ControlHair: Synergizing Physics Simulator and Video Diffusion for Controllable Dynamic Hair Rendering

Weikai Lin[1*✉], Haoxiang Li[2], and Yuhao Zhu[1]

[1]University of Rochester, USA [2]Pixocial Technology, USA
wlin33@ur.rochester.edu, haoxiang@pixocial.com, yzhu@rochester.edu
✉Corresponding author.

Abstract. Hair simulation and rendering are challenging due to complex strand dynamics, diverse material properties, and intricate light–hair interactions. Recent video diffusion models can generate high-quality videos, but they lack fine-grained control over hair dynamics. We present ControlHair, a hybrid framework that integrates a physics simulator with conditional video diffusion to enable precise and controllable dynamic hair rendering. ControlHair adopts a three-stage pipeline: it first encodes physics conditions into per-frame geometry using a simulator, then extracts per-frame control signals, and finally feeds control signals into a video diffusion model to generate videos with desired hair dynamics. This cascaded design decouples physics reasoning from video generation, supports diverse physics, and makes training the video diffusion model easy. Trained on a curated 10K video dataset, ControlHair outperforms text- and pose-conditioned baselines, delivering precisely controlled hair dynamics. We also demonstrate three use cases of ControlHair, including dynamic hairstyle try-on, bullet-time effects, and cinemagraphic. Project page: https://linwk20.github.io/controlhair-web.

Keywords: Hair simulation · Controllable Video diffusion · Physics-informed rendering

1 Introduction

Human hair simulation and rendering are long-standing research problems in computer graphics [2, 13, 25, 31, 45], with applications ranging from films and games [5, 17] to VR/AR and digital humans [46, 49]. However, hair is exceptionally complex, involving tens of thousands of interacting strands as well as intricate material properties and light scattering. This complexity makes simulation and rendering challenging and computationally demanding.

Video diffusion models [11, 21, 30, 40, 51] have demonstrated the ability to generate high-quality videos rivaling professional production. Beyond common text- and image-conditioned generation, recent works have explored fine-grained

* This work was primarily conducted during an internship at Pixocial Technology.

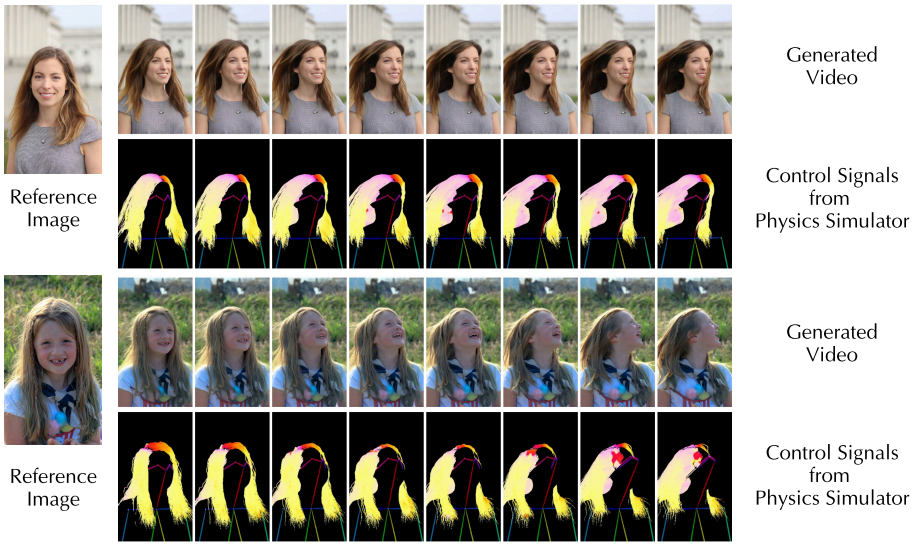

Fig. 1: Given a reference image, ControlHair converts physics conditions (e.g., hair stiffness, wind, or human motion) into per-frame control signals using a physics simulator. These signals, combined with a reference image, guide a diffusion model to generate photorealistic videos with controlled hair dynamics.

and multi-modal controls to enable practical usecases. Examples include conditioning video generation with human pose [26,42,43], lighting [52], or speech and music [3,39]. This suggests a promising direction for dynamic hair rendering: one can formulate it as a conditioned generation problem, where physics parameters serve as input conditions and the output is a video with desired hair dynamics.

Fig. 1 illustrates the overall setting: user-specified physics conditions are converted into geometry controls that guide video generation from a reference image. However, none of the existing conditional video diffusion methods support fine-grained control over dynamic hair rendering. This limitation arises from two challenges: (1) There are numerous physics parameters that influence hair dynamics, including intrinsic hair properties (e.g., mass, stiffness, damping) and external forces (e.g., gravity, wind). There is still no general encoding mechanism that can accommodate the diverse conditions. (2) training such conditional models would require videos annotated with ground-truth physics. However, physics like hair property or wind direction/strength are difficult to infer from videos.

To overcome the above challenges and enable dynamic hair rendering using diffusion models, we first formulate hair simulation and rendering as an image animation problem in the context of conditional video generation (Sec. 3). We then introduce ControlHair, a video diffusion framework for physically-based hair rendering, targeting portrait and digital-human hair rendering.

ControlHair uses a physics simulator to convert physical parameters into per-frame geometry, which then conditions the diffusion model. This design

makes ControlHair agnostic to the specific physics input and reduces annotation cost, since training only requires per-frame geometry rather than hard-to-infer physical parameters. ControlHair consists of three key components:

Simulator-based Physics Encoding. Physics conditions for hair rendering span a wide range of intrinsic parameters and external forces, often specific to the scene, task, or physics model. A straightforward approach is to directly feed these parameters into a diffusion model. However, this introduces two key challenges. First, there may be tens of parameters, making it difficult for the diffusion model to learn the effects of different physics factors. Second, any change in input physics requires redesigning the diffusion model, recollecting training data, and retraining the model, which limits flexibility and generality.

ControlHair addresses above issues by cascading a physics simulator before the diffusion model. We first estimate a 3D hair model from the input image. The simulator then takes physics parameters as input and use them to generate per-frame hair geometry (Sec. 4.1). This simulation+diffusion hybrid approach has two advantages. First, it offloads dynamics generation to the simulator, relaxing the burden of physics reasoning in the diffusion model. Second, we unify the diffusion model input as per-frame geometry. This decouples the diffusion model from specific physics settings: whenever the physics conditions change, we only need to swap the simulator, without redesigning downstream modules.

