Tex4D: Zero-shot 4D Scene Texturing with Video Diffusion Models

Jingzhi Bao [1] **Xueting Li** [2] **Ming-Hsuan Yang** [3]

Abstract

3D meshes are extensively employed in movies, games, AR, and VR for their efficiency in animation and minimal memory footprint, leading to the creation of a vast number of mesh sequences. However, creating dynamic textures for these mesh sequences to model the appearance transformations remains labor-intensive for professional artists. In this work, we present **Tex4D**, a zero-shot approach that creates multi-view and temporally consistent dynamic mesh textures by integrating the inherent 3D geometry knowledge with the expressiveness of video diffusion models. Given an untextured mesh sequence and a text prompt as inputs, our method enhances multi-view consistency by synchronizing the diffusion process across different views through latent aggregation in the UV space. To ensure temporal consistency, such as lighting changes, wrinkles, and appearance transformations, we leverage prior knowledge from a conditional video generation model for texture synthesis. Straightforwardly combining the video diffusion model and the UV texture aggregation leads to blurry results. We analyze the underlying causes and propose a simple yet effective modification to the DDIM sampling process to address this issue. Additionally, we introduce a reference latent texture to strengthen the correlation between frames during the denoising process. To the best of our knowledge, Tex4D is the first method specifically designed for 4D scene texturing. Extensive experiments demonstrate its superiority in producing multi-view and multi-frame consistent dynamic textures for mesh sequences.

1. Introduction

3D meshes are widely used in modeling, computer-aided design (CAD), animation, and computer graphics due to their low memory footprint and efficiency in animation. Visual artists, game designers, and movie creators build numerous animated mesh sequences for visual applications. However, creating vivid videos involves complex post-processing steps, such as creating dynamic textures for appearance transformations, as shown in Fig. 1. These steps are labor-intensive and require specialized expertise by artists.

Figure 1. **Tex4D application.** Our synthesized dynamic textures can be easily integrated into graphics pipelines.

On the other hand, recent advancements in generative models have democratized content creation and demonstrated impressive performance in image and video synthesis. For instance, video generation models (Ho et al., 2022; Esser et al., 2023; Li et al., 2023; He et al., 2022; Yu et al., 2023a; Zhou et al., 2022; Hong et al., 2022; Yang et al., 2024; Zhang et al., 2023b; Xing et al., 2023; Chen et al., 2023c; 2024) trained on large-scale video datasets (Bain et al., 2021; Schuhmann et al., 2021) allow users to create realistic video clips from various inputs such as text prompts, images, or geometric conditions. However, these text-to-video generation models, which are trained solely on 2D data, often struggle with spatial consistency when applied to multi-view image generation (Tang et al., 2023; Shi et al., 2023b; Liu et al., 2023a; Weng et al., 2023; Long et al., 2023; Shi et al., 2023a; Kwak et al., 2023; Tang et al., 2024; Voleti et al., 2024) or 3D object texturing (Cao et al., 2023; Liu et al., 2023b; Richardson et al., 2023; Huo et al., 2024).

To address these limitations, two main approaches have been developed. One approach (Richardson et al., 2023; Chen et al., 2023b; Cao et al., 2023) focuses on resolving multi-view inconsistency in static 3D object texturing by synchronizing multi-view image diffusion processes. While these methods produce multi-view consistent textures for static 3D objects, they do not address the challenge of generating dynamically changing textures for mesh sequences. Another approach (Guo et al., 2023a; Lin et al., 2024; Peng et al., 2024) aims to generate video clips based on the rendering (e.g., depth, normal or UV maps) of an untextured mesh sequence. To encourage temporal consistency, these

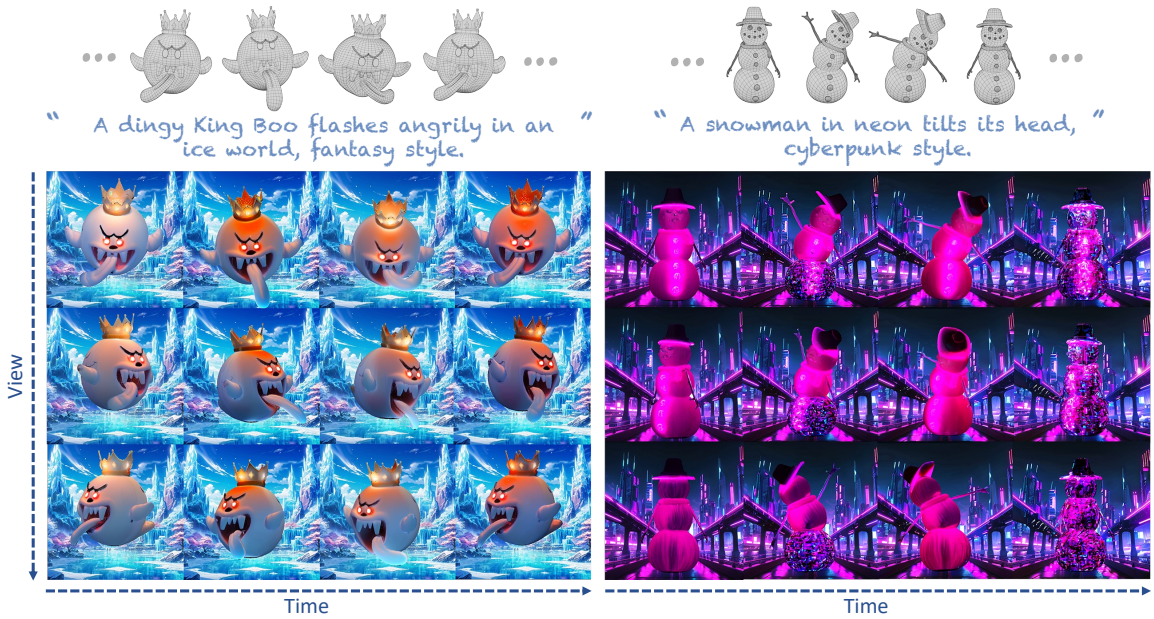


Figure 2. Given an untextured mesh sequence and a text prompt as inputs (Left), **Tex4D** generates multi-view, dynamic textures. On the right, we show renderings of the textured meshes from three views and four timestamps. Zoom in to view the texture details.

methods modify the attention mechanism in 2D diffusion models and utilize inherent correspondences in a mesh sequence to facilitate feature synchronization between frames. Although these techniques can be adapted for multi-view image generation by treating camera pose movement as temporal motion, they usually produce inconsistent 3D texturing due to insufficient exploitation of 3D geometry priors.

In this paper, we introduce a novel task: 4D scene texturing. Given an animated untextured 3D mesh sequence and a text prompt, our goal is to generate dynamic textures that are both temporally and multi-view consistent. We aim to texture 4D scenes while capturing temporal variations, such as lighting and appearance changes, to produce vivid visual results—a key requirement in downstream tasks like character generation. Unlike existing works, we fully leverage 3D geometry knowledge from the mesh sequence to enforce multi-view consistency. Specifically, we develop a method that synchronizes the diffusion process from different views through latent aggregation in the UV space. To ensure temporal consistency, we employ prior knowledge from a conditional video generation model for texture sequence synthesis and introduce a reference latent texture to enhance frame-to-frame correlations during the denoising process. However, naively integrating the UV texture aggregation into the video diffusion process causes the variance shift problem, leading to blurry results. To resolve this issue, we propose an effective modification to the DDIM (Song et al., 2020) sampling process by rewriting the equation. Our method is computationally efficient thanks to its zero-shot nature. The textured mesh sequence can be rendered from any camera view, thus supporting various applications in content creation. Our key contributions are:

- We present **Tex4D**, a zero-shot pipeline for generating high-fidelity dynamic textures that are temporally and multi-view consistent, utilizing text-to-video diffusion models and mesh sequence controls.
- To leverage priors from existing video diffusion models, we develop an effective modification to the DDIM sampling process to address the variance shift issue caused by multi-view texture aggregation and design a background learning module.
- We introduce a reference UV blending mechanism to establish correlations during the denoising steps, addressing self-occlusions, and synchronizing the diffusion process in invisible regions.
- Our method is not only computationally efficient, but also demonstrates comparable if not superior performance to various state-of-the-art baselines.

2. Related Work

Video Stylization and Editing. Video diffusion models have shown remarkable performance in the field of video generation. These models learn motions and dynamics from large-scale video datasets using 3D-UNet to create high-quality, realistic, and temporally coherent videos. Although these approaches show compelling results, the generated videos lack fine-grained control, inhibiting their application in stylization and editing. To solve this issue, inspired by ControlNet (Zhang et al., 2023a), SparseCtrl (Guo et al., 2023a) trains a sparse encoder from scratch using frame masks and sparse conditioning images as input to guide the video diffusion model. CTRL-Adapter (Lin et al., 2024) proposes a trainable intermediate adapter to connect the

features between ControlNet and video diffusion models.

Meanwhile, (Tumanyan et al., 2023) observed that the spatial features of T2I models play an influential role in determining the structure and appearance, Text2Video-Zero (Khachatryan et al., 2023) uses a frame-warping method to animate the foreground object by T2I models and (Wu et al., 2023; Ceylan et al., 2023; Qi et al., 2023) propose utilizing self-attention injection and cross-frame attention to generate stylized and temporally consistent video using DDIM inversion (Song et al., 2020). Subsequently, numerous works (Zhang et al., 2023c; Cai et al., 2024; Yang et al., 2023; Geyer et al., 2023; Eldesokey & Wonka, 2024) generate temporally consistent videos utilizing T2I diffusion models by spatial latent alignment without training. However, the synthesized videos usually show flickerings due to the empirical correspondences, such as feature embedding distances and UV maps, which are insufficient to express the continuous content in the latent space. Another line of work (Singer et al., 2022; Bar-Tal et al., 2022; Blattmann et al., 2023; Xu et al., 2024; Guo et al., 2023b) is to train additional modules on large-scale video datasets to construct feature mappings, for example, Text2LIVE (Bar-Tal et al., 2022) applies test-time training with the CLIP loss, and MagicAnimate (Xu et al., 2024) introduced an appearance encoder to retain intricate clothes details.

Texture Synthesis. With the rapid development of foundation models, researchers have focused on applying their generation capability and adaptability to simplify the process of designing textures and reduce the expertise required. To incorporate the result 3D content with prior knowledge, earlier works (Khalid et al., 2022; Michel et al., 2021; Chen et al., 2022) jointly optimize the meshes and textures from scratch with the simple semantic loss from the pre-trained CLIP (Radford et al., 2021) to encourage the 3D alignment between the generated results and the semantic priors. However, the results show apparent artifacts and distortion because the semantic feature cannot provide fine-grained supervision during the generation of 3D content.

DreamFusion (Poole et al., 2022) and similar models (Lin et al., 2023; Wang et al., 2023; Po & Wetzstein, 2024; Metzer et al., 2022; Chen et al., 2023a) distill the learned 2D diffusion priors from the pre-trained diffusion models (Rombach et al., 2021) to synthesize the 3D content by Score Distillation Sampling (SDS). These methods render 2D projections of the 3D asset parameters and compare them against reference images, iteratively refining the 3D asset parameters to minimize the discrepancy of the target distribution of 3D shapes learned by the diffusion model. Although these approaches enable people without expertise to generate detailed 3D content by textual prompt, their results are typically over-saturated and over-smoothed, hindering their application in actual cases. Another line of optimization-based methods (Yu et al., 2023b; Zeng et al., 2024; Bensadoun et al., 2024) turned to fuse 3D shape information, such as vertex positions, depth maps, and normal maps, with the pre-trained diffusion model by training separate modules on 3D datasets. Still, they require a specific UV layout process to achieve plausible results.

