PhysAnimator: Physics-Guided Generative Cartoon Animation

Tianyi Xie[1,2*] Yiwei Zhao[1†] Ying Jiang[2] Chenfanfu Jiang[2]
[1] Netflix, [2] UCLA
tianyixie77@ucla.edu yiweiz@netflix.com {yingjiang, cffjiang}@ucla.edu

Figure 1. **PhysAnimator** is a novel framework that combines physics principles with video diffusion models to generate high-quality animations from static anime illustrations, allowing users to specify external forces or rigging points for custom effects.

Abstract

Creating hand-drawn animation sequences is labor-intensive and demands professional expertise. We introduce PhysAnimator, a novel approach for generating physically plausible meanwhile anime-stylized animation from static anime illustrations. Our method seamlessly integrates physics-based simulations with data-driven generative models to produce dynamic and visually compelling animations. To capture the fluidity and exaggeration characteristic of anime, we perform image-space deformable body simulations on extracted mesh geometries. We enhance artistic control by introducing customizable energy strokes and incorporating rigging point support, enabling the creation of tailored animation effects such as wind interactions. Finally, we extract and warp sketches from the simulation sequence, generating a texture-agnostic representation, and employ a sketch-guided video diffusion model to synthesize high-quality animation frames. The resulting animations exhibit temporal consistency and visual plausibility, demonstrating the effectiveness of our method in creating dynamic anime-style animations. See our project page for more demos: https://xpandora.github.io/PhysAnimator/.

1. Introduction

Dynamic visual effects are essential to the immersive quality of 2D animation. From the subtle sway of a character's hair to the fluid motion of garments in response to

* Work done during the internship at Netflix.
† Corresponding author.

wind, realistic and pleasing dynamics create a captivating visual experience. Traditionally, these effects are achieved through meticulous, hand-drawn techniques, where animators painstakingly draw each frame to bring these dynamic elements to life. This process is labor-intensive and requires not only artistic skill but also a deep understanding of natural forces and environmental interactions.

To alleviate the challenges of manual animation, researchers have explored both traditional and data-driven approaches. Traditional animation tools [31, 58] provide interactive systems that assist artists in creating animated illustrations based on established principles of 2D animation. These methods often rely on user-provided stroke inputs to specify motion trajectories and utilize geometry constraints to produce deformation-based animation. However, such approaches typically assume simple inputs like lineart or drawings with separated layers, limiting their applicability to more complex, in-the-wild anime illustrations that feature intricate textures and details. In contrast, data-driven video generative models [2, 46, 71] offer a promising alternative by leveraging neural synthesis to generate dynamic effects directly from images, bypassing the need for manual sketching or layered inputs. Recent methods even enable interactive motion design, allowing users to specify object trajectories via drag-based inputs [49, 55, 67]. These approaches typically rely on predicting a sequence of optical flow fields to drive the desired motion and warp the frames accordingly. However, the quality of the generated results is often limited by inaccuracies in the predicted optical flow, which frequently exhibits artifacts due to a lack of geometric understanding and physical constraints. As a result, imprecise motion estimation can lead to noticeable distortions and unsatisfactory visual quality.

Recognizing the limitations of both traditional and purely data-driven methods, we introduce PhysAnimator, a novel framework that integrates physics-based animation with data-driven generative models to synthesize visually compelling dynamic animations driven by environmental forces such as wind or rigging from static anime illustrations. Our approach combines the physical consistency of simulation-based methods with the flexibility and expressiveness of generative models, overcoming the drawbacks of previous approaches. To achieve this, we model objects of interest in the anime illustration as deformable bodies, capturing the fluidity and exaggerated motion characteristic of anime. We solve the motion equations to compute a sequence of optical flow fields that represent consistent dynamics. Users can enhance these animations with customized energy strokes, defining the effects of external forces such as wind. To render the motion dynamics as high-quality frames, we first extract and warp the sketch using the optical flow fields to generate a texture-agnostic video sequence. We then apply a sketch-guided video diffusion model to colorize the sketch sequence based on the reference illustration, ensuring stylistic coherence. Finally, to incorporate dynamic effects that cannot be easily captured by physical simulation, we employ a data-driven cartoon interpolation model, enriching the results with complementary, expressive dynamics. In summary, our contributions are as follows:

- We introduce a novel framework that combines physics-based simulations with data-driven generative models, specifically targeting animations driven by environmental forces and rigging controls, achieving both physical consistency and stylistic expressiveness.
- We develop an image-space deformable body simulation technique that models anime objects as deformable meshes, capturing fluid and exaggerated dynamics.
- We leverage a sketch-guided video diffusion model to render simulation dynamics as high-quality frames, combined with a cartoon interpolation model to introduce additional dynamic effects beyond physical laws.

2. Related Work

2.1. Video Diffusion Models

Building on the success of image diffusion models [10, 53], recent work has introduced video diffusion models to streamline video synthesis from text prompts or images, significantly reducing labor and time costs compared to traditional commercial video editing and creation tools [23, 72]. These approaches either extend pre-trained image diffusion models by incorporating temporal mixing layers in various forms [3, 11, 13, 56] or train video diffusion models with temporal layers from scratch on large-scale text-video paired datasets [20, 79]. However, using text prompts or a single image as input provides limited control over fine-grained aspects of video generation, making it challenging to define complex structural attributes such as spatial layouts, poses, and shapes [17, 74]. To allow for more detailed control over object motion and camera movements, additional guidance in the form of motion trajectories [6, 28, 55, 74], pose [77], depth [21], and optical flow [17] has been integrated into video diffusion models to produce more controllable videos. These powerful video diffusion models have also been applied to various downstream tasks, such as video editing [16, 35], image animation [12, 66, 71], video understanding [38, 54, 61], video interpolation [27, 70] and 3D reconstruction and generation [8, 18, 36, 43, 60]. Nevertheless, these data-driven approaches usually produce artifacts due to a lack of geometric understanding and physical constraints.