Per-frame Control Signals Extraction. After obtaining the per-frame hair geometry, we convert it into control signals for the video diffusion model (Sec. 4.2). We first project the 3D geometry onto 2D images along the desired camera trajectory. From these projections, we extract per-frame hair strand maps [56] and human poses [50] as control signals, which guide the video diffusion model on a per-frame basis. This control-signal design is motivated by three reasons. First, by projecting 3D geometry to 2D, we can control the poses of the generated frames. Second, strand maps and human poses depend only on geometry, which is available from simulation. Third, these per-frame control signals can be easily extracted from 2D frames in real-world videos, making data annotation easy. This greatly reduces the difficulty of training data annotation compared to inferring precise 3D geometry or underlying physics.

Conditioned Video Diffusion for Hair Rendering. Finally, we design a conditioned video diffusion model that takes a reference image (for appearance) and the per-frame control signals to generate desired RGB video (Sec. 4.3). The reference image is encoded using Wan encoder [40], while the per-frame control signals are encoded with a 3D convolution neural network. For training, we collect a large set of high-quality real-world videos containing hair and annotate them with strand maps and human poses to construct input control signals. The reference image is sampled from the video. The model is then trained to map the reference image and control signals to RGB videos. To leverage the prior of pretrained models, our model is finetuned from pretrained model using LoRA [14], which adapts the model with low-rank updates to the weight matrices.

Unlike purely data-driven models that lack fine-grained control over the physics in generated videos, ControlHair conditions on user-defined physics

simulation, enabling detailed, interpretable, and editable dynamics. Experiments show that ControlHair outperforms existing video diffusion models in controllable dynamic hair rendering. We further showcase its applicability with diverse use cases, including dynamic hair try-on, bullet-time effects, and cinemagraphic animation. Our contributions are summarized as follows:

- We formulate controllable dynamic hair rendering as a conditional video generation problem with physics inputs.
- We introduce the first video diffusion framework that incorporates a physics simulator to support fine-grained hair-related physics control.
- We design a control signals extraction pipeline to bridge simulators and diffusion models, enabling pose control and simplifying training data annotation.
- We design a conditional video diffusion model for hair rendering and construct a large-scale, high-quality training set.

2 Related Work

2.1 Controllable Video Diffusion Models

Video diffusion has achieved significant quality improvements through both data and computational scaling, with powerful closed-source models [9, 10, 30, 32] and open-source counterparts [21, 40, 51, 57]. Among them, the Wan series [40] has achieved extraordinary quality. Our diffusion model builds on the Wan 2.1 architecture, with modified interfaces and extra encoder modules to enable controllable dynamic hair rendering. Beyond text and image conditions, recent works incorporate additional control signals such as human pose [42,43], sketch [18,48], camera trajectory [12], audio [3, 39], style [52], and lighting [22]. PhysAnimator [47] uses a simulator to guide dynamics generation, but is limited to 2D mesh deformation. None of the existing works investigate control over the *physical dynamics* of fine human hair structures. Our work uniquely combines video diffusion with a physics simulator to encode hair-related physics.

2.2 Hair Simulation and Rendering

Human hair simulation and rendering are long-standing challenges in computer graphics. Simulation must account for tens of thousands of interacting strands, while rendering requires handling complex materials and light-scattering effects.

Various physics simulation methods have been proposed for hair dynamics, such as mass-spring chains [33], rigid multi-body chains [1], discrete elastic rods [2], Material Point Method [6], hybrid approaches [13], and learning-based methods [24]. These simulations are built on different physics models tailored for different tasks. Ideally, we should select the most suitable model given a task. However, the diversity of physics parameters across models makes it difficult to design a single diffusion model for all possible physics parameters input. For hair rendering, prior works have explored physically based [27, 29, 31, 58] and

Table 1: Input conditions for conditional video generation.

Reference Image	Hair Properties	External Forces	Camera Trajectory
Human, hair, environment lighting	Mass, stiffness, damping, thickness	Wind, gravity, friction, human motion	Pose per frame, viewpoint changes

data-driven methods [25, 45]. However, they either require complex modeling or fail to produce photorealistic results.

Our framework is compatible with arbitrary physics parameters and simulators because we unify the diffusion model input as per-frame geometry. This allows us to switch freely across different simulation models for a given task without redesigning the diffusion model. For hair rendering, we adopt video diffusion priors to avoid complex hair material and light modeling, while producing photorealistic and temporally consistent results.

Related hair-avatar systems, such as HHAvatar [23], DGH [41], and HAAR [37], focus on reconstructing or generating hairstyles, and are complementary to ControlHair's focus on controllable dynamic hair rendering.

3 Problem Formulation

We first formulate dynamic hair rendering within a conditional video generation framework. This formulation extends naturally to physics-conditioned generation of other deformable objects. Concretely, we model it as image animation problem:the inputs include a reference image I_{ref}, physics parameters $\mathcal{P}$, and a camera trajectory $\mathcal{T}_{\text{cam}}$. The output is a video V that preserves the reference appearance while following the dynamics dictated by $\mathcal{P}$ and camera trajectory defined by $\mathcal{T}_{\text{cam}}$, which can be written as:

$$\mathrm{V} = \mathcal{G}(\mathrm{I}_{\text{ref}}, \mathcal{P}, \mathcal{T}_{\text{cam}}) \tag{1}$$

where $\mathcal{G}$ denotes the conditional video generator. We now specify the physics parameters $\mathcal{P}$ that govern hair dynamics. We categorize them into two groups: intrinsic hair properties ($\mathcal{P}_{\text{hair}}$) and external forces ($\mathcal{F}$). The former characterizes hair geometry and physical attributes that determine how strand responses to external stimuli (e.g. mass, stiffness). The latter captures external forces in the scene such as wind or gravity. We summarize the conditions and examples in Tbl. 1. Note that the physics parameters $\mathcal{P}$ are not fixed, as they depend on the task and the underlying physical modeling principles. We can rewrite the formulation in Eqn. 1 as:

$$\mathrm{V} = \mathcal{G}\big(\mathrm{I}_{\text{ref}},\, \mathcal{P}_{\text{hair}},\, \mathcal{F}, \mathcal{T}_{\text{cam}}\big) \tag{2}$$

where $\mathcal{G}$ must fuse I_{ref}, $\mathcal{P}_{\text{hair}}$, $\mathcal{F}$, and $\mathcal{T}_{\text{cam}}$ to produce a video in which appearance, hair dynamics, and camera motion are faithfully controlled. However, directly feeding these conditions into $\mathcal{G}$ makes generation difficult. To address