Recently, TexFusion (Cao et al., 2023) and numerous zero-shot methods (Liu et al., 2023b; Richardson et al., 2023; Huo et al., 2024) have shown significant success in generating globally consistent textures without additional 3D datasets. Based on depth-aware diffusion models, they sequentially inpaint the latents in the UV domain to ensure the spatial consistency of latents observed across different views. Then, they decode the latents from multiple views and finally synthesize the RGB texture through back projection.

However, these methods generate static 3D assets and overlook temporal changes in visual presentations, such as videos. To our knowledge, this is the first approach to synthesize multi-view dynamic textures for mesh sequences, enabling appearance transformations.

3. Preliminaries

Video Diffusion Prior. In this paper, we adopt CTRL-Adapter (Lin et al., 2024) as our prior model to provide dynamic information. CTRL-Adapter aims to adapt a pre-trained text-to-video diffusion model to conditions for various types of images, such as depth or normal map sequences. The key idea behind CTRL-Adapter is to leverage a pre-trained ControlNet (Zhang et al., 2023a) and to align its latents with those of the video diffusion model through a learnable mapping module. Intuitively, the video diffusion model generates temporally consistent video frames that capture dynamic elements like character motions and lighting, while the ControlNet further enhances this capability by allowing the model to condition on geometric information, such as depth and normal map sequences. This makes CTRL-Adapter particularly effective in providing a temporally consistent texture prior to our 4D scene texturing task. Specifically, we leverage the depth-conditioned CTRL-Adapter model. Given a sequence of depth images denoted as $\{D_1, ..., D_K\}$ and a text prompt $\boldsymbol{\mathcal{P}}$, CTRL-Adapter (denoted as $\boldsymbol{\mathcal{C}}$) synthesizes a frame sequence F by $F = \boldsymbol{\mathcal{C}}(\{D_1, ..., D_K\}, \boldsymbol{\mathcal{P}})$.

DDIM Sampling. DDIM (Song et al., 2020) is a widely used sampling method in diffusion models due to its superior efficiency and deterministic nature compared to DDPM (Ho et al., 2020). To enhance numerical stability and prevent temporal color shifts in video diffusion, numerous models (Zhang et al., 2023b; Ho et al., 2022) employ a learning-based sampling technique known as v-prediction (Salimans & Ho, 2022). At each denoising step, the sampling process

for the latents (denoted as $\boldsymbol{z}_t$) can be described as follows:

$$\begin{aligned} \boldsymbol{z}_{t-1} &= \sqrt{\alpha_{t-1}} \cdot \hat{\boldsymbol{z}}_0(\boldsymbol{z}_t) + \sqrt{1-\alpha_{t-1}} \cdot \boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t), \\ \hat{\boldsymbol{z}}_0(\boldsymbol{z}_t) &= \frac{\boldsymbol{z}_t - \sqrt{1-\alpha_t} \cdot \boldsymbol{\epsilon}_\theta}{\sqrt{\alpha_t}}, \quad \boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t) = \boldsymbol{\epsilon}_\theta, \end{aligned} \tag{1}$$

where α_t is the noise variance at time step t, $\boldsymbol{\epsilon}_\theta$ is the estimated noise from the U-Net denoising module, which is expected to follow $\mathcal{N}(0, \mathcal{I})$, and $\hat{\boldsymbol{z}}_0(\boldsymbol{z}_t)$ denotes the predicted original sample (i.e., the latents at timestep 0). After the v-parameterization, the predicted original sample $\hat{\boldsymbol{z}}_0(\boldsymbol{z}_t)$ and the predicted epsilon $\boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t)$ are computed as follows:

$$\begin{aligned} \hat{\boldsymbol{z}}_0(\boldsymbol{z}_t) &= \sqrt{\alpha_t} \cdot \boldsymbol{z}_t - \sqrt{1-\alpha_t} \cdot \boldsymbol{\epsilon}_\theta, \\ \boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t) &= \sqrt{\alpha_t} \cdot \boldsymbol{\epsilon}_\theta + \sqrt{1-\alpha_t} \cdot \boldsymbol{z}_t. \end{aligned} \tag{2}$$

We leverage an enhanced DDIM sampling process in video diffusion models, along with a multi-view consistent texture aggregation mechanism to synthesize 4D textures.

4. Method

Given an untextured mesh animation and a text prompt, our goal is to generate a multi-view and multi-frame consistent texture sequence for each mesh that aligns with both the text description and motion cues while capturing the dynamics from video diffusion models.

To optimize computational efficiency, we uniformly sample K key frames from the video and synthesize textures for these keyframes. Textures for the remaining frames are then generated by interpolating the key frame textures. Formally, given K animated meshes at the keyframes ($\{M_1, ..., M_K\}$), along with a text description $\boldsymbol{\mathcal{P}}$, our method produces temporally and spatially consistent UV maps denoted as $\{UV_1, ..., UV_K\}$, in a zero-shot manner.

Previous texture generation methods (Richardson et al., 2023; Chen et al., 2023b; Cao et al., 2023) typically inpaint and update textures sequentially using pre-defined camera views in an incremental manner. However, these approaches rely on view-dependent depth conditions and lack global spatial consistency, often resulting in visible discontinuities in the assembled texture map. This issue arises from error accumulation during the autoregressive view update process, as noted by (Bensadoun et al., 2024). To resolve these issues, rather than processing each view independently, recent methods (Liu et al., 2023b) propose to generate multi-view textures simultaneously through diffusion. In this work, we similarly leverage the UV space as an intermediate representation to ensure multi-view consistency.

4.1. Overview

As shown in Fig. 3, given a sequence of K meshes, we start by rendering the mesh at V predefined, uniformly sampled camera poses to obtain multi-view depth images (denoted as $\{D_{1,1}, ..., D_{1,K}, D_{2,1} ..., D_{V,K}\}$), which serve as the geometric conditions. To generate textures for each mesh, we initialize $V \times K$ noise images sampled from a Normal distribution (denoted as $\{\boldsymbol{z}^{1,1}, ..., \boldsymbol{z}^{1,K}, \boldsymbol{z}^{2,1}, ..., \boldsymbol{z}^{V,K}\}$). Additionally, we initialize an extra noise map sequence $\{\boldsymbol{z}_b^1, ..., \boldsymbol{z}_b^K\}$ for the backgrounds learning. This noise map corresponds to the texture of a plane mesh that is composited with the foreground object at each diffusion step (See Sec. 4.3). Next, for each view $v \in \{1, ..., V\}$, we apply the video diffusion model (Lin et al., 2024) discussed in Sec. 3 to simultaneously denoise all latents and obtain multi-frame consistent images as $\{I^{1,v}, ..., I^{K,v}\} = \boldsymbol{\mathcal{C}}(\{D_{1,v}, ..., D_{K,v}\}, \boldsymbol{\mathcal{P}})$, where $\boldsymbol{\mathcal{P}}$ is the provided text prompt. Finally, we un-project and aggregate all denoised multi-view images for each mesh to formulate temporally consistent UV textures.

Applying the video diffusion model independently to each camera view often results in multi-view inconsistencies. Inspired by (Liu et al., 2023b; Huo et al., 2024; Zhang et al., 2024), we aggregate the multi-view latents of each mesh in the UV space to merge observations across different views at each denoising step, and then render latent from the latent texture to ensure multi-view consistency. Furthermore, we composite the rendered foreground latents with the background latents at each diffusion step (discussed in Sec. 4.2), which is essential to exploit prior in the video diffusion model (see Fig. 11). Nonetheless, such a simple aggregation method introduces blurriness in the final results. In Sec. 4.3, we analyze the underlying causes and propose a simple yet effective method to enhance the denoising process. Additionally, we create a reference UV to handle self-occlusions and further improve temporal consistency in Sec. 4.4.

4.2. Multi-view Latents Aggregation in the UV Space

We describe the aggregation of multi-view latents in the UV space. For frame $k \in \{1, ..., K\}$, we aggregate the multi-view latents $\{\boldsymbol{z}^{1,k}, \ldots, \boldsymbol{z}^{V,k}\}$ in the UV space by:

$$\mathcal{T}^k\left(\boldsymbol{z}^k\right) = \frac{\sum_{v=1}^{V} \mathcal{R}^{-1}(\boldsymbol{z}^{v,k}, c_v) \odot \cos\left(\theta^v\right)^\alpha}{\sum_{v=1}^{V} \cos\left(\theta^v\right)^\alpha}, \tag{3}$$

where $\mathcal{R}^{-1}$ represents the inverse rendering operator that un-projects the latents to the UV space, thus $\mathcal{R}^{-1}(\boldsymbol{z}^{v,k}, \boldsymbol{c_v})$ produces a partial latent UV texture from view v, $\cos(\theta^v)$ is the cosine map buffered by the geometry shader, recording the cosine value between the view direction and the surface normal for each pixel, α is a scaling factor, and $\boldsymbol{c_v}$ denotes one of the predefined cameras. After multi-view latents aggregation, we obtain multi-view consistent latents by rendering the aggregated UV latent map using $\tilde{\boldsymbol{z}}^{v,k} = \mathcal{R}\left(\mathcal{T}^k; \boldsymbol{c}_v\right)$, where $\mathcal{R}$ is the rendering operation.

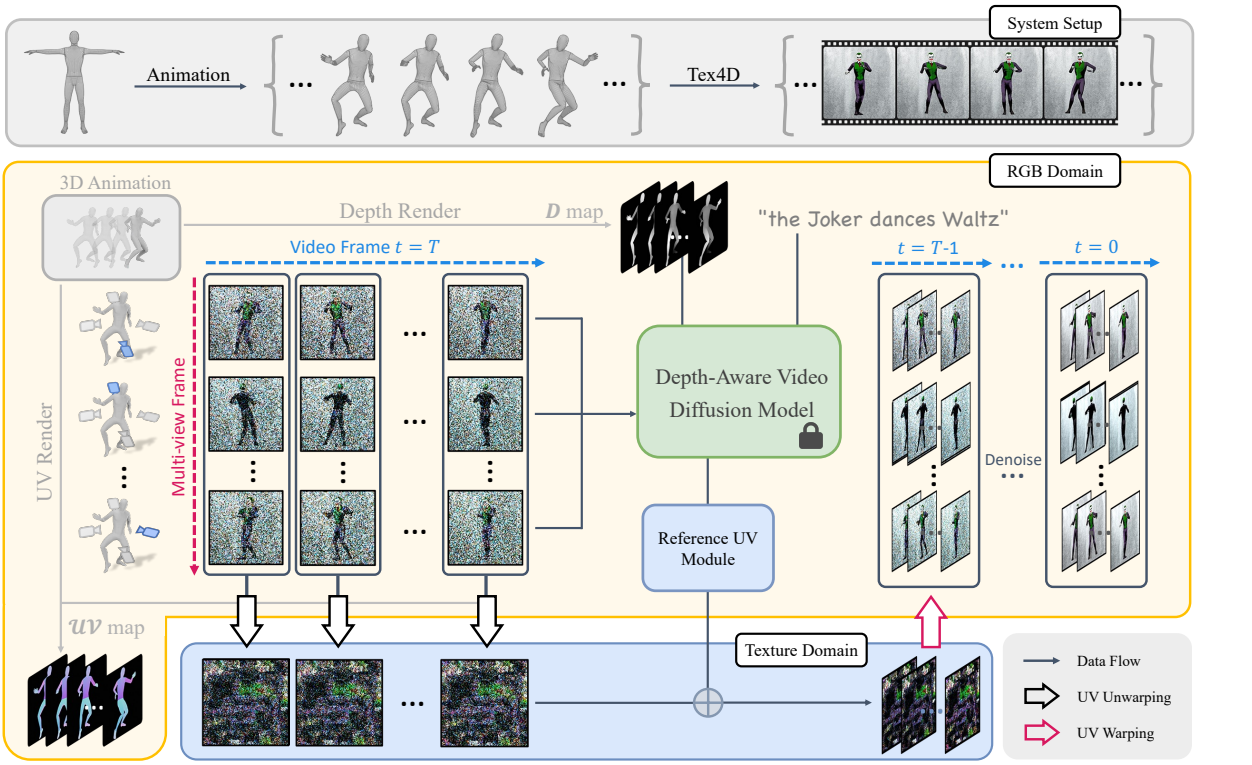


Figure 3. **Overview.** Given a mesh sequence and a text prompt as inputs, Tex4D generates a UV-parameterized texture sequence that is both globally and temporally consistent. At each diffusion step, latent views are aggregated into UV space, followed by multi-view latent texture diffusion to ensure global consistency. To maintain temporal coherence and address self-occlusions, a Reference UV Blending module is applied at each step. Finally, the latent textures are back-projected and decoded to produce RGB textures for each frame.

