EverAnimate: Minute-Scale Human Animation via Latent Flow Restoration

Wuyang Li[*,†] **Yang Gao**[*] **Mariam Hassan** **Lan Feng**
Wentao Pan **Po-Chien Luan** **Alexandre Alahi**[†]

École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
Project Page: `https://everanimate.github.io/homepage/`

Abstract

We propose **EverAnimate**, an efficient post-training method for long-horizon animated video generation that preserves visual quality and character identity. Long-form animation remains challenging because highly dynamic human motion must be synthesized against relatively static environments, making chunk-based generation prone to accumulated drift: (i) low-level quality drift, such as progressive degradation of static backgrounds, and (ii) high-level semantic drift, such as inconsistent character identity and view-dependent attributes. To address this issue, **EverAnimate** restores drifted flow trajectories by anchoring generation to a persistent latent context memory, consisting of two complementary mechanisms. *(i) Persistent Latent Propagation* maintains a context memory across chunks to propagate identity and motion in latent space while mitigating temporal forgetting. *(ii) Restorative Flow Matching* introduces an implicit restoration objective during sampling through velocity adjustment, improving within-chunk fidelity. With only lightweight LoRA tuning, **EverAnimate** outperforms state-of-the-art long-animation methods in both short- and long-horizon settings: at 10 seconds, it improves PSNR/SSIM by 8%/7% and reduces LPIPS/FID by 22%/11%; at 90 seconds, the gains increase to 15%/15% and 32%/27%, respectively.

1 Introduction

Animating human characters from pose sequences is a fundamental problem in motion transfer, with broad applications in virtual avatars, content creation, and motion capture. Benefiting from the increasing capacity of video Diffusion Transformer (DiT) [1, 2], recent methods have substantially improved the realism and controllability of human animation, making it increasingly feasible to synthesize videos that are both visually plausible and temporally coherent.

Existing works [3–10] first extract abstracted motion representations (e.g., 2D skeletons) from videos to mitigate identity leakage and then use them to animate the reference image. Building upon this, some works focus on designing **enhanced motion representations** that incorporate cues such as depth, 3D pose, or human parsing maps [6, 8]. Beyond motion itself, another line of research aims to enable **more flexible controls** without pose retargeting [11, 12], including animation with large body-scale differences and spatial correspondence mismatches [13]. In addition, some works consider facial expression and audio for broader applications in short films [14].

Despite their impressive results, existing methods remain constrained to relatively short generation horizons, typically producing clips of only a few seconds. Recent works [12, 14] attempt to extend animation length through autoregressive, chunk-wise generation. However, the achievable extension

[*] Equal contribution.
[†] Correspondence to: Wuyang Li and Alexandre Alahi.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

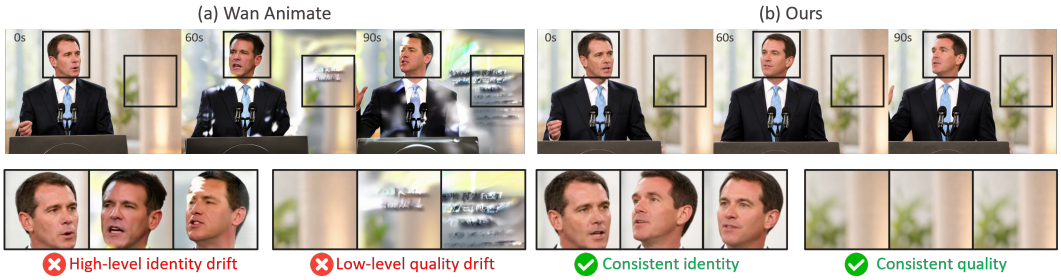


Figure 1: **(a)** Existing human animation methods primarily suffer from two types of drift: low-level quality degradation and high-level identity change. **(b)** Our method alleviates both issues, achieving stable animation. The bottom row provides zoomed-in views of the facial region and the background.

length remains limited, only producing hundreds of frames (see Fig. 7). More importantly, even with commonly adopted anti-drifting methods, such as attention sinks [15][1], error recycling [16], and sliding-window [12], these approaches still accumulate errors over time and suffer from fast quality drift. Consequently, they struggle to generate minute-level animations while maintaining visual fidelity and temporal coherence throughout the entire sequence, as shown in Fig. 1a.

To study this issue, we begin with an intuitive observation: the core challenge of long-form animation lies in the motion heterogeneity between the background and human: *Articulated human motion evolves rapidly, while much of the surrounding scene remains comparatively stable.* This heterogeneity causes two forms of drift (Fig. 1a). **(i) Low-level quality drift:** Repeated cross-chunk conditioning progressively introduces and propagates texture degradation, especially in temporally stable backgrounds. **(ii) High-level identity drift:** Semantically key attributes such as identity, facial appearance, and clothing details gradually degrade, particularly under substantial motion.

To understand these issues, we conduct an empirical analysis (see Sec. 3), revealing two main reasons. **(i)** Repeated latent-to-pixel reconstruction progressively damages visual details during cross-chunk propagation, particularly in temporally static regions, leading to quality drift. **(ii)** Limited semantic memory, e.g., attention sinks [15], is visually helpful but insufficient for reliably anchoring long horizons, leading to identity drift. Such memory acts only as a positive signal, specifying what to preserve without identifying or correcting drift. These findings suggest a latent-space principle for stable animation: *the DiT should propagate semantic memory directly in latent space across chunks, while being equipped with an intrinsic restoration ability in latent flow to correct within-chunk drift.*

Motivated by these findings, we propose **EverAnimate**, an efficient post-training framework for generating minute-scale long animation videos while preserving both visual quality and character identity (Fig. 1b). EverAnimate introduces implicit flow restoration during latent flow propagation, which is further anchored by context memory, comprising two key components. ***(i) Persistent Latent Propagation*** maintains semantic consistency across generated chunks via multi-view latent memory, thereby avoiding repeated destructive reconstruction and strengthening cross-chunk continuity. ***(ii) Restorative Flow Matching*** enables a built-in restorative ability to actively correct emerging drift implicitly without explicitly perturbing conditional images, thereby improving within-chunk visual fidelity. With only lightweight LoRA tuning, EverAnimate outperforms state-of-the-art methods on both short and long generation. In summary, the contributions of this work are as follows.

- We identify two major types of accumulated drift in long-form human animation, empirically reveal the limitations of image-space continuation and attention sinks (see Sec. 3), and derive a latent-space principle for stable long-horizon video generation.
- We propose **EverAnimate**, an efficient post-training framework for long-form human animation, built on the principle that semantic memory is propagated autoregressively across chunks, while emerging drift is corrected through intrinsic restoration.
- EverAnimate consists of two complementary components: *(i) Persistent Latent Propagation*, which strengthens cross-chunk semantic continuity through latent continuation and multi-view identity memory, and *(ii) Restorative Flow Matching*, which improves within-chunk visual fidelity by encouraging drift correction during sampling.

[1]Refer to the use of the user-provided reference frame to guide the generation of all chunks.