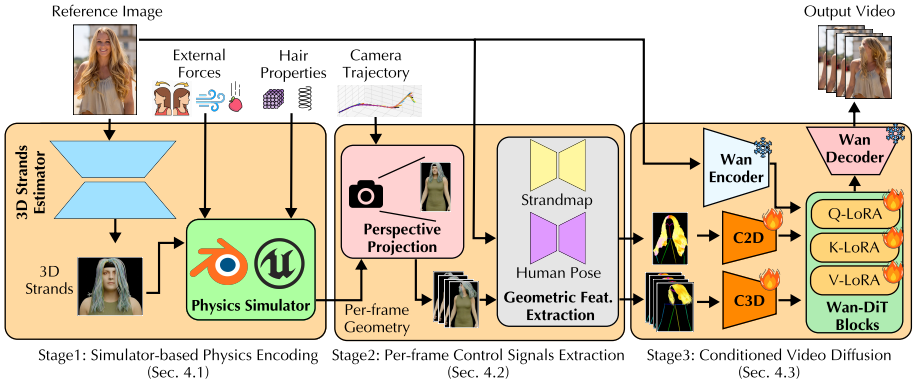

Fig. 2: ControlHair enables controllable dynamic hair rendering via three stages: (1) Simulator-based Physics Encoding (Sec. 4.1), (2) Per-frame Control Signal Extraction (Sec. 4.2), and (3) Conditioned Video Diffusion for Hair Rendering (Sec. 4.3).

this, people usually employ some form of encoder that transforms the conditions into control signals better suited for $\mathcal{G}$. For instance, one of the most widely adopted approaches, ControlNet [53], unifies all conditions into control signals that share the same shape as the target output, i.e., the video V in our formulation. Based on this idea, we can rewrite Eqn. 2 as:

$$\mathrm{V} = \mathcal{G}\big(E_I(\mathrm{I}_{\text{ref}}),\, E_{\text{hair}}(\mathcal{P}_{\text{hair}}),\, E_F(\mathcal{F}),\, E_T(\mathcal{T}_{\text{cam}})\big) \tag{3}$$

where E_I, E_{hair}, E_F, E_T denote encoders that transform the corresponding input conditions into control signals consumed by $\mathcal{G}$. Prior work has studied encoding the reference image for appearance control [40] and encoding camera trajectories for pose control [12]. However, accurately encoding dynamics-related physics, such as $\mathcal{P}_{\text{hair}}$ and $\mathcal{F}$, remains unexplored. The difficulty arises because of two reasons: (1) the effects of physics parameters are often highly entangled, making it difficult for a model to disentangle their effects. For example, lower damping, weaker gravity, or stronger wind can all lead to similar visual outcomes, such as elevated hair strands. (2) The parameter set $\mathcal{P}$ may vary across physics models designed for different tasks, which makes it challenging to design a single diffusion model that handles all inputs. Based on these reasons, an ideal condition encoding mechanism must both ease model learning and remain flexible to accommodate diverse physics inputs.

4 Method

In ControlHair, we investigate physics encodings that is both general and precise, which also reduce training data annotation cost. ControlHair adopts a three-stage pipeline as shown in Fig. 2. First, we encode physics conditions into per-frame hair geometry using a physics simulator (Sec. 4.1). Second, we

extract per-frame control signals by (1) projecting the geometry onto the 2D image along the camera trajectory and (2) extracting strand maps and human poses (Sec. 4.2). Finally, the per-frame control signals, together with the reference frame, are fed into a video diffusion model to generate the output video (Sec. 4.3).

We can also map stages in ControlHair to Eqn. 3. The first two stages encode physics conditions and camera trajectories to produce control signals, corresponding to $E_{\text{hair}}, E_F, E_T$. The final stage employs E_I from Wan 2.1 [40] to encode the reference image and uses a video diffusion model as $\mathcal{G}$ that jointly processes all encoded control signals. We describe each stage in detail below.

4.1 Simulator-based Universal Physics Encoding

Motivation. Encoding physics into video generation is challenging. First, many parameters have highly entangled effects. For example, decreasing hair mass, increasing wind force, or reducing gravity can all amplify hair motion, making it difficult to disentangle their contributions and learn the correct physics. Second, different target effects often require different physical modelings, which introduce distinct physics parameters. It is therefore difficult to design a universal video diffusion that can handle arbitrary parameters across diverse physics models.

To address the above challenges, ControlHair cascades a physics simulator before running video diffusion (Stage 1 in Fig. 2). The simulator takes as input user-defined physics parameters, i.e., intrinsic hair properties and external forces (Tbl. 1), together with a 3D hair strand model estimated from the reference image. It then performs time-step simulations on the 3D hair model to infer per-frame hair geometry. The process can be formulated as:

$$\mathrm{H}_{1:\mathrm{T}} = \mathcal{S}(\mathrm{H}, \mathcal{P}_{\text{hair}}, \mathcal{F}), \tag{4}$$

where H is the 3D hair model estimated from the reference image, $\mathcal{P}_{\text{hair}}$, $\mathcal{F}$ represent hair intrinsic properties and external forces, respectively, and $\mathcal{S}$ is a physics simulator $\mathrm{H}_{1:\mathrm{T}}$ is per-frame hair geometry over T frames.

We encode $\mathcal{P}_{\text{hair}}$ by configuring hair-related physical parameters in the simulator. For external forces $\mathcal{F}$, standard settings cover most cases, such as wind strength/direction and gravity. To account for the effect of human motion (a component of $\mathcal{F}$) on hair dynamics, we instantiate a digital human in the simulator, attach hair strands to its scalp, and drive the character with the desired motions. This setup allows human motion to directly influence hair behavior. The simulator finally outputs $\mathrm{H}_{1:\mathrm{T}}$. The $\mathrm{H}_{1:\mathrm{T}}$ is then encoded into per-frame control signals (Sec. 4.2), which will be used to guide the video generation (Sec. 4.3). Note that we are free to switch among different simulators ($\mathcal{S}$) and physics conditions ($\mathcal{P}_{\text{hair}}$, $\mathcal{F}$) as long as the output of the simulator is per-frame hair geometry.

This cascaded, decoupled design offers two key advantages. First, it delegates the physics reasoning to simulator, leaving the video diffusion model to focus on photorealistic inpainting based on simulated geometry, thereby reducing its burden. Second, it unifies diverse physics parameters into a common representation: per-frame hair geometry. As a result, users can modify the parameter set for

different tasks simply by switching to a suitable simulator, without altering the downstream design. However, estimating 3D hair strands from a single image may introduce geometric errors, as it is an under-constrained problem. These errors could propagate into the generated video. Fortunately, we observe that the diffusion model, guided by the reference image, often corrects mismatches caused by this imperfect reconstruction. We further discuss this in Sec. 6.