4.3. Multi-frame Consistent Texture Generation

The aggregation process discussed above yields multi-view consistent latents $\{\tilde{\boldsymbol{z}}^{v,k}\}$ for the denoising steps. However, this simple aggregation and projection strategy leads to a blurry appearance, as shown in Fig. 6(b). This issue arises primarily because the aggregation process depicted in Eq. 3 derails the DDIM denoise process. Specifically, the estimated noise $\boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t)$ for each step in Eq. 1 is expected to follow $\mathcal{N}(0,\mathcal{I})$, but Eq. 3 indicates that after aggregating multi-view latents, the expected norm of variance of the noise distribution would be less than $\mathcal{I}$. We denote this as the "variance shift" issue caused by the texture aggregation.

To resolve this issue, we rewrite the estimated noise ϵ_θ as the combination of the t-step latent $\boldsymbol{z}_t$ and the estimated latent $\hat{\boldsymbol{z}}_0(\boldsymbol{z}_t)$ at step 0. The v-paramaterized predicted epsilon $\boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t)$ in Eq. 2 is can be equivalently expressed as:

$$\begin{aligned}\boldsymbol{\epsilon}_{\boldsymbol{\theta}} &= \left(\sqrt{\alpha_t}\cdot\boldsymbol{z}_t - \hat{\boldsymbol{z}}_0(\boldsymbol{z}_t)\right)/\sqrt{1-\alpha_t}\\ \boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t) &= \sqrt{\alpha_t}\cdot\boldsymbol{\epsilon}_{\boldsymbol{\theta}} + \sqrt{1-\alpha_t}\cdot\boldsymbol{z}_t\\ &= \sqrt{\frac{\alpha_t}{1-\alpha_t}}\cdot\left(\sqrt{\alpha_t}\boldsymbol{z}_t - \hat{\boldsymbol{z}}_0(\boldsymbol{z}_t)\right) + \sqrt{1-\alpha_t}\cdot\boldsymbol{z}_t.\end{aligned} \tag{4}$$

In practice, we carry out this denoising technique in the UV space. Specifically, we first compute the original texture map (i.e., texture map at step 0, denoted as $\hat{\mathcal{T}}_0$) by aggregating the predicted original multi-view image latents through Eq. 3. The noisy latent texture map at time step t (denoted as $\mathcal{T}_t$) can be similarly computed. We denoise in step t by:

$$\begin{aligned}\mathcal{T}_{t-1} = \sqrt{\alpha_{t-1}}\cdot\hat{\mathcal{T}}_0 + \sqrt{1-\alpha_{t-1}}\cdot\\ \left(\sqrt{\frac{\alpha_t}{1-\alpha_t}}\cdot\left(\sqrt{\alpha_t}\mathcal{T}_t - \hat{\mathcal{T}}_0\right) + \sqrt{1-\alpha_t}\cdot\mathcal{T}_t\right).\end{aligned} \tag{5}$$

Through experimentation, we observe that background optimization plays a crucial role in fully exploiting the prior within the video diffusion model. As shown in Fig. 6(c), using a simple white background leads to blurry results. This may be attributed to a mismatch between the white-background images and the training dataset, which likely contains fewer such examples, affecting the denoising process. To resolve this issue, we compute the final latents as the combination of the foreground latent $\tilde{\boldsymbol{z}}_{t-1}$ projected from the aggregated UV latents and the residual background latent $\boldsymbol{z}_{b,t-1}$ denoised by diffusion models. Specifically, we composite the estimated latents in the $t-1$ step as follows:

$$\boldsymbol{z}_{t-1} = \tilde{\boldsymbol{z}}_{t-1}\odot\boldsymbol{\mathcal{M}}_{\text{fg}} + \boldsymbol{z}_{b,t-1}\odot\left(1-\boldsymbol{\mathcal{M}}_{\text{fg}}\right), \tag{6}$$

$$\tilde{\boldsymbol{z}}_{t-1}, \boldsymbol{\mathcal{M}}_{\text{fg}} = \mathcal{R}\left(\mathcal{T}_{t-1}; \boldsymbol{c}_v\right), \tag{7}$$

where $\boldsymbol{\mathcal{M}}_{\text{fg}}$ represents the foreground mask of the mesh.

To summarize, our diffusion process starts with $K\times(V+1)$ randomly initialized noise maps sampled (i.e.,

$\{\boldsymbol{z}_T^{1,k}, \ldots, \boldsymbol{z}_T^{V,k}\}$, for foreground, $\{\boldsymbol{z}_b^1, ..., \boldsymbol{z}_b^K\}$ for background) and denoise them into images simultaneously. At each denoising step t with the key frame k, we derive the estimated noises $\{\boldsymbol{\epsilon}_{t-1}^{1,k}, \ldots, \boldsymbol{\epsilon}_{t-1}^{V,k}\}$ using the video diffusion model and calculate the estimated original latent $\{\hat{\boldsymbol{z}}_0^{1,k}, \ldots, \hat{\boldsymbol{z}}_0^{V,k}\}$ by Eq. 1. Then, we use Eq. 3 to aggregate the latents onto UV space. Next, we utilize Eq. 5 to take the diffusion step in the UV space, and render the synchronized latents $\{\tilde{\boldsymbol{z}}_{t-1}^{1,k}, \ldots, \tilde{\boldsymbol{z}}_{t-1}^{V,k}\}$ from latent UVs $\{\mathcal{T}_{t-1}^1, \ldots, \mathcal{T}_{t-1}^K\}$ to ensure multi-view consistency. Finally, we composite the denoised latent with the latents at step $t-1$ according to foreground masks by Eq. 6 and Eq. 7.

4.4. Reference UV Blending

While the video diffusion model ensures temporal consistency for latents from each view, consistency can sometimes diminish after aggregation in the texture domain. This issue primarily stems from the view-dependent nature of the depth conditions and the limited resolution of latents, which can lead to distortions when features from different camera angles are combined onto the UV texture. Additionally, self-occlusion during mesh animation often results in a loss of information in invisible regions.

To address these challenges, we propose a reference UV map to enhance correlations between latent textures across frames. Specifically, the reference UV map is constructed by sequentially combining latent textures over time, with each new texture filling only the empty texels of the reference UV map. Then, each texture is blended using the reference UV $\mathcal{T}_{\mathcal{UV}}$ with a mask $\boldsymbol{\mathcal{M}}_{\mathcal{UV}}$ that labels the visible region:

$$\mathcal{T}_t^k = \big((1-\lambda)\cdot\mathcal{T}_t^k + \lambda\cdot\mathcal{T}_{\mathcal{UV}}\big)\odot\boldsymbol{\mathcal{M}}_{\mathcal{UV}}^k + \mathcal{T}_{\mathcal{UV}}\odot\Big(1-\boldsymbol{\mathcal{M}}_{\mathcal{UV}}^k\Big) \tag{8}$$

where λ is the blending weight for the reference UV in the visible region, while the invisible region is simply replaced with the reference texture. We empirically set the blending weight to 0.2 during our experiments.

5. Experiments

Datasets. We source our datasets from two primary repositories: human motion diffusion outputs and the Mixamo and Sketchfab websites. We employ the text-to-motion diffusion model (Tevet et al., 2023) to compare our approach with LatentMan (Eldesokey & Wonka, 2024). For comparison with Generative Rendering (Cai et al., 2024), we obtain animated characters from the Mixamo platform and render them with different motions. Specifically, we first use Blender (Community, 2024) to extract meshes, joints, skinning weights, and animation data from the FBX files. Then, we apply linear blend skinning to animate the meshes. For meshes without UV maps, we utilize XATLAS to parameterize the mesh and unwrap the UVs.

Baselines. To our knowledge, no existing studies tackle multi-view consistent video generation guided by untextured mesh sequences as our method does. We adopt six recent methods, rendering the input (untextured mesh renders and depth maps) based on their configurations to establish baselines, including video stylization methods and video generation methods with various control mechanisms. PnP-Diffusion (Tumanyan et al., 2023) is an image style transfer method conditioned on the DDIM inversion attention feature of the input image. We extend the method to stylize videos on a frame-by-frame basis for comparison, aligning with previous work (Geyer et al., 2023). Built upon cross-frame attention, Text2Video-Zero (Khachatryan et al., 2023) guides the video by warping latents to enhance video dynamics implicitly. We leverage its official extension, which includes support for depth control. TokenFlow (Geyer et al., 2023), Generative Rendering (Cai et al., 2024), and LatentMan (Eldesokey & Wonka, 2024) study frame relations in latent space and establish feature correspondences through nearest neighbor matching and DensePose features. Gen-1 (Esser et al., 2023) is a video-to-video model that learns the structure of input videos and transforms the untextured mesh renders into stylized outputs. Given the lack of the source code for Generative Rendering, we utilize the experimental results presented in their video demos for qualitative comparison. Additionally, we compare our method with the texture generation method Text2Tex (Chen et al., 2023b).

Evaluation Metrics. Quantitatively evaluating multi-view consistency and temporal coherence remains challenging. We conduct a user study to assess overall performance, including the appearance, temporal coherence and spatial consistency, and the fidelity to prompt based on human preference. Additionally, we measure multi-view coherence using Fréchet Video Distance (FVD) (Unterthiner et al., 2018), a video-level metric for temporal coherence utilized in prior works (Li et al., 2024; Xie et al., 2024).

5.1. Qualitative Results

We present qualitative evaluation in Fig. 7. Generative Rendering, TokenFlow, and Text2Video-Zero, which rely on T2I diffusion models with cross-frame attention mechanisms, exhibit noticeable flickering compared to other methods. This issue stems partly from the empirical and implicit correspondence mapping used to enforce interframe latent consistency, as the correspondences in the latent space may not precisely align with those in the RGB space. In contrast, our approach interpolates the frames between key frame textures in RGB space, eliminating artifacts caused by latent manipulation. PnP-Diffusion edits frames independently and generates detailed and sophisticated appearances but suffers from poor spatio-temporal consistency due to the loss of temporal correlations in the latent space. While Gen-1 produces high-quality videos, it fails to maintain

Figure 4. **Qualitative Results.** Our method generates multi-view consistent dynamic textures with a diverse set of styles and prompts.