2.2. Physics-based Animation

Researchers have explored various methods for animation authoring. Xing et al. [68] extend local similarity tech-

niques to global similarity, enabling automatic sketch completion based on previous frames. Building on this, Peng et al. [51] propose a keyframe-based sculpting system that autocompletes user edits using an intuitive brushing interface. Willett et al. [65] tackle the challenge of textured anime images by segmenting the illustration into layers for different objects and allowing users to provide scribbles as trajectory guides. Leveraging animation principles, some works [31, 45] use geometry-based deformations to create stylized animation effects such as "squash" and "stretch". Similarly, Xing et al. [69] employ physics-based simulations and introduce energy brushes to generate elemental dynamics [15], such as smoke and fire. Other approaches [1, 9, 29] incorporate physical laws like gravity, collision, and elasticity to animate deformable characters, enabling the efficient creation of complex scenes. For instance, Jones et al. [30] introduces an example-based plasticity model based on linear blend skinning for animating the failure of near-rigid, man-made materials. Coros et al. [9] exploit two rest-pose adaptation strategies using only internal energy to animate curve, shell, and solid-based characters. Physics-based modeling has also proven effective in adding secondary dynamics [64, 75] to enhance details in rigged animations. Despite their effectiveness, these 2D animation methods typically assume simple inputs, such as sketches, or require layered separation, which limits their applicability for animating complex, in-the-wild anime illustrations.

2.3. Generative Image Dynamics

With the advancement of generative models, there is growing interest in leveraging these powerful techniques to synthesize dynamic animations from static images, guided by motion features extracted from various user inputs. These inputs can be sparse, such as text prompts [7, 41, 71], trajectories [42, 55, 62, 67, 78], or camera movements [62, 73], or dense, like reference videos [28, 63, 80]. To achieve controllable dynamics, prior works often incorporate ControlNet [76] into image or video generative models during the decoding stage, utilizing motion features such as Canny edges, depth maps [17], 2D Gaussian maps [67], and optical flow maps [55]. In addition, several methods employ specialized motion fusion modules [41, 62, 78], spatial or temporal attention mechanisms [28, 63, 73], and feature injection strategies [42] to guide the generation of controllable dynamics. However, these approaches typically lack physics-based supervision, which can result in animations that violate physical laws or fail to align with user intentions [47]. Recently, PhysGen [44] integrated a rigid-body physics simulator to generate physics-consistent dynamics for foreground objects in a given image. While effective, PhysGen is limited to 2D rigid-body motions, making it unsuitable for the fluid, elastic effects commonly seen in anime, such as the waving of cloth or hair, which do not conform to rigid-body dynamics. In this work, we address these limitations by leveraging deformable-body dynamics and a sketch-guided video diffusion model to create high-quality, physically consistent animation sequences that capture the fluidity and elasticity characteristic of anime.

3. Method

Given a reference anime illustration I_0, our goal is to generate a sequence of stylized video frames $\{\hat{I}_1, \hat{I}_2, ..., \hat{I}_T\}$ with natural, user-guided motions. To achieve this, we introduce PhysAnimator, a novel framework that combines video diffusion models with physics-based animation. We start by segmenting the objects of interest in a given anime illustration and creating a 2D triangulated mesh representation. Using this geometry, we obtain dynamics through image-space deformable body simulation, generating a sequence of optical flow fields $\{\mathcal{F}_{0\to1}, \mathcal{F}_{0\to2}, ..., \mathcal{F}_{0\to T}\}$. To offer user control, we enable interactive inputs through customizable energy strokes for defining external forces and rigging points for specifying desired trajectories. To render dynamics as high-quality frames, we extract the sketch from the input illustration and warp it using the computed optical flow fields, yielding a sequence of dynamic sketch frames $\{S_1, S_2, ..., S_T\}$. These sketches, together with the input illustration, are fed into a video diffusion model with a sketch-guided ControlNet [76], synthesizing a vivid animated sequence. Finally, an optional data-driven cartoon interpolation model [70] can be applied to enhance the anime style dynamics of the results by selecting keyframes from the animated sequence as input, yielding the final output frames $\{\hat{I}_1, \hat{I}_2, ..., \hat{I}_T\}$. An overview of our proposed framework is shown in Fig. 2.

3.1. Preliminaries

Latent Video Diffusion Model Most video diffusion models build on the latent diffusion model (LDM) framework [53], which uses Variational Autoencoder [32] (VAE) to map input images into a latent space. In this space, data is transformed into Gaussian noise via a forward diffusion process, which the model learns to reverse through denoising. In the forward diffusion process, the latent code z_0 is perturbed as:

$$z_t = \sqrt{\bar{\alpha}_t} z_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \epsilon \sim \mathcal{N}(0, I), \tag{1}$$

where $\bar{\alpha}_t = \prod_{i=1}^{t}(1-\beta_t)$ with β_t controlling the noise strength coefficient at step t. The denoising model ϵ_θ is trained to recover z_t by minimizing the objective function:

$$L_\epsilon = \|\epsilon - \epsilon_\theta(z_t; c, t)\|_2^2, \tag{2}$$

where θ denotes the learnable network parameters and c represents conditioning input (e.g. text prompts or images).

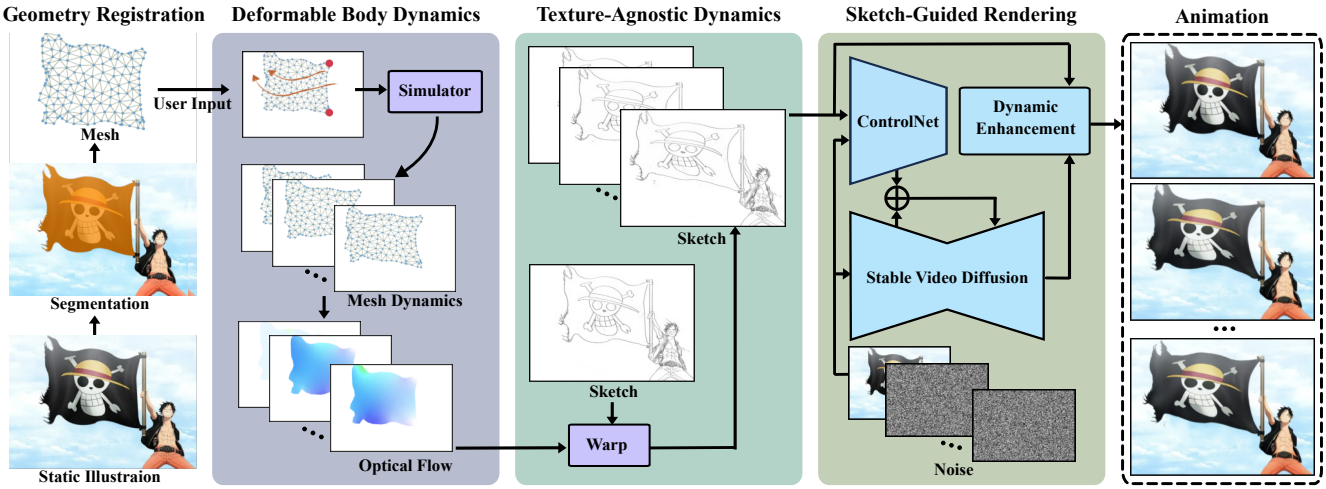


Figure 2. **Method Overview.** We begin by segmenting the object and creating a triangulated deformable mesh. Physics-based simulations are then used to generate dynamic optical flow fields, with users given the option to guide the motion through customizable energy strokes(shown as orange arrows) and rigging points(shown as red dots). The extracted sketch is warped using the computed optical flow and refined with a sketch-guided video diffusion model, producing a smooth, stylized animation sequence. Optionally, a cartoon interpolation model can further be applied to enhance the animation with expressive dynamics.

