TIME-TO-MOVE: TRAINING-FREE MOTION CONTROLLED VIDEO GENERATION VIA DUAL-CLOCK DENOISING

Anonymous authors
Paper under double-blind review

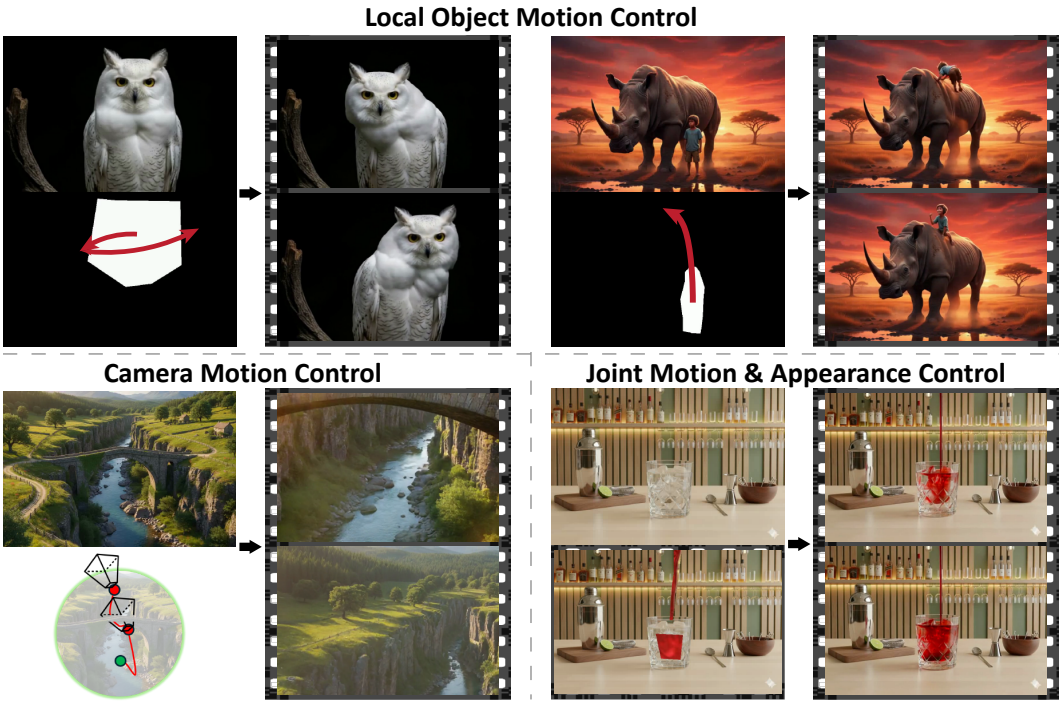


Figure 1: **Qualitative results of Time-to-Move on various tasks.**

ABSTRACT

Diffusion-based video generation can create realistic videos, yet existing image- and text-based conditioning fails to offer precise motion control. Prior methods for motion control typically rely on displacement-based conditioning and require model-specific fine-tuning, which is computationally expensive and restrictive. We introduce Time-to-Move (TTM), a training-free, plug-and-play framework for motion- and appearance-controlled video generation with image-to-video (I2V) diffusion models. Our key insight is to use crude reference animations, obtained through user-friendly manipulations such as cut-and-drag or depth-based reprojection, as direct motion guidance, analogous to using coarse layout input in image editing. To integrate these signals, we adapt SDEdit to the video domain while anchoring the appearance with image conditioning. We further propose *dual-clock denoising*, a region-dependent strategy that enforces strong alignment in motion-specified regions and grants flexibility elsewhere, balancing fidelity to user intent with natural dynamics. This lightweight modification of the sampling process incurs no additional training or runtime cost and is compatible with any backbone. Extensive experiments on object and camera motion benchmarks show that TTM matches or exceeds existing training-based baselines in realism and motion control. Beyond this, TTM introduces a unique capability: precise appearance control through pixel-level conditioning, exceeding the limits of text-only prompting. Please visit our anonymous demo page.

1 INTRODUCTION

Diffusion-based video generators have recently achieved remarkable visual quality, yet their controllability remains limited. Image-to-video (I2V) models partially alleviate this limitation by conditioning on a single input frame, which gives users direct control over the appearance of the generated video. However, *motion* control remains largely prompt-driven which is often unreliable, coarse, and insufficiently fine-grained for interactive use. To address this gap, a practical generative video system should provide an interface that defines both *what* moves and *where* it moves, ensuring realistic, temporally coherent motion while preserving the appearance of the input image. Such fine-grained control would enable interactive content authoring, post-production, and animation prototyping, where creators require precise, local adjustments with fast feedback. Existing approaches for controllable motion in generation typically encode user intent through auxiliary control signals such as optical flow and then *heavily fine-tune* a generator to ingest this motion conditioning (Burgert et al., 2025; Geng et al., 2025). Such methods are computationally expensive to train, often compromise the quality of the original model, and remain model-specific, requiring architectural modifications to incorporate the controls. This motivates a framework that can be applied to *off-the-shelf* video diffusion backbones without expensive tuning or additional data.

We introduce Time-to-Move (TTM), a *training-free*, architecture-agnostic, plug-and-play inference procedure for video diffusion models that matches the speed of standard generation. We observe that crude animation inputs, for instance, created by simple cut-and-drag manipulation or by straightforward reprojection of the image into novel views using estimated depth, can serve as a useful proxy for the intended target. Such references capture coarse structure and convey the desired motion, while remaining easy to generate and flexible enough to be made as specific or detailed as the user wishes. To transform these crude signals into realistic videos, we draw on *SDEdit* (Meng et al., 2021), which shows that coarse structure can be imposed by adding noise to the timestep where the layout is determined. By analogy, we hypothesize that noising the synthetic reference video to the point where *motion* is established by the video diffusion model can embed the intended dynamics. Indeed, this strategy successfully injects motion, but at this noise level fidelity to the reference appearance is lost. To mitigate this, we turn to *image*-conditioned video diffusion models, which preserve the identity and scene details of the initial frame and thereby maintain appearance consistency throughout the generated sequence.

Even with appearance preserved, motion cues remain uneven across regions. In the synthetic guiding reference video, some regions, such as dragged objects, may contain strong user-specified dynamics, while others remain unspecified. These unconstrained regions are not meant to stay static, but rather to adapt naturally in support of the intended movement. This allows the model to adhere closely to a specified motion where it exists while allowing greater freedom to invent plausible dynamics elsewhere. To this end, we introduce a novel *region-dependent dual-clock denoising process*, which assigns one of two distinct SDE timesteps to different regions across frames: strong alignment for user-specified motion and weaker alignment for unconstrained areas, thus allowing spatially varying conditioning strength. To realize this effect without retraining the model, we employ a simple yet effective diffusion blending strategy akin to (Avrahami et al., 2022; Lugmayr et al., 2022).

Unlike prior approaches that rely solely on (either sparse or dense) displacement fields as a guiding signal, our method is conditioned directly on the reference video itself. This provides a richer supervisory signal: In regions where alignment is strongly enforced, we not only constrain motion, but can also dictate appearance attributes such as color, shape, or style. As a result, TTM enables *joint control of motion and appearance*, extending the conditioning space beyond flow-only interfaces. We exploit this capability to support appearance-sensitive prompting in tandem with motion control, for example, animating an object along a user-specified trajectory while simultaneously changing its color (See Figure 7). In summary, our work makes the following contributions:

- **Training-free motion control with crude animation.** Simple user-provided animations (e.g., cut-and-drag or depth reprojection) act as effective motion proxies. Adapting SDEdit-style noise injection to video diffusion, and anchoring appearance with image conditioning, transforms these coarse signals into realistic motion without training.
- **Region-dependent dual-clock denoising.** We introduce a denoising process with two noise schedules: strong alignment in motion-specified regions and weaker alignment elsewhere. This dual-clock design provides spatially varying conditioning without retraining.

- **Joint motion–appearance control.** Conditioning on full reference frames, rather than motion trajectories alone, enables simultaneous control of motion and appearance, a capability previously limited to ambiguous text prompting.

Extensive experiments show that TTM consistently ranks among the best performing methods on both object and camera motion benchmarks, outperforming even training-based baselines. Our approach is training-free and plug-and-play, validated across three different I2V backbones, and is as efficient as standard video sampling.