2 Related Work

2.1 Human Animation

Early work relied on video-to-video translation [3, 4], with motion-transfer formulations that articulated dynamics explicitly [17]. Recent works solve this by first extracting intermediate motion representation, e.g., 2D poses, and then animating a reference image with image-to-video generation, which improves visual fidelity and controllability, including AnimateDiff [5], MagicAnimate [6], and Animate Anyone [7], as well as variants that strengthen pose conditioning and temporal consistency [13, 18, 19]. Video-DiT-based approaches further scale pose-guided synthesis with unified backbones, e.g., UniAnimate-DiT [20], RealisDance-DiT [21], StableAnimator [22], and Wan-Animate [14], and other works [11]. Additionally, some works explore broader motion representations, e.g., 3D skeletons [23], multimodality [24, 25], and human parsing [6, 8]. In parallel, audio-driven body and portrait animation focuses on speech alignment and facial dynamics, with representative works like [26–28]. Despite strong clip-level quality, most methods focus on short horizons, and longer sequences are explored via pose-aware long generation [29, 30], while identity drift [31] and background degradation [32] remain challenging. In contrast, we address the train-test asymmetry in minute-scale, pose-driven animation and jointly reduce identity drift and background degradation.

2.2 Long-form Video Generation

Recent video foundation models have extended the effective temporal context through increasingly effective spatiotemporal compression [2, 33–38], by scaling the model and data scale. Nevertheless, autoregressive extrapolation beyond the training horizon still suffers from a train-test mismatch, leading to exposure bias, accumulated errors, and forgetting. A complementary line of work, therefore, studies long-horizon continuation and drift control. Early methods rely on trajectory guidance or continuation heuristics [15, 39, 40]. More recent approaches redesign the rollout procedure and training objective. Diffusion Forcing [41] and CausVid [42] bridge bidirectional and autoregressive denoising. Self Forcing [43], Rolling Forcing [44], and LongLive [45] mitigate exposure bias through self-conditioned rollouts and attention sink. FramePack [46] breaks causality by predicting the future anchor. Recently, SVI [16], Helios [47], and Matrix-Game 3.0 [48] enable extended generation through error restoration. LongCat-Video [49] incorporates video extension during pre-training, while memory-based formulations such as MALT [50], PFP [51], and WorldMem [52] preserve long-range information with latent or state-space memories. In contrast to generic long-video generation, our work targets minute-scale, pose-guided human animation, where drift arises from both the pose-conditioned motion structure and the visual synthesis process.

3 Preliminaries and Motivation

Problem Setup. Given a reference image I_{ref} and a target pose-control sequence $C_{\mathrm{pose}} = \{P_\ell\}_{\ell=1}^{T}$, pose-guided human animation aims to synthesize a video $V = \{I_\ell\}_{\ell=1}^{T}$ that follows the target poses while preserving the identity and appearance of I_{ref}. Here P_ℓ denotes the pose map at frame ℓ, which is distinct from the RGB frame I_ℓ. In DiT-based models, the video VAE encoder $\mathcal{E}(\cdot)$ maps V into latent codes $X = \mathcal{E}(V)$, and the video VAE decoder $\mathcal{D}(\cdot)$ reconstructs the video as $V = \mathcal{D}(X)$. We denote by G_θ the conditional sampling procedure induced by the DiT vector field v_θ. Then single-clip generation can be written as $X = G_\theta(\mathcal{E}(I_{\mathrm{ref}}), C_{\mathrm{pose}})$. For long-video extension, the pose-control sequence is divided into N consecutive chunks $\{C_{\mathrm{pose}}^{(n)}\}_{n=1}^{N}$, where each chunk contains L poses. The first chunk is generated from the reference image, i.e., $X^{(1)} = G_\theta(\mathcal{E}(I_{\mathrm{ref}}), C_{\mathrm{pose}}^{(1)})$. For chunk $n \geq 2$, existing methods decode the previous latent chunk $X^{(n-1)}$ into video, take its last carry-over frames $I_L^{(n-1)}$, re-encode it, and use it as the carry-over condition:

$$X^{(n)} = G_\theta\Big(\mathcal{E}(I_L^{(n-1)}), C_{\mathrm{pose}}^{(n)}\Big), \quad V^{(n-1)} = \mathcal{D}(X^{(n-1)}).$$

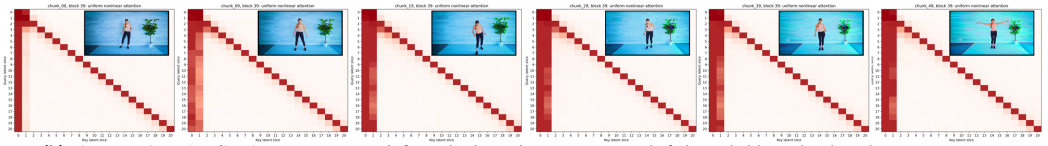
We present the single-frame carry-over case for simplicity, which can be extended to sliding windows.

Problem Analysis. We analyze the state-of-the-art method Wan-Animate [14][2] from the perspectives of VAE and DiT representations. To mitigate long-range drift, Wan-Animate introduces a persistent

[2]Empirically, we find that Wan-Animate performs best among prior animation baselines for long-range generation because of its attention-sink design; see the qualitative comparison.



(a) VAE Error Visualization: Repeated encoding & decoding introduce cumulative errors in both pure-color backgrounds and RGB frames.



(b) DiT Attention Visualization: Existing anti-drift methods, such as attention sink, fail to reliably anchor long-horizon generation.

Figure 2: **Illustration of errors in long-range animation videos.** We visualize (a) VAE round-trip reconstructions and (b) DiT self-attention maps across video chunks at different lengths. In each attention map, the 1st *col.* (highlighted in red) shows how video tokens attend to the global reference frame. *Although most tokens correctly attend to this reference (i.e., forming an attention sink [15, 45]), the generated video still exhibits progressive degradation over time, failing to anchor the long-horizon generation.* The 2nd *col.* shows attention to the last frame of the preceding chunk.

identity reference I_{ref} as an attention sink throughout chunk-wise generation. Specifically, the n-th chunk is generated under the additional condition of I_{ref}:

$$X^{(n)} = G_\theta\Big(\mathcal{E}(I_L^{(n-1)}), C_{\text{pose}}^{(n)}, \mathcal{E}(I_{\text{ref}})\Big),$$

where the re-encoded carry-over frame $\mathcal{E}(I_L^{(n-1)})$ provides inter-chunk continuity, and $\mathcal{E}(I_{\text{ref}})$ is a persistent anchor (sink). However, drift remains significant, revealing two key findings (see Fig. 2).

Finding 1: Repeated frame-level VAE round-trips inevitably accumulate drift. We first consider an idealized setting in which the DiT introduces no additional error on temporally static regions, such as the background, and predicts identical residuals across chunks. Under this assumption, any degradation can be attributed solely to the standard carry-over pipeline, which repeatedly decodes the previous latent chunk, extracts the last frame, and re-encodes it for the next chunk. In Fig. 2, this repeated VAE round-trip causes visible degradation even for static content, including both flat-color images and realistic animation frames. The error accumulates gradually, evolving from mild color distortion to obvious visual artifacts. This observation suggests that *image-space continuation is fundamentally ill-suited for long-horizon animation, and that cross-chunk semantics should instead be propagated autoregressively in latent space*. Existing bidirectional DiTs, however, do not naturally expose a persistent latent state that can be reused across chunks without additional design.

Finding 2: Attention sinks alone cannot fully prevent semantic and visual drift. We test Wan-Animate by using the persistent reference image as an attention sink [15] to globally anchor identity and appearance across chunks. However, as shown in Fig. 2(b), noticeable drift still emerges over long-horizon generation, despite most tokens correctly attending to the reference frame and forming a clear attention sink. We attribute this limitation to three factors. **(i)** A single reference frame cannot provide sufficient information, e.g., multi-view cues, required to preserve identity under substantial changes in pose and viewpoint. **(ii)** Compared with autoregressive DiTs, bidirectional DiT generation involves longer chunks and denser token interactions, which fundamentally dilute the anchoring effect of a single sink token. **(iii)** Attention sinks act only as passive reference signals: they indicate what should be preserved but lack sufficient signals to correct the trajectory once drift occurs.