After denoising, the VAE decoder reconstructs the latent code back into the image space. The latent video diffusion model (LVDM) extends the image LDM to videos by incorporating temporal modules to maintain temporal consistency.

Deformable Body Dynamics Mathematically, the dynamics of a continuum deformable body is described by a time-dependent continuous deformation map $\boldsymbol{x} = \phi(\boldsymbol{X}, t)$, which maps the undeformed material space Ω^0 to the deformed world space Ω^t at time t. The deformation gradient $\boldsymbol{F} = \frac{\partial \phi}{\partial \boldsymbol{X}}$ encodes the local deformation such as scaling, rotation, and shearing. In the context of a discretized setting, e.g. 2D meshes, the deformation map for each triangle can be expressed as

$$\phi_i(\boldsymbol{X}) = \boldsymbol{F}_i \boldsymbol{X} + \boldsymbol{b}_i, \tag{3}$$

where $\boldsymbol{b}_i \in \mathbb{R}^2$ accounts for translation of i-th triangle, and the deformation gradient $\boldsymbol{F}_i \in \mathbb{R}^{2\times 2}$ is assumed to be constant [40]. Undergoing deformation, the deformed body aims to recover its rest shape via resisting forces. Continuum mechanics model this behavior by first defining an energy density function $\Psi(\boldsymbol{F}(\boldsymbol{x}))$, which measures the strain energy per unit undeformed volume. The total potential energy for a deformable body is then obtained as

$$E(\boldsymbol{x}) = \sum_{i=1}^{N} \Psi(\boldsymbol{F}_i) V_i, \tag{4}$$

where V_i denotes volume of i-th triangle. The internal resisting force is then defined as the negative gradient of the potential energy with respect to the vertex position

$$\boldsymbol{f}_{\text{int}}(\boldsymbol{x}) = -\frac{\partial E(\boldsymbol{x})}{\partial \boldsymbol{x}}. \tag{5}$$

The deformable body dynamics is governed by Newton's Second Law as

$$\frac{d^2 \boldsymbol{x}}{dt^2} = \boldsymbol{M}^{-1}(\boldsymbol{f}_{\text{int}}(\boldsymbol{x}) + \boldsymbol{f}_{\text{ext}}(\boldsymbol{x})), \tag{6}$$

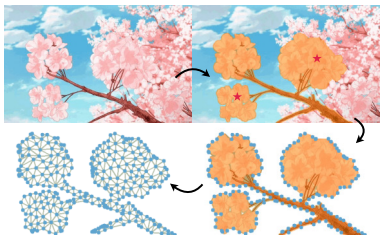
which can be solved numerically. Here M is the mass matrix that represents the masses of all vertices and $\boldsymbol{f}_{\text{ext}}$ refers to external forces, such as gravity.

3.2. Physics-based Animation

To generate dynamic animations from a single illustration, a straightforward approach is to use existing controllable video diffusion models [55, 67]. However, these purely data-driven methods often struggle due to a lack of physical understanding, leading to unrealistic results. To overcome this limitation, we incorporate physics-based animation to generate physically consistent and plausible motions.

Geometry Registration The first step to animate objects of interest in a given illustration is to establish a geometric basis. While previous works [24, 59] have explored predicting 3D geometry from single-view images, these methods are typically constrained to reconstructing real-world objects and do not generalize to in-the-wild anime images, since characters and objects in anime often lack an inherent 3D representation due to their stylized and flat nature. As shown in prior works on cartoon

animation [31, 69], leveraging 2D animation techniques proves effective for generating anime-style dynamic effects. Inspired by this, we focus on extracting 2D meshes for the objects of interest. To achieve this, we first utilize the Segment Anything Model (SAM) [33, 52] to obtain segmentation masks for each target object, guided by user-specified query points. Along the contours of each segmentation mask, we uniformly sample boundary points, defining the outline of the intended mesh. We then employ conforming Delaunay triangulation [37] with these boundary constraints to generate well-structured triangular meshes, creating a simulation-ready format for subsequent animation.



Deformable Body Model Anime scenes often feature dynamic environmental effects such as external forces that interact with characters' clothing, hair, or other elements. Additionally, techniques like "squash" and "stretch" are commonly used to convey motion and energy, adding expressiveness to animated objects. Motivated by these stylized dynamics, we model anime objects as deformable bodies, allowing them to capture the fluidity and exaggerated motion characteristic of anime. A key property of deformable bodies is their ability to change shape in response to external forces and attempt to return to their original rest shape upon deformation. To represent this physical behavior, we employ the Fixed Corotated constitutive model [57], which defines the energy density as:

$$\Psi(\boldsymbol{F}) = \mu\|\boldsymbol{F} - \boldsymbol{R}\|_F^2 + \frac{\lambda}{2}(\det(\boldsymbol{F}) - 1)^2, \tag{7}$$

where μ, λ are the Lamé parameters, and $\boldsymbol{R}$ is the rotational part of $\boldsymbol{F}$ computed via polar decomposition. The first term models the stretching and compression resistance for the individual spatial directions and the second term describes resistance to volume change. The resulting internal forces $\boldsymbol{f}_{\text{int}}$ can then be derived via Eq. (5).

Interactive Animation While $\boldsymbol{f}_{\text{int}}$ governs the inherent physical behavior, $\boldsymbol{f}_{\text{ext}}$ allows for user-defined interactions. Inspired by the concept of energy strokes [15, 69], we introduce customizable energy strokes that carry flow particles. These particles move along the user-specified strokes, propagating external forces to nearby vertices of the deformable mesh. This enables the creation of tailored animation effects, such as simulating wind interactions. Additionally, we incorporate rigging point support, allowing animators to anchor specific regions or guide them along predefined trajectories, offering enhanced control and flexibility. Using the specified energy strokes and deformable modeling, we solve the motion equation (Eq. (6)) to evolve the dynamics. We employ the semi-implicit Euler method to compute the deformation map $\phi_i(X, t)$ at each time step t for every triangle. For each pixel $\boldsymbol{X}_p$ in the reference image, if it lies within a triangle $\mathcal{T}_i$, we assign the displacement vector as $\boldsymbol{d}(\boldsymbol{p}) = \phi_i(\boldsymbol{X}_p, t) - \boldsymbol{X}_p$; otherwise, we assign a zero vector. Collecting displacement vector $\boldsymbol{d}(\boldsymbol{p})$ of all pixels yields the optical flow $\mathcal{F}_{0\to t}$, which defines the displacement fields for the reference illustration I_0 at time t.