2 Related Work

Learning Motion Control in Video Generation. A common strategy in motion control is to *learn* a trajectory-conditioned representation and fuse it throughout the network. Concretely, methods inject user trajectories via multi-scale fusion of trajectory maps in U-Net blocks (Yin et al., 2023), parameter-efficient LoRA modules that decouple camera and object motion (Li et al., 2025b), or motion-patch tokens integrated across transformer blocks (Zhang et al., 2025a); ATI encodes point tracks as Gaussian-weighted latent features (Wang et al., 2025), TrackGo inserts auxiliary branches into SVD's temporal self-attention (Zhou et al., 2025; Blattmann et al., 2023), and MotionPro uses region-wise trajectories plus a motion mask to distinguish object vs. camera motion (Zhang et al., 2025b). Other methods, like ours, incorporate explicit motion-based cues rather than relying solely on learned injection. DragAnything extracts entity representations from first-frame diffusion features and injects trajectory conditioning via a ControlNet-style branch. Go-with-the-Flow (Burgert et al., 2025), aligned with our architecture-agnostic aim, warps diffusion noise with optical flow; unlike our method (and the approaches above), it additionally fine-tunes the base model so that temporally correlated noise yields controllable motion.

Training-free motion-controllable video generation. Several approaches avoid additional training by reusing pretrained models. Recent *text-to-video* (T2V) attempts manipulate attention, TrailBlazer edits spatial and temporal attention early in denoising (Ma et al., 2024); PEEKABOO gates regions with masked spatio-temporal attention (Jain et al., 2024); and FreeTraj adds low-frequency noise shaping with attention biases to follow boxes (Qiu et al., 2024). However, these T2V methods bind boxes to text, which cannot specify fine part-level motion, and do not allow precise appearance control or in-place animation of a given image. Video-MSG (Li et al., 2025a) proposes a training-free guidance scheme where an MLLM-generated video sketch is inverted to initialize the noise of a T2V backbone, which is then denoised into the final video. Like TTM, it is training-free and backbone-agnostic, but Video-MSG operates in a text-to-video setting, requires inversion, and uses a single globally structured noise field shared across all spatial regions. In parallel to these approaches, motion *transfer* methods generate a video by applying motion from a driving sequence to a still image, but require a suitable reference video (Jeong et al., 2024; Yatim et al., 2024; Pondaven et al., 2025). Targeting *I2V* without reference videos, SG-I2V (Namekata et al., 2024) enforces cross-frame consistency by replacing each frame's spatial self-attention keys/values with those of the first, then optimizes the latent with a box-restricted similarity loss, and re-injects high-frequency detail via an FFT. Although zero-shot and conceptually aligned with our concept of aligning moved objects to their first-frame representation, This method is demonstrated on SVD-specific layers, so generality for other backbones is unclear; moreover, as shown in Sec. 4.1, it often induces unintended camera motion. Finally, (Yu et al., 2024) proposes a training-free trajectory-guided I2V using gated self-attention for layout-conditioned control, temporal attention for propagation, and a Motion Afterimage Suppression step. Its modular inpainting design inherits limitations—fidelity tied to T2I inpainting and grounding tokens, heuristic handling of large displacements, while also being designed for a specific video generation model.

Heterogeneous Denoising. Selective or asynchronous denoising has been explored in several contexts. Kim et al. (2025) reformulate inpainting with element-wise noise schedules and spatial timestep embedding, enabling region-asynchronous denoising while adapting a pretrained model via LoRA. SVNR (Pearl et al., 2023) addresses spatially variant sensor noise by training with per-pixel timesteps and starting the reverse process directly from the noisy input. Diffusion Forcing (DF) introduces temporal heterogeneity by assigning each token (e.g., a video frame) its own noise level during training, and at sampling uses a 2D scheduling matrix over time and noise levels so tokens

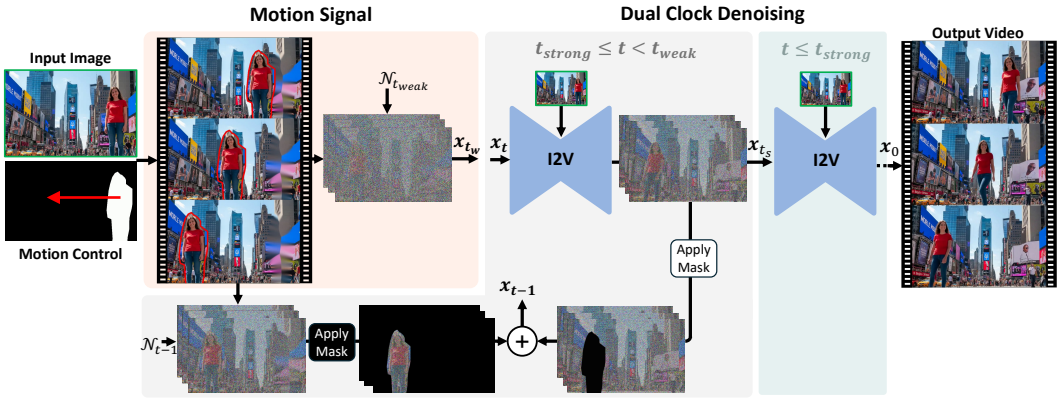


Figure 2: **Overview of Time-to-Move.** Given an input image and a motion instruction, a mask marks the region under strong control. A motion signal is then generated automatically and, together with the image, conditions an image-to-video (I2V) diffusion model. During sampling, denoising starts at different noise levels—lower inside the mask to enforce the specified motion, and higher outside to allow natural deviations in the (typically static) background. Joint sampling then yields a realistic video that preserves input details while accurately following the motion control.

are denoised at different rates (Chen et al., 2024). In contrast to RAD, SVNR, and DF, our approach is training-free: We impose region-specific schedules directly at inference. RePaint (Lugmayr et al., 2022) is also training-free, but repeatedly re-noises unmasked regions and denoises solely the mask, so only part of the image is actively denoised. Our method heterogeneously denoises the entire image, eliminating RePaint's resampling loops since no region is excluded.

3 METHOD

Our goal is to enable precise motion control in generative video models. Inspired by *SDEdit*, which injects coarse edits into images via noising and denoising, we treat a crude warped animation (Sec. 3.1) as the video analogue of such edits and adapt *SDEdit* to inject intended motion into video diffusion (Sec. 3.2). To avoid the loss of identity that occurs when noising alone drives the process, we opt for *image conditioning*, anchoring the generation to the clean first frame so that the appearance is preserved throughout the sequence. Building on these foundations, we introduce a novel dual-clock denoising process (Sec. 3.3) that assigns different noise levels to distinct regions, allowing spatially varying motion guidance. An overview of these components is shown in Fig. 2. Finally, our procedure naturally extends to appearance control, allowing simultaneous specification of both dynamic and visual attributes.

Problem Formulation. Our method takes as input (i) a single image $I \in \mathbb{R}^{3\times H\times W}$, (ii) a coarse, user-specified warped reference video with F frames, $V^w \in \mathbb{R}^{F\times 3\times H\times W}$, and (iii) a binary mask video $M \in \{0,1\}^{F\times H\times W}$ indicating, for each frame, the regions where stronger appearance and motion guidance are desired. The objective is to generate a realistic video $x_0 \in \mathbb{R}^{F\times 3\times H\times W}$ that maintains fidelity to the input image while accurately following the prescribed motion.

3.1 MOTION SIGNAL

We begin by describing how the motion signal V^w is generated. To facilitate user-friendly interaction, the user selects a region in the first frame to produce an initial binary mask M_0, then specifies a coarse motion by dragging this region along a trajectory, yielding the sequence M. This defines a piecewise-smooth displacement field within the masked region, which induces per-frame warps of the input image. The warped video V^w is obtained by warping I, with identity mapping outside the mask, with further details explained in Sec. 5.1. Although presented here as dragging, both $V^w$and M can be constructed in multiple ways. For example, in Sec. 4.2 we show that V^w can also be produced by pixel-wise warping of the input image according to monocular depth estimation. As

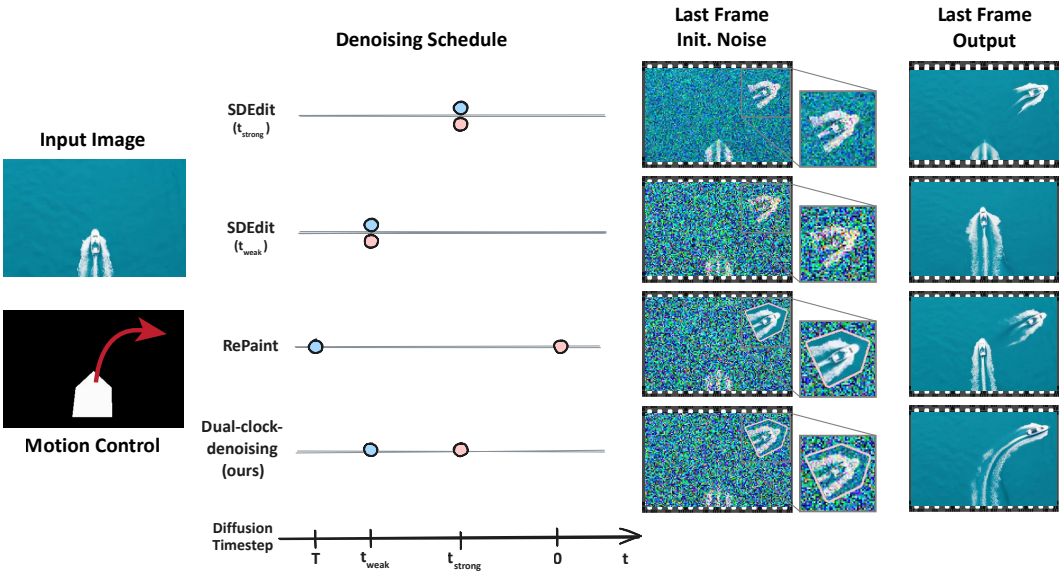


Figure 3: **Region-dependent denoising strategies.** SDEdit (single clock): low noise levels over-constrain the video, suppressing non-masked region dynamics; high noise levels improve realism but drift from the prescribed motion. RePaint (foreground override): motion is enforced in the object, but uncontrolled regions exhibit artifacts such as duplication. Dual-clock (ours): masked regions follow the intended motion with strong fidelity, while the background denoises more freely, yielding realistic dynamics without artifacts.