Remark. These observations point to two requirements for stable long-form animation: (i) propagate cross-chunk semantics (motion/identity/appearance) directly in latent space to prevent forgetting, and (ii) actively correct drift during sampling. Accordingly, we maintain a persistent latent state with *short-term* motion memory and *long-term* identity memory, and we add an ODE-based trajectory restoration to progressively pull deviated states back on track. By avoiding repeated image-space carry-over frames [16], our method improves the anti-drifting and relieves inter-chunk flicker.

4 Method

Overview. Fig. 3 illustrates the overall workflow. During training, we use two adjacent chunks to optimize the model: $V^{(1)}$ provides context memory M_{ctx} and $V^{(2)}$ is the target chunk for the

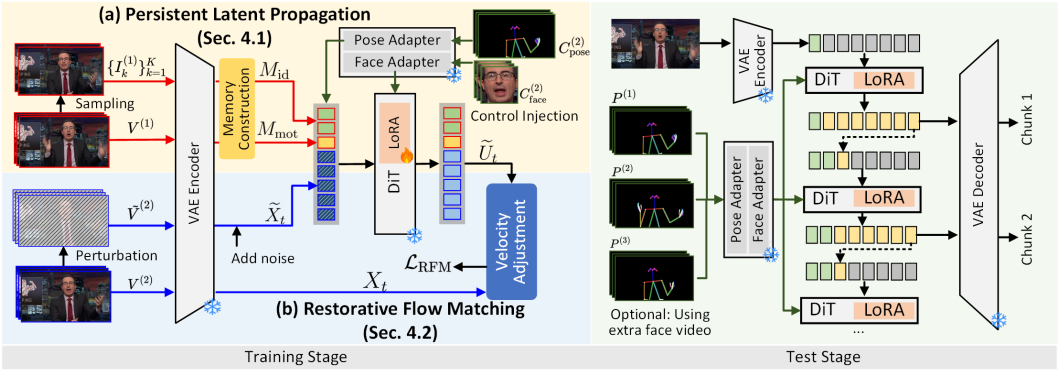


Figure 3: Overview of EverAnimate. **Train:** from a context chunk $V^{(1)}$, we extract motion/identity memories $M_{\text{id/mot}}$ and train the model to generate the next chunk $V^{(2)}$ with restorative flow matching. **Test:** We roll out chunk-by-chunk in the latent space without decoding frames between chunks.

current generation. *(a) Persistent Latent Propagation* constructs motion and identity memory that will be propagated from $V^{(1)}$ to $V^{(2)}$, preventing the high-level drift. *(b) Restorative Flow Matching* encourages the generation flow to recover once the in-chunk trajectory deviates from the clean path, solving the low-level drift. In inference, after generating the current video chunk, we reuse the video latent to guide the next-chunk generation without autoregressively decoding and encoding frames.

4.1 Persistent Latent Propagation

Given the context chunk $V^{(1)}$, we propose to establish a context memory M_{ctx} to generate $V^{(2)}$. This consists of a motion memory M_{mot} that preserves short-term temporal continuity across adjacent chunks, and a global identity memory M_{id} that anchors multi-view identity across all chunks.

Memory Construction. Let $X^{(1)}, X^{(2)} \in \mathbb{R}^{T_z \times H \times W \times C}$ be the clean video latents of the context chunk $V^{(1)}$ and the target chunk $V^{(2)}$, where T_z is the temporally compressed length produced by the video VAE $\mathcal{E}$. We extract both memories from $V^{(1)}$. The motion memory only needs to bridge adjacent chunks, so we keep the last r latent slices. The identity memory should stay useful under pose/viewpoint changes, so we encode a small set of sampled frames:

$$M_{\text{mot}} = X^{(1)}[T_z - r + 1 : T_z], \qquad M_{\text{id}} = \left\{ \mathcal{E}\left(\mathcal{T}_{\text{id}}(I_k^{(1)}) \right) \right\}_{k=1}^{K}, \tag{1}$$

where $\{I_k^{(1)}\}_{k=1}^{K} = \text{RandomSample}(V^{(1)}, K)$. We use random multi-view sampling so that, at test time, users can provide an arbitrary set of reference views while reducing systematic view-to-view bias in the identity memory. However, we find that directly using memory will *spatially affect generation with a context bias* (see Fig. 4a). To solve this, we propose a simple yet effective ***memory augmentation*** $\mathcal{T}_{\text{id}}$ that applies mild identity-preserving spatial augmentation, e.g., random translation and rescaling, in training to prevent spatial biases of memory context. This breaks the undesirable spatial association between memory and generated frames, thereby mitigating context bias (Fig. 4b).

Memory and Control Injection. We inject the memories and controls into the DiT input in two steps. First, we form the context tokens by concatenating motion/identity memories (plus a null pad to match temporal length). For Wan-style backbones, we build up the full memory as follows,

$$M_{\text{ctx}} = \text{Concat}_t(M_{\text{mot}}, M_{\text{id}}, X_{\text{pad}}), \tag{2}$$

where X_{pad} is a null latent block so that M_{ctx} has temporal length T_z. We condition the generation of $V^{(2)}$ on $C^{(2)} = \left\{ C^{(2)}_{\text{pose}}, C^{(2)}_{\text{face}} \right\}$,

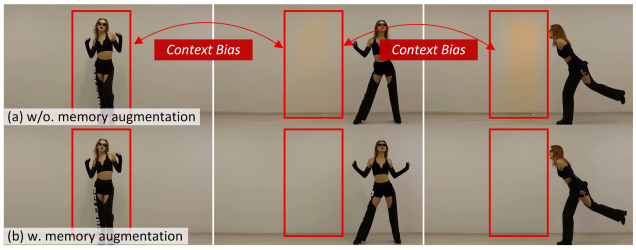


Figure 4: Effects of memory augmentation $\mathcal{T}_{\text{id}}$.

with $C^{(2)}_{\text{pose}} = \{P^{(2)}_{\ell}\}_{\ell=1}^{L}$ aligned to $V^{(2)}$ and $C^{(2)}_{\text{face}}$ the face guidance, following [14]. Second, we inject the pose into the target latent, then concatenate it with the context tokens to form the final DiT input using the pose adapter $\mathcal{E}_{\text{pose}}$. This process can be written as follows,

$$\widehat{X}^{(2)}_t = X^{(2)}_t + \mathcal{E}_{\text{pose}}\left(C^{(2)}_{\text{pose}}\right), \qquad H^{(2)}_t = \text{Concat}_{\text{ch}}\left(\widehat{X}^{(2)}_t, M_{\text{ctx}}\right). \tag{3}$$

Face guidance is encoded by a lightweight face adapter $\mathcal{E}_{\text{face}}(\cdot)$ and injected into intermediate DiT blocks via cross-attention [14]. For brevity, we write the conditioned vector field as $v_\theta(\cdot, t \mid C^{(2)})$. In the next subsection, M_{ctx} denotes the memory tokens, $\widehat{X}^{(2)}_t$ is the pose-guided target latent, and $H^{(2)}_t$ denotes the final DiT input. Then, the model aims to predict a velocity field that restores them residually for memory tokens, while the remaining tokens are used to generate the target video.

