.3, +10.6]$, exact object sign-flip $p{=}0.041$) for the amortized constant and $+6.5$px ($[+0.4, +12.8]$, $p{=}0.088$) for the online estimator on CausVid, and $+48.7$px ($[+6.5, +93.6]$, $p{=}0.070$) for the amortized constant on CF, against $+19.0$px ($p{=}0.0020$) and $+67.8$px ($p{=}0.0020$) for the corresponding matched-preview rows.

Let

$$\text{innov}_b = \frac{c_{b+1} - c_b}{F} - \text{rate}_b$$

be block b's innovation: realized minus commanded per-latent-frame displacement, in the same units as δ_{prior}, read causally from the tracker after block b finalizes. The estimate updates as

$$\widehat{\delta}_b = \lambda\, \widehat{\delta}_{b-1} + (1-\lambda)\, \text{innov}_b, \qquad \text{rate}_{b+1} = \frac{c_{\text{tot}} - c_{b+1}}{B_{b+1}} - g\, \widehat{\delta}_b,$$

a 2D vector applied componentwise with $\lambda = 0.5$. Block 0 is never warped, so it is issued uncorrected with $\text{rate}_0 = 0$, and the estimate is initialized as $\widehat{\delta}_0 \equiv \text{innov}_0$. Here c_0 is the centroid on latent frame 0, where the object is detected, and c_1 is the centroid on the last frame of block 0. Block 0 therefore spans only $F-1$ latent-frame intervals, while innov_0 is normalized by F like every later block. On CausVid ($F = 3$), the initialization thus underestimates block 0's per-frame drift by a factor of $2/3$. On CF ($F = 1$), c_1 and c_0 are measured on the same frame, so $\text{innov}_0 \approx 0$ and the estimator effectively starts from zero. Its first estimate that reflects measured drift is $\widehat{\delta}_1$, which weights block 1's innovation by $1-\lambda$ and first corrects block 2. Online evaluations use $g = 1.0$ on CausVid and $g = 1.5$ on CF.

Item-level intervals for the gain curve. Table 11 reports object-clustered reductions. For continuity with prior work's convention we also give the item-level ($n{=}40$) trajPathErr bootstrap intervals for the same rows: CausVid 53.5 $[45.5, 62.4]$, 42.4 $[36.5, 48.8]$, 34.5 $[29.7, 39.6]$, 34.1 $[29.7, 38.7]$, 37.7 $[32.7, 43.0]$ at $g{=}0, 0.5, 1, 1.5, 2$; CF 158.8 $[126.0, 195.0]$, 113.1 $[84.2, 148.3]$, 100.7

[71.9, 134.6], 91.0 [61.4, 125.7], 107.8 [70.4, 157.2], 116.8 [82.8, 162.3] at g=0, 0.5, 1, 1.5, 2, 2.5. All nonzero gains improve trajPathErr over g=0 in item-level paired tests ($p < 10^{-5}$). This amortized constant is a per-backbone average: applying it unchanged to the curved benchmark or to an unseen seed does only partially transfer (trajPathErr 64.7 → 48.6px on the curved set and 49.1 → 48.9px on an unseen seed, neither significant), consistent with the disturbance tracking the specific noise realization (Appendix H); refitting fresh from the new seed's own tune half recovers most of the benefit, so its practical value depends on a modest per-seed tune set being available.

D Structural and design ablations

D.1 Structural variants

	trajPathErr ↓	FDA ↑	netCos ↑	Sharp. ↑	DOVER ↑
CausVid					
WF-b	77.9	0.699	0.657	1110.6	0.674
WF-v	74.4	0.702	0.636	**1142.8**	0.674
WF-r	**61.5**	**0.762**	**0.953**	1043.6	**0.677**
CF					
WF-b	**166.4**	0.620	0.317	**491.0**	0.609
WF-v	207.8	**0.673**	0.569	485.2	**0.610**
WF-r	181.2	0.647	**0.705**	476.5	0.607

Table 12: Structural ablation, 150px, on the full 80-item benchmark. Campaign configurations and splits are recorded in Table 4.

Three structural variants compare open-loop displacement, tracker memory, and closed-loop replanning. **WF-b** applies a single fixed displacement vector uniformly over the feathered mask (open loop). **WF-v** swaps the per-frame SAM2 image predictor for the video predictor with a persistent memory bank, for robustness through occlusion and deformation. **WF-r** keeps WF-v's video-memory tracker and adds closed-loop, shrinking-horizon replanning as rate computation, rate = (target − pos)/frames remaining, spreading whatever error exists over the frames left.

D.2 Candidate regeneration and disturbance decomposition

A causal backbone can regenerate a block that misses its target without regenerating the entire clip. Drawing up to 4 candidates and keeping the closest reduces error from 53.5 to 47.9px with 3.36 attempts per block and moves the slope only from 0.67 to 0.53. Per-block telemetry also shows candidates missing in the same direction: a block commanded to move 42.4px realizes 106.4, 106.9, and 101.4px across three draws. Thus, in this example, resampling does not remove the disturbance associated with the prompt, seed, and timestep. Decomposing the disturbance into components along and orthogonal to the request shows that the warp halves, but does not eliminate, orthogonal drift (59.1 → 29.7px on CausVid, 160.8 → 72.0px on CF). netCos does not capture this component, partly explaining why it saturates near 0.97 while substantial path error remains (53.5px on CausVid and 158.8px on CF).