4.2 Per-frame Control Signals Extraction

After obtaining per-frame hair geometry $\mathrm{H}_{1:\mathrm{T}}$ from the simulator, we convert it into control signals for video diffusion. Here, the inputs include $\mathrm{H}_{1:\mathrm{T}}$ and a human model sequence $\mathrm{M}_{1:\mathrm{T}}$, defined by the sequence of human poses in $\mathcal{F}$. This stage then outputs per-frame control signals for the video diffusion model, which control both hair dynamics and human motion. Inspired by recent controllable generation [42, 43, 53], we define the control signals to have the same shape as the output video ($\mathrm{T} \times \mathrm{H} \times \mathrm{W}$), thereby providing pixel- and frame-wise control.

We impose two requirements on the control signals: (1) they must reflect the desired camera trajectory $\mathcal{T}_{\mathrm{cam}}$ to allow camera pose control, and (2) they should depend only on geometry, as the physics simulator produces purely geometric results. ControlHair generates the control signals in two steps (Stage 2 in Fig. 2): (1) trajectory-aware 3D-to-2D perspective projection and (2) geometric feature extraction. The first step ensures that the camera trajectory is encoded, and the second ensures the control signals depend only on geometry.

Trajectory-aware perspective projection. To generate control signal for frame t, we first project the hair geometry and human model at time t to 2D image using the camera pose $\mathrm{T}_{\mathrm{cam,t}}$, producing a projected image $\mathrm{I}_{\mathrm{proj,t}}$. Formally:

$$\mathrm{I}_{\mathrm{proj,t}} = \mathit{\Pi}(\mathrm{H}_{\mathrm{t}}, \mathrm{M}_{\mathrm{t}}; \mathrm{T}_{\mathrm{cam,t}}) \tag{5}$$

where $\mathit{\Pi}(\cdot)$ denotes the perspective projection, which projects 3D objects onto a 2D image with the same resolution as the target video. H_{t} is the simulated hair geometry at time t, and M_{t} represents the human model at time t, decided by the human pose at time t. This operation is applied to all frames along the camera trajectory, yielding $\mathrm{I}_{\mathrm{proj,1:T}}$. This design enables explicit control over camera pose, as validated by the bullet-time effect in Fig. 7.

Geometric feature extraction. After obtaining $\mathrm{I}_{\mathrm{proj,1:T}}$, we extract geometric features as control signals. For hair, we use strand map [56], which resemble orientation map but disambiguate directional conflicts. For human pose, we use sparse 2D Dwpose encodings [50]. Since the simulated 3D geometry is available at inference time, strand maps can also be extracted analytically, which may be more efficient. However, since analytical strand maps are unavailable for the training data, we use estimated strand maps as in HairStep [56] and apply the same estimator at inference time to maintain training-inference consistency.

The two features are then merged into a three-channel RGB control image using a hair mask B_{hair} that indicates where hair occludes the human body.

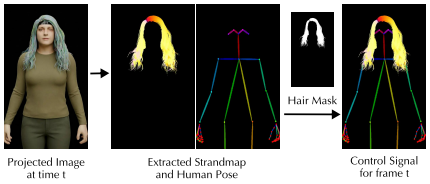

Fig. 3: Example of extracting control signals from a projected image. Control signals are then fed into the diffusion model.

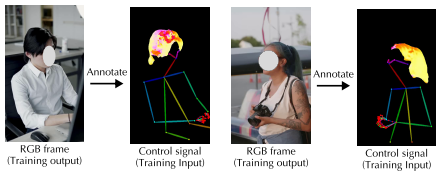

Fig. 4: Training data annotation. Extracting our control signals from videos is easier than inferring the physics.

Formally, the geometric extraction G can be written as:

$$\mathrm{C_t} = G(\mathrm{I_{proj,t}}) = \Phi_{\mathrm{strand}}(\mathrm{I_{proj,t}}) \cdot B_{\mathrm{hair}} + \Phi_{\mathrm{pose}}(\mathrm{I_{proj,t}}) \cdot (1 - B_{\mathrm{hair}}) \tag{6}$$

where $\Phi_{\mathrm{strand}}(\cdot)$ and $\Phi_{\mathrm{pose}}(\cdot)$ denote the feature extraction processes for strand map and human pose, respectively, both are neural networks in our implementation. $\mathrm{C_t}$ is the control signal for frame t which has same resolution as the output video. We show an example of control signal extraction in Fig. 3. We obtain $\mathrm{C_{1:T}}$ by applying this extraction to all projected frames $\mathrm{I_{proj,1:T}}$, which serve as per-frame control signals for video diffusion model (Sec. 4.3).

4.3 Conditioned Video Diffusion for Hair Rendering

With per-frame control signals $\mathrm{C_{1:T}}$ and the reference image $\mathrm{I_{ref}}$, we use a conditional video diffusion model $\mathcal{G}$ to generate the video $\mathrm{V_{1:T}}$ (Stage 3 in Fig. 2):

$$\mathrm{V_{1:T}} = \mathcal{G}\big(E_I(\mathrm{I_{ref}}),\, E_{\mathrm{C3D}}(\mathrm{C_{1:T}}), E_{\mathrm{C2D}}(G(\mathrm{I_{ref}}))\big) \tag{7}$$

where $\mathrm{C_{1:T}}$ provides frame-wise geometric constraints, while the reference image $\mathrm{I_{ref}}$, encoded by E_I, specifies appearance constraints. We use Wan 2.1 encoder [40] as E_I. Before feeding the control signals into the diffusion backbone, we encode them using a 3D convolutional network E_{C3D}. In addition, we extract geometric features (Eqn. 6) from the reference image, encode them with a 2D convolutional network E_{C2D}. Our design follows previous controllable diffusion models [42, 43, 53]. All encodings are fed into a Wan 2.1 diffusion transformer (DiT), which generates video with controlled hair dynamics and human motion.

Data Annotation Made Easy. No existing diffusion model supports our control signals. To address this, we fine-tune a pretrained model with paired control signals and RGB videos. Through this training, the model learns to map control signals, together with a reference image, into video. To obtain such data, we annotate RGB videos with control signals and sample one frame from each video as the reference image. Specifically, we run geometric feature extraction (Eqn. 6) on every video frame, with B_{hair} estimated by a hair segmentation network. This gives us the control signals. We show annotation in Fig. 4.