demonstrated in our demo page, we support additional interactions, such as rotation and scaling of the selected region, which integrate seamlessly into the same formulation. While such warped animations are visually unrealistic, they faithfully capture the user-intended object placement and temporal structure. We exploit these properties by using them as a guiding signal for the video diffusion model during generation, and note that V^w can also encode appearance modifications, such as color changes, within the same framework.

3.2 SDEdit Adaptation for Motion Injection

A key observation of our approach is that unlike prior methods, which extract only flow fields from warped videos (Burgert et al., 2025), we treat the warped video itself as the guiding signal. Inspired by the role of strokes in image SDEdit (Meng et al., 2021), these crude animations provide a coarse but explicit user instruction for motion. We therefore adapt SDEdit to the video setting by directly using the warped animation V^w as the guiding input. We initialize sampling from a noisy version of the warped reference, $x_{t^*} \sim q(x_{t^*} \mid V^w)$, Previous publication (Shaulov et al., 2025) shows that coarse motion is determined early in the denoising trajectory; By noising V^w to t^*, the intended dynamics are injected at precisely this stage. If we were to apply this procedure in a text-conditioned video diffusion model, the fidelity to the input image would be quickly lost: The model's only knowledge of appearance comes from the noised V^w, so fine details cannot be preserved. To overcome this limitation, we instead opt for an *image-conditioned* video diffusion model, which anchors generation to the clean first frame I. The resulting sampling process, $x_0 \sim p_\theta(x_0 \mid x_{t^*}, I)$, faithfully integrates the motion guidance from V^w while preserving identity and appearance throughout the generated sequence.

3.3 Region-Dependent Dual-Clock Denoising

SDEdit employs a single noising timestep t^* to corrupt the reference signal before denoising. In our setting, this creates a trade-off. The warped video V^w contains regions where motion is explicitly specified (masked areas), alongside regions without explicit instruction. For the masked regions, we want the generated video to closely follow the prescribed motion. In unmasked areas, we do not want them to stay static; instead, they should adapt naturally to support the motion. For example, in Fig. 3, when the boat is cut and dragged to follow a trajectory, the wake of the boat should modify

Figure 4: **Qualitative comparison on MC-Bench** Competing methods exhibit artifacts (red), whereas TTM achieves clean placement and appearance consistency.

Method	Training Free?	CTD$_{\downarrow}$	BG–Obj CTD$_{\uparrow}$	Dynamic Degree$_{\uparrow}$	Subject Consistency$_{\uparrow}$	Background Consistency$_{\uparrow}$	Motion Smoothness$_{\uparrow}$	Aesthetic Quality$_{\uparrow}$	Imaging Quality$_{\uparrow}$
SVD-Based Models									
DragAnything	✗	10.645	**50.885**	**0.981**	0.956	0.942	0.983	0.531	0.554
SG-I2V*	✓	**5.796**	12.042	0.803	0.976	0.953	0.991	0.553	0.621
MotionPro	✗	8.685	24.485	0.422	**0.979**	**0.975**	**0.993**	**0.559**	**0.617**
Ours	✓	7.967	35.340	0.427	**0.979**	0.967	**0.993**	0.548	**0.617**
CogVideoX-Based Models with Longer Generated Videos									
GWTF$_{\gamma=0.7}$	✗	32.548	86.614	0.736	0.963	0.965	0.989	0.517	0.539
GWTF$_{\gamma=0.5}$	✗	27.844	**87.708**	**0.764**	0.958	0.963	0.988	0.513	0.539
Ours	✓	**13.665**	70.608	0.357	**0.980**	**0.977**	**0.995**	**0.531**	**0.579**

Table 1: **Quantitative results on MC-Bench object motion control.**

accordingly, even though it was not directly manipulated. With a single timestep, SDEdit cannot accommodate this asymmetry. If t^* is small, the denoised video adheres closely to the warped signal but inherits artifacts such as frozen backgrounds (top row). If t^* is large, the results look realistic but drift away from the intended motion (second row). We therefore conjecture that different regions require different effective noising levels: masked regions demand *strong adherence* to the motion signal, achieved with less noising (t_{strong}), while unmasked regions benefit from *weaker enforcement*, achieved with increased noising (t_{weak}).

The challenge is that standard pretrained diffusion models assume inputs corrupted by a single uniform noise level cannot directly accommodate region-dependent noising, shifting the input distribution off-manifold. To overcome this, we propose *dual-clock denoising*. Given a mask M, we noise the warped video reference V^w to timestep t_{weak} and initialize the denoising process. At each denoising step t with $t_{\text{strong}} \leq t < t_{\text{weak}}$, we override the masked region with the corresponding region of the warped video noised to $t-1$. This constrains the masked regions to follow the intended trajectory, while the background is free to denoise more aggressively and achieve realism. Once $t = t_{\text{strong}}$, we stop overriding and continue the standard sampling process, allowing the model to refine both regions for a coherent result. Let x_t denote the noisy sample at timestep t, and $\hat{x}_{t-1}(x_t, t)$ the denoiser prediction. T he update rule is

$$x_{t-1} \leftarrow (1-M) \odot \hat{x}_{t-1}(x_t, t, I) + M \odot x^w_{t-1},$$

where x^w_{t-1} is the warped reference video noised to timestep $t-1$.

Efficiency and Applicability. Our method is a lightweight modification to standard sampling that adds no extra computation over regular video diffusion and is in fact computationally faster than vanilla inference, since TTM runs the core denoising process only up to $t_{\text{weak}} < T$ instead of all T steps, while the remaining motion and masking operations incur negligible overhead. It is entirely training-free and plug-and-play for image-conditioned I2V models; In experiments, it integrates with three backbones, demonstrating broad applicability.

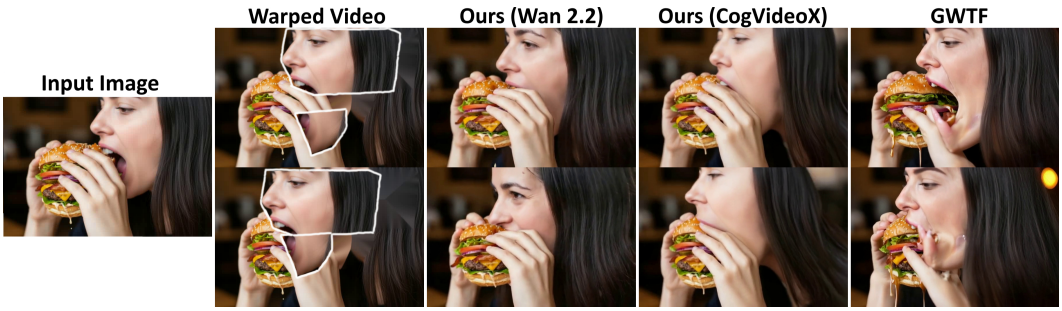


Figure 5: **Comparison on a challenging cut-and-drag example.** GWTF exhibits strong artifacts under large motion (right); TTM follows the prescribed motion realistically across various models.

Method	MSE$_\downarrow$	FID$_\downarrow$	LPIPS$_\downarrow$	SSIM$_\uparrow$	CLIP Cons.$_\uparrow$	Optical flow$_\downarrow$
GWTF$_{\gamma=0.5}$	0.033	25.990	0.371	0.526	0.981	76.714
GWTF$_{\gamma=0.7}$	0.042	28.483	0.370	0.410	**0.985**	81.738
Warped	0.025	33.443	0.339	0.560	0.981	65.494
Ours	**0.022**	**21.966**	**0.332**	**0.586**	0.983	**60.558**

Table 2: **Quantitative results on DL3DV camera motion control.**

4 Experiments

We evaluated TTM in three complementary settings: single-object motion control (Sec. 4.1), camera motion control (Sec. 4.2), and joint motion–appearance editing (Sec. 4.3). These cover the primary modes of user intent: animating a selected object, inducing global motion via viewpoint changes, and modifying the appearance of scene elements. For the first two, we report quantitative benchmarks and qualitative comparisons against state-of-the-art training-based and training-free baselines. For appearance editing, where no standard benchmark exists, we present qualitative results highlighting capabilities unique to our approach. We also demonstrate plug-and-play generality across multiple I2V backbones (Sec. 4.4) and analyze the dual-clock schedule via ablations (App. A). Demonstrations are included in anonymous demo page.