D.3 Shrinking- vs. fixed-horizon replanning

The shrinking-horizon rule divides outstanding error by every frame left, so its denominator collapses toward one block as the clip ends and residual drift is forced into an increasingly aggressive late correction. A fixed-window receding-horizon controller instead replans over a constant W-block window that slides forward, tapering only in the last W blocks.

The fixed window reduces end-of-clip acceleration from 1.36× to 1.11× (p=8.2×10^{-5}), with similar trajectory metrics across the 20 paired items.

	Shrinking	Fixed-window	paired p
End-of-clip behavior			
tail/rest step ratio ↓	1.364	1.110	8.2×10^{-5}
tail step mag. (px/frame)	5.76	5.25	—
Trajectory fidelity			
trajPathErr ↓	61.9	64.3	0.52
netCos ↑	0.964	0.969	0.71
FDA ↑	0.765	0.670	0.32

Table 13: Shrinking-horizon (WF-r default) vs. a fixed-window receding-horizon replanner (W=3 blocks), CausVid, 150px, n=20 paired items. "tail/rest step ratio" is mean per-frame centroid step over the final 20% of the clip divided by that over the rest; 1.0 means no end-of-clip acceleration.

E HORIZON AND STREAMING BEHAVIOR

E.1 HORIZON SWEEP

	1×	2×	4×	8×
CausVid (WF-r)				
trajPathErr ↓	52.6	82.3	146.7	312.6
Unguided trajPathErr ↓	63.6	110.5	192.1	382.0
netCos ↑	0.852	0.972	0.786	0.909
DOVER-t ↑	0.563	0.561	0.534	0.482
CF (WF-r)				
trajPathErr ↓	142.5	319.1	328.8	394.6
DOVER-t ↑	0.494	0.414	0.272	0.217
SG-I2V, chained (partial)				
trajPathErr ↓	23.1	50.5	-	-
DOVER-t ↑	0.365	0.224	-	-

Table 14: Horizon sweep at 1/2/4/8× the main sweep's length (21/42/84/168 latent frames; 81/165/333/669 decoded frames), 10-object subset, 2 directions ($n = 20$ per tier for CF and CausVid). DOVER-t averages guided clips only. Requested displacements are 100/200/400/800px, respectively, approximately preserving speed. The stationary-object path-error references are 50/100/200/400px. SG-I2V's chained generation did not finish a 20-item pass at any tier within 12h wall-clock.

Both backbones complete all four tiers. CF's path error grows by $2.77\times$ for an $8\times$ increase in requested displacement, while DOVER-t decreases; the first doubling nevertheless increases error by $2.24\times$.

The completed CausVid rerun reduces mean path error relative to its paired unguided rollout by 17.3%, 25.5%, 23.6%, and 18.2% at the four tiers. Its error grows from 52.6 to 312.6px ($5.94\times$), with a $2.13\times$ increase between the final two tiers. The $8\times$ error is below the stationary-object reference of 400px, but mean endpoint error remains 695.2px for the 800px request despite netCos 0.909. These descriptive, single-seed comparisons vary duration and requested displacement together.

E.2 MID-STREAM TARGET UPDATES

The switch benchmark runs on CausVid over a 10-item subset. The requested direction changes at the temporal midpoint, right for the first half and down for the second, each segment carrying its own 150px target, with the switch at pixel frame 41 of 81. Path error is scored separately on each segment, the post-switch segment being re-anchored at the position and frame at which the switch occurred so it is scored against its own ramp. Settling time counts frames after the switch until the object's per-frame velocity realigns with the new target (cosine > 0.5) and stays realigned for 3 consecutive frames.

threshold	run	smoothing	settled	mean	distribution
0.3	1	–	10/10	2.3	0,1,1,2,2,2,3,3,4,5
0.3	3	–	10/10	5.1	0,1,1,2,2,3,5,7,15,15
0.5	1	–	9/10	5.2	0,1,2,3,3,3,3,5,27
0.5	**3**	**–**	**8/10**	**9.5**	0,1,3,3,7,8,27,27
0.5	3	3-frame	9/10	5.3	0,1,2,2,3,3,4,6,27
0.7	3	–	8/10	15.8	2,3,8,11,15,27,27,33
0.7	5	–	5/10	17.8	3,8,12,33,33

The bold row is the setting matching the $\cos > 0.5$ threshold used for the directional-agreement rate elsewhere in this paper (§3.2); this is a separate, binary criterion from the continuous FDA score reported in the main tables. Across the sweep the mean ranges from 2.3 to 17.8 frames and the number of items that never settle from 0 to 5 of 10.

F LOCALIZED ARTIFACT MEASURES

Sharpness and DOVER are global scores and may miss damage confined to a small region, such as a seam where the object vacates its source location, smeared or duplicated texture in the disoccluded region, or identity drift in the controlled object. We add two measures, computed on the same paired guided/base clips without additional generation. Outside-mask preservation measures collateral change to scene content away from both objects. The identity proxy targets appearance drift of the controlled object. Neither isolates the vacated or disoccluded region: the base clip's foreground mask covers that region, so outside-mask preservation excludes it by construction. Seam and disocclusion artifacts therefore remain assessed only qualitatively.

Outside-mask preservation. The completed dynamic-mask evaluation compares 160 guided/base pairs, 40 per configuration, excluding the dilated union of independently tracked foreground masks. Table 15 reports a 40px margin; Table 16 gives the complete margin sweep. These replace the fixed-initial-mask, zeroed-input background diagnostics. The retained LPIPS bins exclude foreground pixels only at their own resolution. Their features are computed on unmasked frames, and deeper AlexNet layers have receptive fields (about 195px at conv5) larger than the widest 120px margin. Background LPIPS can therefore still reflect foreground and disocclusion content near the object. Background PSNR is strictly pixel-local.