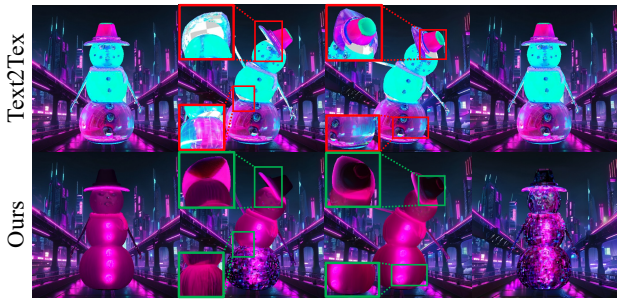


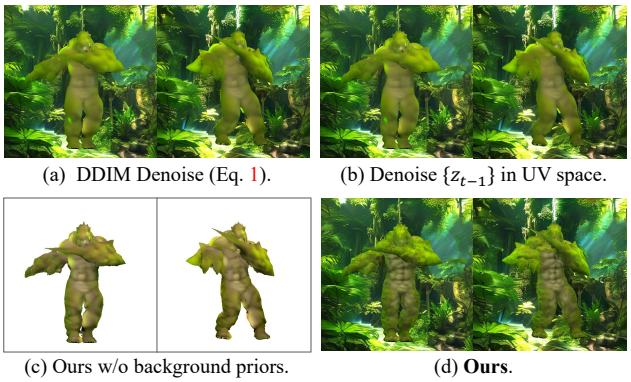
Figure 5. **Comparison with texture generation method.** We compare our method against texture generation method Text2Tex (Chen et al., 2023b), which shows empty texels in invisible regions.

multi-view consistency.

Furthermore, we present multi-view results showcasing a variety of styles and prompts in Fig. 4. Our method, driven by video diffusion models, effectively accounts for the styles and captures temporal variations over time. As shown in Fig. 5, Tex4D effectively handles the invisible regions compared with the traditional texture generation method Text2Tex (Chen et al., 2023b).

5.2. Quantitative Evaluation

To quantitatively assess the effectiveness of our proposed method, we follow prior research (Eldesokey & Wonka, 2024; Geyer et al., 2023; Esser et al., 2023) and conduct a comprehensive A/B user study. Our study involved 67 participants who provided a total of 1104 valid responses based on six different scenes drawn from six previous works, with each scene producing videos from two different views. During each evaluation, participants were presented with rendered meshes and depth conditions viewed from two angles, serving as motion references. They were shown a pair of videos: one generated by our approach and the other from a baseline method. Participants were asked to select the method that exhibited superior performance in three criteria: 1) appearance quality, 2) spatial and temporal consistency, and 3) fidelity to the prompts. Table 1 summarizes the preference percentage of our method over the baseline methods. Our method significantly surpasses state-of-the-art methods by a large margin. In addition, our method achieves lower FVD that demonstrates better multi-view consistency in generated video clips.



(a) DDIM Denoise (Eq. 1). (b) Denoise $\{z_{t-1}\}$ in UV space.

(c) Ours w/o background priors. (d) **Ours**.

Figure 6. **Ablation studies on the multi-view denoise algorithm and backgrounds.** (a) Denoise by DDIM mechanism (Eq. 1). (b) Denoise views $\{z_{t-1}\}$ and project to UV space. (c) Denoise with a white background. (d) Our full algorithm.

5.3. Ablation Study

Ablation for texture aggregation. In Fig. 6 (a) and (b), we present two alternative texture aggregation methods. In Fig. 6 (a), we un-project $\hat{z}_0(z_t)$ and $\epsilon_\theta(z_t)$ into UV space for aggregation. In Fig. 6 (b), we map z_{t-1} to the UV space. Both these two approaches show inferior results compared to our method, which verifies the effectiveness of the proposed texture aggregation algorithm.

Ablation for UV blending module. In Sec. 4.4, we propose a reference UV blending schema to resolve the temporal inconsistency caused by latent aggregation. To validate the effectiveness of this mechanism (See Sec. 4.4), we conduct an ablation study by disabling the reference UV blending module (setting λ to 0). As shown in Fig. 8, without the UV blending module, our method generates textures with noticeable distortions on the Joker's face over time.

Ablation for background priors. Sec. 4.3 discusses the importance of including a plausible background prior. To verify the effectiveness of this design, we replace the learn-

Table 1. **Quantitative evaluation**. We present FVD values and a comparison highlighting the percentage of user preference for our approach against other methods. Our method shows the best spatio-temporal consistency as measured by the FVD metric (Unterthiner et al., 2018). Users consistently favored Tex4D over all baselines.

Method	FVD (↓)	Appearance Quality	Spatio-temporal Consistency	Consistency with Prompt
Text2Video-Zero	3078.94	89.33%	91.78%	91.55%
PnP-Diffusion	1390.04	86.42%	87.18%	89.74%
TokenFlow	1330.43	92.31%	86.84%	93.42%
Gen-1	3114.26	70.27%	75.00%	77.78%
LatentMan	2811.23	86.57%	86.57%	81.82%
Ours	**1303.14**	-	-	-

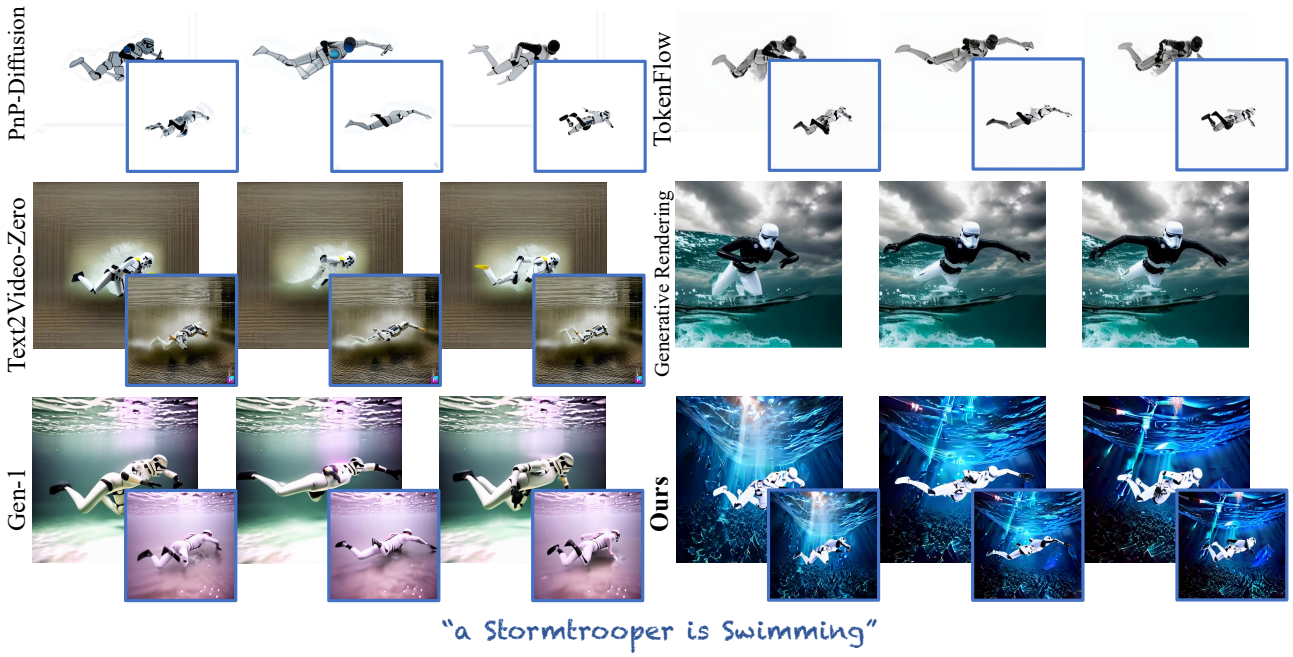


Figure 7. **Qualitative comparisons of multi-view video generation.** We compare our method against PnP-diffusion (Tumanyan et al., 2023), TokenFlow (Geyer et al., 2023), Text2Video-Zero (Khachatryan et al., 2023), Generative Rendering (Cai et al., 2024) (from their video demo), and Gen-1 (Esser et al., 2023). We generate videos in the front view and the side view (blue box) on Mixamo dataset. Our method generates vivid videos that align with the textual prompts while preserving spatial consistency.

Figure 8. **Ablation study on the reference UV blending module.** Without this module, the generated textures may lose consistency over time, as highlighted in the red boxes.

able background latents with an all-white background while keeping all other parts unchanged. Fig. 6 (c) illustrates that this ablation experiment produces significantly blurrier textures compared to our full method, highlighting the importance of background learning.

6. Conclusions

In this paper, we present Tex4D, a zero-shot approach that generates multi-view, multi-frame consistent dynamic textures for untextured, animated mesh sequences based on a text prompt. By incorporating texture aggregation in the UV space within the diffusion process of a conditional video diffusion model, we ensure both temporal and spatial coherence in the generated textures. To leverage priors from existing video diffusion models, we develop an effective modification to the DDIM sampling process to address the variance shift issue caused by multi-view texture aggregation and design a background learning module. Additionally, we enhance temporal consistency by introducing a reference UV map and developing a dynamic background learning framework to produce fully textured 4D scenes. Extensive experiments show that our method can synthesize realistic and consistent 4D textures, demonstrating its superiority against state-of-the-art baselines.

References

Bain, M., Nagrani, A., Varol, G., and Zisserman, A. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021. 1

Bar-Tal, O., Ofri-Amar, D., Fridman, R., Kasten, Y., and Dekel, T. Text2live: Text-driven layered image and video editing. In *ECCV*, pp. 707–723, 2022. 3

Bensadoun, R., Kleiman, Y., Azuri, I., Harosh, O., Vedaldi, A., Neverova, N., and Gafni, O. Meta 3d texturegen: Fast and consistent texture generation for 3d objects. *arXiv preprint arXiv:2407.02430*, 2024. 3, 4

Blattmann, A., Rombach, R., Ling, H., Dockhorn, T., Kim, S. W., Fidler, S., and Kreis, K. Align your latents: High-resolution video synthesis with latent diffusion models. In *CVPR*, 2023. 3

Cai, S., Ceylan, D., Gadelha, M., Huang, C.-H., Wang, T., and Wetzstein, G. Generative rendering: Controllable 4d-guided video generation with 2d diffusion models. In *CVPR*, 2024. 3, 6, 8

Cao, T., Kreis, K., Fidler, S., Sharp, N., and Yin, K. Texfusion: Synthesizing 3d textures with text-guided image diffusion models. In *ICCV*, 2023. 1, 3, 4, 12

Ceylan, D., Huang, C.-H., and Mitra, N. J. Pix2video: Video editing using image diffusion. In *ICCV*, 2023. 3

Chen, D. Z., Li, H., Lee, H.-Y., Tulyakov, S., and Nießner, M. Scenetex: High-quality texture synthesis for indoor scenes via diffusion priors. *arXiv preprint arXiv:2311.17261*, 2023a. 3

Chen, D. Z., Siddiqui, Y., Lee, H.-Y., Tulyakov, S., and Nießner, M. Text2tex: Text-driven texture synthesis via diffusion models. *arXiv preprint arXiv:2303.11396*, 2023b. 1, 4, 6, 7, 15, 16

Chen, H., Xia, M., He, Y., Zhang, Y., Cun, X., Yang, S., Xing, J., Liu, Y., Chen, Q., Wang, X., Weng, C., and Shan, Y. Videocrafter1: Open diffusion models for high-quality video generation. *arXiv preprint arXiv:2310.19512*, 2023c. 1

Chen, H., Zhang, Y., Cun, X., Xia, M., Wang, X., Weng, C., and Shan, Y. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. *arXiv preprint arXiv:2401.09047*, 2024. 1

Chen, Y., Chen, R., Lei, J., Zhang, Y., and Jia, K. Tango: Text-driven photorealistic and robust 3d stylization via lighting decomposition. *NeurIPS*, 35:30923–30936, 2022. 3

