Vivid-ZOO: Multi-View Video Generation with Diffusion Model

Bing Li* † **Cheng Zheng*** **Wenxuan Zhu*** **Jinjie Mai** **Biao Zhang**
Peter Wonka **Bernard Ghanem**

King Abdullah University of Science and Technology

https://hi-zhengcheng.github.io/vividzoo/

Abstract

While diffusion models have shown impressive performance in 2D image/video generation, diffusion-based Text-to-Multi-view-Video (T2MVid) generation remains underexplored. The new challenges posed by T2MVid generation lie in the lack of massive captioned multi-view videos and the complexity of modeling such multi-dimensional distribution. To this end, we propose a novel diffusion-based pipeline that generates high-quality multi-view videos centered around a dynamic 3D object from text. Specifically, we factor the T2MVid problem into viewpoint-space and time components. Such factorization allows us to combine and reuse layers of advanced pre-trained multi-view image and 2D video diffusion models to ensure multi-view consistency as well as temporal coherence for the generated multi-view videos, largely reducing the training cost. We further introduce alignment modules to align the latent spaces of layers from the pre-trained multi-view and the 2D video diffusion models, addressing the reused layers' incompatibility that arises from the domain gap between 2D and multi-view data. In support of this and future research, we further contribute a captioned multi-view video dataset. Experimental results demonstrate that our method generates high-quality multi-view videos, exhibiting vivid motions, temporal coherence, and multi-view consistency, given a variety of text prompts.

1 Introduction

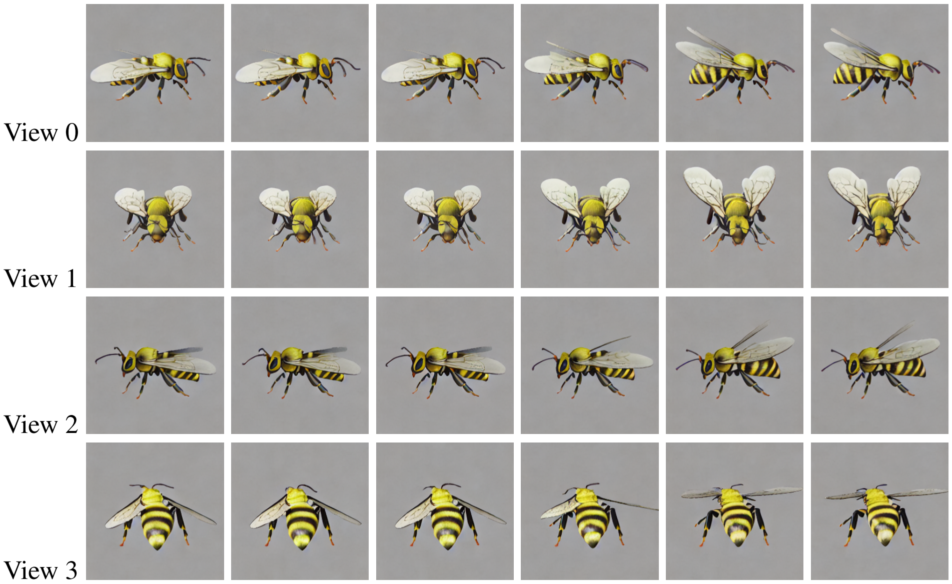
Multi-view videos capture a scene/object from multiple cameras with different poses simultaneously, which are critical for numerous downstream applications [5, 55, 62, 39, 40] such as AR/VR, 3D/4D modeling, media production, and interactive entertainment. More importantly, the availability of such data holds substantial promise for facilitating progress in research areas such as 4D reconstruction [44, 48], 4D generation [3, 49], and long video generation [9, 101] with 3D consistency. However, collecting multi-view videos often requires sophisticated setups [1] to synchronize and calibrate multiple cameras, resulting in a significant absence of datasets and generative techniques for multi-view videos.

In the meantime, diffusion models have shown great success in 2D image/video generation. For example, 2D video diffusion models [6, 23, 28, 81] generate high-quality 2D videos by extending image diffusion models [74, 83]. Differently, multi-view image diffusion models [80, 34, 53, 93] are proposed to generate multi-view images of 3D objects, which have demonstrated significant impact in 3D object generation [45], 3D reconstruction [66], and related fields. However, to the best of our knowledge, no other works have explored Text-to-Multi-view-Video (T2MVid) diffusion

*Equal contributions. † Corresponding author.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

https://doi.org/10.52202/079017-1987



Text prompt: a yellow and black striped wasp bee, 3d asset

Figure 1: The proposed Vivid-ZOO generates high-quality multi-view videos of a dynamic 3D object from text. Each row illustrates six frames drawn from a generated video for one viewpoint.

models. Motivated by recent 2D video and multi-view image diffusion models, we aim to propose a diffusion-based method that generates multi-view videos of dynamic objects from text (see Fig. 1).

Compared to 2D video generation, T2MVid generation poses two new challenges. First, modeling multi-view videos is complex due to their four-dimensional nature, which involves different viewpoints as well as the dimensions of time and space (2D). Consequently, it is nontrivial for diffusion models to model such intricate data from scratch without extensive captioned multi-view video datasets. Second, there are no publicly available large-scale datasets of captioned multi-view videos, but it has been shown that billions of text and 2D image pairs are essential for powerful image diffusion models [74, 76, 83]. For example, Stable Diffusion [76] is trained on the massive LAION-5B dataset [78]. Unlike downloading 2D images available on the Internet, collecting a large quantity of multi-view videos is labor-intensive and time-consuming. This challenge is further compounded when high-quality captioned videos are needed, hindering the extension of diffusion models to T2MVid generation.

In this paper, instead of the labor-intensive task of collecting a large amount of captioned multi-view video data, we focus on the problem of enabling diffusion models to generate multi-view videos from text using only a comparable small dataset of captioned multi-view videos. This problem has not been taken into account by existing diffusion-based methods (*e.g.*, [6][23]). However, studies have revealed that naively fine-tuning a large pre-trained model on limited data can result in overfitting [30, 75, 115]. Our intuition is that we can factor the multi-view video generation problem into viewpoint-space and time components. The viewpoint-space component ensures that the generated multi-view videos are geometrically consistent and aligned with the input text, and the temporal component ensures temporal coherence. With such factorization, a straightforward approach is to leverage large-scale multi-view image datasets (*e.g.*, [51] [67]) and 2D video datasets (*e.g.*, Web10M [4]) to pre-train the viewpoint-space component and temporal component, respectively. However, while this approach can largely reduce the reliance on extensive captioned multi-view videos, it remains costly in terms of training resources.