ControlHair makes training data annotation easier because inferring underlying physics from videos is challenging. To enable high-quality training, we

collect 30K real-world videos containing hair. We crop each video to focus the upper body and hair, and filter samples by resolution and aspect ratio. After annotation, we obtain a curated dataset of 10K videos. **Training.** To leverage priors from pretrained diffusion models, we initialize our model with pretrained weights from UniAnimate-DiT [43] and fine-tune it with our annotated dataset to learn the mapping from new control signals to videos. During training, we keep the Wan encoder and decoder [40] fixed. We fully fine-tune only the convolutional networks. For the DiT backbone, we apply LoRA [14], which adapts the model by adding only low-rank components to the weights. The motivation is that the DiT backbone contains most of the parameters and prior knowledge; using low-rank fine-tuning not only reduces computation but also prevents overfitting to our dataset.

5 Evaluation

5.1 Experimental Setup

Dataset. We train ControlHair on a dataset (Sec. 4.3) covering diverse videos. To characterize the training data, we annotate the visual appearance and motion of randomly sampled videos and report in the Supplementary Material B.

For image animation evaluation, we use long-hair images from FFHQ-Wild [20] to assess rich hair dynamics, since short hair exhibits limited motion. For control-signal-to-video evaluation (Sec. 5.3), we generate ∼100 candidate videos with wind- or human-motion-driven hair dynamics using Google VEO 2 [10], filter visually similar clips, and select 10 diverse videos as the evaluation set.

Implementation Details. We describe the design choices of each module in Fig. 2:

- In **Stage 1**, we adopt DiffLocks [34] to extract 3D hair strands from a single image, and use Blender's particle system [7] to simulate hair dynamics. The simulator exposes configurable physical parameters, including hair properties such as mass, stiffness, and damping, and external forces such as wind, human motion, and gravity. We use these parameters as user controls for better controllability and keep unspecified parameters at their default values. Unless otherwise specified, we use mass = 0.1, stiffness = 6, damping = 9, wind strength = 10, and gravity = 1 (scale factor) as a stable default setting. For specific cases, users may tune these parameters, and future work could use vision-language models to predict suitable values.
- In **Stage 2**, we use Blender's camera system for perspective projection. The strand maps are extracted with a pretrained U-Net from HairStep [56], and 2D human poses are estimated using DWPose [50].
- In **Stage 3**, we follow the network design of UniAnimate-DiT [43] and adopt the encoder, decoder, and DiT from Wan 2.1 [40]. We set the output video resolution to $81 \times 832 \times 480$ (T×H×W) for all experiments.

During training of diffusion model, we fix the encoder and decoder from Wan, fully finetune the convolutional network, and train the DiT with LoRA [14] (rank

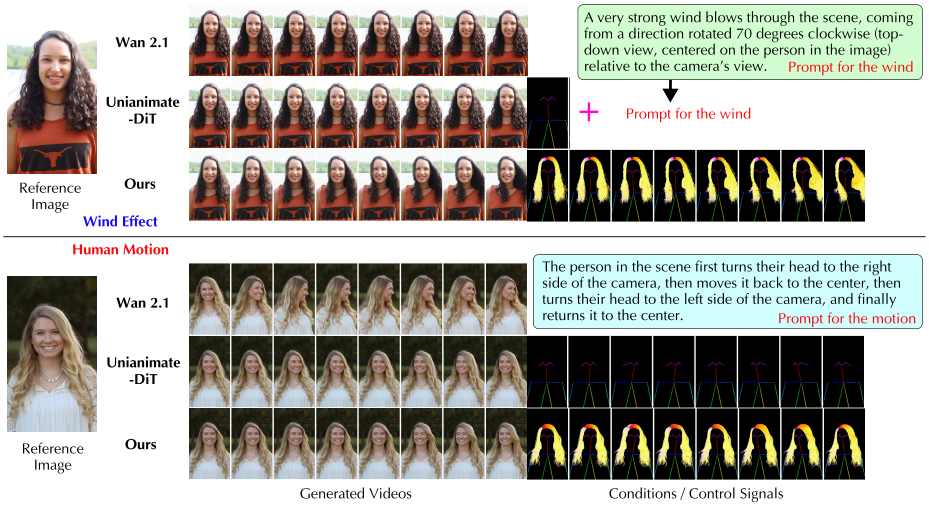

Fig. 5: Qualitative comparison with baselines under strong wind (top) and human motion (bottom). Only ControlHair generates desired control of hair dynamics.

= 128) to enable hair-conditioned video diffusion. Training is performed with a learning rate of 1×10^{-4}, a batch size of 4, and 10K iterations, taking three days on four H100 80GB PCIe GPUs.

Runtime Analysis. We report the runtime of ControlHair on a machine equipped with an AMD EPYC 9354 CPU and an NVIDIA RTX Pro 6000 Max-Q GPU for generating an 81-frame video. Hair reconstruction takes 22 seconds, Blender simulation takes 326 seconds, control-signal extraction takes 21 seconds, and video diffusion takes 853 seconds. The full pipeline therefore takes 1222 seconds, or about 20 minutes, per video. The current system is designed for offline generation rather than real-time rendering. The main runtime bottlenecks are Blender simulation, which lacks GPU acceleration in our setup, and high-quality video diffusion inference. We treat real-time interactive inference as future work.

Baselines. Since no prior video generation method targets controllable hair physics, we compare ControlHair with two intuitive alternatives to show that ControlHair enables more precise and detailed control over hair dynamics:

- Wan2.1 [40]: a strong video diffusion model that supports image-to-video generation conditioned on text prompts. We provide detailed text prompts to describe the physics.
- UniAnimate-DiT [43]: a image-to-video generation framework conditioned on per-frame human pose and text.

Table 2: Quantitative comparison between generated and ground-truth results.

Method	PSNR ↑	SSIM ↑	LPIPS ↓
Wan 2.1 [40]	11.65	0.518	0.499
UniAnimate-DiT [43]	15.78	0.637	0.390
ControlHair (Ours)	**17.15**	**0.668**	**0.370**

5.2 Qualitative Comparison

We evaluate ControlHair against baselines under two physics settings. Setting 1 applies strong wind to the hair, while Setting 2 applies head motion that induces hair dynamics The results are shown in Fig. 5.