4.2 Restorative Flow Matching

Given the memory-anchored input, we train the denoising flow not only to follow the clean flow trajectory but also to recover from small intra-trajectory deviations during rollout.

Flow Matching (FM). We first recall the standard FM formulation (Fig. 5a) for the target chunk $V^{(2)}$. Since the chunk index is fixed in this subsection, we drop the superscript $\cdot^{(2)}$ for readability. Let $X_0 \sim \mathcal{N}(0, I)$ denote the Gaussian source endpoint and let $X_1 = \mathcal{E}(V^{(2)})$ denote the clean latent endpoint. Standard FM defines the linear interpolant and its target velocity as follows,

$$X_t = (1-t)X_0 + tX_1, \quad U_t = X_1 - X_0. \tag{4}$$

After memory and control injection, the DiT receives $H^{(2)}_t$ from Eq. (3). The corresponding FM objective can be written as

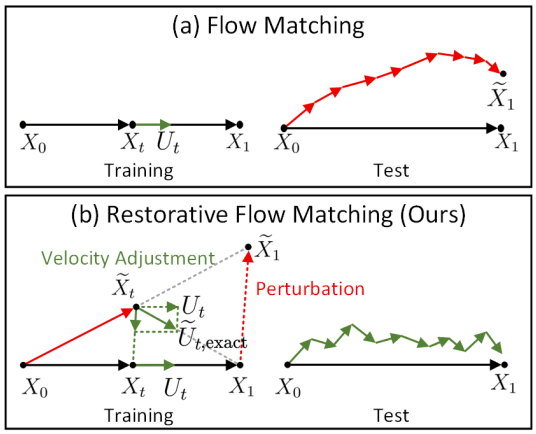


Figure 5: **RFM illustration.** Comparison between our RFM and the standard FM baseline.

$$\mathcal{L}_{\text{FM}} = \mathbb{E}\left[\left\| v_\theta\left(H^{(2)}_t, t \mid C^{(2)}\right) - U_t \right\|_2^2\right]. \tag{5}$$

This objective trains the vector field to transport Gaussian noise toward the clean data manifold under the same memory/control pathway used at inference time. It works well when the trajectory stays close to the clean path, but autoregressive long-video generation often encounters nearby yet imperfect states that standard FM does not explicitly train the model to correct.

From FM to Restorative FM (RFM) with Velocity Adjustment. During long-horizon rollout, each chunk reuses self-generated history, so small errors can propagate across chunks, leading to a drifted trajectory. Recent long-video methods [16, 47, 48, 53] simulate this mismatch by directly perturbing the autoregressive carry-over signal. We instead keep the propagated motion latent M_{mot} unchanged, simulate endpoint drift, and explicitly adjust the velocity (Fig. 5b). As a result, the model learns an intrinsic restoration ability that alleviates cross-chunk flicker. Let ξ denote a random perturbation, let $\mathcal{T}_\xi$ be a perturbation operator with $\mathcal{T}_0$ equal to the identity map, and define

$$\widetilde{V} = \mathcal{T}_\xi(V^{(2)}), \qquad \widetilde{X}_1 = \mathcal{E}(\widetilde{V}), \qquad \widetilde{X}_t = (1-t)X_0 + t\widetilde{X}_1. \tag{6}$$

We then replace the target state X_t with $\widetilde{X}_t$ while keeping the same memory and control pathways. Concretely, we form the pose-injected target latent $\widetilde{\widehat{X}}_t = \widetilde{X}_t + \mathcal{E}_{\text{pose}}(C_{\text{pose}})$ and the corresponding DiT input $\widetilde{H}_t = \text{Concat}_{\text{ch}}(\widetilde{\widehat{X}}_t, M_{\text{ctx}})$. The model is therefore exposed only to perturbed in-chunk states, rather than perturbed transmitted context, which more effectively mitigates drift.

To derive the restorative target, we ask for the unique constant velocity that transports the current perturbed state $\widetilde{X}_t$ to the clean endpoint X_1 over the remaining interval $[t, 1]$. Under the same linear-flow constraint used by standard FM, the continuation from time t to time 1 is

$$\widetilde{X}_{s,\text{con}} = \frac{1-s}{1-t}\widetilde{X}_t + \frac{s-t}{1-t}X_1, \qquad s \in [t, 1], \tag{7}$$

which satisfies $\widetilde{X}_{t,\mathrm{con}} = \widetilde{X}_t$ and $\widetilde{X}_{1,\mathrm{con}} = X_1$. Its endpoint-consistent velocity is constant:

$$\widetilde{U}_{t,\mathrm{exact}} := \frac{d\widetilde{X}_{s,\mathrm{con}}}{ds} = \frac{X_1 - \widetilde{X}_t}{1-t}. \tag{8}$$

When $\widetilde{X}_t = X_t$, namely when no perturbation is applied, this expression reduces to the standard FM velocity $X_1 - X_0$. Substituting Eq. (6) into Eq. (8) yields

$$\widetilde{U}_{t,\mathrm{exact}} = \underbrace{(X_1 - X_0)}_{U_t} + \underbrace{\frac{1}{1-t}(X_t - \widetilde{X}_t)}_{\text{restoration term}}. \tag{9}$$

Eq. (9) shows that RFM can be written as *the standard FM velocity plus a correction that pulls the perturbed state back toward the clean path.*

However, we find that the exact coefficient $\frac{1}{1-t}$ is poorly conditioned and grows rapidly as $t \to 1$ in the low-noise region, where the state is already close to the data manifold. In practice, this makes the correction term disproportionately large near the clean endpoint, leading to unstable targets, over-aggressive supervision, and potential model divergence (see the blue curve in Fig. 6). We therefore propose to *reschedule the exact coefficient* with a bounded time weight $\lambda(t)$,

$$\widetilde{U}_t = U_t + \lambda(t)(X_t - \widetilde{X}_t), \tag{10}$$

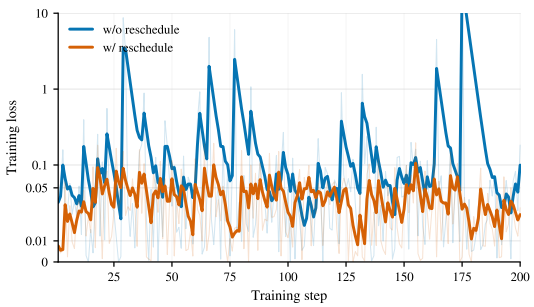


Figure 6: **Effects of the reschedule.** Comparison of the training stability with and without rescheduling $\lambda(t)$, i.e., Eq. (9) vs. Eq. (11). Our rescheduling design, shown in orange, can stabilize training.

where $\lambda(t)$ follows a bounded bell-shaped schedule. A simple choice is Gaussian rescheduling

$$\lambda(t) = \frac{\exp\!\big(-\beta(t-\frac{1}{2})^2\big) - \exp(-\beta/4)}{1 - \exp(-\beta/4)}, \qquad \beta > 0, \tag{11}$$

which peaks in the intermediate regions and smoothly decays near both ends of the trajectory. The intuition is that both extremes are less worth correcting: **(i)** in the high-noise region, the state is dominated by Gaussian noise, so the perturbation contributes little semantic signal and heavy restoration is unnecessary; **(ii)** in the low-noise region, the noise component is small and most deviation has already been corrected by earlier steps, so additional restoration is weak and can over-constrain the clean endpoint. This makes the restorative supervision strongest where perturbations are informative and numerically well-behaved, while avoiding excessive correction near the clean endpoint to stabilize training (see Fig. 6, orange curve). This also follows the loss-reweighting used in FM training [54]. Reusing the conditioning set in Eq. (3), we optimize

$$\mathcal{L}_{\mathrm{RFM}} = \mathbb{E}\left[\left\| v_\theta\big(\widetilde{H}_t, t \mid C\big) - \widetilde{U}_t \right\|_2^2\right]. \tag{12}$$

This formulation subsumes standard flow matching: when $\xi = 0$, we have $\widetilde{X}_1 = X_1$, the restorative term vanishes, and $\widetilde{U}_t = U_t$. When $\xi \neq 0$, the model is trained to follow a nearby but distorted trajectory while retaining a bounded directional pull toward the clean endpoint.