4.1 Object Motion Control

We evaluate TTM for object-level motion control. The inputs are a single source image, a binary mask of the target object, and a 2D trajectory defining the desired motion. We benchmark on MC-Bench (Zhang et al., 2025b) under its official protocol. Notably, the benchmark masks are coarse human-annotated brush regions rather than pixel-accurate segmentations, effectively simulating realistic and noisy user-provided inputs. Further details of the evaluation protocol and implementation are provided in App. D.1.

Baselines. We compare against both training-based methods—DragAnything (Wu et al., 2024), MotionPro (Zhang et al., 2025b), and Go-With-the-Flow (GWTF) (Burgert et al., 2025)—and the training-free SG-I2V (Namekata et al., 2024). Results are grouped by backbone: SVD (hybrid conv/attention, $\sim$1.5B parameters) and CogVideoX (Diffusion Transformer, 5B parameters). We apply our backbone-agnostic method to both architectures, denoting them as TTM$_{\text{SVD}}$ and TTM$_{\text{Cog}}$. For fairness, we report GWTF with both recommended noise-degradation values ($\gamma \in \{0.5, 0.7\}$).

Metrics. We evaluate motion adherence and perceptual quality. For adherence, we use MC-Bench's CoTracker Distance (CTD) for object trajectories and BG–Obj CTD to detect unintended background co-motion. For perceptual quality, we adopt VBench (Huang et al., 2024), a reference-free suite of automated video metrics. See App. D.1 for more details.

Results. Tab. 1 summarizes the results. Across both backbones, TTM attains the lowest CoTracker distance (best adherence to the prescribed motion), excluding SG-I2V. On SVD, our VBench quality matches MotionPro, with minor metric trade-offs, and surpasses DragAnything and SG-I2V on

Figure 6: **Qualitative comparison of camera-motion control.** GWTF drifts from the target camera path, while TTM leverages the warped reference to enforce motion, yielding smooth, artifact-free results beyond simple depth warping.

most measures. TTM's dynamic degree is lower than DragAnything and SG-I2V: we attribute this to DragAnything often inducing unintended scene motion and local deformations (Fig. 4), whereas SG-I2V frequently triggers camera co-motion, moving the whole scene rather than just the object (e.g., a rightward pan in the same figure, where the camera shifts right and the moon exits the frame). This effect—also noted by Burgert et al. (2025)—is reflected in SG-I2V's substantially lower BG–Obj CTD, indicating strong object–background co-motion. On the CogVideoX backbone, TTM achieves substantially stronger adherence to motion conditioning and higher scores on nearly all video-quality metrics compared to GWTF. The only exception is the "dynamic" score, where GWTF reports higher values; however, these gains often come at the cost of scene deformations and inconsistencies, as evident from the background- and subject-consistency metrics in Tab. 1 and in Fig. 5. Overall, TTM exceeds the performance of both training-based and training-free baselines on most metrics, while remaining entirely training-free.

Qualitative Examples. In Fig. 4, we present a representative example from the MC-Bench benchmark, using SVD as the common I2V backbone. Competing methods introduce noticeable artifacts (highlighted in red), while our TTM produces clean foreground placement at the intended location and preserves fidelity to the first-frame appearance. Additional videos and benchmark results are provided in Fig. 5, in App. E and in our anonymous demo page.

4.2 Camera Motion Control

We evaluate TTM on synthesizing realistic videos from a single image under prescribed camera motion. Following GWTF, we use a subset of DL3DV-10K (Ling et al., 2024), which contains static-scene videos with per-frame camera annotations. From the first frame, we estimate metric depth with DepthPro (Bochkovskii et al., 2025), back-project to a 3D point cloud, and reproject along the prescribed motion to construct a reference video. Pixels that are not assigned a value by the warp (i.e., holes) are filled by copying the color of the nearest valid warped pixel. The collection of these inpainted pixels constitutes the mask M. We evaluate 150 sequences with 49 target views each, comparing generated results against the original frames at the same viewpoints. Further details appear in App. D.

Baselines. We benchmark against GWTF, the leading prior method for camera-motion control. The protocol for constructing depth-based warped videos is identical to that used in our approach; however, GWTF further extracts optical flow from them to synthesize noise warping.

Metrics. With ground-truth videos available, we evaluate frame-level alignment using MSE, LPIPS (Zhang et al., 2018), and SSIM (Wang et al., 2004). Motion consistency is assessed by the MSE between RAFT-estimated optical flows (Teed & Deng, 2020) of generated and ground-truth videos. Distributional similarity is assessed with FID (Heusel et al., 2017) between all generated and original frames. Temporal consistency is measured as the average CLIP (Radford et al., 2021) cosine similarity between consecutive generated frames.

Results. We compare against baselines in Tab. 2, Fig. 6 and on our project webpage. Our method delivers the best camera-motion control, outperforming baselines in motion fidelity and pixel qual-

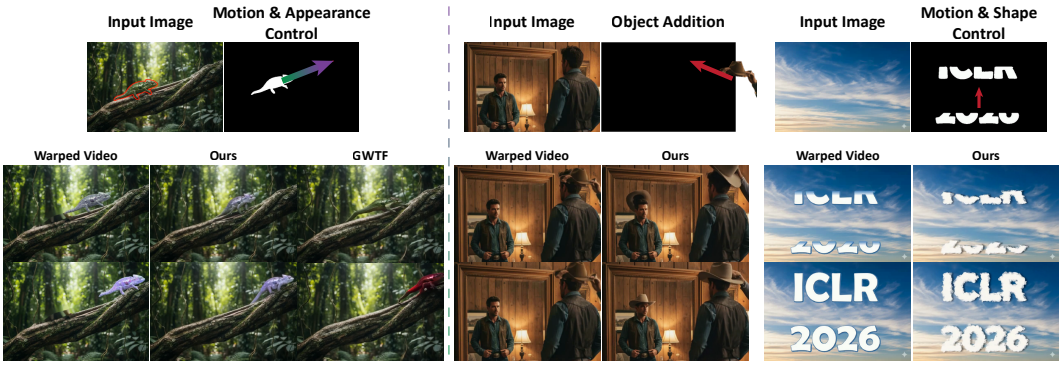


Figure 7: **Joint motion and appearance control.** TTM leverages a user-provided warped reference to control both motion and per-pixel appearance in diverse tasks, with per-task details given in 4.3.

ity: vs. the best GWTF variant, pixel MSE drops by $\mathbf{33}\%$ and FID by $\mathbf{15.5}\%$; optical-flow MSE also decreases, indicating better temporal alignment across the scene.

Qualitative Examples. Fig. 6 compares TTM with GWTF on an input images and user-specified camera trajectories. GWTF struggles with long motions as it relies on noise warping for scene consistency and either drifts from the prescribed path. In contrast, TTM precisely follows the target camera motion and preserves identity across frames, yielding smooth, realistic sequences. Depth warping is shown as coarse guidance; TTM removes its tearing and holes while retaining the intended motion. Additional results appear in App. E.2 and anonymous demo page.

4.3 Appearance Control

Beyond motion, TTM enables pixel-level appearance specification across the scene. By conditioning on full reference frames, the crude animation constrains both *where* objects move and *how* they look. Users can guide motion and evolving appearance jointly, without retraining or additional cost. In contrast, prior methods rely on trajectories and text alone, limiting them to ambiguous appearance changes. In Fig. 7 we illustrate three setups: (i) *Motion and appearance control*: a chameleon follows a user-drawn trajectory while changing from green to purple. For comparison, GWTF is run with optical flow and a text prompt describing the desired color (see App. D.3); our method preserves both motion and appearance, whereas GWTF fails to satisfy both constraints. (ii) *Object insertion*: conditioning on full frames allows adding new objects. We place a hat on a cowboy looking in a mirror; the hat blends naturally into the scene and appears consistently in the reflection. (iii) *Joint motion and shape control*: TTM preserves the intended graphic deformations while harmonizing appearance with the scene as clouds are revealed.

4.4 Plug-and-Play Model Adaptation

With video generators evolving rapidly and parameter counts rising, adding motion control *without retraining* becomes especially valuable. Beyond SVD and CogVideoX, TTM applies *as is* to any image-to-video diffusion model. We demonstrate this on the recently released WAN 2.2[1] (14B parameters) (Wan et al., 2025): with only a brief adaptation, TTM enables both local object control and explicit camera-motion conditioning. In Figs. 1, 5, and 6, as well as on the demo page, we present a set of challenging examples. By contrast, GWTF achieves motion control only after fine-tuning CogVideoX-5B with warped-noise training, requiring $\sim$7,680 A100-80GB GPU-hours.