Instead, we explore a new question: *can we jointly combine and reuse the layers of pre-trained 2D video and multi-view image diffusion models to establish a T2MVid diffusion model*? The large-scale pre-trained multi-view image diffusion models (*e.g.*, MVdream [80]) have learned how to model multi-view images, and the 2D temporal layers of powerful pre-trained video diffusion models (*e.g.*, AnimateDiff [23]) learned rich motion knowledge. However, new challenges are posed. We observe that naively combining the layers from these two kinds of diffusion models leads to poor generation results. More specifically, the training data of multi-view image diffusion models are mainly rendered from synthetic 3D objects (*e.g.*, Objaverse [17, 16]), while 2D video diffusion models are mainly trained on real-world 2D videos, posing a large domain gap issue.

To bridge this gap, we propose a novel diffusion-based pipeline, namely, Vivid-ZOO, for T2MVid generation. The proposed pipeline effectively connects the pre-trained multi-view image diffusion model [80] and 2D temporal layers[2] of the pre-trained video model by introducing two kinds of layers, named 3D-2D alignment layers and 2D-3D alignment layers, respectively. The 3D-2D alignment layers are designed to align features to the latent space of the pre-trained 2D temporal layers, and the introduced 2D-3D alignment layers project the features back. Furthermore, we construct a comparable small dataset consisting of 14,271 captioned multi-view videos to facilitate this and future research line. Although our dataset is much smaller compared to the billion-scale 2D image dataset (LAION [78]) and the million-scale 2D video dataset (e.g., WebVid10M [4]), our pipeline allows us to effectively train a large-scale T2MVid diffusion model using such limited data. Extensive experimental results demonstrate that our method effectively generates high-quality multi-view videos given various text prompts.

We summarize our contributions as follows:

- We present a novel diffusion-based pipeline that generates high-quality multi-view videos from text prompts. This is the first study on T2MVid diffusion models.
- We show how to combine and reuse the layers of the pre-trained 2D video and multi-view image diffusion models for a T2MVid diffusion model. The introduced 3D-2D alignment and 2D-3D alignment are simple yet effective, enabling our method to utilize layers from the two diffusion models across different domains, ensuring both temporal coherence and multi-view consistency.
- We contribute a multi-view video dataset that provides multi-view videos, text descriptions, and corresponding camera poses, which helps to advance the field of T2MVid generation.

2 Related work

2D video diffusion model. Many previous approaches have explored autoregressive transformers (*e.g.*, [18, 29, 107]), physical models [104] or GANs (*e.g.*, [8, 56, 77, 41]) for video generation. Recently, more and more efforts have been devoted to diffusion-based video generation [21, 26, 65, 94, 99, 102, 110, 121, 37], inspired by the impressive results of image diffusion models [11, 12, 74, 76, 83, 115].

The amount of available captioned 2D video data is significantly less than the vast number of 2D image-text pairs available on the Internet. Most methods [7, 22, 23, 90, 91, 92] extend pre-trained 2D image diffusion models to video generation to address the challenge of limited training data. Some methods employ pre-trained 2D image diffusion models (*e.g.*, [76]) to generate 2D video from texts in a zero-shot manner [36][118] or using few-shot tuning strategies [103]. These methods avoid the requirement of large-scale training data. Differently, another research line is to augment pre-trained 2D image diffusion models with various temporal modules or trainable parameters, showing impressive temporal coherence performance. For example, Ho et al. [28] extend the standard image diffusion architecture by inserting a temporal attention block. Animatediff[23] and AYL [7] freeze 2D image diffusion model and solely train additional motion modules on large-scale datasets of captioned 2D videos such as WebVid10M [4]. In addition, image-to-2D-video generation methods [6, 71, 105, 117] are proposed based on diffusion models to generate a monocular video from an image. Methods [95] focus on controllable video generation through different conditions such as pose and depth. MotionCtrl [98] and Direct-a-video [109] can generate videos conditioned by the camera and object motion. CameraCtrl [24] can also control the trajectory of a moving camera for generated videos. However, these text-to-2D-video diffusion models are designed for monocular video generation, which does not explicitly consider the spatial 3D consistency of multi-view videos.

Multi-view image diffusion model. Recent works have extended 2D image diffusion models for multi-view image generation. Zero123 [51] and Zero123++ [79] propose to fine-tune an image-conditioned diffusion model so as to generate a novel view from a single image. Inspired by this,

[2] For clarification, we add “2D" when referring to the layers of the 2D video diffusion models, while we add “3D" when referring to the multi-view image diffusion model.

https://doi.org/10.52202/079017-1987

many novel view synthesis methods [19, 32, 52, 53, 59, 93, 97, 100, 108, 112, 120] are proposed based on image diffusion models. For example, IM3D [59] and Free3D [120] generate multiple novel views simultaneously to improve spatial 3D consistency among different views. Differently, a few methods [13, 33, 89] adapt pre-trained video diffusion models (*e.g.*, [6]) to generate multi-view images from a single image. MVDream [80] presents a text-to-multi-view-image diffusion model to generate four views of an object each time given a text, while SPAD [34] generates geometrically consistent images for more views. Richdreamer [67] trains a diffusion model to generate depth, normal, and albedo.

4D generation using diffusion models. Many approaches [47, 51, 64, 69, 79, 80, 85, 96, 2] have exploited pre-trained diffusion models to train 3D representations for 3D object generation via score distillation sampling [64]. Recently, a few methods [3, 49, 70, 82, 113] leverage pre-trained diffusion models to train 4D representations for dynamic object generation. For example, Ling *et al.*[49] represent a 4D object as Gaussian spatting [35], while Bahmani *et al.*[3] adopt a NeRF-based representation [60, 61, 86]. Then, pre-trained 2D image, 2D video, and multi-view image diffusion models are employed to jointly train the 4D representations. In addition, diffusion models are used to generate 4D objects from monocular videos [14, 31]. Diffusion4D [46] presents a diffusion model that generates an orbital video around 4D content, and 4Diffusion [114] presents a video-conditioned diffusion model that generates MV videos from a monocular video. Different from all these methods, our approach focuses on presenting a T2MVid diffusion model. LMM [116] generates 3D motion for given 3D human models. DragAPart [43] can generate part-level motion for articulated objects. Unlike our method, Kuang *et al.*[38] focuses on generating multiple videos of the same scene given multiple camera trajectories.

