TransAnimate: Taming Layer Diffusion to Generate RGBA Video

Anonymous CVPR submission

Paper ID

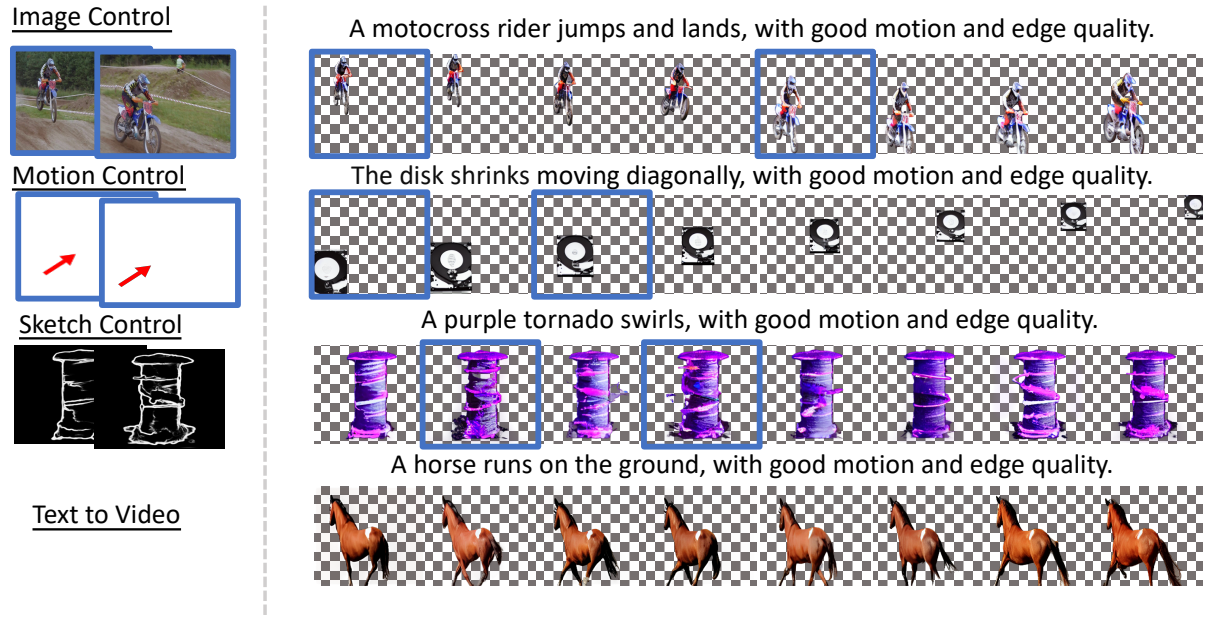


Figure 1. **RGBA Video Generation with TransAnimate.** By utilizing pre-trained text-to-transparent image models, the motion-guided control mechanism, and the proposed dataset, TransAnimate enables high quality generation and effective control of video content.

Abstract

Text-to-video generative models have made remarkable advancements in recent years. However, generating RGBA videos with alpha channels for transparency and visual effects remains a significant challenge due to the scarcity of suitable datasets and the complexity of adapting existing models for this purpose. To address these limitations, we present TransAnimate, an innovative framework that integrates RGBA image generation techniques with video generation modules, enabling the creation of dynamic and transparent videos. TransAnimate efficiently leverages pre-trained text-to-transparent image model weights and combines them with temporal models and controllability plug-ins trained on RGB videos, adapting them for controllable RGBA video generation tasks. Additionally, we introduce an interactive motion-guided control mechanism, where directional arrows define movement and colors adjust scaling, offering precise and intuitive control for designing game effects. To further alleviate data scarcity, we have developed a pipeline for creating an RGBA video dataset, incorporating high-quality game effect videos, extracted foreground objects, and synthetic transparent videos. Comprehensive experiments demonstrate that TransAnimate generates high-quality RGBA videos, establishing it as a practical and effective tool for applications in gaming and visual effects.

1. Introduction

Text-to-video generation models have achieved remarkable progress in recent years, enabling the creation of dynamic and visually engaging content widely applied in video editing, image animation, and motion customization. Furthermore, methods incorporating control signals like optical flow, pose skeletons, and multimodal inputs have been proposed to achieve more precise video generation guidance. Alpha channels play a vital role in producing high-quality visual effects, as they allow transparent elements such as smoke, fire, and light to seamlessly integrate into complex

scenes. This capability is particularly critical in game development, where transparency effects are central to creating immersive and realistic experiences. However, generating controllable RGBA videos with special effects remains highly desirable yet challenging.

Despite breakthroughs in video generation models, RGBA video synthesis remains underdeveloped. This dilemma stems from dual constraints: (1) the lack of large-scale RGBA video datasets severely limits algorithmic exploration, and (2) existing solutions like LayerDiffuse, while capable of generating static transparent images, cannot be seamlessly integrated with video generation models. The core challenge lies in leveraging massive RGB video pretraining resources and RGBA image generation models to build a data-efficient transparent video framework—an urgent problem we address.

This paper presents TransAnimate, a framework that unifies transparency modeling and video generation through three synergistic innovations. First, we establish an RGBA video synthesis pipeline and construct a foundational RGBA video dataset. Second, we combine AnimateDiff's motion modeling capabilities, LayerDiffuse's transparency generation expertise, and sparse-control video methods by fine-tuning adaptation layers on limited RGBA video data. Third, we design motion control mechanisms tailored for game visual effect artists, enabling pixel-precise control through directional arrows for trajectory specification and hue parameters for effect scaling.