3.3. Generative Rendering

In this section, we describe how to render the video sequence $\{\hat{I}_1, \hat{I}_2, ..., \hat{I}_T\}$ given the reference image I_0 and the optical flow sequence $\{\mathcal{F}_{0\to1}, \mathcal{F}_{0\to2}, \ldots, \mathcal{F}_{0\to T}\}$ derived from simulation.

Sketch-Guided Rendering While the video sequence can be directly generated by warping the reference image I_0 using optical flow fields, the resulting frames often exhibit black hole artifacts caused by occlusions. To address this problem, we extract the sketch of I_0, denoted as S_0, and obtain the sketch at time t as follows:

$$S_t = \mathcal{W}(S_0, \mathcal{F}_{0\to t}, w_{0\to t}), \tag{8}$$

where $\mathcal{W}$ represents the forward-warping operator, and $w_{0\to t}$ denotes the warping weights for each pixel in I_0. In forward warping, multiple pixels may map to the same 2D location in the output frame [48], potentially leading to artifacts or distortions if the weights $w_{0\to t}$ are not properly defined. Inspired by [42], we set the pixel weight as $w_{0\to t}(\boldsymbol{p}) = \|\mathcal{F}_{0\to t}(\boldsymbol{p})\|_2$, giving higher importance to pixels with larger motion, typically corresponding to foreground objects.

Next, we leverage a video diffusion model to generate the rendered frames $\{I_1^r, I_2^r, ..., I_T^r\}$ using the obtained sketch sequence $\{S_1, S_2, ..., S_T\}$. The diffusion model uses the reference image I_0 as an input, and we employ a ControlNet [76] with the sketches as control signals, guiding the generation process to ensure that the results align with the sketch inputs. We observed that during the inference time, due to segmentation inaccuracies, the optical flow warping may introduce unintended distortions, or miss parts of the object contour, generating imperfect sketches. To address this, we apply Gaussian blur to the input sketches at both training and inference time, which smooths out inconsistencies. The video diffusion model is then capable of refining the results, leveraging its generative capabilities to handle imperfections and produce coherent outputs.

Complementary Dynamics Unlike motions in the real world, dynamic effects in animation do not strictly adhere to

physical laws and can not be fully captured by 2D animation methods. In the industrial animation pipeline, artists typically begin by creating a series of keyframes that define the primary motion trajectory, followed by drawing in-between frames to ensure smooth and fluid transitions. Inspired by this workflow, we leverage a data-driven module to enhance the physics-based animated results. Specifically, we select keyframes from the sketch-guided rendering results, forming the keyframe sequence $\{I^r_0, I^r_n, I^r_{2n}, ..., I^r_{in}\}$, where n denotes the gap between keyframes. We then employ a cartoon interpolation video diffusion model [70], which synthesizes intermediate frames $\{\hat{I}_{in+1}, \hat{I}_{in+2}, ..., \hat{I}_{i(n+1)-1}\}$ between each adjacent keyframe pair $(I^r_{in}, I^r_{i(n+1)})$. For keyframes, we simply assign $\hat{I}_{in} = I^r_{in}$. This approach allows us to introduce expressive, data-driven complementary dynamics that go beyond what can be achieved through physics-based animation alone.

4. Experiments

In this section, we conduct a comprehensive comparison of our method against existing video diffusion models and demonstrate that our approach generates high-quality and physically plausible animations.

Implementation Details We implement our deformable body simulator using Taichi [25]. During interactive animation, users can adjust the Lamé parameters μ and λ to control the characteristics of the objects based on their specific needs. For generative rendering, we follow LVCD [26] to train a sketch-guided ControlNet for the stable video diffusion model [4], using blurred sketches as control signals to address potential imperfections in the predicted sketch sequence. When utilizing ToonCrafter [70] for generating in-between frames, similar to its original implementation, we also train an additional sketch-guided ControlNet, but set the control scale to 0.1 during inference. We also set $n = 15$ for the in-betweening step. This configuration helps the generated frames follow the rough motion indicated by the sketches while introducing extra stylized animation details.

Dataset We build our training dataset using the Sakuga-42M Dataset [50], which consists of 1.4 million animation video clips. To ensure high-quality training samples, we filter out clips with fewer than 24 frames and dynamics scores outside the range of 0.05 to 0.7. Next, we extract sketches from the selected clips using the method from [5], resulting in a final dataset of 380,000 pairs of sketches and corresponding video sequences.

Baseline We compare our method against two categories of video generation approaches. The first category includes state-of-the-art Image-to-Video (I2V) models, such as Cinemo [46] and DynamiCrafter [71], which take an input image and use text prompts to guide the motion dynamics of the generated videos. For these models, we generate prompts describing the image content and expected motion using ChatGPT-4V. The second category, including Motion-I2V [55] and Drag Anything [67], usually trains an additional motion-control module for video generative models such that, given a single image input, a user-specified trajectory can be provided to control the movement of objects in the generated frames. To ensure a fair comparison, we extract trajectories from our animated results and use them as input for these methods.

4.1. Quantitative Evaluation

Given the absence of established benchmarks for anime-style image-to-video generation, we follow Motion-I2V [55] and construct a test set comprising 20 anime images with stylized dynamic elements such as swaying hair, clothing, and plants. For each method, we generate 10 video samples per image, resulting in a total of 200 videos per method. To quantitatively evaluate the generated results, we use the Fréchet Inception Distance (FID) [22] to measure the similarity between the generated frames and the reference images. Additionally, we employ VideoScore [19] to assess multiple aspects of video quality, including visual quality, temporal consistency, dynamic degree, and factual consistency. The factual consistency evaluates the consistency of the video content with common sense and factual knowledge. We exclude the text alignment score as our method and the trajectory-controlled approach do not involve text input.