3 Multi-view video diffusion model

Problem definition. Our goal for T2MVid generation is to generate a set of multi-view videos centered around a dynamic object from a text prompt. Motivated by the success of diffusion models in 2D video/image generation, we aim to design a T2MVid diffusion model. However, T2MVid generation is challenging due to the complexity of modeling multi-view videos and the difficulty of collecting massive captioned multi-view videos for training.

We address the above challenges by exploring two questions. (1) Can we design a diffusion model that effectively learns T2MVid generation, yet only needs a comparable small dataset of multi-view video data? (2) Can we jointly leverage, combine, and reuse the layers of pre-trained 2D video and multi-view image diffusion models to establish a T2MVid diffusion model? Addressing these questions can reduce the reliance on large-scale training data and decrease training costs. However, this question remains unexplored for diffusion-based T2MVid.

Overview. We address the above questions by factoring the T2MVid generation problem over viewpoint-space and time. With the factorization, we propose a diffusion-based pipeline for T2MVid generation (see Fig 2), including the multi-view spatial modules and multi-view temporal modules. Sec 3.1 describes how we adapt a pre-trained multi-view image diffusion model as the multi-view spatial modules. Multi-view temporal modules effectively leverage temporal layers of the pre-trained 2D video diffusion model with the newly introduced 3D-2D alignment layers and 2D-3D alignment layers (Sec 3.2). Finally, we describe training objectives in Sec 3.3 and the dataset construction to support our pipeline for T2MVid generation in Sec 3.4.

3.1 Multi-view spatial module

Our multi-view spatial modules ensure that the generated multi-view videos are geometrically consistent and aligned with the input text. Recent multi-view image diffusion models [34, 80] generate high-quality multi-view images by fine-tuning Stable Diffusion and modifying its self-attention layers. We adopt the architecture of Stable Diffusion for our multi-view spatial modules. Furthermore, we leverage a pre-trained multi-view image diffusion model based on Stable Diffusion by reusing its pre-trained weights in our spatial modules, which avoids training from scratch and reduces the training cost. However, the self-attention layers of Stable Diffusion are not designed for multi-view videos. We adapt these layers for multi-view self-attention as below.

https://doi.org/10.52202/079017-1987

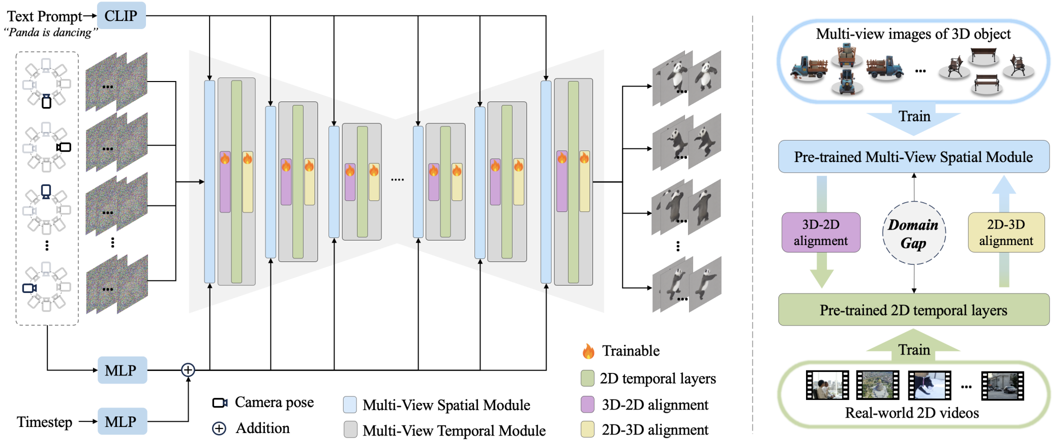


Figure 2: Overview of the proposed Vivid-ZOO. **Left**: Given a text prompt, our diffusion model generates multi-view videos. Instead of training from scratch, the multi-view spatial module reuses the pre-trained multi-view image diffusion model, and the multi-view temporal module leverages the 2D temporal layers of the pre-trained 2D video diffusion model to enforce temporal coherence. **Right**: Jointly reusing the pre-trained multi-view image diffusion model and temporal 2D layers poses new challenges due to the large gap between their training data (multi-view images of synthetic 3D objects versus real-world 2D videos). We introduce 3D-2D alignment and 2D-3D alignment to address the domain gap issue.

Multi-view self-attention. We inflate self-attention layers to capture geometric consistency among generated multi-view videos. Let $\mathbf{F} \in \mathbb{R}^{b \times K \times N \times d \times h \times w}$ denote the 6D feature tensor of multi-view videos in the diffusion model, where b, K, N, d and $h \times w$ are batch size, view number, frame number, feature channel and spatial dimension, respectively. Inspired by [34, 80], we reshape $\mathbf{F}$ into a shape of $(b \times N) \times d \times (K \times h \times w)$, leading to a batch of feature maps $\tilde{\mathbf{F}}^n$ of 2D images, where $(b \times N)$ is the batch size, $\tilde{\mathbf{F}}^n$ denotes a feature map representing all views at frame index n, and $(K \times h \times w)$ is the spatial size. We then feed the reshaped feature maps $\tilde{\mathbf{F}}^n$ into self-attention layers. Since $\tilde{\mathbf{F}}^n$ consists of all views at frame index n, the self-attention layers learn to capture geometrical consistency among different views. We also inflate other layers of stable diffusion (see Appendix E) so that we can reuse their pre-trained weight.

Camera pose embedding. Our diffusion model is controllable by camera poses, achieved by incorporating a camera pose sequence as input. These poses are embedded by MLP layers and then added to the timestep embedding, following MVdream [80]. Here, our multi-view spatial module reuses the pre-trained multi-view image diffusion model MVDream [80].

3.2 Multi-view temporal module

Besides spatial 3D consistency, it is crucial for T2MVid diffusion models to maintain the temporal coherence of generated multi-view videos simultaneously. Improper temporal constraints would break the synchronization among different views and introduce geometric inconsistency. Moreover, training a complex temporal module from scratch typically requires a large amount of training data.

Instead, we propose to leverage the 2D temporal layers of large pre-trained 2D video diffusion models (*e.g.*, [23]) to ensure temporal coherence for T2MVid generation. These 2D temporal layers have learned rich motion priors, as they have been trained on millions of 2D videos (*e.g.*, [4]). Here, we employ the 2D temporal layers of AnimateDiff [23] due to its impressive performance in generating temporal coherent 2D videos.