While large-scale RGB video datasets contain millions of samples, the scarcity of RGBA video data remains a significant bottleneck. To address this limitation, we employ three complementary data collection strategies: (a) curating high-quality videos from game designers to capture authentic transparency properties, (b) extracting foreground object videos from instance segmentation data to enhance motion diversity, and (c) synthesizing controllable transparent videos using parametric transformations such as translation and scaling. Each strategy has distinct advantages and limitations: (a) provides superior visual quality but lacks category diversity, (b) introduces diverse motion patterns but suffers from incomplete foregrounds and imperfect edges, and (c) ensures precise edges and broad category coverage but is limited in motion complexity. Additionally, we manually curate high-quality results from models trained on these datasets and incorporate them as supplementary training data in an iterative refinement process. To balance these trade-offs, we propose a positive trigger strategy, which assigns distinct learnable tokens to each data source. This approach helps mitigate the impact of imperfect data distributions by preventing the model from absorbing unwanted features, thereby improving overall learning efficiency.

Architecturally, we first augment LayerDiffuse with temporal modules initialized using AnimateDiff's RGB-pretrained weights, eliminating costly training from scratch. Subsequent fine-tuning on RGBA video data enables effective adaptation of motion priors from AnimateDiff. For controllable generation, we repurpose RGB-pretrained SparseCtrl [13] with input-layer fine-tuning—experiments show minimal parameter adjustments suffice for high-consistency control.

Addressing game developers' needs, we implement a vector-chroma joint control system: Motion vectorization enables 8-directional trajectory control with velocity parameterization, while chromatic scaling regulates effect magnitudes. This integrated control mechanism significantly enhances flexibility for game visual effects creation, empowering designers to produce high-quality customizable effects with unprecedented efficiency and precision. By unifying transparency modeling and motion dynamics, our method achieves diverse RGBA video generation while maintaining controllability comparable to RGB video approaches.

Our contributions are summarized as follows:

- We propose TransAnimate, a framework for generating temporally coherent layered transparent videos and effects.
- We develop a dataset pipeline integrating game effects, segmented foregrounds, and synthetic videos, refined via iterative enhancement and a positive trigger strategy to mitigate distribution mismatches.
- We introduce novel directional-chromatic controls enabling pixel-accurate motion specification for game effects.

2. Related Work

Text-to-video generation. Recent advancements in text-to-video (T2V) generation [6, 15, 18, 21, 34, 44] have predominantly utilized diffusion models [16, 28, 32, 36], celebrated for their training stability and robust open-source ecosystems. A foundational milestone in this domain is the Video Diffusion Model [17], which extends a 2D image diffusion framework to handle video data by jointly training on image and video datasets from scratch. Building upon this, subsequent approaches leverage pre-trained image generators, such as Stable Diffusion [33], by integrating temporal layers into the existing 2D architectures and fine-tuning on expansive video datasets [1]. Among innovative methods, Align-Your-Latents [3] achieves efficient T2V conversion by aligning noise maps sampled independently for each frame, while AnimateDiff [12] incorporates a modular motion layer, enabling the generation of high-quality animations on customized image generation backbones [35]. To address temporal coherence challenges, Lumiere [2] eliminates the need for a temporal super-resolution module by directly generating videos with consistent frame rates. Further notable advancements include adopting scalable trans-

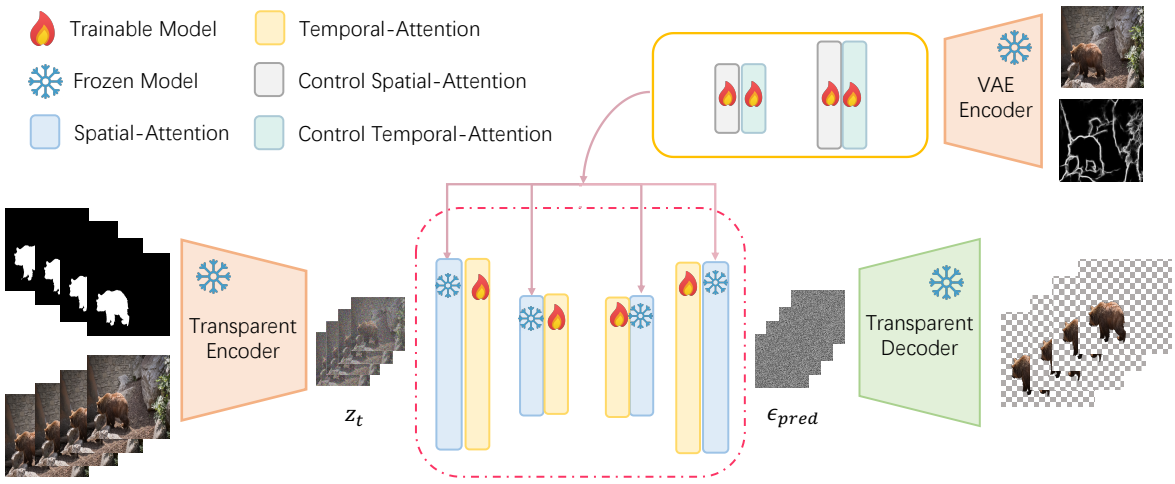
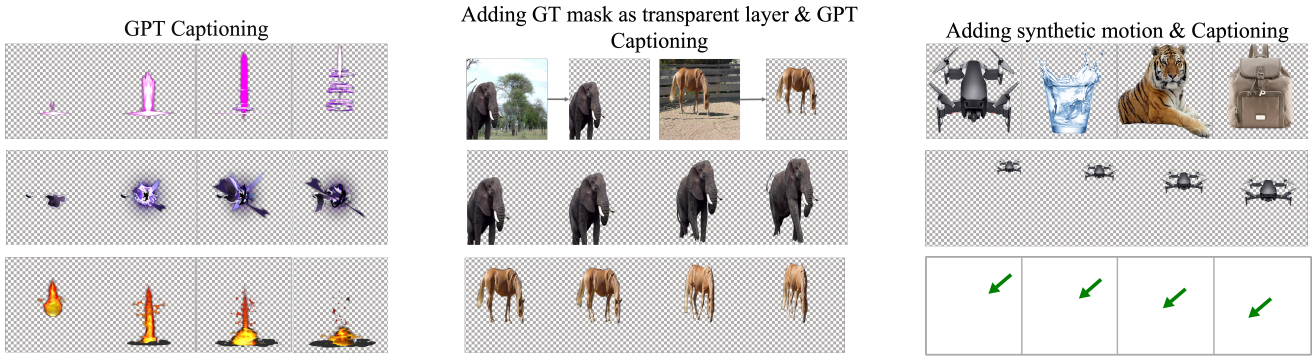