The quantitative evaluation results, presented in Tab. 1, demonstrate that our method outperforms baseline approaches across most metrics, including visual quality, temporal consistency, and factual consistency, demonstrating the overall high quality of our generated videos. As shown in Fig. 3, Cinema tends to generate static videos that closely resemble the reference image, leading to better FID scores. While the dynamic degree score for DragAnything is notably higher than other methods, this is due to its tendency to misinterpret motion control as camera movement, causing shifts in the entire image space and resulting in an inflated dynamics score. Although our method's dynamic degree score is slightly lower than that of DynamiCrafter and Motion-I2V, these methods often exhibit implausible, distorted motions that create an illusion of increased dynamics. In contrast, our approach produces geometry-consistent motions and achieves high factual consistency, demonstrating the realistic motion dynamics enabled by our physics-based modeling. In addition, following [14, 34], we conducted an user study adopting a two-alternative forced choice (2AFC) protocol, where participants are asked to choose the preferred video based on temporal consistency, visual quality, motion plausibility, and overall feeling given

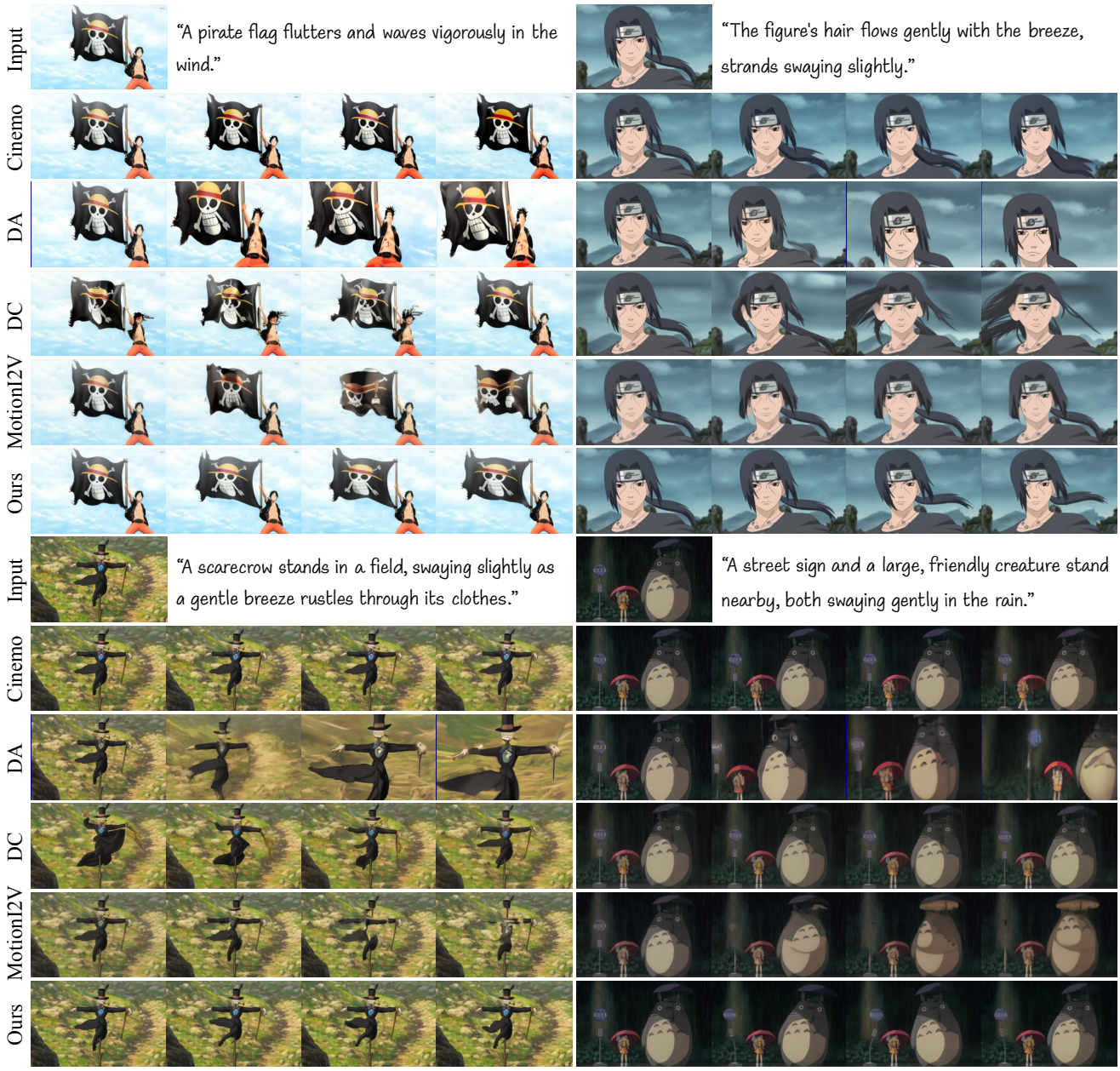


Figure 3. **Qualitative Comparison**. We compare our results against Cinemo [46], Drag Anything [67], DynamiCrafter [71] and MotionI2V [55]. Text prompts for Cinemo and DynamiCrafter are generated using ChatGPT-4V, while trajectories for Drag Anything and MotionI2V are extracted from our animated results.

two videos (one from our method and one from baselines). The user preference results, presented in Tab. 2, show that our proposed method consistently outperforms the baselines across all evaluation criteria.

4.2. Qualitative Comparison

We present a qualitative comparison with baseline methods in Fig. 3. Cinemo often produces static results with minimal motion dynamics. Drag Anything frequently misinterprets the motion trajectory as a camera movement, resulting in unintended dynamic sequences. DynamiCrafter, while generating larger motions, struggles to maintain the geometry of input objects, leading to noticeable distortions. MotionI2V has difficulty preserving the appearance of the input content, often yielding unsatisfactory results. In contrast, our physics-guided approach ensures both physically plausible motions and high-quality rendering, preserving the geometry and visual consistency of the input.

Table 1. **Quantitative Comparisons.** We report FID to evaluate the similarity between the reference image and generated frames. VSVQ, VSTC, VSDD, and VSFC represent scores for visual quality, temporal consistency, dynamic degree, and factual consistency, respectively, as measured by VideoScore [19].

Methods	FID↓	VSVQ↑	VSTC↑	VSDD↑	VSFC↑
Cinemo [46]	49.5	2.85	2.80	2.42	2.58
DragAnything [67]	148.9	2.77	2.45	2.97	2.52
DynamiCrafter [71]	94.9	2.78	2.68	2.53	2.51
Motion-I2V [55]	121.8	2.70	2.50	2.66	2.39
Ours	90.4	2.89	2.86	2.48	2.64

Table 2. **User Study.** We show the user preference for our method over the baseline methods in terms of visual quality (VQ), temporal consistency (TC), motion plausibility (MP), and overall feeling.