5 Cut-and-Drag GUI

To make Time-to-Move accessible beyond scripted experiments, we provide a lightweight "cut-and-drag" user interface that, given an input image, creates the warped reference video and the mask

[1] https://github.com/Wan-Video/Wan2.2?tab=readme-ov-file

video annotated by the user. Given an input frame, the user selects one or more objects by drawing polygons around them, or by using a single click with Segment Anything (SAM) (Kirillov et al., 2023) for automatic full-object masks. Each selected region can then be dragged over time in a sequence of segments, with per-segment controls for rotation, uniform scaling, and simple hue-based recoloring. The GUI interpolates these user-defined key poses into a crude warped animation and automatically builds the corresponding binary mask sequence. External images can also be imported and animated with the same cut-and-drag operations, enabling object insertion. Fig. 8 provides an illustration of the GUI's workflow.

5.1 Warping

We parameterize the motion in a forward manner, while rendering the warped video with standard backward warping. Let $M_0 \in \{0,1\}^{H\times W}$ be the initial object mask, either defined by the user using the interactive GUI or taken directly from MC-Bench. The dragging interaction induces, for each frame $t \in \{0,\dots,F-1\}$, a 2D transform ϕ_t acting on pixel coordinates $x \in \mathbb{R}^2$. In the MC-Bench setting this transform reduces to a pure translation with displacement $\Delta_t \in \mathbb{R}^2$,

$$\phi_t(x) = x + \Delta_t.$$

The warped guidance video is then obtained by sampling from the input image at inverse-transformed coordinates,

$$V_t^w(x) = I\big(\phi_t^{-1}(x)\big),$$

for pixels belonging to the moving foreground, while background pixels retain their original value. To handle occlusions and disocclusions, we remove the masked region from I and fill it once using a simple nearest-neighbour inpainting procedure that propagates nearby background colors. The moving foreground "sprite" extracted from I is then rendered at each frame t using ϕ_t and composited onto this background. For general interactive use, the same mechanism extends beyond pure translation: rotation and uniform scaling correspond to using ϕ_t as a similarity transform rather than just a shift, but the backward-warp-and-composite procedure remains unchanged. In practice, the user specifies only a few key poses in the GUI, and the intermediate transforms ϕ_t are interpolated so that the sprite motion and the resulting warped video evolve gradually.

6 Conclusions, Limitations and Future Directions

We introduced a training-free framework for motion and appearance control in I2V diffusion models. By extending the SDEdit principle to videos, we treat warped reference animations as direct motion guidance, while image conditioning preserves fidelity to the input. To balance strict adherence in user-specified regions while enabling natural adaptation in the remaining regions, we proposed region-dependent dual-clock denoising, a plug-and-play strategy that produces realistic and faithful generations. Our method has several limitations. Although our framework adapts seamlessly to different I2V backbones, the dual-clock scheme still requires tuning of $(t_{\text{weak}}, t_{\text{strong}})$ for each model. Identity preservation is restricted to content visible in the first frame; objects entering later cannot be anchored beyond what is implicitly recovered during denoising. Finally, our framework requires full object masks when specifying motion, unlike some motion-prompting methods that are explicitly trained to operate from partial markings. Nevertheless, our method remains robust to imperfect masks, as demonstrated in MC-Bench. For camera-motion control, as in other recent free-form motion-control methods, our setup relies on an off-the-shelf monocular depth estimator to successfully construct the crude 3D animation. Our framework accommodates extensions beyond our current implementation. In particular, the dual-clock scheme could be generalized to support multiple regions, soft masks, or smoother noise schedules, offering more fine-grained control. We leave richer appearance edits (e.g., stylistic transformations), exploration of alternative warping schemes, articulated motion, and long-horizon video generation for future work.

7 Ethics Statement

We affirm adherence to the ICLR Code of Ethics. Our work uses publicly available benchmarks (e.g., MC-Bench, DL3DV) and author-crafted synthetic examples that were released for research; we did not collect new personal data. Where underlying datasets may include human subjects, consent

and licensing follow the original publications. We will release code and configuration sufficient to reproduce results without redistributing third-party imagery.

Because our method enables fine-grained motion and appearance control, it could be misused to create misleading or harmful content (e.g., deepfakes). We caution that deployments should follow ethical standards and applicable laws, and we recommend safeguards such as provenance/watermarking hooks, content filtering, and clear usage guidelines that prohibit impersonation and privacy violations. Outputs may reflect biases in upstream backbones and prompts; we avoid sensitive-attribute claims and encourage task-specific bias checks before high-stakes use. The authors report no conflicts of interest. In the process of writing this paper, we used the aid of Large Language Models (LLMs) to assist and polish writing.

8 Reproducibility statement.

This work is fully reproducible. Our method is fully specified in Sec. 3, with evaluation protocols in Sec. 4 and implementation details in App. D.1. We provide an anonymized code release for the WAN 2.2 implementation in the supplementary materials, together with exemplar configuration files. For SVD, CogVideoX, and other baselines we rely on publicly available checkpoints.

References

Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18208–18218, 2022.

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

Aleksei Bochkovskii, Amaël Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R. Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2410.02073.

Ryan Burgert, Yuancheng Xu, Wenqi Xian, Oliver Pilarski, Pascal Clausen, Mingming He, Li Ma, Yitong Deng, Lingxiao Li, Mohsen Mousavi, et al. Go-with-the-flow: Motion-controllable video diffusion models using real-time warped noise. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 13–23, 2025.

Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2024.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Daniel Geng, Charles Herrmann, Junhwa Hur, Forrester Cole, Serena Zhang, Tobias Pfaff, Tatiana Lopez-Guevara, Yusuf Aytar, Michael Rubinstein, Chen Sun, et al. Motion prompting: Controlling video generation with motion trajectories. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 1–12, 2025.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 6629–6640, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

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

Yash Jain, Anshul Nasery, Vibhav Vineet, and Harkirat Behl. Peekaboo: Interactive video generation via masked-diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8079–8088, 2024.

Hyeonho Jeong, Geon Yeong Park, and Jong Chul Ye. Vmc: Video motion customization using temporal attention adaption for text-to-video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9212–9221, 2024.

Sora Kim, Sungho Suh, and Minsik Lee. Rad: Region-aware diffusion models for image inpainting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 2439–2448, 2025.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4015–4026, 2023.

Jialu Li, Shoubin Yu, Han Lin, Jaemin Cho, Jaehong Yoon, and Mohit Bansal. Training-free guidance in text-to-video generation via multimodal planning and structured noise initialization. *arXiv preprint arXiv:2504.08641*, 2025a.

Yaowei Li, Xintao Wang, Zhaoyang Zhang, Zhouxia Wang, Ziyang Yuan, Liangbin Xie, Ying Shan, and Yuexian Zou. Image conductor: Precision control for interactive video synthesis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 5031–5038, 2025b.

Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22160–22169, 2024.

Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11461–11471, 2022.

Wan-Duo Kurt Ma, John P Lewis, and W Bastiaan Kleijn. Trailblazer: Trajectory control for diffusion-based video generation. In *SIGGRAPH Asia 2024 Conference Papers*, pp. 1–11, 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. *arXiv preprint arXiv:2108.01073*, 2021.

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. *arXiv preprint arXiv:2411.04989*, 2024.

OpenAI. Gpt-4o system card, 2024. URL `https://arxiv.org/abs/2410.21276`.

Naama Pearl, Yaron Brodsky, Dana Berman, Assaf Zomet, Alex Rav Acha, Daniel Cohen-Or, and Dani Lischinski. Svnr: Spatially-variant noise removal with denoising diffusion. *arXiv preprint arXiv:2306.16052*, 2023.

Alexander Pondaven, Aliaksandr Siarohin, Sergey Tulyakov, Philip Torr, and Fabio Pizzati. Video motion transfer with diffusion transformers. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 22911–22921, 2025.

Haonan Qiu, Zhaoxi Chen, Zhouxia Wang, Yingqing He, Menghan Xia, and Ziwei Liu. Freetraj: Tuning-free trajectory control in video diffusion models. *arXiv preprint arXiv:2406.16863*, 2024.

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

Ariel Shaulov, Itay Hazan, Lior Wolf, and Hila Chefer. Flowmo: Variance-based flow guidance for coherent motion in video generation. *arXiv preprint arXiv:2506.01144*, 2025.

Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pp. 402–419. Springer, 2020.

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

Angtian Wang, Haibin Huang, Jacob Zhiyuan Fang, Yiding Yang, and Chongyang Ma. Ati: Any trajectory instruction for controllable video generation. *arXiv preprint arXiv:2505.22944*, 2025.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

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 *European Conference on Computer Vision*, pp. 331–348. Springer, 2024.