Remark. Recent long-video methods (e.g., SVI [16], Helios [47], and Matrix-Game 3.0 [48]) improve stability by perturbing the conditional image, i.e., an *input-oriented* strategy. While effective for simulating error accumulation, this corrupts the carry-over condition that should remain temporally consistent across chunks, thereby undermining stability and increasing temporal flicker. We instead adopt an *output-oriented* perspective: because per-step generation quality is determined by the latent state X_t, we preserve the propagated motion latent and apply an explicit restorative term to the in-chunk state $\widetilde{X}_t$. In this way, we correct the trajectory without sacrificing cross-chunk continuity.

4.3 Training and Inference

Progressive Training. Training is divided into two stages. *(i) Memory adaptation.* Existing animation models are adapted from image-to-video generators, which only take a single image as the condition.

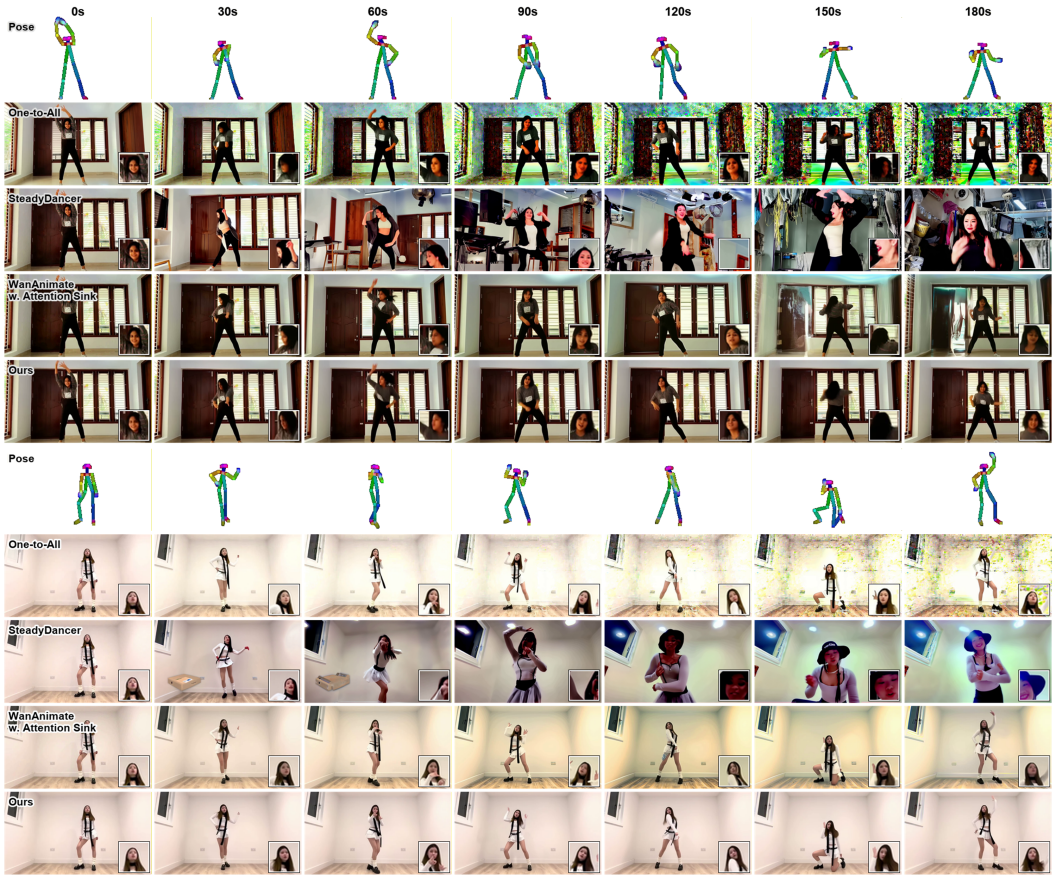


Figure 7: Qualitative comparison with state-of-the-art methods. See supplementary material for videos. The bottom-right region is cropped according to the face regions in the ground-truth videos.

We therefore first adapt the model to the memory condition by generating the current chunk using two types of memory defined in Eq. (1). In this stage, we perturb the motion memory following [16, 47] and optimize the model with standard FM loss, $\mathcal{L}_{\mathrm{FM}}$. *(ii) Anti-drift adaptation.* We then optimize the model with restorative FM loss $\mathcal{L}_{\mathrm{RFM}}$ to improve long-range stability.

Inference. Users can flexibly provide m reference frames, where $1 \leq m \leq K$, to specify the target identity and scene context. If $m = K$, these reference frames are directly used as the K-frame identity memory and are shared by all chunks. If fewer reference frames are provided, e.g., in the single-reference setting, we first generate the initial chunk $V^{(1)}$ using the available reference frame(s), and then randomly sample $K - m$ keyframes from $V^{(1)}$ to complete the identity memory. Once constructed, this identity memory remains fixed and is shared across subsequent chunks. For the first chunk $V^{(1)}$, no previous motion context is available, so the motion latent M_{mot} is zero-padded during generation. For each subsequent chunk $V^{(n)}$ with $n > 1$, the last r latent slices from the previous chunk are propagated as short-term motion memory, as defined in Eq. (1). These latent slices are directly used for conditioning, avoiding the need to decode and re-encode preceding frames.

5 Experiments

Implementation. We build our method on top of Wan-2.2-Animate [14]. During training, we set $K = 4$, corresponding to one reference frame and three additional multi-view reference images, and use $r = 1$ for short-term motion memory. The first and second training stages run for 4,000 and 1,000 iterations, respectively, with 8 GPUs. The LoRA rank and scaling factor α are both set to 128. For restorative training, we randomly apply color shift, sharpness, and saturation perturbations to the target video chunk, following Helios [47]. During inference, we use 20 sampling steps without classifier-free guidance (CFG), using only the user-given reference frame as [14].

Table 1: Quantitative comparison with state-of-the-art methods across different temporal horizons. F-PSNR denotes the face-region PSNR used to evaluate identity correctness and consistency.