Methods	VQ	TC	MP	Overall
Cinemo [46]	86%	83%	82%	81%
DragAnything [67]	93%	91%	89%	91%
DynamiCrafter [71]	84%	78%	76%	81%
Motion-I2V [55]	95%	94%	97%	96%

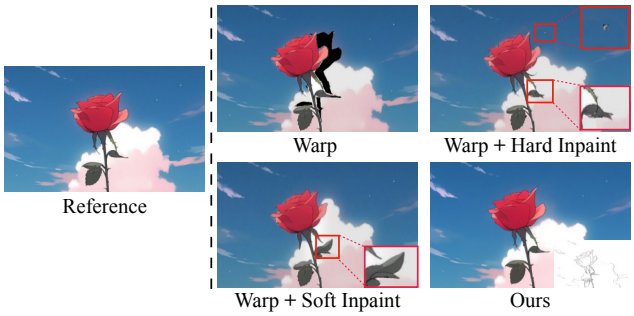


Figure 4. **Sketch-Guided Rendering**. Applying warping and inpainting introduces artifacts due to segmentation inaccuracy. Soft-inpainting [39] reduces these issues but can alter the content. Our sketch-guided rendering method produces high-quality results while preserving image details.

4.3. Additional Qualitative Results

Sketch-Guided Rendering After physics-based animation in our proposed framework, we obtain a sequence of optical flow $\{\mathcal{F}_{0\to1}, \mathcal{F}_{0\to2}, \ldots, \mathcal{F}_{0\to T}\}$. A straightforward approach would be to warp the input image I_0 using these optical flows and apply an inpainting algorithm to fill any occluded areas. However, this often results in suboptimal outputs. The main issue lies in segmentation inaccuracies, which can cause unintended regions to be warped or leave parts of the intended objects static, as shown in Fig. 4. While introducing a soft mask [39] into the inpainting process can help smooth the transition between the newly generated region and the original figure, it may also alter the content, making it diverge from the appearance of the reference image. Additionally, applying inpainting frame by frame may also cause temporal consistency issues. To address this, our method employs a sketch-guided rendering module. We first extract a sparse geometric representation of the image and apply the animation dynamics directly to this sketch-based structure. This sparse representation is more robust to segmentation inaccuracies and helps preserve the intended motion. Our subsequent rendering module then synthesizes high-quality video sequences from the animated sketches, maintaining both temporal consistency and visual fidelity.

Figure 5. **Complementary Dynamics Enhancement**. While physics-based animation maintains geometric consistency, it may lack the fluidity and exaggeration commonly seen in anime. We employ a data-driven interpolation module to enhance the motion dynamics, creating more natural-looking animations that better resemble real anime.

Complementary Dynamics Enhancement While our physics-based animation ensures physically accurate and geometry-consistent motion, it is inherently limited to 2D representations and cannot fully capture 3D effects. Moreover, the exaggerated dynamics often seen in anime do not always conform to strict physical laws. To address these limitations, we incorporate a data-driven cartoon interpolation module [70] to introduce complementary dynamics. As illustrated in Fig. 5, this interpolation module enables fluid contour deformations of clothing during motion, rather than rigidly adhering to static geometry. It also dynamically generates new sketch lines, enhancing the expressiveness of the animation and bringing it closer to the aesthetic of traditional hand-drawn anime.

5. Conclusion

We presented PhysAnimator, a novel framework for generating dynamic and stylized animations from static anime illustrations by integrating physics-based simulations with data-driven generative models. Our approach generates physically plausible, fluid and exaggerated anime-styled motion through deformable body simulations, providing controllability to users through user-guided energy strokes. A sketch-guided video diffusion model ensures high-quality, temporally consistent frames, while a data-driven anime frame interpolation model adds expressive, non-physical dynamics. The experiments show that our proposed method outperforms existing video diffusion methods, offering a powerful tool for creating visually compelling and user-controllable anime-style animations.

Acknowledgements

We thank anonymous reviewers for their insightful comments. We acknowledge support from NSF 2153851. We would like to thank Rahul Garg, Hossein Taghavi, Roshni Cooper, Ritwik Kumar, Boris Chen, Amli Murphy, Oliver Banasiak and Taiki Sakurai for their valuable suggestions and support.

References

[1] Jernej Barbič, Marco da Silva, and Jovan Popović. Deformable object animation using reduced optimal control. In *ACM SIGGRAPH 2009 papers*, pages 1–9. 2009. 3

[2] 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:2311.15127*, 2023. 2

[3] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22563–22575, 2023. 2

[4] Wenhao Chai, Xun Guo, Gaoang Wang, and Yan Lu. Stablevideo: Text-driven consistency-aware diffusion video editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23040–23050, 2023. 6

[5] Caroline Chan, Fredo Durand, and Phillip Isola. Learning to generate line drawings that convey geometry and semantics. 2022. 6

[6] Weifeng Chen, Yatai Ji, Jie Wu, Hefeng Wu, Pan Xie, Jiashi Li, Xin Xia, Xuefeng Xiao, and Liang Lin. Control-a-video: Controllable text-to-video generation with diffusion models. *arXiv:2305.13840*, 2023. 2

[7] Xinyuan Chen, Yaohui Wang, Lingjun Zhang, Shaobin Zhuang, Xin Ma, Jiashuo Yu, Yali Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. Seine: Short-to-long video diffusion model for generative transition and prediction. In *The Twelfth International Conference on Learning Representations*, 2023. 3

[8] Zilong Chen, Yikai Wang, Feng Wang, Zhengyi Wang, and Huaping Liu. V3d: Video diffusion models are effective 3d generators. *arXiv:2403.06738*, 2024. 2

[9] Stelian Coros, Sebastian Martin, Bernhard Thomaszewski, Christian Schumacher, Robert Sumner, and Markus Gross. Deformable objects alive! *ACM Transactions on Graphics (TOG)*, 31(4):1–9, 2012. 3

[10] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 2

[11] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7346–7356, 2023. 2

[12] Rinon Gal, Yael Vinker, Yuval Alaluf, Amit Bermano, Daniel Cohen-Or, Ariel Shamir, and Gal Chechik. Breathing life into sketches using text-to-video priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4325–4336, 2024. 2

[13] Songwei Ge, Seungjun Nah, Guilin Liu, Tyler Poon, Andrew Tao, Bryan Catanzaro, David Jacobs, Jia-Bin Huang, Ming-Yu Liu, and Yogesh Balaji. Preserve your own correlation: A noise prior for video diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22930–22941, 2023. 2

[14] Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Tokenflow: Consistent diffusion features for consistent video editing. *arxiv:2307.10373*, 2023. 6

[15] Joseph Gilland. *Elemental Magic, Volume 2: The Technique of Special Effects Animation*. Routledge, 2012. 3, 5