Danah Yatim, Rafail Fridman, Omer Bar-Tal, Yoni Kasten, and Tali Dekel. Space-time diffusion features for zero-shot text-driven motion transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8466–8476, 2024.

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. *arXiv preprint arXiv:2308.08089*, 2023.

Shoubin Yu, Jacob Zhiyuan Fang, Jian Zheng, Gunnar Sigurdsson, Vicente Ordonez, Robinson Piramuthu, and Mohit Bansal. Zero-shot controllable image-to-video animation via motion decomposition. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 3332–3341, 2024.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric . In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 586–595, Los Alamitos, CA, USA, June 2018. IEEE Computer Society. doi: 10.1109/CVPR.2018.00068. URL `https://doi.ieeecomputersociety.org/10.1109/CVPR.2018.00068`.

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, 2025a.

Zhongwei Zhang, Fuchen Long, Zhaofan Qiu, Yingwei Pan, Wu Liu, Ting Yao, and Tao Mei. Motionpro: A precise motion controller for image-to-video generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 27957–27967, 2025b.

Haitao Zhou, Chuang Wang, Rui Nie, Jinlin Liu, Dongdong Yu, Qian Yu, and Changhu Wang. Trackgo: A flexible and efficient method for controllable video generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 10743–10751, 2025.

A Ablation Study: Dual-Clock Denoising

We ablate the dual-clock denoising scheme, presented in 3.3, using the same evaluation protocol described in 4.1. In TTM, the *first tick* t_{weak} sets the initialization noise level for sampling, while the *second tick* t_{strong} sets when we stop overriding the masked part with the noisy warped reference; after this point, all pixels denoise together. For this section, we evaluate this procedure under different settings. In these experiments, we use different timing ticks, denoting the first as t_1 and the second as t_2

First tick (t_1)	Second tick (t_2)	CoTracker distance	Dynamic degree	Imaging quality
t_{weak}	t_{weak}	27.316	0.265	0.623
t_{strong}	t_{strong}	5.528	0.353	0.620
T	0	2.954	0.411	0.578
t_{weak}	0	2.923	0.404	0.576
t_{strong}	0	2.942	0.353	0.579
T	t_{weak}	29.399	0.254	0.622
T	t_{strong}	9.228	0.430	0.615
t_{weak}	t_{strong}	7.967	0.427	0.617

Table 3: **Dual-Clock Ablation.**

The resulting behaviors under different settings, together with their quantitative outcomes, are summarized below and in Table 3:

Single-clock baseline ($t_1 = t_2$). This implies applying SDEdit on the warped video ($t_{\text{weak}} = t_{\text{strong}}$). When $t_1 = t_2 = t_{\text{weak}}$, too little conditioning is induced: the CoTracker distance is high, reflecting poor motion adherence. When $t_1 = t_2 = t_{\text{strong}}$, non-masked regions become over-constrained to unintended motion, suppressing dynamics (e.g., the background freezes); see Fig. 3, where the boat's foam remains static although the boat moves.

RePaint-style ($t_2 = 0$). Here denoising occurs only outside the masked reference (equivalent to RePaint). As expected, for any t_1 the CoTracker distance drops sharply, since the warped masked region is injected throughout denoising. However, this comes at the cost of Imaging quality: the videos appear nearly perfect in motion adherence but unnatural overall, due to the lack of flexibility inside the mask region.

Unconstrained background ($t_1 = T$). No constraint is applied to non-masked regions. For $t_2 = t_{\text{weak}}$, motion is not enforced and the model tends to generate overly static videos. For $t_2 = t_{\text{strong}}$, performance improves, but tracking error remains unsatisfactory; in practice, this setup often produces duplicate copies of the source object, which harms adherence.

Dual clock (ours). $t_1 = t_{\text{weak}}$, $t_2 = t_{\text{strong}}$. This setting achieves the best overall trade-off, combining strong motion-conditioning adherence (low CoTracker distance) with higher dynamic degree and robust visual quality.

B Ablation Study: Mask Perturbations

Our object-control evaluation on MC-Bench in Sec.4 implicitly measures robustness to inaccurate masks: the dataset provides human-annotated brush masks that are coarse, include background regions, and often miss fine object details, rather than pixel-accurate segmentations. To further validate this, we add an experiment that explicitly perturbs the input masks and reports the resulting performance, complementing the Sec. 4.1. We follow the same experimental setup, but apply morphological erosion and dilation with varying kernel sizes to the MC-Bench masks, simulating under- and over-segmented masks with different boundary characteristics. As shown in Tab. 4, these perturbations lead to only minor changes in all metrics, indicating that our method is robust to mask inaccuracies.

Morpholigcal Operation	Kernel Size	CoTracker distance	Dynamic degree	Imaging quality
-	-	7.967	0.427	0.617
Dilate	3	8.059	0.435	0.618
Dilate	5	8.527	0.425	0.618
Dilate	7	7.898	0.438	0.617
Erode	3	8.499	0.416	0.618
Erode	5	8.276	0.430	0.617
Erode	7	9.125	0.433	0.617

Table 4: **Mask Perturbations Ablation.**

C Ablation Study: Timestep Sensitivity

Tuning a small set of inference-time parameters is a common requirement in diffusion-based methods, analogous to adjusting the guidance scale in classifier-free guidance (CFG) (Ho & Salimans, 2022) or selecting a timestep schedule in SDEdit. To characterize the sensitivity of TTM to its timestep parameters, we perform an ablation in which we vary t_{weak} and t_{strong} around their default values on MC-Bench and measure the resulting performance. As summarized in Tab. 5, smaller t_{strong} values increase motion adherence and reduce the CoTracker distance, but slightly degrade imaging quality, as the object becomes more rigid and less free to adapt. Conversely, larger t_{weak} values (i.e., more initial noise) generally lead to a higher dynamic degree. Overall, the trends are smooth, and our default settings lie in a stable operating regime that provides a favorable trade-off between motion control and visual fidelity.

t_{weak}	t_{strong}	CoTracker distance	Dynamic degree	Imaging quality
38	27	11.571	0.419	0.620
36	27	10.596	0.454	0.619
34	27	9.546	0.416	0.619
38	25	8.576	0.433	0.617
36	25	7.967	0.427	0.617
34	25	8.031	0.419	0.618
38	23	6.500	0.438	0.612
36	23	6.757	0.419	0.612
34	23	6.130	0.414	0.615

Table 5: **Timestep Sensitivity Ablation.** The original experiment used $t_{weak} = 36$ and $t_{strong} = 25$

D Implementation Details

D.1 Object Motion Control

This subsection complements Sec. 4.1 with concise protocol and implementation details:

- **Single-Trajectory.** To avoid ambiguity stemming from masks linked to multiple objects/trajectories in the original MC-Bench dataset, we restrict evaluation to single-trajectory cases (over 91% of the dataset).
- **Input handling.** Inputs are resized to each model's native size and padded to match aspect ratio; after generation, padding is removed and outputs are resized back. Exceptions: MotionPro uses its original benchmark pipeline; DragAnything is run with its default input handling (we observed best results without external resizing).
- **Trajectory scaling**: the 2D trajectory points are affinely transformed with the *same* resize-and-pad mapping applied to the frames. After generation, we remove padding and invert the scaling when mapping tracks back for evaluation, ensuring geometric consistency.
- **Clip length.** Standardized to 16 frames for SVD-based methods (as Zhang et al. (2025b)) and 49 frames for CogVideoX-based methods. Concretely, SVD emits 14 or 25 frames—thus TTM_{SVD} generates 25 and keeps the first 16; DragAnything emits 20 and we keep the first 16; SG-I2V produces 14 and we evaluate the native 14-frame output, which may be slightly favorable to its metrics. If the output has more frames than the provided trajectory, we trim the trajectory to the target length.
- **Pre/post-processing and prompts.** SG-I2V is conditioned on bounding boxes rather than masks, unlike the other methods. Therefore, following Burgert et al. (2025), we supply the tight bounding box of the provided mask. For prompts, SVD-based methods are text-free, while CogVideoX-based methods use the MC-Bench prompts.
- **Mask resizing**: Since both SVD and CogVideoX operate in a downsampled latent space, we project the binary masks to the latent resolution with nearest-neighbor interpolation. For SVD this is spatial-only; for CogVideoX also subsample in time to match temporal compression (nearest-neighbor in time).
- **Hyperparameters.** All methods use T=50 denoising steps. For TTM_{SVD} set $(t_{\text{weak}}, t_{\text{strong}}) = (36, 25)$ and fix MotionPro's motion bucket to 17 (as in their release). For TTM_{Cog} use $(46, 41)$. Other run-time settings follow each method's defaults.