	10s						30s					
Method	PSNR↑	SSIM↑	LPIPS↓	FID↓	V-MAE↓	F-PSNR↑	PSNR↑	SSIM↑	LPIPS↓	FID↓	V-MAE↓	F-PSNR↑
One-to-All	20.730	0.824	0.268	55.969	0.041	16.311	20.453	0.818	0.287	52.033	0.036	16.245
SCAIL	19.602	0.700	0.311	60.734	0.068	15.575	16.499	0.568	0.429	88.485	0.051	13.108
SteadyDancer	20.528	0.736	0.257	49.975	0.050	14.235	16.724	0.617	0.382	61.317	0.049	12.120
UniAnimate-DiT	22.773	0.882	0.192	45.350	0.044	16.401	20.908	0.857	0.238	58.852	**0.032**	15.353
Wan-Animate	23.470	0.839	0.217	42.898	0.042	17.278	21.907	0.787	0.241	40.809	0.039	16.426
Ours	**25.238**	**0.895**	**0.169**	**38.277**	**0.031**	**17.701**	**24.458**	**0.873**	**0.181**	**31.190**	0.033	**17.650**
	60s						90s					
Method	PSNR↑	SSIM↑	LPIPS↓	FID↓	V-MAE↓	F-PSNR↑	PSNR↑	SSIM↑	LPIPS↓	FID↓	V-MAE↓	F-PSNR↑
One-to-All	20.108	0.802	0.314	46.951	0.037	15.495	19.306	0.754	0.352	50.354	0.021	15.492
SCAIL	12.913	0.419	0.545	150.123	0.070	10.846	11.048	0.349	0.602	186.992	0.037	9.937
SteadyDancer	14.344	0.556	0.478	81.423	0.064	11.062	12.997	0.524	0.541	101.645	0.043	10.741
UniAnimate-DiT	19.028	0.807	0.296	71.183	0.040	13.967	17.422	0.745	0.349	83.521	0.027	13.329
Wan-Animate	20.665	0.745	0.282	32.603	0.041	15.920	19.676	0.702	0.323	32.636	0.030	15.855
Ours	**23.857**	**0.855**	**0.194**	**26.241**	**0.027**	**16.927**	**22.646**	**0.810**	**0.220**	**23.981**	**0.027**	**16.623**

Datasets and Metrics. Following One-to-All Animate [11], we train on a combination of the Champ [55], UBC [56], and Seedance [57] videos and 2k self-collected minute-scale videos from Youtube with 480P resolution. For evaluation, given the lack of standardized long-video benchmarks, we report results at multiple target durations, including 10s, 30s, 60s, and 90s at 25 FPS. We benchmark generation quality from three aspects: **(i)** frame-level fidelity using PSNR/SSIM and perceptual similarity using LPIPS; **(ii)** feature-space perceptual quality at the clip level using FID and Video-MAE Distance (V-MAE, computed from VideoMAE features [58]); and **(iii)** long-range identity correctness and consistency using face-region PSNR (F-PSNR).

5.1 Main Results

Qualitative Comparison. Fig. 7 compares generation quality across different rollout horizons. We can find that most models produce visually plausible results at short horizons but gradually deteriorate over time. In contrast, our method can maintain stable quality in the background and human identity without obvious artifacts, demonstrating robustness for long-range generation.

Quantitative Comparison. Tab. 1 shows that our method consistently achieves the best performance across rollout horizons, compared with state-of-the-art pose animation methods [11, 12, 14, 20, 23]. Note that generated videos may exhibit camera motion that differs from the ground truth, making pixel-level metrics such as PSNR and SSIM less reliable. At 10s, it improves over Wan-Animate, increasing PSNR from 23.47 to 25.24 and reducing LPIPS from 0.217 to 0.169. The advantage becomes more pronounced at longer horizons, such as 90s, indicating the stability of our method.

Ablation Study. Tab. 2 reports a 60s ablation. The full model improves over the baseline by +5.39 PSNR, and increases SSIM from 0.543 to 0.855. Removing RFM yields noticeably lower perceptual quality. Removing PLP primarily weakens cross-chunk carry-over, suggesting that memory propagation is important for long-horizon consistency.

Table 2: Ablation study at 60s (480p).

Method	PSNR↑	SSIM↑	LPIPS↓
Baseline	18.472	0.543	0.386
w/o RFM	21.842	0.781	0.273
w/o PLP	22.317	0.809	0.208
Full model	**23.857**	**0.855**	**0.194**

6 Conclusion

We propose EverAnimate, a lightweight post-training framework for long-form pose-guided human animation. Rather than relying on image-space continuation, EverAnimate addresses long-horizon degradation through latent-state control. *(i) Persistent Latent Propagation* maintains reusable latent memory across chunks, preserving identity and motion cues over extended rollouts. *(ii) Restorative Flow Matching* complements this mechanism with a bounded corrective step during sampling, steering perturbed latent trajectories back toward the clean generation path. Together, these designs enable more stable long-form generation and substantially improve consistency and fidelity.

Acknowledgment

This work was supported as part of the Swiss AI Initiative by a grant from the Swiss National Supercomputing Centre (CSCS) under project ID a144 on Alps, Sportradar, Valeo, and Honda R&D Co., Ltd. We would like to express our gratitude to Valentin Gerard, Adrien Lefevre, Zimin Xia, and the Longcat Team for insightful discussions.

References

[1] 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. URL `https://arxiv.org/abs/2503.20314`.

[2] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, Kathrina Wu, Qin Lin, Junkun Yuan, Yanxin Long, Aladdin Wang, Andong Wang, Changlin Li, Duojun Huang, Fang Yang, Hao Tan, Hongmei Wang, Jacob Song, Jiawang Bai, Jianbing Wu, Jinbao Xue, Joey Wang, Kai Wang, Mengyang Liu, Pengyu Li, Shuai Li, Weiyan Wang, Wenqing Yu, Xinchi Deng, Yang Li, Yi Chen, Yutao Cui, Yuanbo Peng, Zhentao Yu, Zhiyu He, Zhiyong Xu, Zixiang Zhou, Zunnan Xu, Yangyu Tao, Qinglin Lu, Songtao Liu, Dax Zhou, Hongfa Wang, Yong Yang, Di Wang, Yuhong Liu, Jie Jiang, and Caesar Zhong. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024. URL `https://arxiv.org/abs/2412.03603`.

[3] Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei A Efros. Everybody dance now. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5933–5942, 2019.

[4] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. *arXiv preprint arXiv:1808.06601*, 2018.

[5] Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023.

[6] Zhongcong Xu, Jianfeng Zhang, Jun Hao Liew, Hanshu Yan, Jia-Wei Liu, Chenxu Zhang, Jiashi Feng, and Mike Zheng Shou. Magicanimate: Temporally consistent human image animation using diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1481–1490, 2024.

[7] Li Hu, Xin Gao, Peng Zhang, Ke Sun, Bang Zhang, and Liefeng Bo. Animate anyone: Consistent and controllable image-to-video synthesis for character animation. *arXiv preprint arXiv:2311.17117*, 2023.

[8] Jeongho Kim, Min-Jung Kim, Junsoo Lee, and Jaegul Choo. Tcan: Animating human images with temporally consistent pose guidance using diffusion models. *arXiv preprint arXiv:2407.09012*, 2024.

[9] Yong Zhong, Min Zhao, Zebin You, Xiaofeng Yu, Changwang Zhang, and Chongxuan Li. Posecrafter: One-shot personalized video synthesis following flexible pose control. In *European conference on computer vision*, pages 243–260. Springer, 2024.

[10] Qijun Gan, Yi Ren, Chen Zhang, Zhenhui Ye, Pan Xie, Xiang Yin, Zehuan Yuan, Bingyue Peng, and Jianke Zhu. Humandit: Pose-guided diffusion transformer for long-form human motion video generation. *arXiv preprint arXiv:2502.04847*, 2025.

[11] Shijun Shi, Jing Xu, Zhihang Li, Chunli Peng, Xiaoda Yang, Lijing Lu, Kai Hu, and Jiangning Zhang. One-to-all animation: Alignment-free character animation and image pose transfer. *arXiv preprint arXiv:2511.22940*, 2025.