However, we observed that naively combining the pre-trained 2D temporal layers with the multi-view spatial module leads to poor results. The incompatibility is due to the fact that the pre-trained 2D temporal layers and the multi-view spatial modules are trained on data from different domains (*i.e.*, real 2D and synthetic multi-view data) that have a large domain gap. To address the domain gap issue, one approach is to fine-tune all 2D temporal layers of a pre-trained 2D video diffusion model on multi-view video data. However, such an approach not only needs to train many parameters but can also harm the learned motion knowledge [30] if a small training dataset is given. We present a multi-view temporal module (see Fig. 3) that reuses and freezes all 2D temporal layers to maintain the learned motion knowledge and introduce the 3D-2D alignment layer and the 2D-3D alignment layer.

3D-2D alignment. We introduce the 3D-2D alignment layers to effectively combine the pre-trained 2D temporal layers with the multi-view spatial module. Recently, a few methods [23, 6] add motion LoRA to 2D temporal attention for personalized/customized video generation tasks. However, our aim is different, *i.e.*, we expect to preserve the learned motion knowledge of 2D temporal layers, such that our multi-view temporal module can leverage the knowledge for ensuring temporal coherence.

Since motion prior knowledge is captured by the pre-trained 2D temporal attention layers, we insert the 3D-2D alignment layers before the 2D temporal attention layers. The 3D-2D alignment layers are learned to align the features into the latent space of the pre-trained 2D temporal layers. Furthermore, inspired by ControlNet [115] and [25], the 3D-2D alignment layers are inserted via residual connections and are zero-initialized, providing an identity mapping at the beginning of training. The process is described as follows:

$$\mathbf{F} = \alpha^{2D}(\mathbf{F}) + \alpha^{3D\rightharpoonup 2D}(\mathbf{F}) \tag{1}$$

where $\alpha^{3D\rightharpoonup 2D}$ is the 3D-2D alignment layer. α^{2D} is the 2D temporal layer followed by the 2D temporal attention layers and we refer to it as *2D in-layer* (see more details in Appendix). The 3D-2D alignment layer is plug-and-play and is simply implemented as an MLP.

Multi-view temporal coherence. We reuse and freeze the pre-trained 2D temporal layers in our multi-view temporal module to ensure the temporal coherence of each generated video. However, the 2D temporal layer is designed to handle 2D videos. We inflate the 2D temporal layer by reshaping the feature $\mathbf{F}$ to the 2D video dimension via the *rearrange* operation [73]. Then, 2D temporal layers $\gamma(\cdot)$ model temporal coherence across frames by calculating the attention of points at the same spatial location in $\mathbf{F}$ across frames for each video:

$$\mathbf{F} = \text{rearrange}(\mathbf{F},\ b\ K\ N\ h\ w\ d \rightarrow (b\ K\ h\ w)\ N\ d) \tag{2}$$

$$\mathbf{F} = \gamma(\mathbf{F}) \tag{3}$$

$$\mathbf{F} = \text{rearrange}(\mathbf{F}, (b\ K\ h\ w)\ N\ d \rightarrow\ b\ K\ N\ h\ w\ d) \tag{4}$$

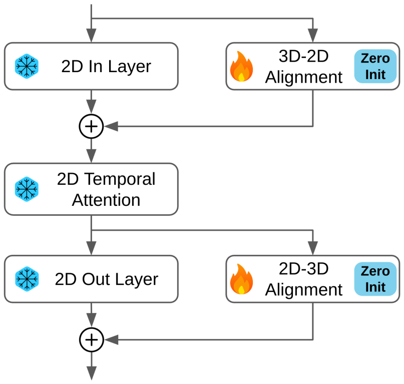


Figure 3: Our multi-view temporal module, where 3D-2D alignment layers are trained to align features to the latent space of the 2D temporal attention layers, and the 2D-3D alignment layers project them back.

2D-3D alignment. We add the 2D-3D alignment layers after 2D temporal attention layers to project the feature back to the feature space of the multi-view spatial modules.

$$\mathbf{F}^a = \beta^{2D}(\mathbf{F}) + \beta^{2D\rightharpoonup 3D}(\mathbf{F}) \tag{5}$$

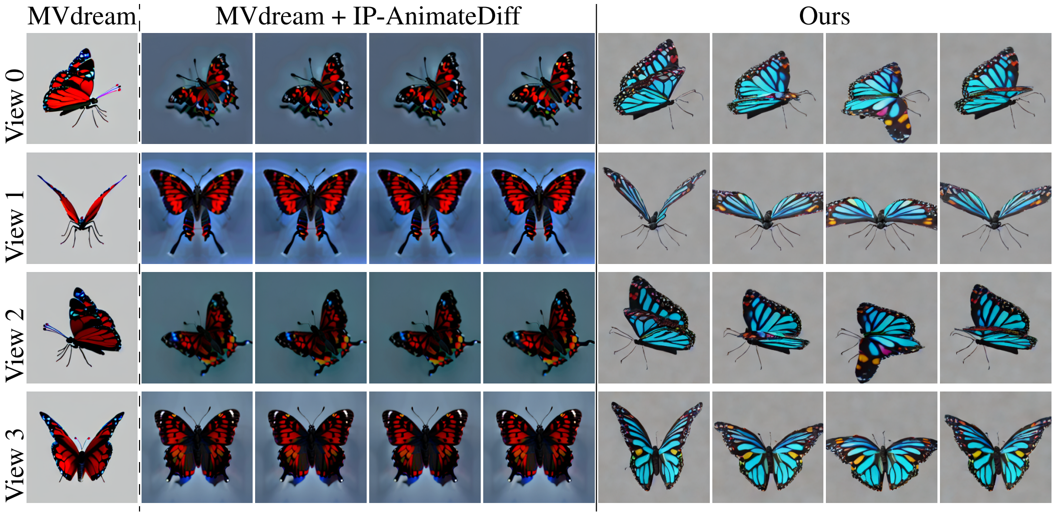
where $\beta^{3D\rightharpoonup 2D}$ is the 2D-3D alignment layer. β^{2D} is the 2D temporal layer following the 2D temporal attention layer. The 2D-3D alignment layers are implemented as an MLP.