Identity proxy. We compute DINOv2 cosine similarities using the fixed bounding box of the initial object mask: guided-first versus guided-last (G_0, G_T), and guided-first versus base-first (G_0, B_0). The latter checks initial appearance agreement. The crop does not follow the object, so motion out of the box confounds the temporal score with background appearance.

cell	LPIPS bg ↓	LPIPS full	PSNR bg ↑	G_0, G_T ↑	G_0, B_0 ↑
WF-r / CausVid	0.114	0.346	26.36	0.727	0.9987
WF-r+FF (g=1.0) / CausVid	0.118	0.351	26.38	0.742	0.9987
WF-r / CF	0.258	0.384	17.98	0.553	0.9980
WF-r+FF (g=1.5) / CF	0.308	0.468	16.87	0.546	0.9979

Table 15: Artifact diagnostics on the held-out half ($n = 40$ per row). Background LPIPS and PSNR use the dynamic foreground-union exclusion with a 40px margin; full-frame LPIPS and identity columns retain their existing measurements. The last two columns report DINOv2 cosine similarity in a fixed initial-mask crop: guided-first versus guided-last (G_0, G_T), and guided-first versus base-first (G_0, B_0).

At a 40px margin, CausVid background LPIPS is 0.114 without feedforward correction and 0.118 with correction, with PSNR approximately 26.4dB in both cases. CF changes more: correction increases background LPIPS from 0.258 to 0.308 and reduces PSNR from 17.98 to 16.87dB. These are descriptive differences over each configuration's retained region. There are 58 empty-mask frame exclusions: 22 for WF-r/CF and 36 for corrected CF, with none for CausVid. All 40 items per row retain evaluable frames. Exclusions remain subject to mask review and can reflect either object exit or tracking failure.

Configuration	Metric	20px	40px	80px	120px
WF-r / CausVid	LPIPS ↓	0.144	0.114	0.099	0.091
	PSNR (dB) ↑	25.92	26.36	26.61	27.81
	Retained (%)	48.6	39.3	24.3	13.3
WF-r+FF / CausVid	LPIPS ↓	0.152	0.118	0.099	0.088
	PSNR (dB) ↑	25.73	26.38	27.12	28.74
	Retained (%)	48.0	38.6	23.6	12.8
WF-r / CF	LPIPS ↓	0.266	0.258	0.257	0.259
	PSNR (dB) ↑	17.86	17.98	18.33	18.72
	Retained (%)	66.6	59.2	46.5	35.9
WF-r+FF / CF	LPIPS ↓	0.321	0.308	0.299	0.291
	PSNR (dB) ↑	16.69	16.87	17.44	18.07
	Retained (%)	59.7	51.5	38.0	27.0

Table 16: Dynamic-mask background sensitivity across square dilation radii. Each metric is averaged over valid frames per item, then over 40 items. Retained denotes the mean valid pixel fraction; the retained feature fractions are smaller under conservative pooling. Feedforward gains are 1.0 for CausVid and 1.5 for CF.

At 120px, only 12.8–13.3% of pixels remain for CausVid, compared with 27.0–35.9% for CF. The CF correction has higher background LPIPS and lower PSNR at every tested margin; the two CausVid configurations remain close.

DTW cross-check. §4.2's trajPathErr uses a constant-rate reference ramp, which charges a clip that follows the right path but eases in or out for timing. Rescoring the same held-out clips with DTW, which allows local time warping, gives 53.5 vs. 33.3px for WF-r and 34.5 vs. 19.3px corrected on CausVid, and 158.8 vs. 121.8 versus 91.0 vs. 71.1px on CF.

G CURVED AND DIAGONAL PATHS

Path	Method	DTW ↓ mean	DTW ↓ median	trajPathErr ↓ mean	trajPathErr ↓ median
diagonal-100	WF-CausVid	36.0	31.8	52.8	52.0
	WF-CF	85.5	75.4	132.8	130.3
	SG-I2V	**14.8**	**13.4**	**23.3**	**22.2**
	FreeTraj	20.9	22.4	25.1	25.9
diagonal-200	WF-CausVid	48.4	44.5	71.2	73.9
	WF-CF	87.8	74.1	132.1	98.6
	SG-I2V	**23.8**	**17.7**	**34.9**	**35.9**
	FreeTraj	31.1	31.8	46.4	44.2
curve-100	WF-CausVid	35.7	32.4	52.5	42.6
	WF-CF	52.0	33.1	83.1	55.6
	SG-I2V	**18.4**	**19.0**	**29.9**	34.3
	FreeTraj	26.3	25.5	43.9	43.7
curve-200	WF-CausVid	54.2	42.3	82.2	71.9
	WF-CF	68.0	28.7	117.8	62.9
	SG-I2V	27.7	24.8	**45.9**	50.2
	FreeTraj	**17.5**	**14.6**	49.6	**47.2**

Table 17: Curved/diagonal extension, before feedforward correction. Warp Forcing: n=20 (10 objects × 2 seeds); SG-I2V/FreeTraj: n=10 (single sample).

This extension tests non-straight paths: 10 objects × 2 path types (45° diagonal; quarter-circle arc sweeping rightward to downward) × 2 magnitudes (100px, 200px net). The controller receives time-indexed waypoints sampled along each path and replans within each segment. trajPathErr compares

each frame with the intended time-indexed piecewise-linear curve; DTW compares with the same curve while allowing unconstrained local time warping.