Figure 2. **Framework Overview.** TransAnimate generates transparent videos by learning motion patterns from videos. A frozen Transparent Encoder extracts features, refined by Temporal Attention and Linear Layers. Pre-trained SparseCtrl weights enable control via motion, sketches, and RGB images. A frozen Transparent Decoder reconstructs transparent frames, enhancing generation with limited RGBA data.

former architectures [26], leveraging spatiotemporal compressed latent spaces, as demonstrated by W.A.L.T.[14] and Sora[4], and utilizing discrete tokens alongside language models for video synthesis, exemplified by VideoPoet [23]. Building on previous RGB video generation approaches, our method introduces text-to-RGBA video generation tailored for game effects design, addressing a significant gap in this domain.

Controllable video generation. Existing text-to-video models often suffer from limited control, as relying solely on text descriptions introduces ambiguity and reduces precision. To address this, various methods incorporate explicit guidance signals. For instance, some approaches use depth maps or skeleton sequences to dictate scene layout or human motion with greater accuracy [7, 8, 11, 20, 37, 45]. Others leverage image-based control signals, which enhance video quality and improve temporal consistency [11, 27]. Despite these advances, controlling camera motion during video generation remains underexplored. AnimateDiff [12] applies LoRA [19] fine-tuning to adapt model weights for specific camera angles. Direct-a-Video [41] incorporates a camera embedder for pose control but supports only basic parameters, limiting its capacity to simple motions like panning. MotionCtrl [39] extends this idea with additional input parameters to enable complex camera trajectories. However, its reliance on fine-tuning components of the diffusion model compromises generalization. In this work, we introduce a novel motion-guided control framework specifically designed for generating transparent videos with predefined motion directions and scales. This method fills a critical gap by offering game effect designers a powerful and flexible tool for creating dynamic, high-quality content that meets the unique demands of their workflows.

Transparent Layer Processing Most existing approaches for transparent layer generation focus primarily on image generation, closely linked to image matting techniques. For instance, PPMatting [5] is a neural network model for image matting, trained from scratch using standard matting datasets. Building on advancements in foundational models, Matting Anything [24] leverages the Segment Anything Model (SAM) [22] as its backbone for matting tasks, while VitMatte [42] utilizes a Vision Transformer (ViT) in a tri-map-based matting framework. Expanding beyond traditional matting, recent innovations explore the integration of layered effects in diffusion models. LayerDiffuse [43] introduces the concept of "latent transparency." This technique encodes alpha channel transparency directly into the latent manifold of a pretrained latent diffusion model by modifying the Variational Autoencoder (VAE) to decode alpha channels, enabling richer transparency effects. Despite these advancements, transparent video generation remains significantly underexplored. SAM-2 [31] integrates a Transformer architecture with streaming storage and memory mechanisms to provide coherent segmentation predictions across video sequences. However, SAM-2 is not explicitly designed for video generation and struggles to create diverse, high-quality transparent layers essential for effects like fire, light, and smoke, which are crucial in game effect design. In this work, we introduce a novel framework tailored for game effect generation, enabling text-to-RGBA video generation with support for various control modalities. Our approach addresses the limitations of existing methods, providing an innovative solution for creating visually compelling and dynamic transparent layers in video content, specifically for gaming



(a). Game effect dataset (b). Foreground dataset (c). Synthesized motion dataset

Figure 3. **Illustration of TransAnimate.** Our dataset consists of (a) Animate Dataset with 3,000 high-quality game effect videos, (b) Foreground Object Videos Dataset with 7,000 segmented videos capturing diverse motion patterns, and (c) Synthesized Transparent Motion Videos with 20,000 generated samples featuring controlled motion transformations. For synthesized dataset, from top to bottom, it represents Motion Caption, Raw Image, Synthetic, and Motion Control.

applications.

3. Method

Our approach enables a latent diffusion model to learn transferable motion priors from video data while leveraging existing text-to-transparent-image generation methods to produce transparent videos. In Section 3.1, we introduce the foundational principles of our method. Section 3.2 details the dataset preparation process. Section 3.3 demonstrates how the proposed positive prompts enhance motion and edge quality. Section 3.4 elaborates on the TransAnimate framework for transparent video generation. Finally, Section 3.5 explores how pretrained SparseCtrl weights are adapted to achieve controllable transparent video generation.

3.1. Preliminary

Stable Diffusion (SD), a widely-used open-source text-to-image (T2I) model, performs the diffusion process in the latent space of a pre-trained autoencoder. This process perturbs the encoded image $z_0 = \mathcal{E}(x_0)$ into z_t at step t by adding noise:

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

where $\bar{\alpha}_t$ governs the noise strength. The denoising UNet ϵ_θ, utilizing an MSE loss, predicts noise while integrating ResNet, self-attention, and cross-attention mechanisms to incorporate text conditions effectively.

AnimateDiff [12] is an extension of T2I models, leverages video data to learn transferable motion priors, enabling animation generation through iterative denoising. The method consists of three core components: a domain adapter, a motion module, and a LoRA-based extension, all seamlessly integrated into the T2I framework during inference.