3.3 Training objectives

We train our diffusion model to generate multi-view videos. Note that we freeze most layers/modules in the diffusion model and only train the 3D-2D and 2D-3D alignment layers during training, which largely reduces the training cost and reliance on large-scale data. Let $\mathcal{X}$ denote the training dataset, where a training sample $\{\mathbf{x}, y, \mathbf{c}\}$ consists of N multi-view videos $\mathbf{x} = \{x\}_1^N$, N corresponding camera poses $\mathbf{c}$, and a text prompt y. The training objective $\mathcal{L}$ on $\mathcal{X}$ is defined as follows:

$$\mathcal{L} = \mathbb{E}_{\mathbf{z}_t^v, y, \epsilon, t}\left[\left\|\epsilon - \epsilon_\theta(\mathbf{z}_t^v, t, \tau_\theta(y), \mathbf{c})\right\|^2\right] \tag{6}$$

where $\tau_\theta(\cdot)$ is a text encoder that encodes the text into text embedding, $\epsilon_\theta(\cdot)$ is the denoising network. $\mathbf{z}_0^v$ is the latent code of a multi-view video sequence and $\mathbf{z}_t^v$ is its noisy code with added noise ϵ.



Text prompt: Beautiful, intricate butterfly, 3d asset.

Figure 4: Comparison on T2MVid generation. Although MVDream generates spatially 3D consistent images among views (the 1st column), MVDream + IP-AnimateDiff breaks the spatial 3D consistency among its generated videos. Instead, our method generates high-quality multi-view videos with large motions while maintaining temporal coherence and spatial 3D consistency.

3.4 Multi-view video dataset

Different from 2D images that are available in vast numbers on the Internet, it is much more difficult and expensive to collect a large amount of multi-view videos centered around 3D objects and corresponding text captions. Recently, multi-view image datasets (*e.g.*, [51, 67]), rendered from synthetic 3D models, have shown a significant impact on various tasks such as novel view synthesis [51, 93], 3D generation (Gaussian Splatting [84], large reconstruction model [50]), multi-view image generation [80] and associated applications. Motivated by this, we resort to rendering multi-view videos from synthetic 4D models (animated 3D models).

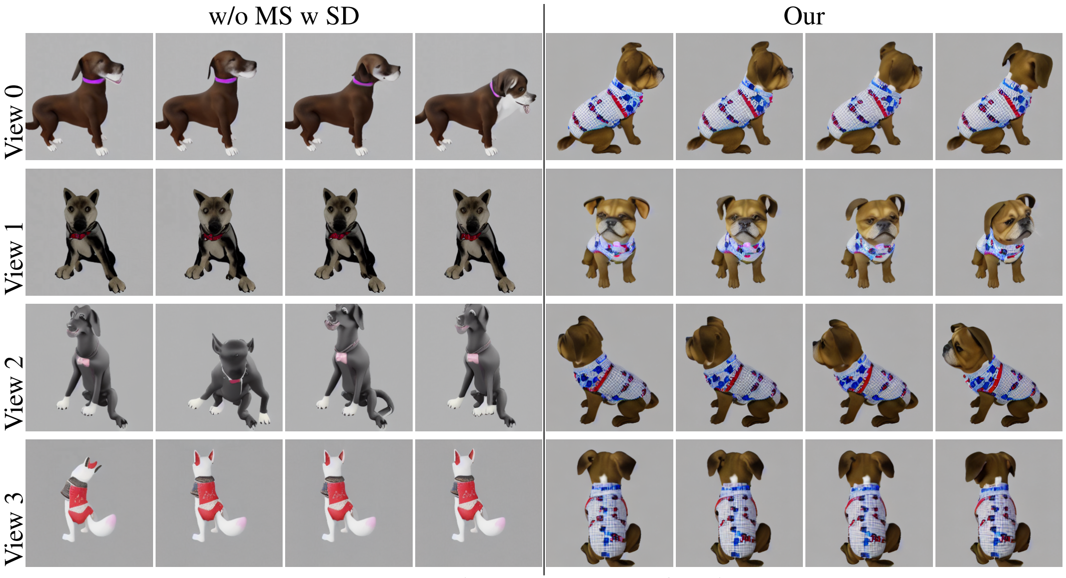
We construct a dataset named MV-VideoNet that provides 14,271 triples of a multi-view video sequence, its associated camera pose sequence, and a text description. In particular, we first select animated objects from Objaverse [17]. Objaverse is an open-source dataset that provides high-quality 3D objects and animated ones (*i.e.*, 4D object). We select 4D objects from the Objaverse dataset and discard those without motions or with imperceptible motions. Given each selected 4D object, we render 24-view videos from it, where the azimuth angles of camera poses are uniformly distributed. To improve the quality of our dataset, we manually filter multi-view videos with low-quality *e.g.*, distorted shapes or motions, very slow or rapid movement. For text descriptions, we adopt the captioning method Cap3D [57, 58] to caption a multi-view video sequence. Cap3D leverages BLIP2 [42] and GPT4 [63] to fuse information from multi-view images, generating text descriptions.

4 Experiments

Implementation details. We reuse the pre-trained MVDream V1.5 in our multi-view spatial module and reuse the pre-trained 2D temporal layers of AnimateDiff V2.0 in our multi-view temporal module. We train our model using AdamW [54] with a learning rate of 10^{-4}. During training, we process the training data by randomly sampling 4 views that are orthogonal to each other from a multi-view video sequence, reducing the spatial resolution of videos to 256×256, and sample video frames with a stride of 3. Following AnimateDiff, we use a linear beta schedule with β_{start} = 0.00085 and β_{end} = 0.012. (Please refer to the Appendix for more details).

Evaluation metrics. Quantitatively evaluating multi-view consistency and temporal coherence remains an open problem for T2MVid generation. We quantitatively evaluate text alignment via CLIP [68] and temporal coherence via Frechet Video Distance (FVD) [87]. Yet, Ge *et al.*[20] pointed out FVD leans more towards per-frame quality than temporal consistency. To compensate for FVD,

https://doi.org/10.52202/079017-1987



Text prompt: a dog wearing an outfit, 3d asset

Figure 5: Visual comparison of the contributions of our multi-view spatial module

we conduct a user study to evaluate the overall performance incorporating text alignment, temporal coherence, and multi-view consistency according to human preference (H. Pref.). CLIP and FVD scores in Tab. 1 are computed from 25 multi-view videos, where most input prompts used to generate these videos are separate from the training set, and only two prompts are from the training set. For ablation study, there are five methods and ten subjects for evaluating human preference, leading to 5×2×10 =100 questionnaires per input text prompt. To reduce the cost, we use input prompts to evaluate human preference in the ablation.