On CausVid, matched-preview feedforward correction reduces the pooled error by 18.4%, with reductions of 32.2% and 29.2% on diagonal-100 and diagonal-200, respectively (Table 18). The gains are smaller on curved paths, particularly curve-200.

Path	Before (px)	Corrected (px)	Reduction
diagonal-100	52.75	35.77	32.2%
diagonal-200	71.20	50.38	29.2%
curve-100	52.47	44.23	15.7%
curve-200	82.18	80.60	1.9%
Pooled	64.65	52.75	18.4%

Table 18: CausVid curved/diagonal results before and after matched-preview feedforward correction (g=1.0), using the saved trajectory-error results. Each path condition contains 20 items (10 objects $\times$ seeds 42 and 123). The pooled row averages all 80 items; percentage reductions are computed relative to the corresponding uncorrected mean, before rounding.

WF-CF's error is consistently higher on the diagonal than the curve at both magnitudes, which superficially suggests curves are easier. Paired item-by-item, diagonal is worse in 17 of 20 pairs (mean $+49.7$px). This comes from the geometry, as our quarter-circle sweeps rightward first and downward only later, while the diagonal moves right and down simultaneously from frame 0. Given CF's direction asymmetry (Appendix I), the curve spends its early frames exploiting CF's easiest direction before requesting any of the harder component, while the diagonal forces CF to reconcile its best and worst directions concurrently for the whole clip. CausVid, which has no such asymmetry, shows no such gap.

H Controller comparisons and transmission

H.1 Causal-transfer experiment

SG-I2V and FreeTraj use their public implementations for the native reference runs. We transplant their defining mechanisms, plus DragStream and QWERTY, into CausVid using released defaults where possible; full-sequence operations become causal/chunk-local and UNet choices map to DiT blocks. SG-I2V's native run uses CF-generated first frames; its transplant uses CausVid's. These comparisons are secondary to the multi-seed transmission and correction results.

The purpose of this experiment is to characterize the tested causal adaptations. Table 2 (main paper, §4.3) retains the accuracy and quality measurements, and Table 19 gives directional tests and optimization-budget diagnostics.

Transfer across mechanisms. Transplanted into CausVid, **FreeTraj's** corrected noise-and-attention mechanism improves only modestly over the balanced-direction null (netCos 0.145, p=0.064 against its paired base clips) while preserving DOVER (0.693 guided vs. 0.675 base). It reaches netCos 0.997 on its native bidirectional backbone (Table 3), showing that the native mechanism does not retain its directional reliability under this causal transfer. **DragStream's** corrected transplant exhibits moderate control (netCos 0.238, p=0.055). **QWERTY** transfers detectably on test10 (netCos 0.277, p=0.027).

These transplant results are conditional on the tested adaptations and hyperparameters. Restricting full-sequence operations to causal chunks and mapping their components into CausVid changes the setting in which the original mechanisms were designed to operate; released defaults may therefore be poorly matched to this environment.

SG-I2V aligns intermediate attention features; the surrogate aligns raw latents. Increasing the surrogate's optimization budget provides a separate test of how control strength affects accuracy and prior transmission.

Mechanism transplanted onto CausVid	trajPathErr ↓	netCos ↑	FDA ↑	p
Mechanism transplants				
FreeTraj, noise-flow + attention	114.9	0.145	0.388	0.0645
SG-I2V, feature alignment (5 it)	65.9	0.428	0.461	0.0059
DragStream, both corrections (5 it)	87.2	0.238	0.329	0.055
QWERTY	72.4	0.277	0.449	0.0273
Raw-latent alignment surrogate				
5 iterations	78.9	0.131	0.520	—
25 iterations	40.6	0.796	0.685	—
50 iterations	42.0	0.762	0.680	—
100 iterations	40.9	0.773	0.683	—
50 iterations, 3× lr	39.7	0.786	0.690	—
Warp Forcing (WF-r)	53.5	0.970	0.648	0.0020

Table 19: Training-free mechanisms re-implemented in CausVid's generation loop (n=40 per row, 10 objects, seed 42, test10) with matched conditioning and seeds. Each test compares a method with its own paired unguided clips. The raw-latent rows evaluate the surrogate's optimization budget. p is a two-sided paired Wilcoxon test on the ten object-averaged guided-minus-base netCos differences. Surrogate budget rows report descriptive accuracy comparisons.

mechanism / backbone	base clips	trajPathErr	slope	95% CI (clustered)	r
Causal backbones					
Raw-latent align. / CausVid, 5 it	10	78.9	+0.73	[+0.64, +0.84]	0.94
WF-r / CausVid	10	53.5	+0.67	[+0.52, +0.92]	0.68
WF-r / CF	10	158.8	+0.56	[+0.14, +1.01]	0.60
Raw-latent align. / CausVid, 25 it	10	40.6	+0.18	[+0.09, +0.34]	0.33
SG-I2V transplant / CausVid	10	65.9	+0.09	[+0.04, +0.66]	0.33
FreeTraj transplant / CausVid†	10	114.9	−0.01	[−0.09, +1.44]	−0.01
DragStream transplant / CausVid	10	87.2	0.06	[−0.11, +0.67]	0.23
QWERTY transplant / CausVid	10	72.4	+0.42	[+0.36, +0.94]	0.66
Native bidirectional backbones					
SG-I2V / SVD	20	29.0	−0.10	[−0.21, +0.21]	−0.15
FreeTraj / VideoCrafter2†	20	23.0	+0.03	[−0.03, +0.07]	0.04

Table 20: Projected guided–base dependence; intervals cluster by object (one base clip supplies four directions). †FreeTraj changes initial noise, weakening this interpretation.