In Setting 1 (top), we add a strong wind to blow the hair of the reference image. The wind direction is rotated 70° clockwise (top-down view) from the camera view, centered on the person. For the baselines, the only way to represent wind is through text prompts. We provided detailed prompts explicitly describing strong wind effects, as shown in Fig. 5. However, even with aggressive wording (e.g., "strong wind"), both baselines fail to move the hair, highlighting the difficulty of controlling fine-grained physical effects via text conditions. In contrast, ControlHair leverages control signals from a physics simulator and is able to drive hair motion in the expected direction.

In Setting 2 (bottom), the user's head is required to first rotate to the right of the camera, then to the left, and finally return to the center. We evaluate two aspects: whether the head motion follows the specified sequence, and whether the hair dynamics are controllable. In Wan2.1, we use detailed text prompt shown in Fig. 5. Although the prompt clearly specifies the intended motion, the generated video shows the head turning only to the right but never to the left, violating the required sequence. UniAnimate-DiT successfully controls the head pose, as it is conditioned on human poses, however, it cannot control hair dynamics and instead relies solely on the diffusion prior. In contrast, ControlHair achieves accurate control of both head motion and hair dynamics by following simulator-provided signals, yielding more controllable results.

5.3 Quantitative Comparison

We now evaluate qualitatively how well ControlHair leverages and follow the control signals. We use the control-signal-to-video reconstruction task, which is widely adopted in controllable generation [15,19,42]. Specifically, we treat videos generated by Google VEO 2 [10] as ground truth (the user-expected video) and extract control signals from them, which are then used to reconstruct the original videos. For Wan2.1, we use manually annotated text prompts as control signals. For UniAnimate-DiT, we include additional per-frame human poses extracted using DWPose [50]. For ControlHair, control signals include per-frame human poses and hair strand maps, which is extracted using Eqn. 6. Our goal is to test if each model can follow the control signals to reconstruct the expected video.

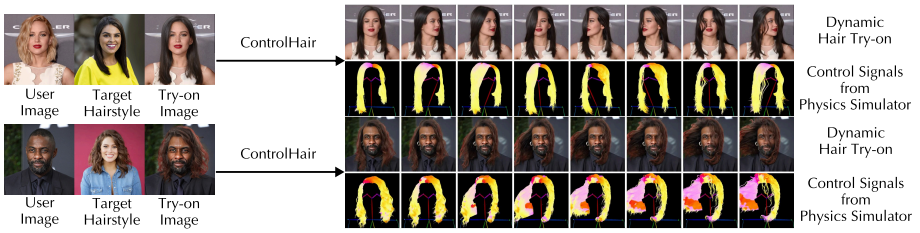

Fig. 6: Dynamic Hair Try-on using ControlHair. We cascade ControlHair after existing static image-based hair try-on frameworks to enable dynamic hair try-on.

The results are shown in Tbl. 2, where we compare the generated and ground-truth videos using widely adopted frame-similarity metrics: PSNR [16], SSIM [44], and LPIPS [54]. The results are averaged across all frames in all videos. Wan 2.1 performs the worst, as text prompts cannot precisely capture per-frame physics. UniAnimate-DiT performs better since it has additional human poses input. Among all methods, ControlHair achieves the best results, achieving improvements of up to 47%, 29%, and 26% in PSNR, SSIM, and LPIPS. This indicates: (1) ControlHair has the highest physics controllability, and (2) ControlHair can leverage hair-dynamics control signals (strand maps).

5.4 Use Cases

To demonstrate the applicability of ControlHair, we showcase three representative use cases: dynamic hair try-on, bullet-time hair effects, and cinemagraphic.

Dynamic Hair Try-On. Hair try-on is an important task: it allows users to preview whether a hairstyle suits them before making changes. However, prior work has focused only on static try-on [4, 55], limiting users' understanding of dynamic hairstyle appearance. To enable dynamic hair try-on, we cascade ControlHair after the static hair try-on system HairFusion [4].

The results are shown in Fig. 6. Given a user image and a target hairstyle, we first apply HairFusion to obtain a static try-on image. We then use ControlHair to introduce dynamics such as head rotations (first row) or wind effects (second row), yielding the final dynamic hair try-on videos. The results show that by combining ControlHair with a static try-on system, we extend static hair try-on to dynamic hair try-on with detailed and controllable dynamics.

Bullet-Time Hair Effects. As discussed in Sec. 4.2, we can control the camera trajectory of video generation by manipulating the perspective projection. This enables bullet-time effect: at a specific moment of physics simulation, we can freeze the simulator to lock the hair geometry and render the scene from different viewpoints. This allows users to observe the hair from different perspectives at the same instant. We showcase this application in Fig. 7 by first applying a left-to-right wind in the physics simulator. We then freeze the simulator. After freezing, we rotate the camera around the person: 20° to the right, followed by 40° to the left. The hair is frozen, and the camera trajectory is well-controlled.

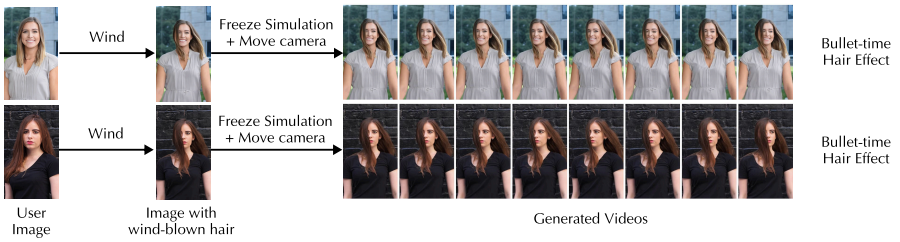

Fig. 7: Bullet-Time Effects. We freeze the physics simulator and rotate the camera around the user to produce the bullet-time effect.

Fig. 8: Cinemagraphic. We animate only the hair and append reverse playback to the video, achieving a cinemagraphic effect.

Cinemagraphic Effects. Beyond animating images, ControlHair can generate unconventional physical effects such as zero gravity, extremely stiff or soft hair, or unusually strong wind. Such effects are difficult to generate using other diffusion models if they are absent from the training data. To showcase this capability, we use ControlHair to generate videos with cinemagraphic effects. Specifically, we design the effect such that the human subject remains static while only the hair is animated, and the video is looping. Such effect is difficult to capture in the real world, and producing them requires post-processing.

To achieve this effect, we fix the human pose in the control signals while applying wind to the hair. Looping is enabled by appending a reversed playback segment to the video. The results are shown in Fig. 8, where the model successfully freezes the human while animating the hair.