4.1 Qualitative and quantitative results

To the best of our knowledge, no studies have explored T2MVid diffusion models before. We establish a baseline method named **MVDream + IP-AnimateDiff** for comparison. **MVDream + IP-AnimateDiff** combines the pre-trained multi-view image diffusion model *MVDream* [80] and the 2D video diffusion model *AnimateDiff* [23], since MVDream generates high-quality multi-view images and AnimateDiff generates temporal coherent 2D videos. Following [119], we combine AnimateDiff with IP-adaptor [111] to enable AnimateDiff to take an image as input.

Given a text prompt, **MVDream + IP-AnimateDiff** generates multi-view videos in two stages, where MVDream generates multi-view images in the first stage, and IP-AnimateDiff animates each generated image from view into a 2D video in the second stage.

Fig. 4 and Tab. 1 show that **MVDream + IP-AnimateDiff** achieves slightly better CLIP values. However, our method outperforms **MVDream + IP-AnimateDiff** by a large margin in FVD and overall performance. **MVDream + IP-AnimateDiff** introduces the noticeable 3D inconsistency among different views. For example, both appearances and motions of the butterfly in the view 0 video are inconsistent with those of view 3. In contrast, our method not only achieves better performance in maintaining multi-view consistency, but also generates larger and more vivid motions for the butterfly, thanks to our pipeline and dataset. In addition, different from MVDream + IP-AnimateDiff employing two kinds of diffusion models and generating results in two stages, our method provides a unified diffusion model generating high-quality multi-view videos in only one stage. Please refer to the Appendix for more results.

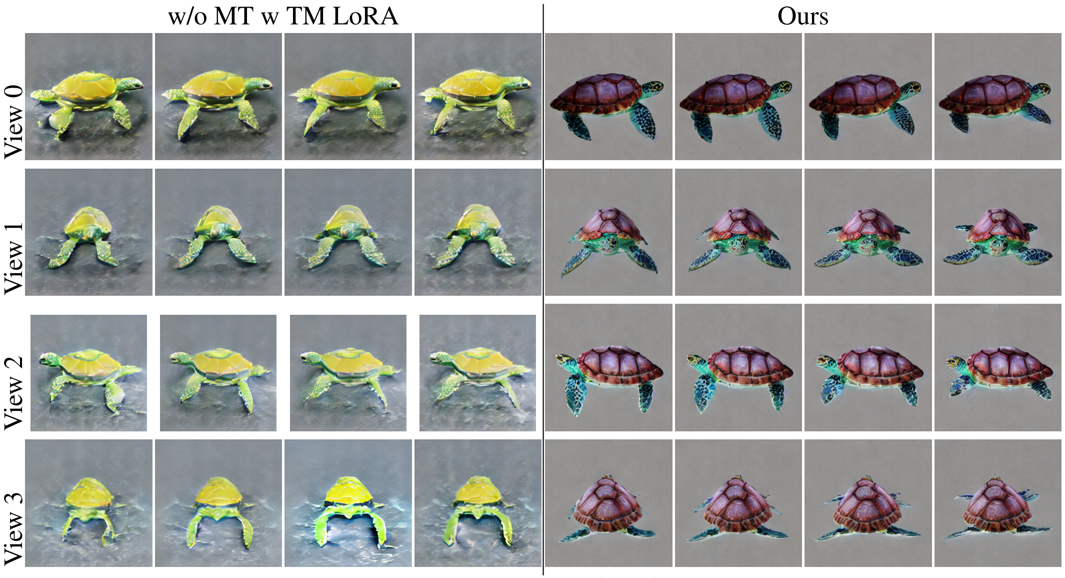
4.2 Ablation study and discussions

We conduct the ablation study to show the effectiveness of the design in our multi-view spatial and temporal modules, as well as the proposed 3D-2D and 2D-3D alignment.

https://doi.org/10.52202/079017-1987

Table 1: Multi-view video generation. Best in bold.

Method	FVD ↓	CLIP ↑	Overall ↑
MVDream + IP-AnimateDiff	2038.66 ± 44.36	**32.71** ± **0.67**	28%
Ours	**1634.28** ± **45.24**	32.24 ± 0.78	**72%**



Text prompt: a sea turtle, 3d asset.

Figure 6: Visual comparison of the contributions of our multi-view temporal module

Design of multi-view spatial module. We build a baseline named **w/o MS w SD** that employs original Stable Diffusion 1.5 [76] as our multi-view spatial module and reuses its pre-trained weights. We also insert the camera embedding into the Stable Diffusion to enable viewpoint control. That is, **w/o MS w SD** is to generate a single-view video (2D) conditioned on input text and camera poses. We train **w/o MS w SD** on our dataset, where single-view videos are used as training data.

Since single-view video generation is much simpler than multi-view video generation, **w/o MS w SD** achieves high performance in video quality. However, **w/o MS w SD** fails to maintain multi-view consistency among different views (see Fig. 5) and has degraded overall generation performance (see Tab. 2). For example, the motion and shapes of the dragon are significantly inconsistent among views. Instead, by simply adapting a pre-trained multi-view image diffusion model as our spatial module, our method effectively ensures multi-view consistency.

Design of multi-view temporal module. Recent methods apply LoRA [30] to the 2D temporal attention layers of a pre-trained 2D video diffusion model and fine-tune only LoRA for personalized and customized 2D video generation tasks [23, 6, 72]. Following these methods, we build a temporal module named **TM LoRA** by inflating 2D temporal layers of AnimateDiff to handle multi-view videos and adding LoRA to the 2D temporal attention layers. We replace our multi-view temporal module with **TM LoRA**, and denote it by **w/o MT w TM LoRA**. Fig. 6 and Tab. 2 shows **w/o MT w TM LoRA** generates low-quality results, despite being fine-tuned on our dataset. Instead, our multi-view temporal module inserts 3D-2D alignment and 2D-3D alignment layers before and after the 2D temporal attention layers, enabling the multi-view temporal module to be compatible with the multi-view spatial module.

Table 2: The ablation study results. The overall performance is assessed by a user study using paired comparison [3, 15].

Method	Overall ↑
w/o MS w SD	44.88%
w/o MT w TM LoRA	11.25%
w/o 3D-2D alignment	53.50%
w/o 2D-3D alignment	54.50%
Ours	**80.25%**

Effect of 3D-2D alignment. We remove the proposed 3D-2D alignment from our model and train the model on our dataset with the same settings. Tab. 2 shows **w/o 3D-2D alignment** degrades

https://doi.org/10.52202/079017-1987

our temporal coherence and video quality performance. Instead, by projecting the feature to the latent space of the pre-trained 2D attention layers, our 3D-2D alignment layer effectively enables the 2D attention layers to align temporally correlated content, ensuring the video quality and temporal coherence.