The main results (§4.3) use within-controller changes to distinguish accuracy from transmission. The following diagnostics examine outcome dispersion, sampling structure, and statistical dependence across the evaluated configurations.

The identity slope $= r \cdot \mathrm{SD}_y/\mathrm{SD}_x$ separates correlation from outcome dispersion. SG-I2V/SVD has $\mathrm{SD}_y = 61.0$px and $\mathrm{SD}_y/\mathrm{SD}_x = 0.68$; FreeTraj/VideoCrafter2 has $\mathrm{SD}_y = 39.3$px and a ratio of 0.79. Their residual SDs are 60.3px and 39.2px, respectively. Both retain substantial outcome variation while showing little linear dependence on unguided displacement.

Varying SG-I2V's requested magnitude on SVD (50/100/200/300px) moves path error from 13.3 to 33.8px with no movement in the slope (+0.01, +0.04, −0.18, −0.11; all CIs contain 0). This manipulation does not reach the 50–80px error range where the causal cells sit, because raising the requested displacement raises the target proportionally and leaves relative accuracy high.

Each object supplies one base rollout and four direction projections, so the 40 held-out items contain 10 independent base clips. We therefore bootstrap whole objects. Table 21 compares these intervals with item-level resampling and shows how the dependence structure affects uncertainty.

Single-seed fits. OLS intervals resample whole objects because four directions share one base clip. The 20-object seed-42 fit gives $y = 0.872x + 165$px ($r = 0.76$, 95% CI $[0.44, 1.11]$).

cell	items	base clips	item-level CI	clustered CI
WF-r / CausVid	40	10	$[+0.41, +1.03]$	$[+0.52, +0.92]$
WF-r / CF	40	10	$[+0.21, +0.97]$	$[+0.14, +1.01]$
Raw-latent align. / CausVid, 5 it	40	10	$[+0.61, +0.88]$	$[+0.64, +0.84]$
Raw-latent align. / CausVid, 25 it	40	10	$[+0.07, +0.33]$	$[+0.09, +0.34]$
SG-I2V transplant / CausVid	40	10	$[+0.05, +0.61]$	$[+0.04, +0.66]$
FreeTraj transplant / CausVid	40	10	$[-0.13, +1.00]$	$[-0.09, +1.44]$
DragStream transplant / CausVid	40	10	$[-0.10, +0.53]$	$[-0.11, +0.67]$
QWERTY transplant / CausVid	40	10	$[+0.31, +0.62]$	$[+0.36, +0.94]$
SG-I2V / SVD	80	20	$[-0.24, +0.14]$	$[-0.21, +0.21]$
FreeTraj / VideoCrafter2	80	20	$[-0.16, +0.21]$	$[-0.03, +0.07]$

Table 21: 95% bootstrap intervals on the slope under two resampling units: individual direction-items, and whole objects (one base clip each). "base clips" counts the separately generated unguided rollouts behind each row.

H.2 Motion-Zero native reference and causal transplant

We evaluate Motion-Zero (Chen et al., 2025) on the test10 objects in four directions ($n = 40$, seed 42, requested displacement 150px). The native reference uses the released implementation on ZeroScope v2 at 384×384, 16 frames, 30 denoising steps, and classifier-free guidance scale 12. We retain the released spatial and temporal guidance settings. Clean videos without box overlays are evaluated.

The CausVid transplant uses 81 decoded frames at 832×480 and maps UNet/CLIP attention constraints to DiT layers 12/13 and T5 token spans. Spatial constraints and temporal cosine losses operate within the current causal chunk, with one gradient update of size 20 at denoising step 0; the initial block remains unchanged.

	trajPathErr ↓	endErr ↓	netCos ↑	DOVER ↑
Motion-Zero transplant	104.86	171.69	0.721	0.6602
WF-r	53.5	131.07	0.970	0.6822
Motion-Zero native reference	74.07	148.69	0.285	0.3682

Table 22: Motion-Zero evaluation. Errors are in native pixels. The native row differs in backbone, resolution, duration, and conditioning.

Motion-Zero improves over its base by 29.4%, and under this transplant WF-r has 49.0% lower guided trajectory error, winning on 31/40 cases, with lower endpoint error and higher netCos and DOVER.

H.3 Seed-aware inference for the 10-seed estimator

Wrong-seed randomization. For each item, the ten centered base values are reassigned across seeds by a uniformly sampled derangement and the within-item slope is refit. The interval summarizes the resulting randomization distribution. Because the centered predictor sums to zero, this construction has

$$\mathbb{E}_\pi[\hat{\beta}_\pi] = -\frac{\hat{\beta}_{\text{matched}}}{S-1} = -\frac{0.523}{9} = -0.0581,$$

up to Monte Carlo error. The observed derangement mean, -0.059 (randomization interval $[-0.143, +0.037]$ over 2000 draws), agrees with this expectation. The matched coefficient of 0.523 lies above this randomization interval, showing that reassignment disrupts the observed positive alignment.