- **VBench.** For CogVideoX-based 49-frame models, we use the long benchmark variant[2].
- **Dynamic Degree.** VBench flags a clip as *dynamic* when the mean of the top 5% RAFT flow magnitudes in a frame exceeds a resolution-scaled threshold $\alpha \cdot \frac{\min(H,W)}{256}$ in at least 25% of sampled frames. The default $\alpha = 6.0$, tuned for VBench's source videos, is too strict for our MC-Bench setting—predominantly static camera with small, localized motions—so nearly all clips are marked static. We therefore set $\alpha = 3.5$ (keeping the 25% rule unchanged), which yields a more meaningful separation of dynamic vs. static.
- **Background–Object CoTracker Distance (BG–Obj CTD).** This metric measures whether the background unintentionally moves together with the controlled object. We run CoTracker on the generated video for both (i) the tracked object trajectory and (ii) a uniform 16×16 grid of points sampled in the first frame. Let $o_t \in \mathbb{R}^2$ denote the object's tracked position at frame t, and $p_{j,t} \in \mathbb{R}^2$ the tracked position of grid point j at frame t. We convert all tracks to displacements from frame 1: $\Delta o_t = o_t - o_1$, $\Delta p_{j,t} = p_{j,t} - p_{j,1}$. For each frame $t \geq 2$ and grid point j, we compute $d_{j,t} = \|\Delta p_{j,t} - \Delta o_t\|_2$ (pixels). The BG–Obj CTD is then the average over frames and grid points:

$$\text{BG–Obj CTD} = \frac{1}{(T-1)J} \sum_{t=2}^{T} \sum_{j=1}^{J} d_{j,t}.$$

  Higher values indicate stronger object–background disentanglement (less co-motion).

D.2 Camera Control on DL3DV

This subsection provides additional details for Sec. 4.2. For our camera control experiments, we use a subset of the DL3DV-10K dataset. The reference warped videos are created at a resolution of 960p using PyTorch3D.

We utilize the official DL3DV camera transition data and align the coordinate systems with a sign flip of the z-axis and a flip of the camera pitch due to convention differences between PyTorch3D and NerfStudio, which was used to create the DL3DV dataset. The point cloud is generated in the original camera frame, with the camera extrinsics derived from parameter estimations and the estimated transition of the first frame. To resolve the inherent depth ambiguity, we perform a binary search to find the transition scale that maximizes the MSE alignment between the warped and original videos. This aligns the transitions to a metric scale consistent with the output of DepthPro.

To select a robust subset for evaluation, we first filter out videos from the 10K subset with an estimated scale of less than 0.3, as these were found to exhibit minimal real camera movement. We then select the 150 scenes with the lowest MSE loss between the warped and ground truth videos. The masks used for the TTM process are generated by first marking pixels with no point cloud contribution as regions denoised freely from t_{weak} and all other pixels are regions denoised freely from t_{strong}. To ensure only regions with dense point cloud data are used for guidance, we apply a morphological "open" operation to the mask using an kernel size of 5. This operation serves to remove isolated noise and expand the non-guidance areas, resulting in a cleaner, more reliable mask. For text guidance, we automatically generate a text prompt for each scene using GPT-4o (OpenAI, 2024), following CogVideoX's protocol[3].

D.3 Appearance Control

For the chameleon example demonstrating joint motion and appearance control, we use the prompt: "A realistic video of a four-legged chameleon walking slowly and naturally from left to right along a thick, textured vine in a lush jungle. Its limbs move in a coordinated, controlled reptilian gait as it adjusts its body to the curve of the vine. The chameleon gradually changes color from green to purple."

[2] https://github.com/Vchitect/VBench/tree/master/vbench2_beta_long

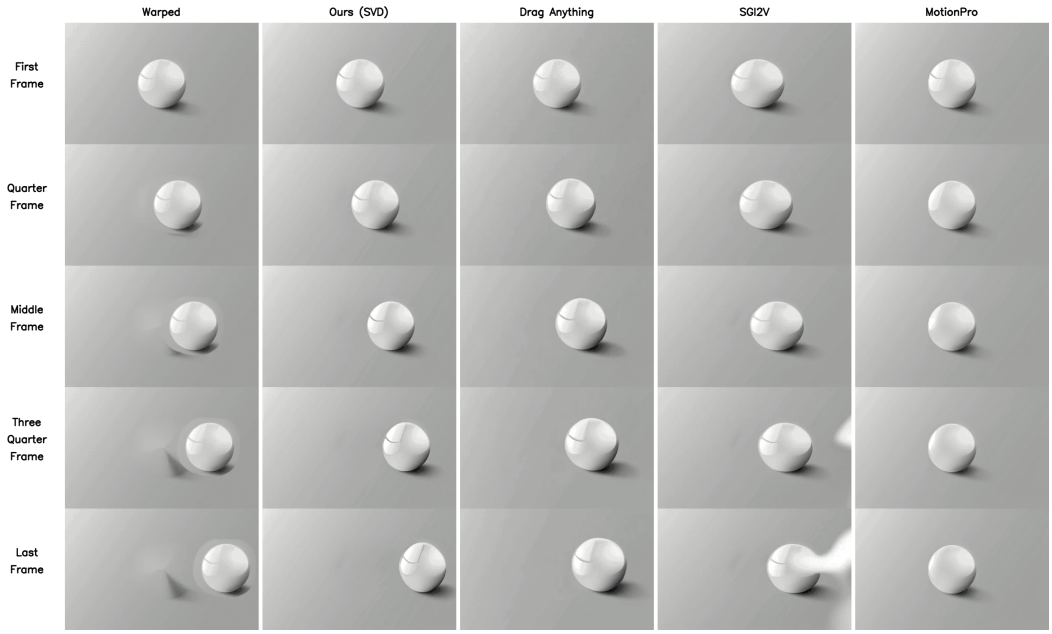
[3] https://github.com/zai-org/CogVideo/blob/main/inference/convert_demo.py

E EXTRA QUALITATIVE COMPARISONS

For the video versions of the comparisons in this paper, as well as additional results, please visit our anonymous demo page.

E.1 QUALITATIVE COMPARISONS FROM MC-BENCH

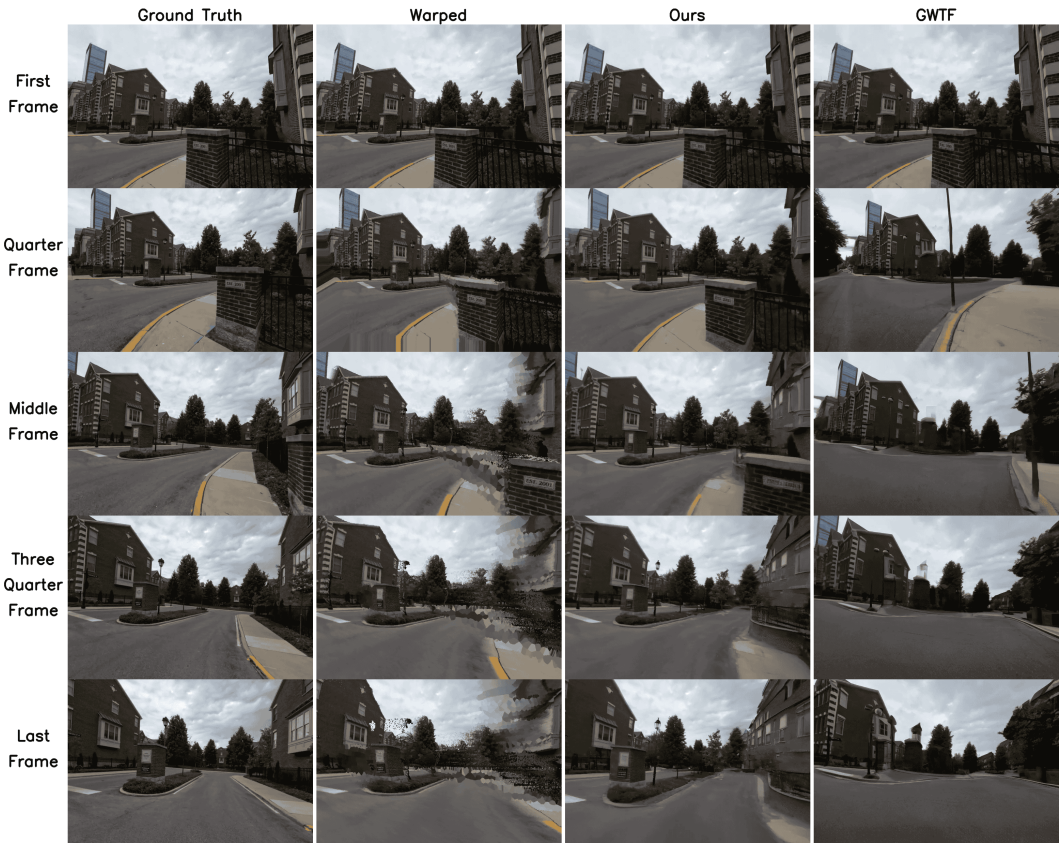
Following the experiment described in Sec. 4.1, we present additional results beyond those shown in Fig. 4, further illustrating our method's performance against leading approaches on the MC-Bench dataset using the SVD backbone.