Community, B. O. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2024. URL http://www.blender.org. 6

Eldesokey, A. and Wonka, P. Latentman: Generating consistent animated characters using image diffusion models. In *CVPR*, pp. 7510–7519, 2024. 3, 6, 7

Esser, P., Chiu, J., Atighehchian, P., Granskog, J., and Germanidis, A. Structure and content-guided video synthesis with diffusion models. In *ICCV*, pp. 7346–7356, 2023. 1, 6, 7, 8

Geyer, M., Bar-Tal, O., Bagon, S., and Dekel, T. Tokenflow: Consistent diffusion features for consistent video editing. *arXiv preprint arXiv:2307.10373*, 2023. 3, 6, 7, 8

Guo, Y., Yang, C., Rao, A., Agrawala, M., Lin, D., and Dai, B. Sparsectrl: Adding sparse controls to text-to-video diffusion models. *arXiv preprint arXiv:2311.16933*, 2023a. 1, 2

Guo, Y., Yang, C., Rao, A., Liang, Z., Wang, Y., Qiao, Y., Agrawala, M., Lin, D., and Dai, B. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arxiv:2307.04725*, 2023b. 3

He, Y., Yang, T., Zhang, Y., Shan, Y., and Chen, Q. Latent video diffusion models for high-fidelity long video generation. *arXiv preprint arXiv:2211.13221*, 2022. 1

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *NeurIPS*, 33:6840–6851, 2020. 3

Ho, J., Chan, W., Saharia, C., Whang, J., Gao, R., Gritsenko, A., Kingma, D. P., Poole, B., Norouzi, M., Fleet, D. J., et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 1, 3, 12

Hong, W., Ding, M., Zheng, W., Liu, X., and Tang, J. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022. 1, 12

Huo, D., Guo, Z., Zuo, X., Shi, Z., Lu, J., Dai, P., Xu, S., Cheng, L., and Yang, Y.-H. Texgen: Text-guided 3d texture generation with multi-view sampling and resampling. *ECCV*, 2024. 1, 3, 4, 12

Khachatryan, L., Movsisyan, A., Tadevosyan, V., Henschel, R., Wang, Z., Navasardyan, S., and Shi, H. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. *arXiv preprint arXiv:2303.13439*, 2023. 3, 6, 8

Khalid, N. M., Xie, T., Belilovsky, E., and Tiberiu, P. Clip-mesh: Generating textured meshes from text using pre-trained image-text models. *SIGGRAPH Aisa*, December 2022. 3

Kwak, J.-g., Dong, E., Jin, Y., Ko, H., Mahajan, S., and Yi, K. M. Vivid-1-to-3: Novel view synthesis with video diffusion models. *arXiv preprint arXiv:2312.01305*, 2023. 1

Li, B., Zheng, C., Zhu, W., Mai, J., Zhang, B., Wonka, P., and Ghanem, B. Vivid-zoo: Multi-view video generation with diffusion model. *arXiv preprint arXiv:2406.08659*, 2024. 6

Li, X., Chu, W., Wu, Y., Yuan, W., Liu, F., Zhang, Q., Li, F., Feng, H., Ding, E., and Wang, J. Videogen: A reference-guided latent diffusion approach for high definition text-to-video generation. *arXiv preprint arXiv:2309.00398*, 2023. 1

Lin, C.-H., Gao, J., Tang, L., Takikawa, T., Zeng, X., Huang, X., Kreis, K., Fidler, S., Liu, M.-Y., and Lin, T.-Y. Magic3d: High-resolution text-to-3d content creation. In *CVPR*, 2023. 3

Lin, H., Cho, J., Zala, A., and Bansal, M. Ctrl-adapter: An efficient and versatile framework for adapting diverse controls to any diffusion model. *arXiv preprint arXiv:2404.09967*, 2024. 1, 2, 3, 4, 12

Liu, Y., Lin, C., Zeng, Z., Long, X., Liu, L., Komura, T., and Wang, W. Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023a. 1, 13

Liu, Y., Xie, M., Liu, H., and Wong, T.-T. Text-guided texturing by synchronized multi-view diffusion. *arXiv preprint arXiv:2311.12891*, 2023b. 1, 3, 4, 12, 13

Long, X., Guo, Y.-C., Lin, C., Liu, Y., Dou, Z., Liu, L., Ma, Y., Zhang, S.-H., Habermann, M., Theobalt, C., et al. Wonder3d: Single image to 3d using cross-domain diffusion. *arXiv preprint arXiv:2310.15008*, 2023. 1

Metzer, G., Richardson, E., Patashnik, O., Giryes, R., and Cohen-Or, D. Latent-nerf for shape-guided generation of 3d shapes and textures. *arXiv preprint arXiv:2211.07600*, 2022. 3

Michel, O., Bar-On, R., Liu, R., Benaim, S., and Hanocka, R. Text2mesh: Text-driven neural stylization for meshes. *arXiv preprint arXiv:2112.03221*, 2021. 3

Peng, B., Wang, J., Zhang, Y., Li, W., Yang, M.-C., and Jia, J. Controlnext: Powerful and efficient control for image and video generation. *arXiv preprint arXiv:2408.06070*, 2024. 1

Po, R. and Wetzstein, G. Compositional 3d scene generation using locally conditioned diffusion. In *2024 International Conference on 3D Vision (3DV)*, pp. 651–663. IEEE, 2024. 3

Poole, B., Jain, A., Barron, J. T., and Mildenhall, B. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv*, 2022. 3

Qi, C., Cun, X., Zhang, Y., Lei, C., Wang, X., Shan, Y., and Chen, Q. Fatezero: Fusing attentions for zero-shot text-based video editing. *arXiv preprint arXiv:2303.09535*, 2023. 3

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *ICML*, pp. 8748–8763, 2021. 3

Richardson, E., Metzer, G., Alaluf, Y., Giryes, R., and Cohen-Or, D. Texture: Text-guided texturing of 3d shapes. In *SIGGRAPH*, pp. 1–11, 2023. 1, 3, 4

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. *arXiv preprint arXiv:2112.10752*, 2021. 3

Salimans, T. and Ho, J. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022. 3

Schuhmann, C., Vencu, R., Beaumont, R., Kaczmarczyk, R., Mullis, C., Katta, A., Coombes, T., Jitsev, J., and Komatsuzaki, A. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021. 1

Shi, R., Chen, H., Zhang, Z., Liu, M., Xu, C., Wei, X., Chen, L., Zeng, C., and Su, H. Zero123++: a single image to consistent multi-view diffusion base model. *arXiv preprint arXiv:2310.15110*, 2023a. 1

Shi, Y., Wang, P., Ye, J., Mai, L., Li, K., and Yang, X. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*, 2023b. 1

Singer, U., Polyak, A., Hayes, T., Yin, X., An, J., Zhang, S., Hu, Q., Yang, H., Ashual, O., Gafni, O., et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022. 3

Song, J., Meng, C., and Ermon, S. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2, 3

Tang, S., Zhang, F., Chen, J., Wang, P., and Furukawa, Y. Mvdiffusion: Enabling holistic multi-view image generation with correspondence-aware diffusion. *arXiv*, 2023. 1

Tang, S., Chen, J., Wang, D., Tang, C., Zhang, F., Fan, Y., Chandra, V., Furukawa, Y., and Ranjan, R. Mvdiffusion++: A dense high-resolution multi-view diffusion model for single or sparse-view 3d object reconstruction. *arXiv preprint arXiv:2402.12712*, 2024. 1

Tevet, G., Raab, S., Gordon, B., Shafir, Y., Cohen-or, D., and Bermano, A. H. Human motion diffusion model. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=SJ1kSyO2jwu. 6

Tumanyan, N., Geyer, M., Bagon, S., and Dekel, T. Plug-and-play diffusion features for text-driven image-to-image translation. In *CVPR*, pp. 1921–1930, June 2023. 3, 6, 8

Unterthiner, T., Van Steenkiste, S., Kurach, K., Marinier, R., Michalski, M., and Gelly, S. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018. 6, 8

Voleti, V., Yao, C.-H., Boss, M., Letts, A., Pankratz, D., Tochilkin, D., Laforte, C., Rombach, R., and Jampani, V. SV3D: Novel multi-view synthesis and 3D generation from a single image using latent video diffusion. In *ECCV*, 2024. 1

Wang, Z., Lu, C., Wang, Y., Bao, F., Li, C., Su, H., and Zhu, J. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. In *NeurIPS*, 2023. 3

Weng, H., Yang, T., Wang, J., Li, Y., Zhang, T., Chen, C., and Zhang, L. Consistent123: Improve consistency for one image to 3d object synthesis. *arXiv preprint arXiv:2310.08092*, 2023. 1

Wu, J. Z., Ge, Y., Wang, X., Lei, S. W., Gu, Y., Shi, Y., Hsu, W., Shan, Y., Qie, X., and Shou, M. Z. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *ICCV*, pp. 7623–7633, 2023. 3

Xie, Y., Yao, C.-H., Voleti, V., Jiang, H., and Jampani, V. SV4D: Dynamic 3d content generation with multi-frame and multi-view consistency. *arXiv preprint arXiv:2407.17470*, 2024. 6

Xing, J., Xia, M., Zhang, Y., Chen, H., Wang, X., Wong, T.-T., and Shan, Y. Dynamicrafter: Animating open-domain images with video diffusion priors. *arXiv preprint arXiv:2310.12190*, 2023. 1

Xu, Z., Zhang, J., Liew, J. H., Yan, H., Liu, J.-W., Zhang, C., Feng, J., and Shou, M. Z. Magicanimate: Temporally consistent human image animation using diffusion model. In *CVPR*, 2024. 3

Yang, S., Zhou, Y., Liu, Z., , and Loy, C. C. Rerender a video: Zero-shot text-guided video-to-video translation. In *SIGGRAPH Aisa*, 2023. 3

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. 1, 12

Yu, S., Sohn, K., Kim, S., and Shin, J. Video probabilistic diffusion models in projected latent space. In *CVPR*, pp. 18456–18466, 2023a. 1

Yu, X., Dai, P., Li, W., Ma, L., Liu, Z., and Qi, X. Texture generation on 3d meshes with point-uv diffusion. In *ICCV*, pp. 4206–4216, 2023b. 3

Zeng, X., Chen, X., Qi, Z., Liu, W., Zhao, Z., Wang, Z., Fu, B., Liu, Y., and Yu, G. Paint3d: Paint anything 3d with lighting-less texture diffusion models. In *CVPR*, pp. 4252–4262, 2024. 3

Zhang, L., Rao, A., and Agrawala, M. Adding conditional control to text-to-image diffusion models. In *CVPR*, pp. 3836–3847, 2023a. 2, 3

Zhang, L., Wang, Z., Zhang, Q., Qiu, Q., Pang, A., Jiang, H., Yang, W., Xu, L., and Yu, J. Clay: A controllable large-scale generative model for creating high-quality 3d assets. *ACM Transactions on Graphics (TOG)*, 2024. 4

Zhang, S., Wang, J., Zhang, Y., Zhao, K., Yuan, H., Qing, Z., Wang, X., Zhao, D., and Zhou, J. I2vgen-xl: High-quality image-to-video synthesis via cascaded diffusion models. *arXiv preprint arXiv:2311.04145*, 2023b. 1, 3, 12

Zhang, Y., Wei, Y., Jiang, D., Zhang, X., Zuo, W., and Tian, Q. Controlvideo: Training-free controllable text-to-video generation. *arXiv preprint arXiv:2305.13077*, 2023c. 3

Zheng, W., Teng, J., Yang, Z., Wang, W., Chen, J., Gu, X., Dong, Y., Ding, M., and Tang, J. Cogview3: Finer and faster text-to-image generation via relay diffusion. *arXiv preprint arXiv:2403.05121*, 2024. 12

Zhou, D., Wang, W., Yan, H., Lv, W., Zhu, Y., and Feng, J. Magicvideo: Efficient video generation with latent diffusion models. *arXiv preprint arXiv:2211.11018*, 2022. 1

A. More Implementation Details

A.1. Implementation Details

We utilize the CTRL-Adapter (Lin et al., 2024), trained on the video diffusion model I2VGen-XL (Zhang et al., 2023b), as the backbone for generation, with the denoising steps set to $T = 50$. Initially, we center the untextured mesh sequence and pre-define six different viewpoints around the Y-axis in the XZ-plane, uniformly sampled in spherical coordinates, along with an additional top view with an elevation angle of zero and an azimuth angle of $30°$. For latent initialization, we first sample Gaussian noise on the latent textures and then render 2D latent samples for each view to improve the coherence of the generated outputs. During denoising, we upscale the latent resolution to 96×96 to reduce aliasing. We empirically set the blending coefficient to 0.2. It takes approximately 30 minutes to generate a video with 24 keyframes taken on an RTX A6000 Ada GPU. We decode the denoised latents in keyframes to RGB images, and then un-project and aggregate these images to transform the latent UV maps to RGB textures as previous works (Liu et al., 2023b; Cao et al., 2023; Huo et al., 2024). Finally, we interpolate the textures of the keyframes at intervals of 3 to synthesize the final video clips.