Bootstrap implementation. In the object cluster bootstrap we draw 20 objects with replacement, carry all four directions and all ten seeds of a drawn object together, and relabel duplicated objects so that a twice-drawn object contributes two separate sets of item means. In the seed cluster bootstrap we draw 10 seeds with replacement and demean each item over the drawn seed multiset, so

a duplicated seed enters the item mean twice. The crossed bootstrap draws objects and seeds independently and takes the object $\times$ seed cells they select, again re-demeaning inside the resample. The wild cluster bootstrap by seed imposes β=0, multiplies each seed's residuals by a Rademacher weight, and enumerates all 2^{10} weightings exactly, inverting the bootstrap-t distribution for the interval. Two-way cluster-robust standard errors use the Cameron-Gelbach-Miller decomposition $V = V_{\text{obj}} + V_{\text{seed}} - V_{\text{obj}\cap\text{seed}}$ with finite-cluster corrections on the two marginal terms and a t_9 reference, the conventional conservative choice at $\min(G)$=10 (Table 23).

Seed fixed effects. The four cardinal directions project each object's base displacement onto $\pm\Delta x, \pm\Delta y$, making the item-demeaned predictor sum to zero within every seed. Additive seed effects are therefore orthogonal to this predictor. The two-way within estimator gives $\beta = 0.5235$, matching the item fixed-effects estimate to machine precision ($|\Delta\beta| < 10^{-14}$). We therefore fit $y_{i,s} = \alpha_i + \gamma_{s,d(i)} + \beta x_{i,s} + \epsilon_{i,s}$, absorbing both item and seed-by-direction effects by alternating demeaning. The coefficient is 0.525, with two-way object$\times$seed clustered 95% CI $[0.26, 0.79]$ ($p = 0.0015$, t_9 reference), compared with 0.523 before this adjustment.

Exact sign-flip enumeration. On demeaned data the no-intercept slope is linear in the per-object contributions $c_o = \sum_{i\in o} \tilde{x}_i\tilde{y}_i$, so flipping object signs gives $\beta^{(r)} = (\sum_o q_o c_o) \,/\, \sum_i \tilde{x}_i^2$ with $q_o \in \{-1, +1\}$, and the full null can be enumerated. Flipping both variables together would leave their product invariant and be a no-op, so we flip the response only. The test is valid under the null that, conditional on the demeaned predictor, an object's demeaned response deviations are sign-symmetric. Enumerating all $2^{20} = 1{,}048{,}576$ assignments, exactly 2 have $|\beta^{(r)}| \geq |\hat{\beta}|$ (the all-$+$ and all-$-$ assignments), giving a two-sided $p = 2/1{,}048{,}576 = 1.9 \times 10^{-6}$ and a one-sided $p = 1/1{,}048{,}576 = 9.5 \times 10^{-7}$; the permutation unit is the object, and the statistic is the refitted within slope. Because this scheme permutes objects, it does not preserve seed-level dependence; the seed-wise wild cluster bootstrap in Table 23 is its seed-aware counterpart, and returns its own floor, $p = 2/2^{10} = 0.0020$.

inference scheme	dependence handled	95% CI on β=0.523
object cluster bootstrap	object	$[+0.29, +0.76]$
seed cluster bootstrap	seed	$[+0.28, +0.67]$
crossed bootstrap (object $\times$ seed)	both, resampled independently	$[+0.22, +0.82]$
two-way cluster-robust (CGM)	both, analytic	$[+0.25, +0.80]$
wild cluster bootstrap by seed	seed, exact 2^{10} enumeration	$[+0.26, +0.79]$
leave-one-seed-out refits	— (stability)	range $[+0.43, +0.58]$

Table 23: Uncertainty on the primary 10-seed item fixed-effects slope under five schemes accounting for object dependence, seed dependence, or both (800 observations, 20 objects, 10 seeds).

seed	pooled slope	95% CI (object-clustered)	r
17	+0.323	$[+0.18, +0.77]$	0.25
99	+0.587	$[+0.35, +0.84]$	0.42
314	+0.587	$[+0.49, +0.81]$	0.56
555	+0.476	$[+0.24, +0.69]$	0.39
777	+0.873	$[+0.44, +1.00]$	0.73
2718	+0.419	$[+0.16, +0.69]$	0.31
31415	+0.561	$[+0.30, +0.78]$	0.39
42	+0.872	$[+0.44, +1.11]$	0.76
7	+0.434	$[+0.21, +0.86]$	0.36
123	+0.416	$[+0.31, +0.60]$	0.34

Table 24: Per-seed expanded pooled slopes on CausVid (the 20-object slope-analysis set), each fit separately with its own object-clustered bootstrap interval. All ten are positive with intervals excluding zero.

Tracker independence. Because both variables in the main regressions come from the same CoTracker3 instance, a systematic tracker bias could in principle inflate the measured coupling even

if the underlying motion were less correlated. Recomputing the WF-r/CausVid benchmark-held-out pooled slope with an independent, classical Lucas-Kanade optical-flow tracker (Shi-Tomasi corners seeded on the identical SAM2 mask, so only the tracking algorithm differs) gives slope 0.665, 95% CI $[0.55, 0.85]$, essentially unchanged from CoTracker3's 0.671, $[0.52, 0.92]$ on the same items. On the 3-seed, 240-observation subset of the item fixed-effects estimator (Appendix A lists the 20-object slope-analysis set), the same swap gives slope 0.452, 95% CI $[0.34, 0.60]$ under Lucas-Kanade against the 3-seed CoTracker3 estimate of 0.641, $[0.30, 0.98]$, still clearly positive and highly significant (sign-flip $p<0.0001$) but smaller. This estimator regresses small within-item, cross-seed deviations, so it is more exposed to measurement-error attenuation from Lucas-Kanade's coarser, non-learned tracking. On that 3-seed subset the positive association therefore survives replacement of the tracker, with a reduced magnitude.