To model temporal dynamics, AnimateDiff inflates the 2D T2I model to handle 3D video data. Video tensors $x \in \mathbb{R}^{b \times c \times f \times h \times w}$ are reshaped to allow the image layers to process frames independently, while a newly introduced motion module captures temporal dependencies across frames.

The motion module in AnimateDiff employs a temporal Transformer enhanced with self-attention blocks and sinusoidal position encodings. Input features are reshaped into sequences along the temporal axis to facilitate inter-frame information exchange:

$$z_{out} = \text{Softmax}(QK^T/\sqrt{c}) \cdot V, \tag{2}$$

where Q, K, V are linear projections of the input features. This design effectively encodes motion priors, enabling the generation of coherent and realistic animations.

3.2. New Dataset for Transparent Video Generation

Achieving high-quality transparent video generation presents three major challenges: (1) ensuring high-fidelity transparent layers, (2) capturing realistic and dynamic motion, and (3) maintaining sufficient diversity in video content. To address these challenges, we construct a comprehensive dataset by collecting and synthesizing three distinct types of video data, ensuring that our network learns from diverse samples with high-quality motion and transparency.

Animate Dataset. The animate dataset consists of 3,000 professionally curated game effect videos created by expert designers. These videos feature complex and visually rich animations, such as explosions, energy transformations, and magical effects. Each video is carefully selected to ensure high transparency quality and realistic motion.

Foreground Object Videos Dataset. To enhance diversity and capture a wide range of motion patterns, we construct a foreground object dataset by extracting 7,000 videos from large-scale video instance segmentation datasets, including VideoMatte240K [25], MeViS [9], YouTube-VOS [40], DAVIS [29], and MOSE [10]. These datasets cover a broad range of object categories, ensuring

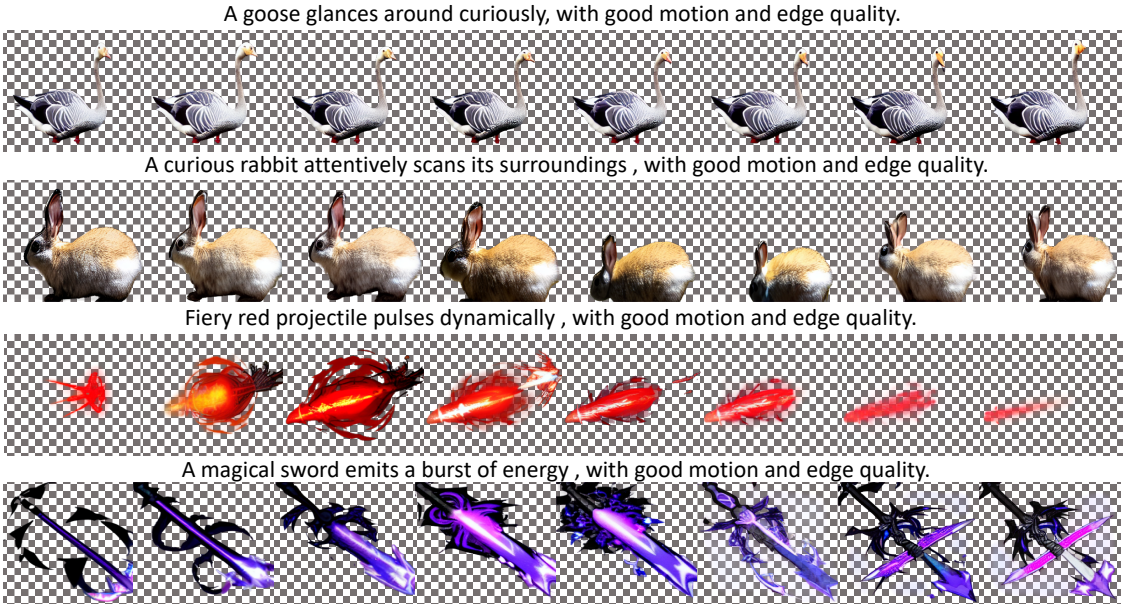


Figure 4. Text-to-RGBA video generation results of TransAnimate.

Dataset	RGBA Edge Quality	Motion Quality	Category Diversity
Game Effects Dataset	✓	✓	✗
Foreground Dataset	✗	✓	✓
Synthetic Dataset	✓	✗	✓
Iterated Dataset	✓	✓	✓

Table 1. Comparison of datasets based on three key quality aspects.

the dataset includes diverse appearances, textures, and motion trajectories. Using video instance segmentation techniques, we extract foreground objects and retain only those exhibiting significant motion dynamics.

Synthesized Transparent Motion Videos. To further improve transparency quality and motion diversity, we generate 20,000 synthetic transparent videos by applying controlled motion transformations to foreground images from predefined transparent classes. These transformations include translation, scaling, and rotation, ensuring that the dataset captures a wide variety of movement behaviors. The motion characteristics are visualized with directional arrows (for translation) and color-coded indicators (green for scaling up, blue for stability, and red for shrinking). Captions are generated based on the transparent class and applied motion transformation, ensuring a strong alignment between textual descriptions and visual content.

Data Iteration We jointly train the TransAnimate model on the aforementioned three datasets, employing a data iteration strategy to manually curate high-quality generated results as supplementary training data. This effectively mitigates the scarcity of RGBA data. The animate dataset ensures high-quality transparent layers and realistic motion, the foreground object dataset provides diversity in motion and appearance, and the synthesized transparent motion dataset further refines transparency and motion control. To address the limitations of different datasets, we adopt a negative trigger strategy to filter out undesirable features, significantly enhancing the performance of transparent video generation.