[12] Jiaming Zhang, Shengming Cao, Rui Li, Xiaotong Zhao, Yutao Cui, Xinglin Hou, Gangshan Wu, Haolan Chen, Yu Xu, Limin Wang, and Kai Ma. Steadydancer: Harmonized and coherent human image animation with first-frame preservation. *arXiv preprint arXiv:2511.19320*, 2025. URL `https://arxiv.org/abs/2511.19320`.

[13] Shuai Tan, Biao Gong, Xiang Wang, Shiwei Zhang, Dandan Zheng, Ruobing Zheng, Kecheng Zheng, Jingdong Chen, and Ming Yang. Animate-x: Universal character image animation with enhanced motion representation. *arXiv preprint arXiv:2410.10306*, 2024.

[14] Gang Cheng, Xin Gao, Li Hu, Siqi Hu, Mingyang Huang, Chaonan Ji, Ju Li, Dechao Meng, Jinwei Qi, Penchong Qiao, et al. Wan-animate: Unified character animation and replacement with holistic replication. *arXiv preprint arXiv:2509.14055*, 2025.

[15] Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. *arXiv preprint arXiv:2309.17453*, 2023.

[16] Wuyang Li, Wentao Pan, Po-Chien Luan, Yang Gao, and Alexandre Alahi. Stable video infinity: Infinite-length video generation with error recycling. In *International Conference on Learning Representations*, volume 2026, pages 23406–23432, 2026.

[17] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. *Advances in neural information processing systems*, 32, 2019.

[18] Shenhao Zhu, Junming Leo Chen, Zuozhuo Dai, Zilong Dong, Yinghui Xu, Xun Cao, Yao Yao, Hao Zhu, and Siyu Zhu. Champ: Controllable and consistent human image animation with 3d parametric guidance. In *European Conference on Computer Vision*, pages 145–162. Springer, 2024.

[19] Yuang Zhang, Jiaxi Gu, Li-Wen Wang, Han Wang, Junqi Cheng, Yuefeng Zhu, and Fangyuan Zou. Mimicmotion: High-quality human motion video generation with confidence-aware pose guidance. *arXiv preprint arXiv:2406.19680*, 2024.

[20] Xiang Wang, Shiwei Zhang, Longxiang Tang, Yingya Zhang, Changxin Gao, Yuehuan Wang, and Nong Sang. Unianimate-dit: Human image animation with large-scale video diffusion transformer. *arXiv preprint arXiv:2504.11289*, 2025.

[21] Jingkai Zhou, Yifan Wu, Shikai Li, Min Wei, Chao Fan, Weihua Chen, Wei Jiang, and Fan Wang. RealisDance-DiT: Simple yet strong baseline towards controllable character animation in the wild. *arXiv preprint arXiv:2504.14977*, 2025.

[22] Shuyuan Tu, Zhen Xing, Xintong Han, Zhi-Qi Cheng, Qi Dai, Chong Luo, and Zuxuan Wu. StableAnimator: High-quality identity-preserving human image animation. In *CVPR*, 2025.

[23] Wenhao Yan, Sheng Ye, Zhuoyi Yang, Jiayan Teng, ZhenHui Dong, Kairui Wen, Xiaotao Gu, Yong-Jin Liu, and Jie Tang. Scail: Towards studio-grade character animation via in-context learning of 3d-consistent pose representations. *arXiv preprint arXiv:2512.05905*, 2025.

[24] aigc apps. Videox-fun: A video generation pipeline for diffusion transformer, 2026. URL `https://github.com/aigc-apps/VideoX-Fun`.

[25] Yang Gao, Wuyang Li, Po-Chien Luan, and Alexandre Alahi. Deformable gaussian occupancy: Decoupling rigid and nonrigid motion with factorized distillation. In *CVPR*, 2026.

[26] Rang Meng, Xingyu Zhang, Yuming Li, and Chenguang Ma. Echomimicv2: Towards striking, simplified, and semi-body human animation. *arXiv preprint arXiv:2411.10061*, 2024. URL `https://arxiv.org/abs/2411.10061`.

[27] Gaojie Lin, Jianwen Jiang, Jiaqi Yang, Zerong Zheng, and Chao Liang. Omnihuman-1: Rethinking the scaling-up of one-stage conditioned human animation models. *arXiv preprint arXiv:2502.01061*, 2025.

[28] Yi Chen, Sen Liang, Zixiang Zhou, Ziyao Huang, Yifeng Ma, Junshu Tang, Qin Lin, Yuan Zhou, and Qinglin Lu. HunyuanVideo-Avatar: High-fidelity audio-driven human animation for multiple characters. *arXiv preprint arXiv:2505.20156*, 2025.

[29] Jingxuan He, Busheng Su, and Finn Wong. PoseGen: In-context LoRA finetuning for pose-controllable long human video generation. *arXiv preprint arXiv:2508.05091*, 2025.

[30] Shen Zheng, Jiaran Cai, Yuansheng Guan, Shenneng Huang, Xingpei Ma, Junjie Cao, Hanfeng Zhao, Qiang Zhang, Shunsi Zhang, and Xiao-Ping Zhang. High-fidelity and long-duration human image animation with diffusion transformer. *arXiv preprint arXiv:2512.21905*, 2025.

[31] Junyoung Seo, Rodrigo Mira, Alexandros Haliassos, Stella Bounareli, Honglie Chen, Linh Tran, Seungryong Kim, Zoe Landgraf, and Jie Shen. Lookahead anchoring: Preserving character identity in audio-driven human animation. *arXiv preprint arXiv:2510.23581*, 2025.

[32] Xiaoyu Liu, Mingshuai Yao, Yabo Zhang, Xianhui Lin, Peiran Ren, Xiaoming Li, Ming Liu, and Wangmeng Zuo. AnimateAnywhere: Rouse the background in human image animation. *arXiv preprint arXiv:2504.19834*, 2025.

[33] 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. *arXiv preprint arXiv:2311.15127*, 2023.

[34] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024.

[35] Wan Team. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

[36] Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024.

[37] Sand.ai. MAGI-1: Autoregressive video generation at scale. *arXiv preprint arXiv:2505.13211*, 2025.

[38] Guibin Chen, Dixuan Lin, Jiangping Yang, Chunze Lin, Junchen Zhu, Mingyuan Fan, et al. SkyReels-V2: Infinite-length film generative model. *arXiv preprint arXiv:2504.13074*, 2025.

[39] 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. URL `https://arxiv.org/abs/2406.16863`.

[40] Po-Chien Luan, Wuyang Li, Yang Gao, and Alexandre Alahi. Social-mamba: Socially-aware trajectory forecasting with state-space models. *arXiv preprint arXiv:2605.15424*, 2026.

[41] Boyuan Chen, Diego Marti Monso, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. In *NeurIPS*, 2024.

[42] Tianwei Yin, Qiang Zhang, Richard Zhang, William T. Freeman, Frédo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models. In *CVPR*, 2025.

[43] Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. In *NeurIPS*, 2025.

[44] Kunhao Liu, Wenbo Hu, Jiale Xu, Ying Shan, and Shijian Lu. Rolling forcing: Autoregressive long video diffusion in real time. In *ICLR*, 2026.

[45] Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Yingcong Chen, Yao Lu, Song Han, and Yukang Chen. Longlive: Real-time interactive long video generation. *arXiv preprint arXiv:2509.22622*, 2025. URL `https://arxiv.org/abs/2509.22622`.