A.2. Denoising Algorithm of Our Method

We present the complete workflow of our method in Algorithm 1. For clarity, we omit the notation for the latent variables $\boldsymbol{z}_b$ representing the background plane texture, as they follow the same scheme as the foreground latents. The reference UV map $\mathcal{T}_{\mathcal{UV}}$ is constructed by progressively combining latent textures over time, with each new texture filling only the unoccupied texels in the reference UV map. We denote this process as "Combine" in the following workflow.

Algorithm 1 Tex4D

Input: UV maps $\mathcal{UV} = \{UV_1, ..., UV_K\}$; depth maps $\boldsymbol{\mathcal{D}} = \{D_{1,1}, ..., D_{1,V}, D_{2,1}, ..., D_{K,V}\}$; text prompt $\boldsymbol{\mathcal{P}}$; CTRL-Adapter model $\boldsymbol{\mathcal{C}}$; rendering operation $\mathcal{R}$; unproject operation $\mathcal{R}^{-1}$; cameras $\boldsymbol{c}$; T diffusion steps; $\mathcal{T}$ latent textures (including foreground and background); λ blending weight; k keyframes

$\mathcal{T}_T \sim \mathcal{N}(\mathbf{0}, \mathcal{I})$ // Sample noise in UV space
$\tilde{\boldsymbol{z}}_T, \boldsymbol{\mathcal{M}}_{\text{fg}} = \mathcal{R}(\mathcal{T}_T; \boldsymbol{c})$
$\boldsymbol{z}_{b,T} \sim \mathcal{N}(\mathbf{0}, \mathcal{I})$
$\boldsymbol{z} = \boldsymbol{z}_T = \tilde{\boldsymbol{z}}_T \odot \boldsymbol{\mathcal{M}}_{\text{fg}} + \boldsymbol{z}_{b,T} \odot (1 - \boldsymbol{\mathcal{M}}_{\text{fg}})$ // Composite latents
For $t = T, \ldots, 1$ **do**
  $\boldsymbol{z}_{b,t-1} \leftarrow \boldsymbol{\mathcal{C}}(\boldsymbol{z}_{b,t}; \boldsymbol{\mathcal{D}}, \boldsymbol{\mathcal{P}})$
  $\boldsymbol{\epsilon}_\theta \leftarrow \boldsymbol{\mathcal{C}}(\boldsymbol{z}_t; \boldsymbol{\mathcal{D}}, \boldsymbol{\mathcal{P}})$ // Estimate noise from $\boldsymbol{\mathcal{C}}$
  $\hat{\boldsymbol{z}}_0(\boldsymbol{z}_t) = \sqrt{\alpha_t} \cdot \boldsymbol{z}_t - \sqrt{1 - \alpha_t} \cdot \boldsymbol{\epsilon}_\theta$
  $\hat{\mathcal{T}}_0, \boldsymbol{\mathcal{M}}_{\mathcal{UV}} \leftarrow \mathcal{R}^{-1}(\hat{\boldsymbol{z}}_0; \boldsymbol{c}, \mathcal{UV})$ // Bake textures by Eq. 3
  $\mathcal{T}_{\mathcal{UV}} = \text{Combine}(\hat{\mathcal{T}}_0; \boldsymbol{\mathcal{M}}_{\mathcal{UV}})$
  For k in $1, ..., K$ **do**
    $\mathcal{T}_{t-1}^k = \sqrt{\alpha_{t-1}} \cdot \hat{\mathcal{T}}_0^k + \sqrt{1 - \alpha_{t-1}} \left(\sqrt{\frac{\alpha_t}{1-\alpha_t}} \cdot (\sqrt{\alpha_t} \mathcal{T}_t^k - \hat{\mathcal{T}}_0^k) + \sqrt{1 - \alpha_t} \cdot \mathcal{T}_t^k \right)$ // Denoise Eq. 5
    $\mathcal{T}_{t-1}^k = \left((1 - \lambda) \cdot \mathcal{T}_{t-1}^k + \lambda \cdot \mathcal{T}_{\mathcal{UV}} \right) \odot \boldsymbol{\mathcal{M}}_{\mathcal{UV}}^k + \mathcal{T}_{\mathcal{UV}} \odot \left(1 - \boldsymbol{\mathcal{M}}_{\mathcal{UV}}^k \right)$ // Blend textures by Eq. 8
  $\tilde{\boldsymbol{z}}_{t-1}, \boldsymbol{\mathcal{M}}_{\text{fg}} = \mathcal{R}\left(\mathcal{T}_{t-1}; \boldsymbol{c}, \mathcal{UV}\right)$
  $\boldsymbol{z}_{t-1} = \tilde{\boldsymbol{z}}_{t-1} \odot \boldsymbol{\mathcal{M}}_{\text{fg}} + \boldsymbol{z}_{b,t-1} \odot (1 - \boldsymbol{\mathcal{M}}_{\text{fg}})$ // Composite latents by Eq. 6
  $\boldsymbol{z} = \boldsymbol{z}_{t-1}$
Output: $\boldsymbol{z}$

A.3. V-Prediction

Tex4D is a zero-shot approach built on a pre-trained conditional video diffusion model, where v-prediction is a technique commonly used in video diffusion models (e.g., I2VGen-XL (Zhang et al., 2023b), Imagen (Ho et al., 2022), CogVideoX (Hong et al., 2022; Yang et al., 2024), CogView3 (Zheng et al., 2024)) to accelerate the training and prevent temporal color shifts. In our method, we utilize CTRL-Adapter (Lin et al., 2024), a conditional video diffusion model that guides video by depth maps trained on the DDIM v-prediction mechanism. Hence, we use v-prediction to ensure the proper functioning of the conditional video diffusion model.

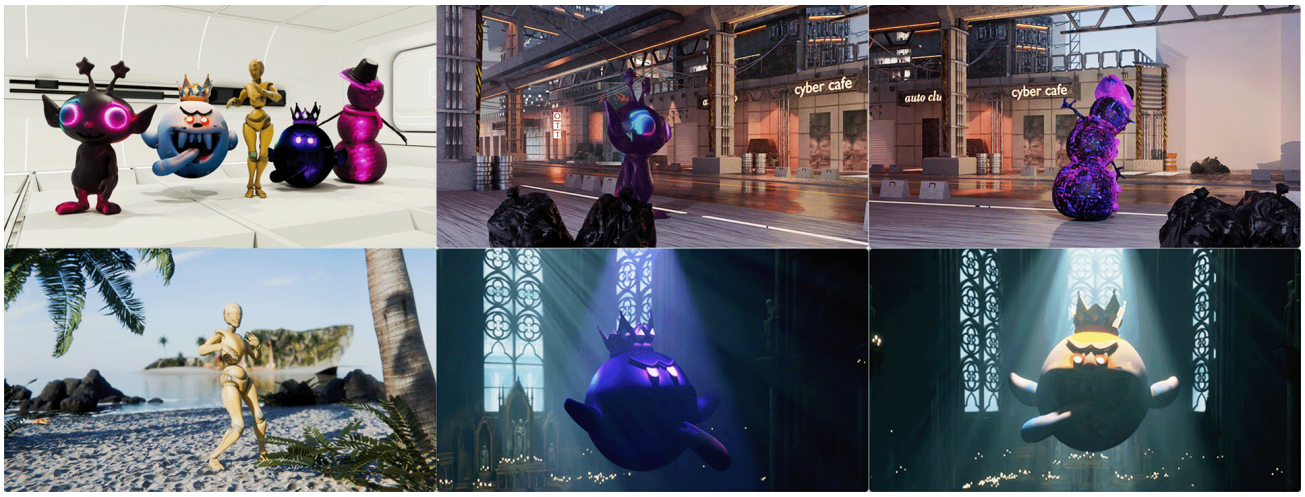


Figure 9. **Tex4D Applications.** Our synthesized dynamic textures can be easily integrated into graphics pipelines. We utilize the shader editor in Blender to animate textures with image sequence nodes. The dynamic textures help technical artists render vivid videos without additional lighting and mesh controls.

B. More Qualitative Results

B.1. Graphics Application and Video Demo

As shown in Fig. 9, Tex4D demonstrates its utility in the graphics pipeline by integrating dynamic texture sequences into Blender for rendering. This integration enables seamless visualization of animated textures directly on 3D models, showcasing Tex4D's capability to handle complex visual dynamics in real-world applications. We highly recommend the reviewers watch our supplementary videos for details.

B.2. Multi-view Results

In Fig. 15, we present additional characters generated by Tex4D, showcasing the method's effectiveness and its ability to generalize across a diverse array of styles and prompts. We also evaluate Tex4D on non-human character animations in Fig. 16, demonstrating its robust generalization capabilities across various types of mesh sequences. In each case, we provide two different views to show that our method can ensure multi-view consistency.

To emphasize the temporal changes in the generated textures, we also design some prompts, for example, 'flashed a magical light', 'dramatic shifts in lighting', 'cyberpunk style' in our experiments. As shown in Fig. 16, the results of 'ghost', 'King Boo' and 'Snowman' validate the effectiveness of our method in generating different level of temporal changes by a variety of textual prompts, while maintaining the consistency both spatially and temporally. Additionally, we provide a supplementary video that includes baseline comparisons and multi-view results for all examples.

B.3. Texture Results

In this section, we present the texture sequences, which are the intermediate results of our pipeline. Our method utilizes XATLAS to unwrap the UV maps from meshes without human labor. XATLAS is a widely used library for mesh parameterization and UV unwrapping, commonly integrated into popular tools and engines, facilitating efficient texture mapping in 3D graphics applications. As shown in Fig. 10, our method seamlessly bakes temporal changes, including lighting variations, wrinkles, and appearance transformations, directly into the textures, removing the need for manual post-production by technical artists.