Effect of 2D-3D alignment. As shown in Tab. 2, “w/o 2D-3D temporal alignment" achieves lower performance with the same training settings due to the removal of 2D-3D temporal alignment. The results indicate that only 3D-2D alignment is insufficient in jointly leveraging the pre-trained 2D temporal layers [23] and the multi-view image diffusion model [80] in our diffusion model. Instead, our 2D-3D alignment projects the features processed by the pre-trained 2D temporal layers back to the latent space of the multi-view image diffusion model, leading to high-quality results.

Training cost. MVDream is trained on 32 Nvidia Tesla A100 GPUs, which takes 3 days, and AnimateDiff takes around 5 days on 8 A100 GPUs. By combining and reusing the layers of MVDream and AnimateDiff, our method only needs to train the proposed 3D-2D alignment and 2D-3D layers, reducing the training cost to around 2 days with 8 A100 GPUs.

5 Conclusions

In this paper, we propose a novel diffusion-based pipeline named Vivid-ZOO that generates high-quality multi-view videos centered around a dynamic 3D object from text. The presented multi-view spatial module ensures the multi-view consistency of generated multi-view videos, while the multi-view temporal module effectively enforces temporal coherence. By introducing the proposed 3D-2D temporal alignment and 2D-3D temporal alignment layers, our pipeline effectively leverages the layers of the pre-trained multi-view image and 2D video diffusion models, reducing the training cost and accelerating the training of our diffusion model. We also construct a dataset of captioned multi-view videos, which facilitates future research in this emerging area.

Acknowledgments and Disclosure of Funding

The research reported in this publication was supported by funding from King Abdullah University of Science and Technology (KAUST) - Center of Excellence for Generative AI, under award number 5940 and the SDAIA-KAUST Center of Excellence in Data Science and Artificial Intelligence. We thank Zhangjie Wu and Mengmeng Xu for their valuable constructive suggestions and help.

https://doi.org/10.52202/079017-1987

References

[1] Liang An, Jilong Ren, Tao Yu, Tang Hai, Yichang Jia, and Yebin Liu. Three-dimensional surface motion capture of multiple freely moving pigs using mammal. *Nature Communications*, 14(1):7727, 2023.

[2] Sherwin Bahmani, Xian Liu, Yifan Wang, Ivan Skorokhodov, Victor Rong, Ziwei Liu, Xihui Liu, Jeong Joon Park, Sergey Tulyakov, Gordon Wetzstein, et al. Tc4d: Trajectory-conditioned text-to-4d generation. *arXiv preprint arXiv:2403.17920*, 2024.

[3] Sherwin Bahmani, Ivan Skorokhodov, Victor Rong, Gordon Wetzstein, Leonidas Guibas, Peter Wonka, Sergey Tulyakov, Jeong Joon Park, Andrea Tagliasacchi, and David B. Lindell. 4d-fy: Text-to-4d generation using hybrid score distillation sampling. *arXiv preprint arXiv:2311.17984*, 2023.

[4] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *IEEE International Conference on Computer Vision*, 2021.

[5] Aayush Bansal, Minh Vo, Yaser Sheikh, Deva Ramanan, and Srinivasa Narasimhan. 4d visualization of dynamic events from unconstrained multi-view videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[6] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. Stable video diffusion: Scaling latent video diffusion models to large datasets, 2023.

[7] 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.

[8] Tim Brooks, Janne Hellsten, Miika Aittala, Ting chun Wang, Timo Aila, Jaakko Lehtinen, Ming-Yu Liu, Alexei A Efros, and Tero Karras. Generating long videos of dynamic scenes. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[9] 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.

[10] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

[11] Junsong Chen, Yue Wu, Simian Luo, Enze Xie, Sayak Paul, Ping Luo, Hang Zhao, and Zhenguo Li. Pixart$-delta$: Fast and controllable image generation with latent consistency models, 2024.

[12] Junsong Chen, Jincheng YU, Chongjian GE, Lewei Yao, Enze Xie, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *The Twelfth International Conference on Learning Representations*, 2024.

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

[14] Wen-Hsuan Chu, Lei Ke, and Katerina Fragkiadaki. Dreamscene4d: Dynamic multi-object scene generation from monocular videos. *arXiv preprint arXiv:2405.02280*, 2024.

[15] Herbert Aron David. *The method of paired comparisons*, volume 12. 1963.

[16] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, Eli VanderBilt, Aniruddha Kembhavi, Carl Vondrick, Georgia Gkioxari, Kiana Ehsani, Ludwig Schmidt, and Ali Farhadi. Objaverse-xl: A universe of 10m+ 3d objects. *arXiv preprint arXiv:2307.05663*, 2023.

[17] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. *arXiv preprint arXiv:2212.08051*, 2022.

[18] Ming Ding, Wendi Zheng, Wenyi Hong, and Jie Tang. Cogview2: Faster and better text-to-image generation via hierarchical transformers. *Advances in Neural Information Processing Systems*, 35:16890–16902, 2022.

[19] Ruiqi Gao*, Aleksander Holynski*, Philipp Henzler, Arthur Brussee, Ricardo Martin-Brualla, Pratul P. Srinivasan, Jonathan T. Barron, and Ben Poole*. Cat3d: Create anything in 3d with multi-view diffusion models. *arXiv*, 2024.

[20] Songwei Ge, Aniruddha Mahapatra, Gaurav Parmar, Jun-Yan Zhu, and Jia-Bin Huang. On the content bias in fréchet video distance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

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

[22] Rohit Girdhar, Mannat Singh, Andrew Brown, Quentin Duval, Samaneh Azadi, Sai Saketh Rambhatla, Akbar Shah, Xi Yin, Devi Parikh, and Ishan Misra. Emu video: Factorizing text-to-video generation by explicit image conditioning, 2023.

[23] Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *International Conference on Learning Representations*, 2024.

[24] Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. Cameractrl: Enabling camera control for text-to-video generation. *arXiv preprint arXiv:2404.02101*, 2024.

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[26] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022.

[27] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851, 2020.

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

[29] 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.

[30] 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. In *International Conference on Learning Representations*, 2022.