[46] Lvmin Zhang, Shengqu Cai, Muyang Li, Gordon Wetzstein, and Maneesh Agrawala. Packing input frame context in next-frame prediction models for video generation. In *NeurIPS*, 2025.

[47] Shenghai Yuan, Yuanyang Yin, Zongjian Li, Xinwei Huang, Xiao Yang, and Li Yuan. Helios: Real real-time long video generation model. *arXiv preprint arXiv:2603.04379*, 2026.

[48] Zile Wang, Zexiang Liu, Jaixing Li, Kaichen Huang, Baixin Xu, Fei Kang, Mengyin An, Peiyu Wang, Biao Jiang, Yichen Wei, et al. Matrix-game 3.0: Real-time and streaming interactive world model with long-horizon memory. *arXiv preprint arXiv:2604.08995*, 2026.

[49] Meituan LongCat Team. LongCat-Video technical report. *arXiv preprint arXiv:2510.22200*, 2025.

[50] Sihyun Yu, Meera Hahn, Dan Kondratyuk, Jinwoo Shin, Agrim Gupta, Jose Lezama, Irfan Essa, David Ross, and Jonathan Huang. Malt diffusion: Memory-augmented latent transformers for any-length video generation. *arXiv preprint arXiv:2502.12632*, 2025.

[51] Lvmin Zhang, Shengqu Cai, Muyang Li, Chong Zeng, Beijia Lu, Anyi Rao, Song Han, Gordon Wetzstein, and Maneesh Agrawala. Pretraining frame preservation in autoregressive video memory compression. *arXiv preprint arXiv:2512.23851*, 2025.

[52] Zeqi Xiao, Yushi Lan, Yifan Zhou, Wenqi Ouyang, Shuai Yang, Yanhong Zeng, and Xingang Pan. WorldMem: Long-term consistent world simulation with memory. *arXiv preprint arXiv:2504.12369*, 2025.

[53] Shuo Chen, Cong Wei, Sun Sun, Ping Nie, Kai Zhou, Ge Zhang, Ming-Hsuan Yang, and Wenhu Chen. Context forcing: Consistent autoregressive video generation with long context. *arXiv preprint arXiv:2602.06028*, 2026.

[54] ModelScope Team. Diffsynth-studio. GitHub repository, 2024. URL `https://github.com/modelscope/DiffSynth-Studio`. Accessed: 2026-05-04.

[55] Shenhao Zhu, Junming Leo Chen, Zuozhuo Dai, Zilong Dong, Yinghui Xu, Xun Cao, Yao Yao, Hao Zhu, and Siyu Zhu. Champ: Controllable and consistent human image animation with 3d parametric guidance. In *European Conference on Computer Vision*, pages 145–162. Springer, 2024.

[56] Polina Zablotskaia, Aliaksandr Siarohin, Bo Zhao, and Leonid Sigal. Dwnet: Dense warp-based network for pose-guided human video generation. *arXiv preprint arXiv:1910.09139*, 2019.

[57] Yu Gao, Haoyuan Guo, Tuyen Hoang, Weilin Huang, Lu Jiang, Fangyuan Kong, Huixia Li, Jiashi Li, Liang Li, Xiaojie Li, et al. Seedance 1.0: Exploring the boundaries of video generation models. *arXiv preprint arXiv:2506.09113*, 2025.

[58] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *Advances in Neural Information Processing Systems*, 2022.

A Discussion

A.1 Limitations

Our method substantially improves temporal robustness for minute-scale pose-guided human video generation under a fixed, chunked rollout setting. However, it does not completely eliminate long-horizon drift in some cases: **errors may still appear in certain samples** when generation is extended to extremely long lengths, especially under out-of-distribution appearances. This might be caused by the limited capacity of LoRA training and data diversity. A further limitation is that the current framework **does not provide explicit control over camera viewpoint or camera trajectory**. The relative motion between the subject and the camera is modeled only implicitly and is largely determined by the training distribution and stochastic generation process. As a result, the method may have limited controllability when users require precise camera movements, consistent viewpoint changes, or coordinated subject-camera motion over long durations.

Future work may further improve scalability, reduce ultra-long-horizon error accumulation, and introduce explicit camera-control mechanisms for specifying viewpoint, camera trajectory, and subject-camera relative motion. It would also be valuable to evaluate the method across a broader range of identities, clothing styles, camera motions, scene layouts, and real-world usage scenarios.

A.2 Broader Impact

Long-form human animation can support a variety of beneficial applications, including creative content production, virtual avatars, education, accessibility, and assistive communication. By improving the temporal consistency of synthesized human motion, the proposed approach may lower the cost and technical barriers of producing character animation and personalized visual media.

A.3 Safeguards and Responsible Release

Given the dual-use potential of high-fidelity human animation generation, we do not support unrestricted deployment. Any release of models, code, or demonstrations should follow a responsible process that may include gated access, explicit use policies, audit logging, and prohibitions on impersonation, non-consensual synthetic media, harassment, and politically deceptive content. Where practical, technical measures should also be adopted, including visible or invisible watermarking, and compatibility with content authentication standards. In addition, each release should provide clear documentation of intended applications and relevant ethical considerations.

A.4 Statistical Significance

Due to the high computational cost of long-horizon video generation and evaluation, we do not include repeated-run confidence intervals or statistical significance tests for all settings. We acknowledge this as a limitation. Nevertheless, we report results consistently across methods under the same evaluation setup, and the observed improvements are supported by both quantitative metrics and qualitative long-form comparisons.

A.5 Ethics and Societal Impacts

Sections A.2 and A.3 discuss the dual-use nature of the proposed technology, including both beneficial applications and potential misuse risks. We explicitly recommend consent, provenance, watermarking, access control, and policies against impersonation and non-consensual synthetic media for any real-world deployment or release. When we open-source the project, we will further strengthen these safeguards through responsible release practices and technical protections wherever feasible.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction summarize the paper's two core contributions (persistent latent propagation and restorative flow matching) and scope the claims to long-horizon pose-guided human animation; see Abstract and Sec. 1.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses limitations including residual drift over very long horizons, scaling of memory/compute with duration, partial reliance on private data, and generalization scope; see Sec. A.1.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper does not present formal theorems/lemmas requiring complete proofs; it provides method derivations in the main text (e.g., the restorative target/velocity derivation).

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Our code and data will be fully released. We have reported the main training and inference hyperparameters, model configuration, evaluation protocol, and implementation details for reproducibility.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code and data will be released.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Training/inference settings (e.g., K, r, iterations, batch size, LoRA rank/α, perturbations, sampling steps) and evaluation horizons/metrics are described.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: The paper presented quantitative results following existing works, which have conveyed statistical information without using error bars.

   Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper has reported the number of GPUs and training iterations.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We reviewed the NeurIPS Code of Ethics and conducted this research in accordance with the guidelines, including avoiding deanonymizing information in the submission.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The paper discusses potential positive impacts and potential misuse risks of long-form human animation, together with responsible-use considerations.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [Yes]

Justification: The paper discusses safeguards and responsible release considerations for dual-use human animation generation.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All the assets used in our paper are clearly mentioned and cited.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.

- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not release new datasets/model weights/code artifacts beyond the paper itself.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This work does not include crowdsourcing experiments or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

Justification: This work does not involve human subject studies requiring IRB approval.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development and experiments do not use LLMs as an important, original, or non-standard component; no LLM usage declaration is required.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.