C. More Ablation Results

Ablation on Background To show the effects of various background latent initialization strategies, we provide additional examples, including the approach used in the texture synthesis method (Liu et al., 2023b) and a background that contrasts sharply with the foreground object, as shown in Fig. 11. In detail, SyncMVD (Liu et al., 2023a) encodes the backgrounds

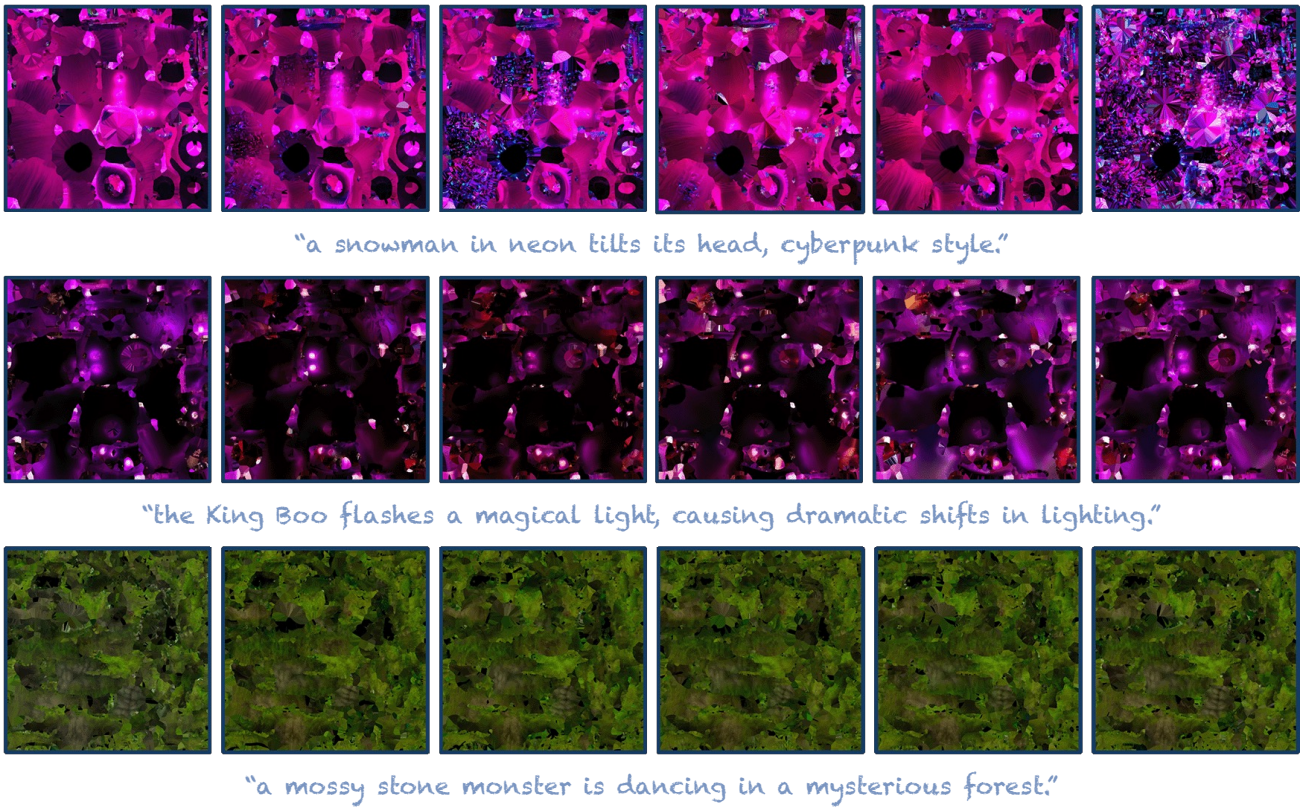


Figure 10. **Visualization of generated textures for mesh sequences.** Our method effectively incorporates temporal changes, such as lighting variations and appearance transformations, directly into the textures, eliminating the need for post-production by technical artists.

with alternative random solid color images. For the high-contrast background experiment, we use the latents obtained from the DDIM inversion of highly contrasted foreground and background to initialize our latents.

D. More Method Comparisons

D.1. Comparison with Depth-Conditioned Video Diffusion Models

While depth-conditioned video diffusion models effectively generate visually compelling results from a single viewpoint, they often struggle to maintain consistent multi-view representations of a single object, such as a character, across different perspectives. To highlight this limitation, we present multi-view results from the depth-conditioned video diffusion model in Fig. 13. The primary cause of this issue is that depth conditions are inherently view-dependent, in contrast to UV maps, which provide global information about the 3D space, enabling a unique mapping for each 3D point across all views.

D.2. Comparison with Textured Mesh Animations

In this section, we highlight the differences between our method and traditional approaches, demonstrating the effectiveness of 4D texturing in capturing temporal variations (e.g., lighting and wrinkles) within mesh sequences to produce vivid visual results. Traditional methods typically involve texturing a base mesh (often called a clay mesh) and animating it using skinning techniques. This animated sequence is then refined by technical artists who control scene lighting or simulate cloth dynamics to achieve the final visual presentation. This process is labor-intensive and demands specialized expertise in cinematic production and technical engines.

In contrast, our method presents a streamlined alternative by directly integrating complex temporal changes into mesh sequences. As shown in Fig. 4, 15 and 16, our approach effectively captures intricate temporal effects such as cloth wrinkles, dynamic lighting, and evolving appearances using textual prompts, significantly simplifying the workflow while maintaining high-quality results.

(a) Background w/ alternative random noise

(b) Background highly contrast w/ foreground

(c) Ours w/o background priors (foreground composited with our background)

Figure 11. **More ablation study on the background priors.** We present three ablations, including the approach used in the texture synthesis method SyncMVD (Liu et al., 2023b), a background that contrasts sharply with the foreground, and without background priors.

We demonstrate the limitations of traditional textured mesh animation in handling complex temporal changes in Fig. 12. Specifically, we employ the Text2Tex (Chen et al., 2023b) to generate the texture for the input mesh in T-pose and render it from multiple viewpoints. To ensure a fair comparison, we composite the rendered results with the background generated by our method. Notably, the 'ghost' and 'snowman' examples exhibit visible seams during animation due to self-occlusions are common appeared in dynamic poses but cannot be accurately predicted during T-pose texture generation. This results in empty texels and disrupts the visual continuity of the animation.

E. User Study

We show each participant 30 pairs of videos synthesized by different methods, capturing the same object from different views. For each pair, each participant is asked three questions in sequence:

- Which method has better appearance quality?
- Which method has better spatial and temporal consistency?
- Which method has better fidelity to the prompts?

F. Limitations and Discussion

One limitation of our method is the lack of seamless integration between the generated textures and the background, resulting in a disjointed appearance where the foreground and background elements may seem artificially stitched together. However, the dynamic textures remain globally consistent and can be directly applied to the downstream tasks, as shown in Fig. 9. To the best of our knowledge, no existing work tackles the foreground and background texture generation together because the task is computationally expensive, and the scene-level dataset is limited. Addressing the scene-level 4D texturing remains an open challenge that we aim to explore in future work. In addition, we notice that our method is relatively computationally intensive compared with other texture synthesis methods. The running time of our method primarily depends on the foundation model CTRL-Adapter, which takes approximately 5 minutes to generate a 24-frame video. We anticipate efficiency improvements with advancements in conditioned video diffusion models to further enhance the practicality of Tex4D.

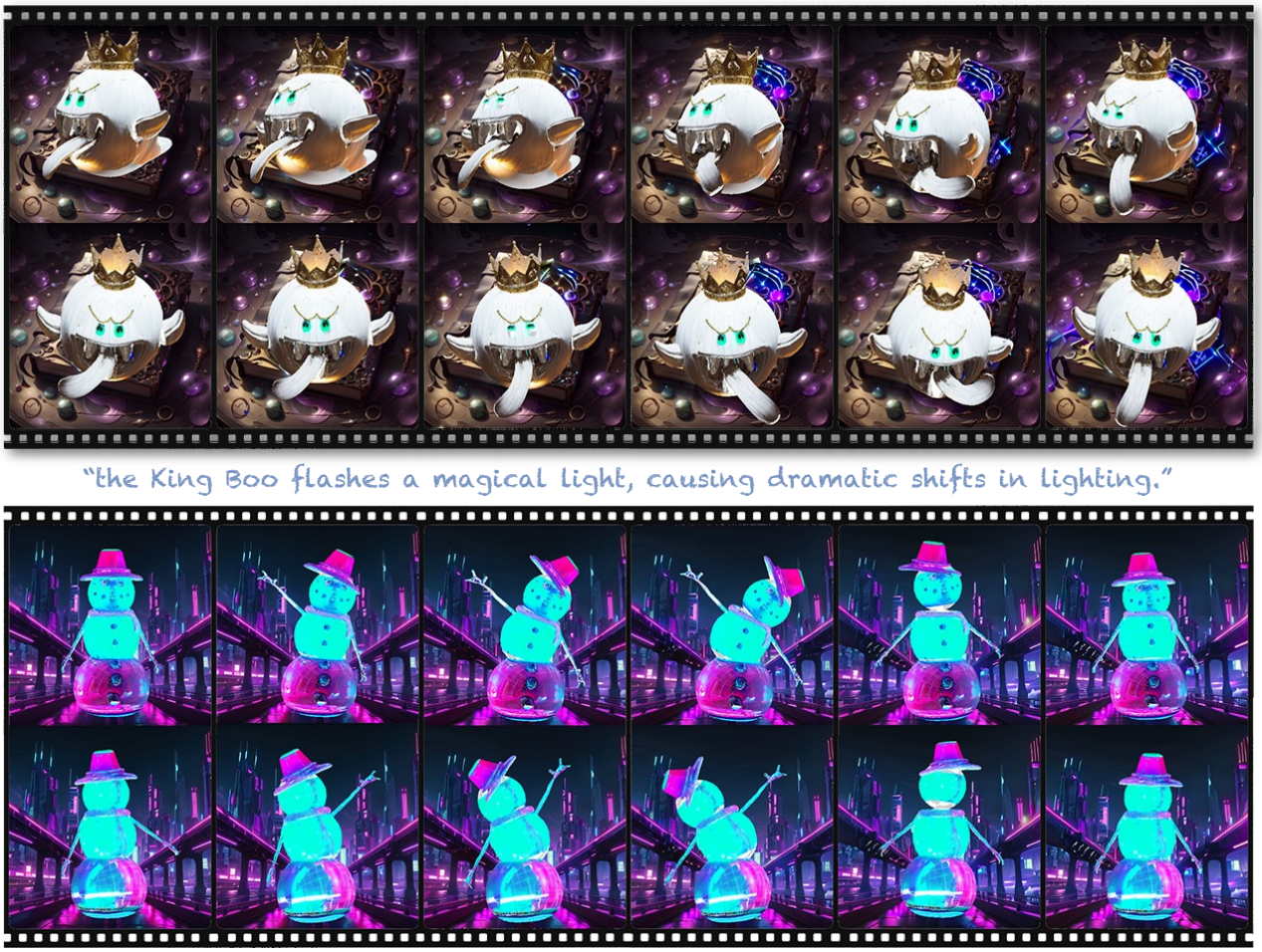


Figure 12. **Results of textured mesh animation.** We present the visual results of Text2Tex (Chen et al., 2023b) with our backgrounds. Text2Tex fails to capture temporal variations between frames and results in empty texels in invisible regions.