3.3. Positive Triggers

As shown in Table 1, the datasets we collect exhibit certain limitations in terms of quality attributes. Specifically, the Foreground Dataset demonstrates strong motion quality but suffers from poor edge quality, while the Synthetic Dataset excels in edge quality but lacks sufficient motion fidelity. To mitigate these issues, we leverage positive triggers during training, explicitly guiding the model to learn high-quality attributes from different datasets.

For instances originating from the Synthetic Dataset, we include positive triggers emphasizing good edge quality, helping the model recognize and reinforce well-defined edges. Conversely, for samples from the Foreground Dataset, we incorporate positive triggers highlighting strong motion attributes, enabling the model to better capture dynamic motion patterns. This targeted approach allows the network to develop a more comprehensive understanding of both motion and edge quality during training.

During inference, we combine positive triggers for both good edge and motion quality, ensuring that the generated results exhibit improved sharpness and dynamic consistency. This strategy enhances the overall quality of

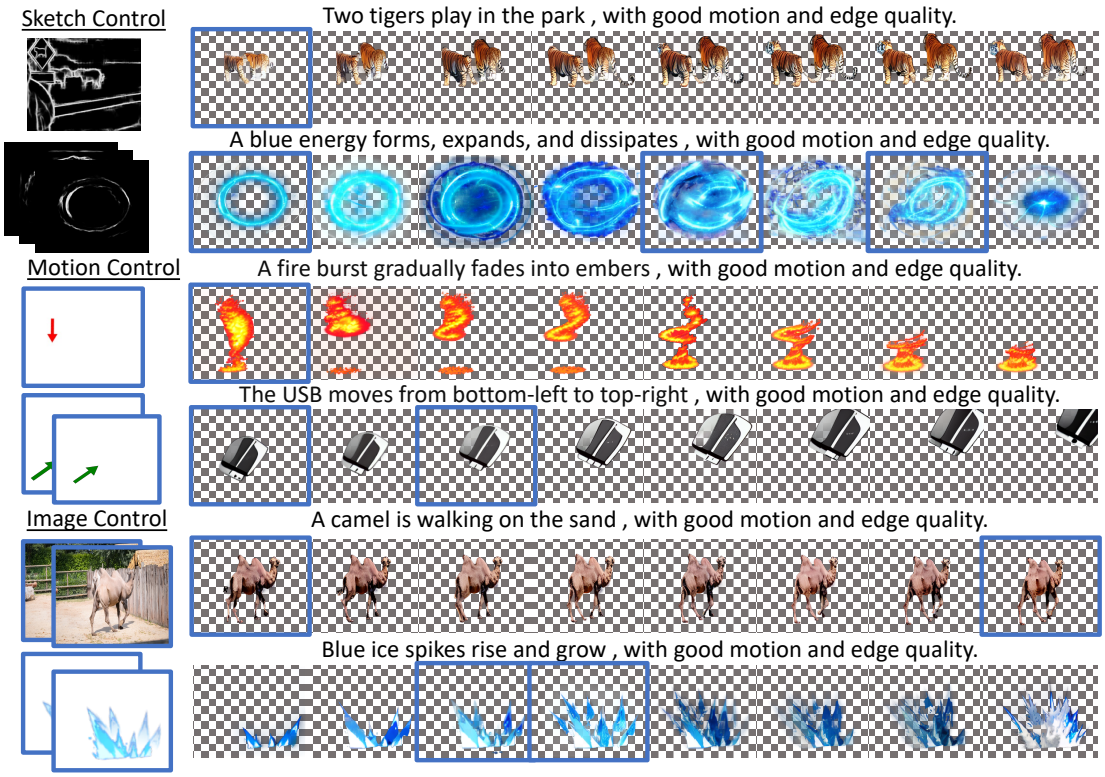


Figure 5. Conditional generation results from TransAnimate. The qualitative results are results with sketch, depth, and RGB image conditions. The input conditions are displayed on the left, while the keyframes guided by these conditions are highlighted with blue borders.

RGBA video generation by bridging the strengths of multiple datasets.

3.4. Text2RGBA Video Generation

Our method introduces a novel approach to RGBA video generation by extending text-to-transparent-image frameworks, particularly by using LayerDiffuse [43]. Unlike prior methods focused solely on RGB, we address the unique requirements of generating transparent animations, crucial for game effects design.

The framework integrates a motion module inspired by AnimateDiff [12], enabling animation through iterative denoising. This module is trained independently while keeping the text-to-transparent-image generation network frozen. By incorporating this learnable module into LayerDiffuse, we achieve temporal animation while preserving the high-quality transparency features of the original model.

To adapt LayerDiffuse for video generation, we extend its architecture to process 5D video tensors $\boldsymbol{x} \in \mathbb{R}^{b\times c\times f\times h\times w}$, where b, c, f, h, and w denote the batch size, channels, frames, height, and width, respectively. Following a network inflation strategy similar to AnimateDiff [3], we allow the image layers to process frames independently by reshaping the temporal axis f into the batch axis during feature extraction and restoring it afterward.

Conversely, the motion module reshapes spatial dimensions (h, w) into the batch axis during temporal processing, which is restored post-processing. This design ensures efficient and independent handling of spatial and temporal dimensions.

The motion module captures temporal dependencies by dividing the reshaped feature map along the temporal axis into a sequence of vectors $\{z_1, z_2, \ldots, z_f\}$, where each vector corresponds to a frame. These vectors are processed through self-attention layers, enabling temporal interactions:

$$z_{out} = \text{Softmax}\left(\frac{QK^\top}{\sqrt{c}}\right)V, \tag{3}$$

where $Q = W^Q z$, $K = W^K z$, and $V = W^V z$ are the query, key, and value projections of the input features. This mechanism ensures temporal coherence by enabling information flow across frames.