Leave-one-object-out. The expanded pooled estimate is insensitive to any single object: a leave-one-object-out refit moves the CausVid slope only within $[0.60, 0.96]$ (full-sample 0.872, 95% CI $[0.44, 1.11]$) and never changes sign. No single high-drift object drives either estimate.

Testing block commitment directly. We vary commitment while holding the controller fixed. In CausVid, each look-back revision restores the boundary before the earliest revised block and regenerates that block and its successors, including their cache entries. Slopes for $W = 0, \ldots, 6$ are $0.671, 0.713, 0.684, 0.914, 0.883, 0.699, 0.532$. The clustered bootstrap trend is -0.0087 per additional block (95% CI $[-0.089, +0.122]$, $p = 0.91$); the $W = 6$ interval is $[0.09, 1.10]$. In Wan2.1, the same global clean-estimate warp is applied to one fully revisable block or three sequentially committed blocks. With delivered displacement matched within 1% ($126 \rightarrow 127$px), the slope changes from 0.888 to 0.940 (difference $+0.052$, paired cluster-bootstrap 95% CI $[-0.007, +0.092]$, $p = 0.066$, 39 clusters). The fully revisable arm demonstrates transmission under whole-clip sampling. Framewise slopes decline smoothly within both causal models. The commitment contrasts leave its contribution uncertain.

Seed-linked and category-linked variation. Wan2.1's unguided displacement has cross-seed correlation $r = 0.10$ for the same object (95% CI includes zero). Within each of four seeds, 65–85% of 17–20 unrelated objects drift in the same horizontal direction (combined $p = 2 \times 10^{-4}$ under independent per-object coin flips). Guided/base slopes range from 0.68 to 0.96. These observations suggest a shared directional component within seeds. Section 4.1 reports the corresponding CausVid randomization analysis.

I CF DIRECTION ASYMMETRY AND CORRECTION

CF exhibits a pronounced rightward motion tendency in the evaluated campaigns, motivating direction-specific checks of correction.

Grouped by poke direction (means $\pm$ item SD): right 0.998 ± 0.002, down 0.743 ± 0.316, up 0.767 ± 0.261, left 0.440 ± 0.879, with 30% of left items ending with net motion opposite the request. CausVid shows none of this (left 0.991, right 0.993).

Every experiment already runs an unguided base pass. Across the 20-object main sweep and an independent 10-object replication, CF's net horizontal displacement is rightward in 100% of cases (means $+360$px and $+181$px), with no exceptions. CausVid shows no comparable bias (70% positive, mean $+63$px, with substantial negative outliers). The bias does not require a motion word: under the same seeds, rightward drift persists at comparable or greater magnitude for the bare prompt "A {object}." than for "A {object} is moving." ($+197$px vs. $+181$px), indicating that it arises from the checkpoint's rollout dynamics.

On the 20-object leftward-request condition (the benchmark's worst case, $dx{=}-150$), correction at the selected gain improves netCos (mean $\pm$ item SD) $0.328 \pm 0.917 \rightarrow 0.583 \pm 0.805$ and trajPathErr $180.6 \rightarrow 117.7$px, with items moving opposite the request dropping 7/20 to 4/20. That a single global gain fixes five failures but breaks two previously successful items suggests that per-item or per-category gain scheduling may be necessary.

J QUALITATIVE SAMPLES

Qualitative extensions. The two-object CF demonstration uses separate SAM2 trackers and fixed rates $d_i = \Delta_i/(T_{\text{lat}}-1)$. The corresponding latent flow fields are summed; overlaps have no priority or occlusion resolution, and opposing commands can cancel. Initialization excludes the first object's mask when locating the second, but later overlaps are unrestricted.

The background demonstration applies $\phi_{\text{bg}}(p) = (1 - w_M(p))\Delta_{\text{bg}}/[8(T_{\text{lat}} - 1)]$, where w_M is the tracked foreground mask resized to the latent grid and feathered inward. This specifies image-plane background translation, with no separate foreground command or 3D camera model. Object/background coupling arises during denoising. The displayed clips are archived runs; the background-tracking scores come from a separate benchmark pass.

FreeTraj appearance diagnostic. A native FreeTraj ablation covers cat, dog, bird, and horse at seed 123, with rightward targets, 81 frames at 512×320, and 50 DDIM steps. Five arms hold prompt, checkpoint, seed, and sampler settings fixed: vanilla generation, both interventions disabled, noise-flow only, attention only, and full guidance. Disabling both interventions closely reproduces vanilla generation (pixel MSE 1.7–4.7). Noise-flow increases the whole-frame fraction of pixels below intensity 32/255 for all four objects; sharpness changes are object-dependent.



Figure 6: Additional qualitative trajectory-following comparison.