[16] Yuchao Gu, Yipin Zhou, Bichen Wu, Licheng Yu, Jia-Wei Liu, Rui Zhao, Jay Zhangjie Wu, David Junhao Zhang, Mike Zheng Shou, and Kevin Tang. Videoswap: Customized video subject swapping with interactive semantic point correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7621–7630, 2024. 2

[17] Yuwei Guo, Ceyuan Yang, Anyi Rao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Sparsectrl: Adding sparse controls to text-to-video diffusion models. In *European Conference on Computer Vision*, pages 330–348. Springer, 2025. 2, 3

[18] Junlin Han, Filippos Kokkinos, and Philip Torr. Vfusion3d: Learning scalable 3d generative models from video diffusion models. In *European Conference on Computer Vision*, pages 333–350. Springer, 2025. 2

[19] Xuan He, Dongfu Jiang, Ge Zhang, Max Ku, Achint Soni, Sherman Siu, Haonan Chen, Abhranil Chandra, Ziyan Jiang, Aaran Arulraj, et al. Videoscore: Building automatic metrics to simulate fine-grained human feedback for video generation. *arXiv:2406.15252*, 2024. 6, 8

[20] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity long video generation. *arXiv:2211.13221*, 2022. 2

[21] Yingqing He, Menghan Xia, Haoxin Chen, Xiaodong Cun, Yuan Gong, Jinbo Xing, Yong Zhang, Xintao Wang, Chao Weng, Ying Shan, et al. Animate-a-story: Storytelling with retrieval-augmented video generation. *arXiv:2307.06940*, 2023. 2

[22] 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. *Advances in neural information processing systems*, 30, 2017. 6

[23] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022. 2

[24] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *arXiv:2311.04400*, 2023. 4

[25] Yuanming Hu, Tzu-Mao Li, Luke Anderson, Jonathan Ragan-Kelley, and Frédo Durand. Taichi: a language for high-performance computation on spatially sparse data structures. *ACM Transactions on Graphics (TOG)*, 38(6): 201, 2019. 6

[26] Zhitong Huang, Mohan Zhang, and Jing Liao. Lvcd: Reference-based lineart video colorization with diffusion models. *arXiv:2409.12960*, 2024. 6

[27] Siddhant Jain, Daniel Watson, Eric Tabellion, Ben Poole, Janne Kontkanen, et al. Video interpolation with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7341–7351, 2024. 2

[28] 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*, pages 9212–9221, 2024. 2, 3

[29] Ben Jones, Jovan Popovic, James McCann, Wilmot Li, and Adam Bargteil. Dynamic sprites: artistic authoring of interactive animations. *Computer Animation and Virtual Worlds*, 26(2):97–108, 2015. 3

[30] Ben Jones, Nils Thuerey, Tamar Shinar, and Adam W Bargteil. Example-based plastic deformation of rigid bodies. *ACM Transactions on Graphics (TOG)*, 35(4):1–11, 2016. 3

[31] Rubaiat Habib Kazi, Tovi Grossman, Nobuyuki Umetani, and George Fitzmaurice. Motion amplifiers: sketching dynamic illustrations using the principles of 2d animation. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 4599–4609, 2016. 2, 3, 5

[32] Diederik P Kingma. Auto-encoding variational bayes. *arXiv:1312.6114*, 2013. 3

[33] 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*, pages 4015–4026, 2023. 5

[34] Nicholas Kolkin, Jason Salavon, and Gregory Shakhnarovich. Style transfer by relaxed optimal transport and self-similarity. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10051–10060, 2019. 6

[35] Max Ku, Cong Wei, Weiming Ren, Harry Yang, and Wenhu Chen. Anyv2v: A tuning-free framework for any video-to-video editing tasks. *arXiv:2403.14468*, 2024. 2

[36] Jeong-gi Kwak, Erqun Dong, Yuhe Jin, Hanseok Ko, Shweta Mahajan, and Kwang Moo Yi. Vivid-1-to-3: Novel view synthesis with video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6775–6785, 2024. 2

[37] Der-Tsai Lee and Bruce J Schachter. Two algorithms for constructing a delaunay triangulation. *International Journal of Computer & Information Sciences*, 9(3):219–242, 1980. 5

[38] Minhyeok Lee, Suhwan Cho, Chajin Shin, Jungho Lee, Sunghun Yang, and Sangyoun Lee. Video diffusion models are strong video inpainter. *arXiv:2408.11402*, 2024. 2

[39] Eran Levin and Ohad Fried. Differential diffusion: Giving each pixel its strength. *arXiv:2306.00950*, 2023. 8

[40] Minchen Li, Chenfanfu Jiang, and Zhaofeng Luo. *Physics-Based Simulation*. 2024. 4

[41] Mingxiao Li, Bo Wan, Marie-Francine Moens, and Tinne Tuytelaars. Animate your motion: Turning still images into dynamic videos. *arXiv:2403.10179*, 2024. 3

[42] Zhengqi Li, Richard Tucker, Noah Snavely, and Aleksander Holynski. Generative image dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24142–24153, 2024. 3, 5

[43] Fangfu Liu, Wenqiang Sun, Hanyang Wang, Yikai Wang, Haowen Sun, Junliang Ye, Jun Zhang, and Yueqi Duan. Reconx: Reconstruct any scene from sparse views with video diffusion model. *arXiv:2408.16767*, 2024. 2

[44] Shaowei Liu, Zhongzheng Ren, Saurabh Gupta, and Shenlong Wang. Physgen: Rigid-body physics-grounded image-to-video generation. In *European Conference on Computer Vision*, pages 360–378. Springer, 2025. 3

[45] Jiaju Ma, Li-Yi Wei, and Rubaiat Habib Kazi. A layered authoring tool for stylized 3d animations. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2022. 3

[46] Xin Ma, Yaohui Wang, Gengyu Jia, Xinyuan Chen, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Cinemo: Consistent and controllable image animation with motion diffusion models. *arXiv:2407.15642*, 2024. 2, 6, 7, 8

[47] Fanqing Meng, Jiaqi Liao, Xinyu Tan, Wenqi Shao, Quanfeng Lu, Kaipeng Zhang, Yu Cheng, Dianqi Li, Yu Qiao, and Ping Luo. Towards world simulator: Crafting physical commonsense-based benchmark for video generation. *arXiv:2410.05363*, 2024. 3

[48] Simon Niklaus and Feng Liu. Softmax splatting for video frame interpolation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5437–5446, 2020. 5

[49] Muyao Niu, Xiaodong Cun, Xintao Wang, Yong Zhang, Ying Shan, and Yinqiang Zheng. Mofa-video: Controllable image animation via generative motion field adaptions in frozen image-to-video diffusion model. *arXiv:2405.20222*, 2024. 2