During training, the encoder, decoder, and 3D U-Net components of LayerDiffuse are frozen, and only the motion module is trained. The diffusion process progressively adds noise to the latent representation $\boldsymbol{x}_t$ over time steps t. The network predicts the added noise using a denoising

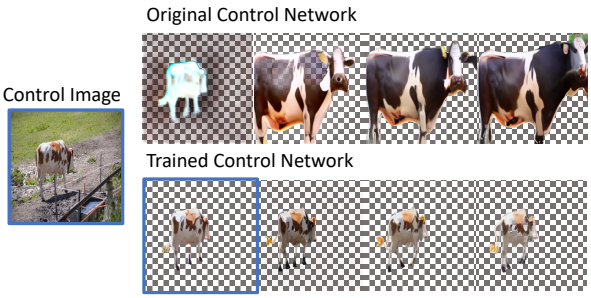


Figure 6. Qualitative comparison of SparseControl's original weights with our trained weights.

	FVD	CLIP	RGBA Quality	Motion Quality
Original control	1400.86	19.22	16.32	18.17
Trained control	**407.41**	**30.57**	**83.62**	**82.28**

Table 2. Quantitative comparison of SparseControl's original weights with our trained weights.

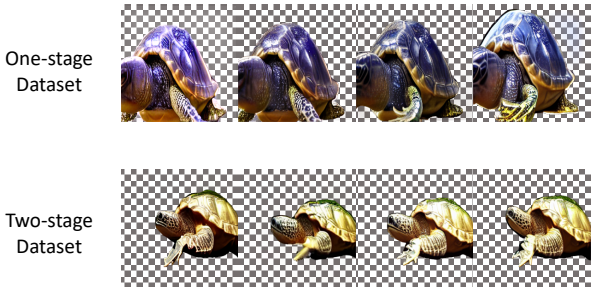


Figure 7. Qualitative comparison of results obtained using one-stage and two-stage training datasets.

	FVD	CLIP	RGBA Quality	Motion Quality
One-stage	485.63	28.55	67.05	69.17
Two-stage	**429.82**	**29.32**	**76.32**	**78.79**

Table 3. Quantitative comparison of results obtained using one-stage and two-stage training datasets.

model ϵ_θ, minimizing the following objective:

$$\mathcal{L} = \mathbb{E}_{\mathcal{E}(x_0^{1:f}),y,\epsilon^{1:f}\sim\mathcal{N}(0,I),t}\left[\|\epsilon - \epsilon_\theta(z_t^{1:f}, t, \tau_\theta(y))\|_2^2\right]. \tag{4}$$

At inference, the pre-trained latent transparency decoder of LayerDiffsion, $\mathcal{D}(\cdot,\cdot)$, reconstructs the video's RGB and alpha channels. Given the adjusted latent representation $\boldsymbol{x}_a$, the decoder outputs the transparent video:

$$[\hat{\boldsymbol{I}}_c\ \hat{\boldsymbol{I}}_\alpha] = \mathcal{D}(\hat{\boldsymbol{I}}, \boldsymbol{x}_a), \tag{5}$$

where $\hat{\boldsymbol{I}}_c$ and $\hat{\boldsymbol{I}}_\alpha$ represent the reconstructed color and alpha channels, respectively. This process ensures high-quality RGBA video outputs suitable for game effects.

3.5. Controllable RGBA Video Generation

This section demonstrates how to apply SparseCtrl [11], which was trained on RGB videos, to RGBA controllable generation, incorporating three types of controls: motion control, RGB image control, and sketch control. Notably, we propose utilizing only the RGB channels of RGBA videos as control inputs, which is more practical for downstream tasks. We reuse the pre-trained weights of SparseCtrl and keep them frozen, while fine-tuning the Adapter layer on RGBA videos to align with TransAnimate Unet. This strategy effectively leverages the prior knowledge of the pre-trained RGB model, enabling RGBA video generation and control even with limited RGBA training data.

Motion-Guided Video Generation. Motion guidance is crucial for intuitive game effect design, enabling creators to precisely control the direction and scale of animations. Existing methods often lack support for such fine-grained functionality. Our motion control mechanism allows users to define the generation direction using arrows and indicate the effect scale using colors. For example, a transparent video effect can be guided to move from left to right while gradually shrinking, providing game designers with enhanced control and adaptability for their creative workflows.

Sketch-to-Video Generation. Sketches offer an intuitive and accessible way for non-professional users to guide T2V generation. SparseCtrl [11] allows users to input sketches to shape video content. A single sketch can define the overall layout, while multiple sketches—such as for the first, last, and key intermediate frames—can guide coarse motion and transitions. This functionality is especially suited for tasks like storyboarding, enabling creators to visualize and iterate on video concepts effortlessly.

Image Animation and Transition. SparseCtrl [11] unifies various video generation tasks, including video prediction, animation, and interpolation, under a single framework leveraging RGB image conditions. Image animation generates videos based on the first frame, while transitions are guided by both the first and last frames. Video prediction uses initial frames to extrapolate motion, and interpolation creates smooth transitions between sparsely provided keyframes. By unifying these tasks, SparseCtrl broadens the applicability of video generation methods, making them versatile tools for diverse creative and practical scenarios.

4. Experiments

4.1. Experiment Setting

Setup. Our method leverages the pre-trained VAE encoder, decoder, and 3D-UNet architecture from LayerDiffuse, with

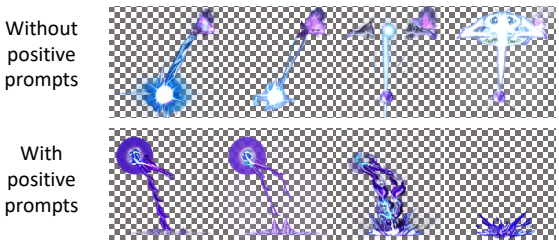


Figure 8. Qualitative comparison of results with and without the proposed positive triggers.

the motion module adapted from AnimateDiff. The training resolution is set to 256×256 for 16 frames. We train the model for a total of 3,000 iterations with a batch size of 16, utilizing two NVIDIA A100 80GB GPUs. The learning rate is set to 1×10^{-5}, ensuring stable and effective optimization.