6 Discussion

ControlHair may also introduce two kinds of errors. First, the 3D hair model estimated from a single input image in Stage 1 can be inaccurate because single-view reconstruction is under-constrained. The video diffusion model can sometimes correct these errors using the reference image, and we show such failure cases in the Supplementary Material D. Future work could mitigate this issue by using multi-view priors [8, 28, 35, 38] to infer more accurate 3D hairstyles.

Second, the performance of the physics simulator affects the final results. If the chosen simulator is imperfect, the generated video may look unrealistic. However, we observe that when the physics simulator is sufficiently accurate, as in the control-signal-to-video tasks (Sec. 5.3), our diffusion model can generate highly realistic results. To further improve simulation quality, one can adopt a more advanced simulator [36] or even a neural simulator learned from data [24].

Extreme Cases. We also tested extreme scenarios such as multiple wind sources combined with drastic human motion. Experimental results demonstrate that ControlHair can handle these extreme cases. We found Blender physics simulator might diverge under certain extreme physics or very short hair; however, this is related to the choice of simulator and is independent of the our framework. We also show full-body and out-of-distribution examples with large motion in the Supplementary Material C; large motion may still cause blurred frames, a common limitation of current video diffusion models.

7 Conclusion

We presented ControlHair, a three-stage pipeline decouples physics reasoning from video generation, integrates physics simulators with video diffusion to enable fine-grained control over the physics. Trained on a curated dataset, ControlHair outperforms baselines in both qualitative and quantitative evaluations. ControlHair also enables diverse applications, including dynamic hairstyle try-on, bullet-time effects, and cinemagraphic. The hybrid design in ControlHair opens possibilities for embedding physics into generative video models, paving the way toward controllable and physically grounded rendering.

References

1. Anjyo, K.i., Usami, Y., Kurihara, T.: A simple method for extracting the natural beauty of hair. In: Proceedings of the 19th annual conference on Computer graphics and interactive techniques. pp. 111–120 (1992)
2. Bergou, M., Wardetzky, M., Robinson, S., Audoly, B., Grinspun, E.: Discrete elastic rods. In: ACM Siggraph 2008 Papers, pp. 1–12 (2008)
3. Chen, Z., Cao, J., Chen, Z., Li, Y., Ma, C.: Echomimic: Lifelike audio-driven portrait animations through editable landmark conditions. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 2403–2410 (2025)
4. Chung, C., Park, S., Kim, J., Choo, J.: What to preserve and what to transfer: Faithful, identity-preserving diffusion-based hairstyle transfer. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 2582–2590 (2025)
5. Epic Games: An early look at next-generation real-time hair and fur. Unreal Engine Tech Blog (2020), `https://www.unrealengine.com/en-US/tech-blog/an-early-look-at-next-generation-real-time-hair-and-fur`, Accessed 23 June 2026
6. Fei, Y., Huang, Y., Gao, M.: Principles towards real-time simulation of material point method on modern gpus. arXiv preprint arXiv:2111.00699 (2021)
7. Foundation, B.: Blender (2025), `http://www.blender.org`, Accessed 23 June 2026

8. Gao, R., Holynski, A., Henzler, P., Brussee, A., Martin-Brualla, R., Srinivasan, P., Barron, J.T., Poole, B.: Cat3d: Create anything in 3d with multi-view diffusion models. arXiv preprint arXiv:2405.10314 (2024)
9. Germanidis, A.: Introducing gen-3 alpha: A new frontier for video generation. `https://runwayml.com/research/introducing-gen-3-alpha` (2024), runway Research. Accessed 23 June 2026
10. Google DeepMind: Veo. `https://deepmind.google/models/veo/` (2025), google DeepMind. Accessed 23 June 2026
11. HaCohen, Y., Chiprut, N., Brazowski, B., Shalem, D., Moshe, D., Richardson, E., Levin, E., Shiran, G., Zabari, N., Gordon, O., et al.: Ltx-video: Realtime video latent diffusion. arXiv preprint arXiv:2501.00103 (2024)
12. He, H., Xu, Y., Guo, Y., Wetzstein, G., Dai, B., Li, H., Yang, C.: Cameractrl: Enabling camera control for text-to-video generation. arXiv preprint arXiv:2404.02101 (2024)
13. Hsu, J., Wang, T., Pan, Z., Gao, X., Yuksel, C., Wu, K.: Sag-free initialization for strand-based hybrid hair simulation. ACM Transactions on Graphics (TOG) **42**(4), 1–14 (2023)
14. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685 **10** (2021)
15. Hu, L.: Animate anyone: Consistent and controllable image-to-video synthesis for character animation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8153–8163 (2024)
16. Huynh-Thu, Q., Ghanbari, M.: Scope of validity of psnr in image/video quality assessment. Electronics letters **44**(13), 800–801 (2008)
17. Iben, H., Meyer, M., Petrovic, L., Soares, O., Anderson, J., Witkin, A.: Artistic simulation of curly hair. In: Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation. pp. 63–71 (2013)
18. Jiang, L., Chen, S., Wu, B., Guan, X., Zhang, J.: Vidsketch: Hand-drawn sketch-driven video generation with diffusion control. arXiv preprint arXiv:2502.01101 (2025)
19. Karras, J., Holynski, A., Wang, T.C., Kemelmacher-Shlizerman, I.: Dreampose: Fashion video synthesis with stable diffusion. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 22680–22690 (2023)
20. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4401–4410 (2019)
21. Kong, W., Tian, Q., Zhang, Z., Min, R., Dai, Z., Zhou, J., Xiong, J., Li, X., Wu, B., Zhang, J., et al.: Hunyuanvideo: A systematic framework for large video generative models. arXiv preprint arXiv:2412.03603 (2024)
22. Liang, R., Gojcic, Z., Ling, H., Munkberg, J., Hasselgren, J., Lin, C.H., Gao, J., Keller, A., Vijaykumar, N., Fidler, S., et al.: Diffusion renderer: Neural inverse and forward rendering with video diffusion models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 26069–26080 (2025)
23. Liao, Z., Xu, Y., Li, Z., Li, Q., Zhou, B., Bai, R., Xu, D., Zhang, H., Liu, Y.: Hhavatar: Gaussian head avatar with dynamic hairs. IEEE Transactions on Pattern Analysis and Machine Intelligence (2025)
24. Lin, G.W.C., Larionov, E., Chen, H.y., Roble, D., Stuyck, T.: Neuralocks: Real-time dynamic neural hair simulation. arXiv preprint arXiv:2507.05191 (2025)