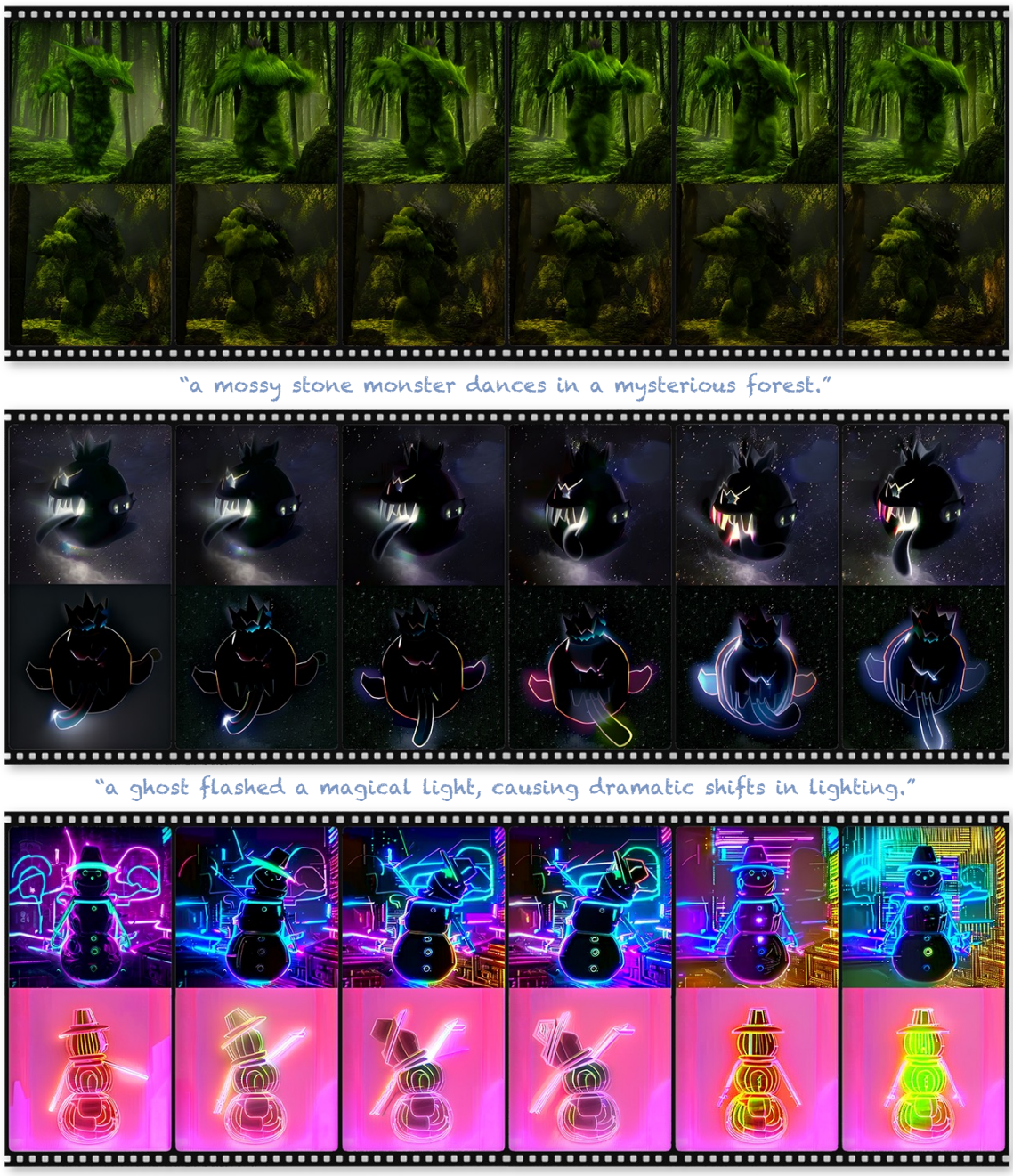


Figure 13. **Multi-view results from conditioned video diffusion models.** The conditioned video diffusion models struggle to maintain consistent multi-view representations of a single object due to the depth condition being view-dependent.

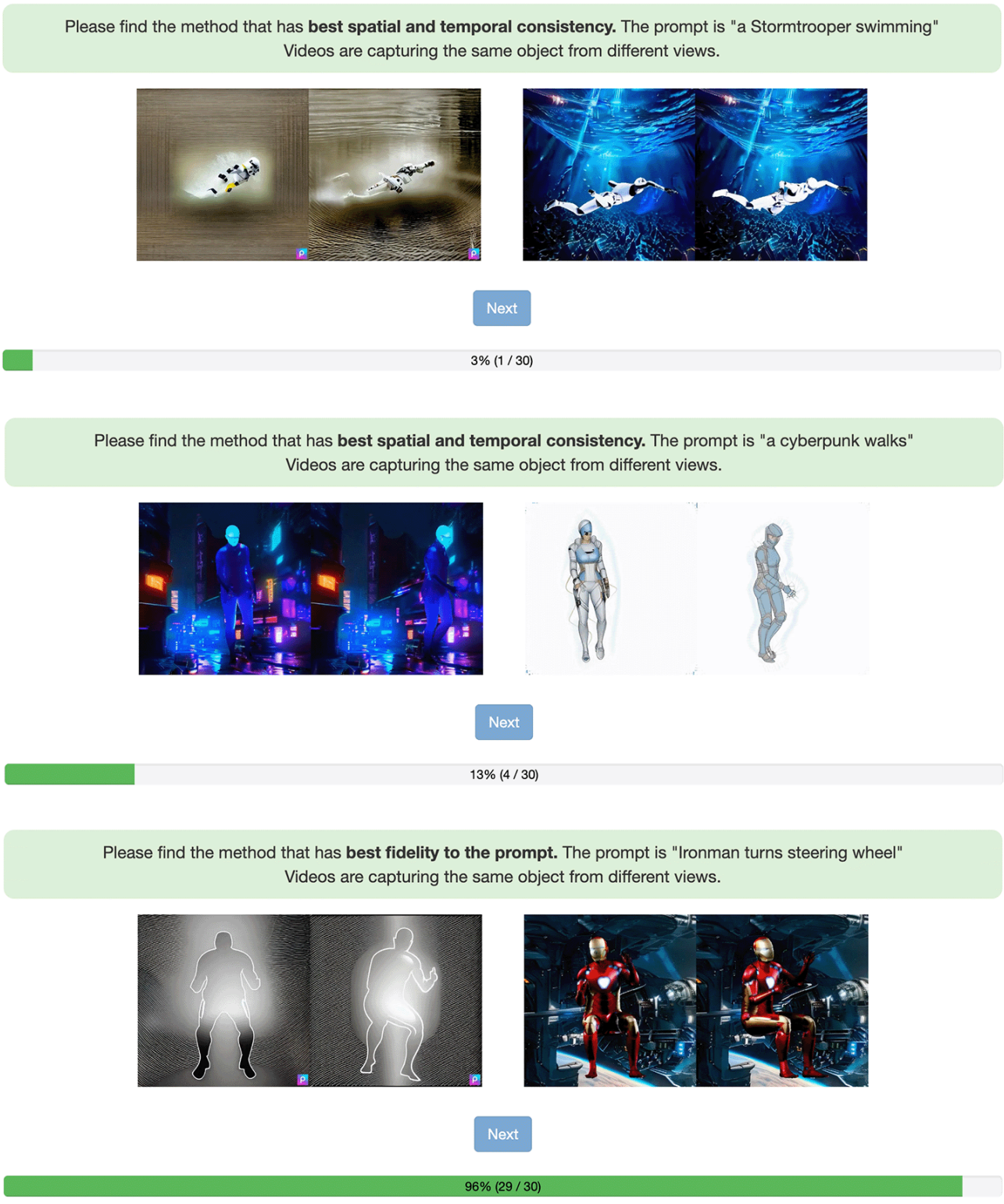


Figure 14. **User Study.** We provide more visual examples and include quantitative results from our user study. We evaluate the videos from three metrics: Appearance Quality, Spatial and Temporal Consistency, and Fidelity to the Prompt.

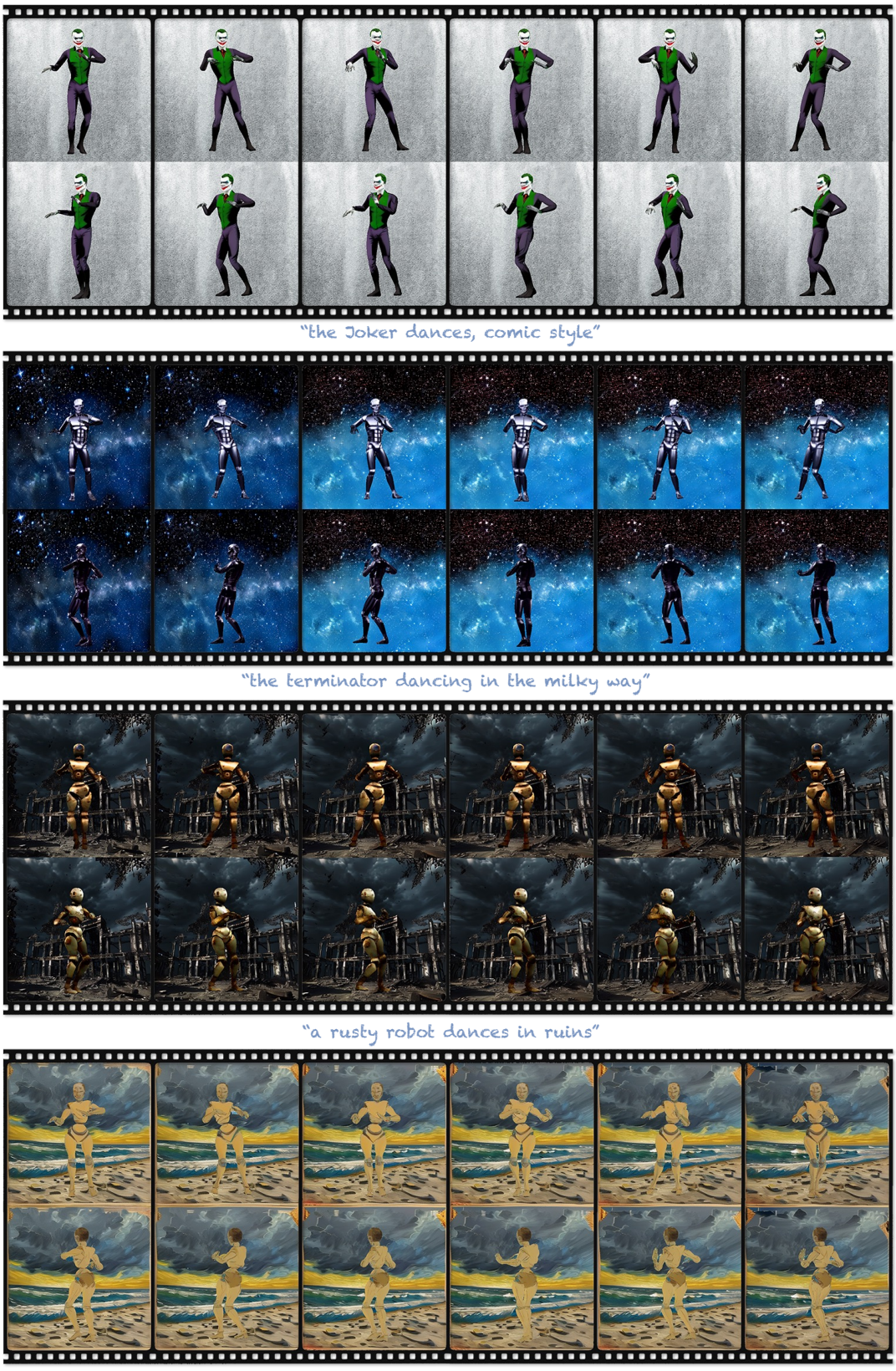


Figure 15. **More qualitative results.** We present the results of Tex4D with brief prompts, demonstrating the ability of Tex4D to generate multi-view consistent textures.

Figure 16. **More qualitative results on non-human character animations.** We present the results of Tex4D with prompts emphasizing the dynamics, demonstrating the ability of Tex4D to capture the dynamics from video diffusion models.