Evaluation Metrics. We evaluate RGBA and motion quality using CLIP Score [30], FVD [38], and a user study. In the user study, 15 participants assess 50 generated videos based on two key aspects: (1) the quality of RGBA edges and (2) the realism and smoothness of motion.

4.2. Qualitative Results

Figure 4 presents qualitative results. These examples illustrate the model's ability to produce high-quality transparency effects while preserving fine details and structural consistency. Furthermore, the results demonstrate the model's strong generalization across various content types, reinforcing its robustness and adaptability.

4.3. Qualitative Controllable Results

Our approach seamlessly integrates with SparseControl to facilitate multi-modal video generation, enabling precise control over object structure, movement, and appearance. Users can provide sketches, motion trajectories, or reference images to guide the generation process. Figure 5 illustrates qualitative results, demonstrating the model's capability to generate transparent images with high fidelity while maintaining user-defined control inputs. These results highlight the model's ability to produce visually compelling outputs and generalize effectively across diverse content.

4.4. Ablative Study

Effectiveness of Control Training. We evaluate the effectiveness of our control method by comparing it with the pre-trained weights provided by SparseControl. As shown in Figure 6 and Table 2, directly applying RGB pre-trained weights fails to preserve structural consistency in controllable RGBA video generation. In contrast, our approach successfully adapts RGB-based control to RGBA videos, achieving superior alignment with the control image while maintaining transparency effects. These results highlight the ability of pre-trained RGB control weights to generalize effectively to RGBA video generation, addressing the challenges posed by the scarcity of RGBA training data.

	FVD	CLIP	RGBA Quality	Motion Quality
w/o Positive Prompts	458.37	28.95	71.58	73.31
w Positive Prompts	**429.827**	**29.32**	**76.32**	**78.79**

Table 4. Quantitative comparison of results with and without the proposed positive triggers.

Two-Stage Training. We examine the impact of a two-stage training strategy on generation quality. As illustrated in Figure 7, this approach enhances dataset diversity and scalability without compromising visual fidelity, resulting in more stable and high-quality video generation. Additionally, the quantitative results in Table 3 confirm that the two-stage training strategy significantly improves the final performance of the generated RGBA videos, demonstrating its effectiveness in refining output quality.

Positive Triggers. We analyze the contribution of the proposed positive prompts strategy in enhancing generation quality, particularly in motion coherence and RGBA edge fidelity. As shown in Figure 8, this technique enriches dataset diversity while preserving structural integrity, leading to more consistent and visually refined outputs. Furthermore, Table 4 provides quantitative evidence that incorporating positive prompts significantly enhances the overall quality of generated RGBA videos, reinforcing its effectiveness as a control mechanism.

5. Conclusion

In this paper, we address the underexplored challenge of RGBA video generation by introducing TransAnimate, a novel framework that combines transparency modeling with motion dynamics to generate high-quality, layered, transparent videos with temporal coherence. To tackle the scarcity of RGBA datasets, we propose a three-step data creation strategy, leveraging high-quality game effect videos, extracted foreground objects, and synthetic transparent videos with controlled motion dynamics. We also introduce a motion-guided control mechanism that enables precise adjustments to motion direction and scaling, tailored to game effect creation. By utilizing RGB channels as control inputs, our approach effectively leverages pre-trained RGB models, enabling robust RGBA video generation and control with limited RGBA data. Our contributions bridge critical gaps in RGBA video generation, providing a unified framework with controllability comparable to RGB methods while extending capabilities to layered transparency. This work opens new possibilities for immersive and customizable content creation in gaming and visual effects.

References

[1] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1728–1738, 2021. 2

[2] Omer Bar-Tal, Hila Chefer, Omer Tov, Charles Herrmann, Roni Paiss, Shiran Zada, Ariel Ephrat, Junhwa Hur, Yuanzhen Li, Tomer Michaeli, et al. Lumiere: A space-time diffusion model for video generation. *arXiv preprint arXiv:2401.12945*, 2024. 2

[3] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22563–22575, 2023. 2, 6

[4] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. 3

[5] Guowei Chen, Yi Liu, Jian Wang, Juncai Peng, Yuying Hao, Lutao Chu, Shiyu Tang, Zewu Wu, Zeyu Chen, Zhiliang Yu, et al. Pp-matting: high-accuracy natural image matting. *arXiv preprint arXiv:2204.09433*, 2022. 3

[6] Haoxin Chen, Menghan Xia, Yingqing He, Yong Zhang, Xiaodong Cun, Shaoshu Yang, Jinbo Xing, Yaofang Liu, Qifeng Chen, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter1: Open diffusion models for high-quality video generation, 2023. 2

[7] Weifeng Chen, Jie Wu, Pan Xie, Hefeng Wu, Jiashi Li, Xin Xia, Xuefeng Xiao, and Liang Lin. Control-a-video: Controllable text-to-video generation with diffusion models. *arXiv preprint arXiv:2305.13840*, 2023. 3

[8] Zhimin Chen, Longlong Jing, Yingwei Li, and Bing Li. Bridging the domain gap: Self-supervised 3d scene understanding with foundation models. *Advances in Neural Information Processing Systems*, 36:79226–79239, 2023. 3

[9] Henghui Ding, Chang Liu, Shuting He, Xudong Jiang, and Chen Change Loy. Mevis: A large-scale benchmark for video segmentation with motion expressions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2694–2703, 2023. 4