25. Luo, H., Ouyang, M., Zhao, Z., Jiang, S., Zhang, L., Zhang, Q., Yang, W., Xu, L., Yu, J.: Gaussianhair: Hair modeling and rendering with light-aware gaussians. arXiv preprint arXiv:2402.10483 (2024)
26. Ma, Y., He, Y., Cun, X., Wang, X., Chen, S., Li, X., Chen, Q.: Follow your pose: Pose-guided text-to-video generation using pose-free videos. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 4117–4125 (2024)
27. Marschner, S.R., Jensen, H.W., Cammarano, M., Worley, S., Hanrahan, P.: Light scattering from human hair fibers. ACM Transactions on Graphics (TOG) **22**(3), 780–791 (2003)
28. Miao, K., Agrawal, H., Zhang, Q., Semeraro, F., Cavallo, M., Gu, J., Toshev, A.: Dsplats: 3d generation by denoising splats-based multiview diffusion models. arXiv preprint arXiv:2412.09648 (2024)
29. Moon, J.T., Walter, B., Marschner, S.: Efficient multiple scattering in hair using spherical harmonics. In: ACM SIGGRAPH 2008 papers, pp. 1–7 (2008)
30. OpenAI: Sora. `https://openai.com/sora/` (2024), Accessed 23 June 2026
31. Petrovic, L., Henne, M., Anderson, J.: Volumetric methods for simulation and rendering of hair. Pixar Animation Studios **2**(4), 1–6 (2005)
32. Polyak, A., Zohar, A., Brown, A., Tjandra, A., Sinha, A., Lee, A., Vyas, A., Shi, B., Ma, C.Y., Chuang, C.Y., et al.: Movie gen: A cast of media foundation models. arXiv preprint arXiv:2410.13720 (2024)
33. Rosenblum, R.E., Carlson, W.E., Tripp III, E.: Simulating the structure and dynamics of human hair: modelling, rendering and animation. The Journal of Visualization and Computer Animation **2**(4), 141–148 (1991)
34. Rosu, R.A., Wu, K., Feng, Y., Zheng, Y., Black, M.J.: Difflocks: Generating 3d hair from a single image using diffusion models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 10847–10857 (2025)
35. Shi, Y., Wang, P., Ye, J., Long, M., Li, K., Yang, X.: Mvdream: Multi-view diffusion for 3d generation. arXiv preprint arXiv:2308.16512 (2023)
36. SideFX: Houdini. `https://www.sidefx.com/` (2025), Accessed 23 June 2026
37. Sklyarova, V., Zakharov, E., Hilliges, O., Black, M.J., Thies, J.: Text-conditioned generative model of 3d strand-based human hairstyles. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4703–4712 (2024)
38. Tang, L., Jia, M., Wang, Q., Phoo, C.P., Hariharan, B.: Emergent correspondence from image diffusion. Advances in Neural Information Processing Systems **36**, 1363–1389 (2023)
39. Tian, L., Hu, S., Wang, Q., Zhang, B., Bo, L.: Emo2: End-effector guided audio-driven avatar video generation. arXiv preprint arXiv:2501.10687 (2025)
40. Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., et al.: Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314 (2025)
41. Wang, J., Xu, Y., Tretschk, E., Wang, Z., Ianina, A., Bozic, A., Neumann, U., Tung, T.: Dgh: Dynamic gaussian hair. Advances in Neural Information Processing Systems **38**, 45560–45594 (2026)
42. Wang, X., Zhang, S., Gao, C., Wang, J., Zhou, X., Zhang, Y., Yan, L., Sang, N.: Unianimate: Taming unified video diffusion models for consistent human image animation. arXiv preprint arXiv:2406.01188 (2024)
43. Wang, X., Zhang, S., Tang, L., Zhang, Y., Gao, C., Wang, Y., Sang, N.: Unianimate-dit: Human image animation with large-scale video diffusion transformer. arXiv preprint arXiv:2504.11289 (2025)

44. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)
45. Wang, Z., Nam, G., Stuyck, T., Lombardi, S., Cao, C., Saragih, J., Zollhöfer, M., Hodgins, J., Lassner, C.: Neuwigs: A neural dynamic model for volumetric hair capture and animation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8641–8651 (2023)
46. Ward, K., Galoppo, N., Lin, M.C.: A simulation-based vr system for interactive hairstyling. In: IEEE Virtual Reality Conference (VR 2006). pp. 257–260. IEEE (2006)
47. Xie, T., Zhao, Y., Jiang, Y., Jiang, C.: Physanimator: Physics-guided generative cartoon animation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 10793–10804 (2025)
48. Xing, J., Liu, H., Xia, M., Zhang, Y., Wang, X., Shan, Y., Wong, T.T.: Tooncrafter: Generative cartoon interpolation. ACM Transactions on Graphics (TOG) **43**(6), 1–11 (2024)
49. Xu, Y., Chen, B., Li, Z., Zhang, H., Wang, L., Zheng, Z., Liu, Y.: Gaussian head avatar: Ultra high-fidelity head avatar via dynamic gaussians. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1931–1941 (2024)
50. Yang, Z., Zeng, A., Yuan, C., Li, Y.: Effective whole-body pose estimation with two-stages distillation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4210–4220 (2023)
51. Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., et al.: Cogvideox: Text-to-video diffusion models with an expert transformer. arXiv preprint arXiv:2408.06072 (2024)
52. Ye, Z., Huang, H., Wang, X., Wan, P., Zhang, D., Luo, W.: Stylemaster: Stylize your video with artistic generation and translation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 2630–2640 (2025)
53. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 3836–3847 (2023)
54. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 586–595 (2018)
55. Zhang, Y., Zhang, Q., Song, Y., Zhang, J., Tang, H., Liu, J.: Stable-hair: Real-world hair transfer via diffusion model. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 10348–10356 (2025)
56. Zheng, Y., Jin, Z., Li, M., Huang, H., Ma, C., Cui, S., Han, X.: Hairstep: Transfer synthetic to real using strand and depth maps for single-view 3d hair modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12726–12735 (2023)
57. Zheng, Z., Peng, X., Yang, T., Shen, C., Li, S., Liu, H., Zhou, Y., Li, T., You, Y.: Open-sora: Democratizing efficient video production for all. arXiv preprint arXiv:2412.20404 (2024)
58. Zinke, A., Yuksel, C., Weber, A., Keyser, J.: Dual scattering approximation for fast multiple scattering in hair. In: ACM SIGGRAPH 2008 papers, pp. 1–10 (2008)