Warped
Ours (SVD)
Drag Anything
SGI2V
MotionPro
First Frame
Quarter Frame
Middle Frame
Three Quarter Frame
Last Frame

E.2 QUALITATIVE COMPARISONS FROM DL3DV

Following the experiment described in Sec. 4.2, we present qualitative results for camera-motion control on the DL3DV benchmark, comparing our method with GWTF given an input image, its monocular depth estimate, and a depth-warped video. These examples demonstrate superior performance in maintaining the intended camera motion and overall visual fidelity.

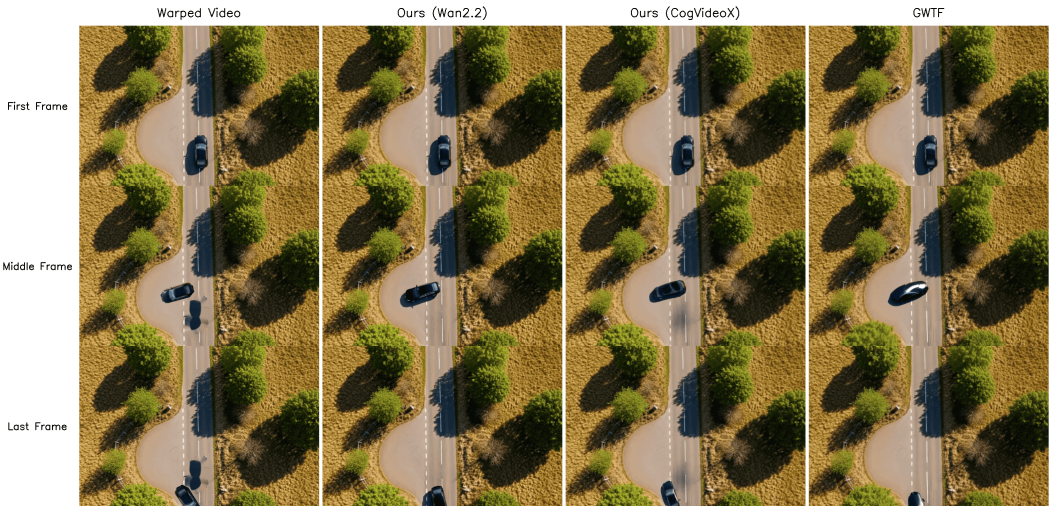

Ground Truth
Warped
Ours
GWTF
First Frame
Quarter Frame
Middle Frame
Three Quarter Frame
Last Frame

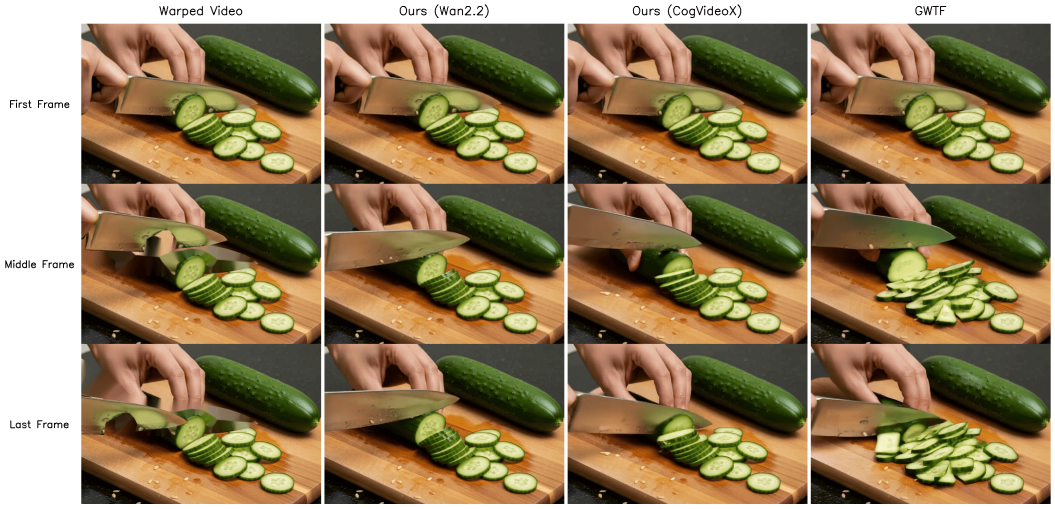

Ground Truth
Warped
Ours
GWTF
First Frame
Quarter Frame
Middle Frame
Three Quarter Frame
Last Frame

F Challenging User-Created Examples

Generation. To produce the examples shown in Fig. 5, Fig. 6, and on the demo page, we collected 53 test cases, hand-crafted by users, spanning both object-motion and camera-motion control. For each case, the initial reference frame was generated with Gemini (Comanici et al., 2025), and object-control inputs were specified via a GUI adapted from the interface introduced in (Burgert et al., 2025). We will publicly release these examples at a later date.

Additional Results As explained in Sec. 4.4, we leverage the plug-and-play nature of our method to run on the recently released WAN2.2. Below we present additional "cut-and-drag" examples that complement Fig. 5. These real-world cases illustrate scenarios in which the current state-of-the-art baseline, GWTF, often struggles to produce coherent results.




Warped Video
Ours (Wan2.2)
Ours (CogVideoX)
GWTF
First Frame
Middle Frame
Last Frame


Warped Video
Ours (Wan2.2)
Ours (CogVideoX)
GWTF
First Frame
Middle Frame
Last Frame

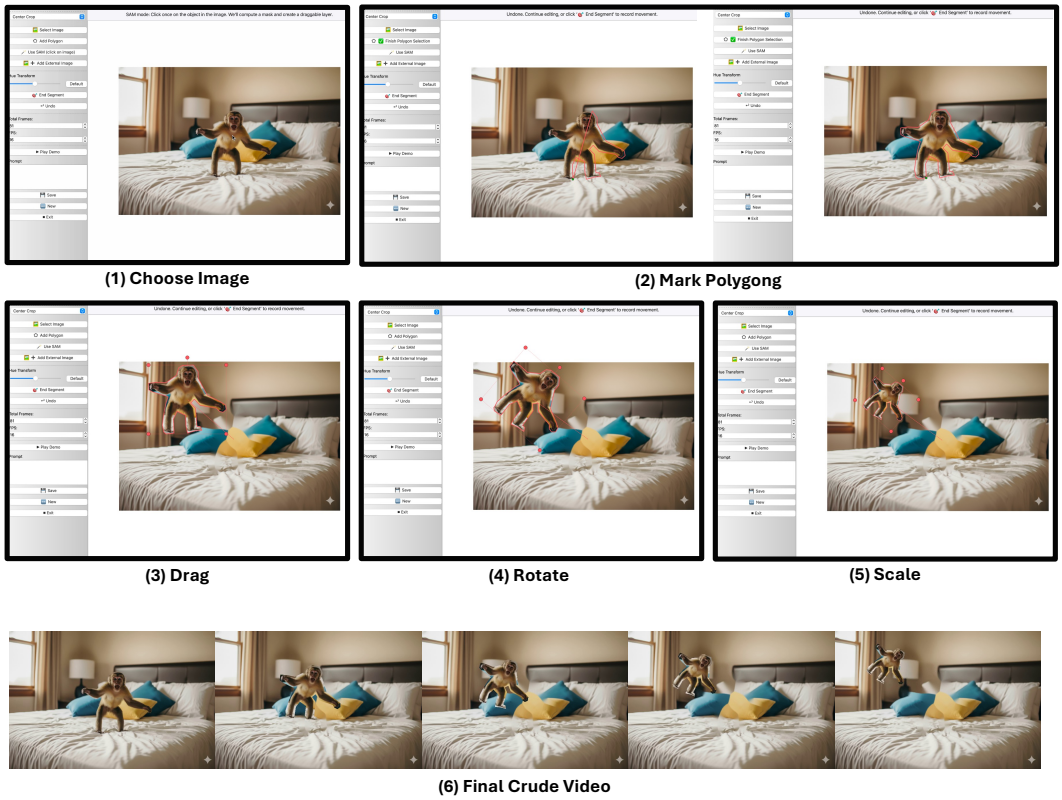


Figure 8: Example interaction with our cut-and-drag GUI. (1) The user selects an input image. (2) A region (e.g., the monkey) is defined via a polygon. (3) The object is dragged to define a motion segment. The final key pose is then refined by applying controls for (4) rotation and (5) uniform scaling. (6) The GUI interpolates these key poses, automatically generating the warped video and corresponding mask sequence, where the object gradually moves, rotates, and scales to its final position; these serve as the final motion signal for the TTM framework.