[10] Henghui Ding, Chang Liu, Shuting He, Xudong Jiang, Philip HS Torr, and Song Bai. Mose: A new dataset for video object segmentation in complex scenes. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 20224–20234, 2023. 4

[11] Yuwei Guo, Ceyuan Yang, Anyi Rao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Sparsectrl: Adding sparse controls to text-to-video diffusion models. *arXiv preprint arXiv:2311.16933*, 2023. 3, 7

[12] Yuwei Guo, Ceyuan Yang, Anyi Rao, Yaohui Wang, Yu Qiao, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023. 2, 3, 4, 6

[13] Yuwei Guo, Ceyuan Yang, Anyi Rao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Sparsectrl: Adding sparse controls to text-to-video diffusion models. In *European Conference on Computer Vision*, pages 330–348. Springer, 2024. 2

[14] Agrim Gupta, Lijun Yu, Kihyuk Sohn, Xiuye Gu, Meera Hahn, Li Fei-Fei, Irfan Essa, Lu Jiang, and José Lezama. Photorealistic video generation with diffusion models. *arXiv preprint arXiv:2312.06662*, 2023. 3

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

[16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 2

[17] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *arXiv:2204.03458*, 2022. 2

[18] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022. 2

[19] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 3

[20] Li Hu, Xin Gao, Peng Zhang, Ke Sun, Bang Zhang, and Liefeng Bo. Animate anyone: Consistent and controllable image-to-video synthesis for character animation. *arXiv preprint arXiv:2311.17117*, 2023. 3

[21] Johanna Karras, Aleksander Holynski, Ting-Chun Wang, and Ira Kemelmacher-Shlizerman. Dreampose: Fashion video synthesis with stable diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22680–22690, 2023. 2

[22] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023. 3

[23] Dan Kondratyuk, Lijun Yu, Xiuye Gu, José Lezama, Jonathan Huang, Rachel Hornung, Hartwig Adam, Hassan Akbari, Yair Alon, Vighnesh Birodkar, et al. Videopoet: A large language model for zero-shot video generation. *arXiv preprint arXiv:2312.14125*, 2023. 3

[24] Jiachen Li, Jitesh Jain, and Humphrey Shi. Matting anything. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1775–1785, 2024. 3

[25] Shanchuan Lin, Andrey Ryabtsev, Soumyadip Sengupta, Brian L Curless, Steven M Seitz, and Ira Kemelmacher-Shlizerman. Real-time high-resolution background matting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8762–8771, 2021. 4

[26] Xin Ma, Yaohui Wang, Gengyun Jia, Xinyuan Chen, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte: Latent diffusion transformer for video generation. *arXiv preprint arXiv:2401.03048*, 2024. 3

[27] Haomiao Ni, Changhao Shi, Kai Li, Sharon X Huang, and Martin Renqiang Min. Conditional image-to-video generation with latent flow diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18444–18455, 2023. 3

[28] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023. 2

[29] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017. 4

[30] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 8

[31] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 3

[32] Xuqian Ren, Wenjia Wang, Dingding Cai, Tuuli Tuominen, Juho Kannala, and Esa Rahtu. Mushroom: Multi-sensor hybrid room dataset for joint 3d reconstruction and novel view synthesis. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4508–4517, 2024. 2

[33] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 2

[34] Ludan Ruan, Yiyang Ma, Huan Yang, Huiguo He, Bei Liu, Jianlong Fu, Nicholas Jing Yuan, Qin Jin, and Baining Guo. Mm-diffusion: Learning multi-modal diffusion models for joint audio and video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10219–10228, 2023. 2

[35] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22500–22510, 2023. 2

[36] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2

[37] Matias Turkulainen, Xuqian Ren, Iaroslav Melekhov, Otto Seiskari, Esa Rahtu, and Juho Kannala. Dn-splatter: Depth and normal priors for gaussian splatting and meshing. *arXiv preprint arXiv:2403.17822*, 2024. 3

[38] Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Gelly. Fvd: A new metric for video generation. 2019. 8

[39] Zhouxia Wang, Ziyang Yuan, Xintao Wang, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. Motionctrl: A unified and flexible motion controller for video generation. *arXiv preprint arXiv:2312.03641*, 2023. 3

[40] Ning Xu, Linjie Yang, Yuchen Fan, Dingcheng Yue, Yuchen Liang, Jianchao Yang, and Thomas Huang. Youtube-vos: A large-scale video object segmentation benchmark. *arXiv preprint arXiv:1809.03327*, 2018. 4

[41] Shiyuan Yang, Liang Hou, Haibin Huang, Chongyang Ma, Pengfei Wan, Di Zhang, Xiaodong Chen, and Jing Liao. Direct-a-video: Customized video generation with user-directed camera movement and object motion. *arXiv preprint arXiv:2402.03162*, 2024. 3

[42] Jingfeng Yao, Xinggang Wang, Shusheng Yang, and Baoyuan Wang. Vitmatte: Boosting image matting with pretrained plain vision transformers. *Information Fusion*, 103: 102091, 2024. 3

[43] Lvmin Zhang and Maneesh Agrawala. Transparent image layer diffusion using latent transparency. *arXiv preprint arXiv:2402.17113*, 2024. 3, 6

[44] Shiwei Zhang, Jiayu Wang, Yingya Zhang, Kang Zhao, Hangjie Yuan, Zhiwu Qin, Xiang Wang, Deli Zhao, and Jingren Zhou. I2vgen-xl: High-quality image-to-video synthesis via cascaded diffusion models. *arXiv preprint arXiv:2311.04145*, 2023. 2

[45] Yabo Zhang, Yuxiang Wei, Dongsheng Jiang, Xiaopeng Zhang, Wangmeng Zuo, and Qi Tian. Controlvideo: Training-free controllable text-to-video generation. *arXiv preprint arXiv:2305.13077*, 2023. 3