[50] Zhenglin Pan, Yu Zhu, and Yuxuan Mu. Sakuga-42m dataset: Scaling up cartoon research. *arXiv:2405.07425*, 2024. 6

[51] Mengqi Peng, Li-yi Wei, Rubaiat Habib Kazi, and Vladimir G Kim. Autocomplete animated sculpting. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, pages 760–777, 2020. 3

[52] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv:2408.00714*, 2024. 5

[53] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2, 3

[54] Jiahao Shao, Yuanbo Yang, Hongyu Zhou, Youmin Zhang, Yujun Shen, Matteo Poggi, and Yiyi Liao. Learning temporally consistent video depth from video diffusion priors. *arXiv:2406.01493*, 2024. 2

[55] Xiaoyu Shi, Zhaoyang Huang, Fu-Yun Wang, Weikang Bian, Dasong Li, Yi Zhang, Manyuan Zhang, Ka Chun Cheung, Simon See, Hongwei Qin, et al. Motion-i2v: Consistent and controllable image-to-video generation with explicit motion modeling. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 2, 3, 4, 6, 7, 8

[56] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv:2209.14792*, 2022. 2

[57] Alexey Stomakhin, Russell Howes, Craig A Schroeder, and Joseph M Teran. Energetically consistent invertible elasticity. In *Symposium on Computer Animation*, 2012. 5

[58] Qingkun Su, Xue Bai, Hongbo Fu, Chiew-Lan Tai, and Jue Wang. Live sketch: Video-driven dynamic deformation of static drawings. In *Proceedings of the 2018 chi conference on human factors in computing systems*, pages 1–12, 2018. 2

[59] Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pages 1–18. Springer, 2025. 4

[60] Vikram Voleti, Chun-Han Yao, Mark Boss, Adam Letts, David Pankratz, Dmitry Tochilkin, Christian Laforte, Robin Rombach, and Varun Jampani. Sv3d: Novel multi-view synthesis and 3d generation from a single image using latent video diffusion. In *European Conference on Computer Vision*, pages 439–457. Springer, 2025. 2

[61] Qian Wang, Abdelrahman Eldesokey, Mohit Mendiratta, Fangneng Zhan, Adam Kortylewski, Christian Theobalt, and Peter Wonka. Zero-shot video semantic segmentation based on pre-trained diffusion models. *arXiv:2405.16947*, 2024. 2

[62] Zhouxia Wang, Ziyang Yuan, Xintao Wang, Yaowei Li, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. Motionctrl: A unified and flexible motion controller for video generation. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 3

[63] Yujie Wei, Shiwei Zhang, Zhiwu Qing, Hangjie Yuan, Zhiheng Liu, Yu Liu, Yingya Zhang, Jingren Zhou, and Hongming Shan. Dreamvideo: Composing your dream videos with customized subject and motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6537–6549, 2024. 3

[64] Nora S Willett, Wilmot Li, Jovan Popovic, Floraine Berthouzoz, and Adam Finkelstein. Secondary motion for performed 2d animation. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, pages 97–108, 2017. 3

[65] Nora S Willett, Rubaiat Habib Kazi, Michael Chen, George Fitzmaurice, Adam Finkelstein, and Tovi Grossman. A mixed-initiative interface for animating static pictures. In *Proceedings of the 31st annual ACM symposium on user interface software and technology*, pages 649–661, 2018. 3

[66] Ronghuan Wu, Wanchao Su, Kede Ma, and Jing Liao. Aniclipart: Clipart animation with text-to-video priors. *arXiv:2404.12347*, 2024. 2

[67] 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*, pages 331–348. Springer, 2025. 2, 3, 4, 6, 7, 8

[68] Jun Xing, Li-Yi Wei, Takaaki Shiratori, and Koji Yatani. Autocomplete hand-drawn animations. *ACM Transactions on Graphics (TOG)*, 34(6):1–11, 2015. 2

[69] Jun Xing, Rubaiat Habib Kazi, Tovi Grossman, Li-Yi Wei, Jos Stam, and George Fitzmaurice. Energy-brushes: Interactive tools for illustrating stylized elemental dynamics. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, pages 755–766, 2016. 3, 5

[70] Jinbo Xing, Hanyuan Liu, Menghan Xia, Yong Zhang, Xintao Wang, Ying Shan, and Tien-Tsin Wong. Tooncrafter: Generative cartoon interpolation. *arXiv:2405.17933*, 2024. 2, 3, 6, 8

[71] Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Wangbo Yu, Hanyuan Liu, Gongye Liu, Xintao Wang, Ying Shan, and Tien-Tsin Wong. Dynamicrafter: Animating open-domain images with video diffusion priors. In *European Conference on Computer Vision*, pages 399–417. Springer, 2025. 2, 3, 6, 7, 8

[72] Zhen Xing, Qijun Feng, Haoran Chen, Qi Dai, Han Hu, Hang Xu, Zuxuan Wu, and Yu-Gang Jiang. A survey on video diffusion models. *ACM Computing Surveys*, 2023. 2

[73] Shiyuan Yang, Liang Hou, Haibin Huang, Chongyang Ma, Pengfei Wan, Di Zhang, Xiaodong Chen, and Jing Liao. Direct-a-video: Customized video generation with user-directed camera movement and object motion. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–12, 2024. 3

[74] 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:2308.08089*, 2023. 2

[75] Jiayi Eris Zhang, Seungbae Bang, David IW Levin, and Alec Jacobson. Complementary dynamics. *arXiv:2009.02462*, 2020. 3

[76] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 3, 5

[77] 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:2406.19680*, 2024. 2

[78] Zhenghao Zhang, Junchao Liao, Menghao Li, Long Qin, and Weizhi Wang. Tora: Trajectory-oriented diffusion transformer for video generation. *arXiv:2407.21705*, 2024. 3

[79] Zhongwei Zhang, Fuchen Long, Yingwei Pan, Zhaofan Qiu, Ting Yao, Yang Cao, and Tao Mei. Trip: Temporal residual learning with image noise prior for image-to-video diffusion models. In *Proceedings of the IEEE/CVF Conference*

on Computer Vision and Pattern Recognition, pages 8671–8681, 2024. 2

[80] Rui Zhao, Yuchao Gu, Jay Zhangjie Wu, David Junhao Zhang, Jiawei Liu, Weijia Wu, Jussi Keppo, and Mike Zheng Shou. Motiondirector: Motion customization of text-to-video diffusion models. *arXiv:2310.08465*, 2023. 3