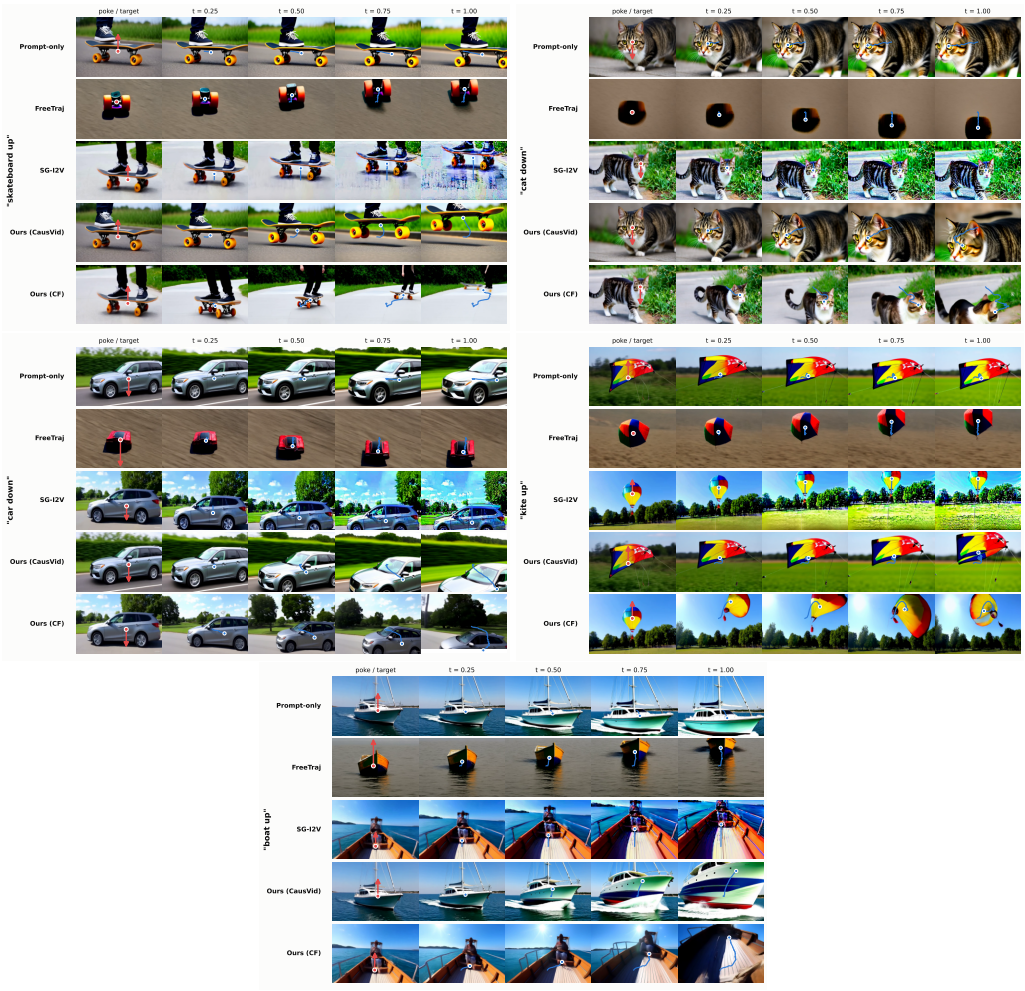


Figure 7: Additional qualitative trajectory-following comparison. (continued)

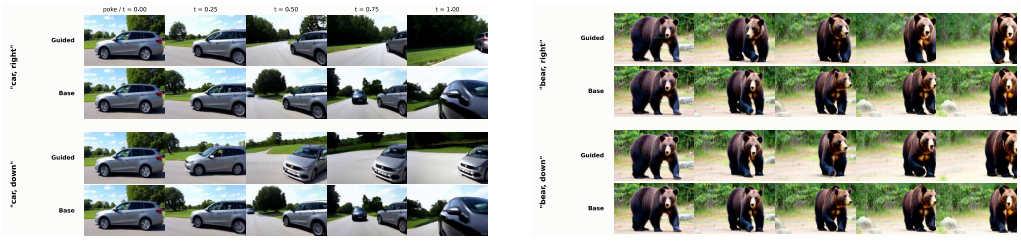


Figure 8: Background/camera motion induced jointly with an object poke. On a 10-item pass tracking a fixed background point grid against the requested pan, netCos $= 0.999$ and binary directional-agreement of 0.934 (see §3.2 for the definitions).

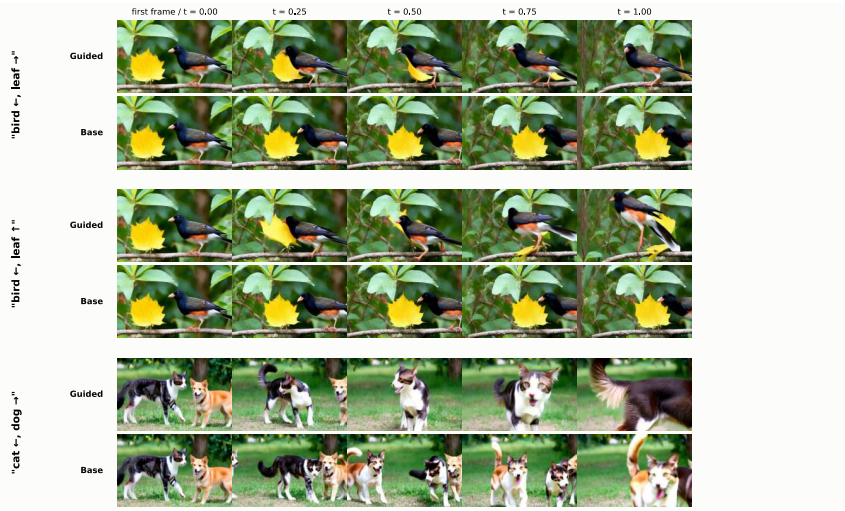


Figure 9: Multi-object control with independently specified per-object trajectories, reported qualitatively only.