[31] Yanqin Jiang, Li Zhang, Jin Gao, Weiming Hu, and Yao Yao. Consistent4d: Consistent 360° dynamic object generation from monocular video. In *The Twelfth International Conference on Learning Representations*, 2024.

https://doi.org/10.52202/079017-1987

[32] Yifan Jiang, Hao Tang, Jen-Hao Rick Chang, Liangchen Song, Zhangyang Wang, and Liangliang Cao. Efficient-3dim: Learning a generalizable single-image novel-view synthesizer in one day. In *The Twelfth International Conference on Learning Representations*, 2024.

[33] Philip Torr Junlin Han, Filippos Kokkinos. Vfusion3d: Learning scalable 3d generative models from video diffusion models. *arXiv preprint arXiv:2403.12034*, 2024.

[34] Yash Kant, Ziyi Wu, Michael Vasilkovsky, Guocheng Qian, Jian Ren, Riza Alp Guler, Bernard Ghanem, Sergey Tulyakov, Igor Gilitschenski, and Aliaksandr Siarohin. Spad : Spatially aware multiview diffusers, 2024.

[35] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):1–14, 2023.

[36] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15954–15964, 2023.

[37] Jihwan Kim, Junoh Kang, Jinyoung Choi, and Bohyung Han. Fifo-diffusion: Generating infinite videos from text without training, 2024.

[38] Zhengfei Kuang, Shengqu Cai, Hao He, Yinghao Xu, Hongsheng Li, Leonidas Guibas, and Gordon. Wetzstein. Collaborative video diffusion: Consistent multi-video generation with camera control. In *arXiv*, 2024.

[39] Bing Li, Chia-Wen Lin, Boxin Shi, Tiejun Huang, Wen Gao, and C-C Jay Kuo. Depth-aware stereo video retargeting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6517–6525, 2018.

[40] Bing Li, Chia-Wen Lin, Cheng Zheng, Shan Liu, Junsong Yuan, Bernard Ghanem, and C-C Jay Kuo. High quality disparity remapping with two-stage warping. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2269–2278, 2021.

[41] Bing Li, Yuanlue Zhu, Yitong Wang, Chia-Wen Lin, Bernard Ghanem, and Linlin Shen. Anigan: Style-guided generative adversarial networks for unsupervised anime face generation. *IEEE Transactions on Multimedia*, 24:4077–4091, 2021.

[42] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pages 19730–19742, 2023.

[43] Ruining Li, Chuanxia Zheng, Christian Rupprecht, and Andrea Vedaldi. Dragapart: Learning a part-level motion prior for articulated objects. *arXiv preprint arXiv:2403.15382*, 2024.

[44] Tianye Li, Mira Slavcheva, Michael Zollhöfer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard Newcombe, and Zhaoyang Lv. Neural 3d video synthesis from multi-view video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5521–5531, June 2022.

[45] Zhiqi Li, Yiming Chen, Lingzhe Zhao, and Peidong Liu. Controllable text-to-3d generation via surface-aligned gaussian splatting. 2024.

[46] Hanwen Liang, Yuyang Yin, Dejia Xu, Hanxue Liang, Zhangyang Wang, Konstantinos N Plataniotis, Yao Zhao, and Yunchao Wei. Diffusion4d: Fast spatial-temporal consistent 4d generation via video diffusion models. *arXiv preprint arXiv:2405.16645*, 2024.

[47] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 300–309, 2022.

https://doi.org/10.52202/079017-1987

[48] Haotong Lin, Sida Peng, Zhen Xu, Tao Xie, Xingyi He, Hujun Bao, and Xiaowei Zhou. High-fidelity and real-time novel view synthesis for dynamic scenes. In *SIGGRAPH Asia Conference Proceedings*, 2023.

[49] Huan Ling, Seung Wook Kim, Antonio Torralba, Sanja Fidler, and Karsten Kreis. Align your gaussians: Text-to-4d with dynamic 3d gaussians and composed diffusion models. *arXiv preprint arXiv:2312.13763*, 2023.

[50] Minghua Liu, Chao Xu, Haian Jin, Linghao Chen, Mukund Varma T, Zexiang Xu, and Hao Su. One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *Advances in Neural Information Processing Systems*, 36, 2024.

[51] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9298–9309, 2023.

[52] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023.

[53] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. *arXiv preprint arXiv:2310.15008*, 2023.

[54] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

[55] Jian-Guang Lou, Hua Cai, and Jiang Li. A real-time interactive multi-view video system. In *Proceedings of the 13th annual ACM international conference on Multimedia*, pages 161–170, 2005.

[56] Pauline Luc, Aidan Clark, Sander Dieleman, Diego de Las Casas, Yotam Doron, Albin Cassirer, and Karen Simonyan. Transformation-based adversarial video prediction on large-scale data. *arXiv preprint arXiv:2003.04035*, 2020.

[57] Tiange Luo, Justin Johnson, and Honglak Lee. View selection for 3d captioning via diffusion ranking. *arXiv preprint arXiv:2404.07984*, 2024.

[58] Tiange Luo, Chris Rockwell, Honglak Lee, and Justin Johnson. Scalable 3d captioning with pretrained models. *arXiv preprint arXiv:2306.07279*, 2023.

[59] Luke Melas-Kyriazi, Iro Laina, Christian Rupprecht, Natalia Neverova, Andrea Vedaldi, Oran Gafni, and Filippos Kokkinos. Im-3d: Iterative multiview diffusion and reconstruction for high-quality 3d generation, 2024.

[60] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[61] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (TOG)*, 41:1 – 15, 2022.

[62] Armin Mustafa, Hansung Kim, Jean-Yves Guillemaut, and Adrian Hilton. General dynamic scene reconstruction from multiple view video. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[63] OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell,

https://doi.org/10.52202/079017-1987

Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2024.

[64] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *The Eleventh International Conference on Learning Representations*, 2023.

[65] Chenyang Qi, Xiaodong Cun, Yong Zhang, Chenyang Lei, Xintao Wang, Ying Shan, and Qifeng Chen. Fatezero: Fusing attentions for zero-shot text-based video editing. *arXiv:2303.09535*, 2023.

[66] Guocheng Qian, Jinjie Mai, Abdullah Hamdi, Jian Ren, Aliaksandr Siarohin, Bing Li, Hsin-Ying Lee, Ivan Skorokhodov, Peter Wonka, Sergey Tulyakov, and Bernard Ghanem. Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

[67] Lingteng Qiu, Guanying Chen, Xiaodong Gu, Qi zuo, Mutian Xu, Yushuang Wu, Weihao Yuan, Zilong Dong, Liefeng Bo, and Xiaoguang Han. Richdreamer: A generalizable normal-depth diffusion model for detail richness in text-to-3d. *arXiv preprint arXiv:2311.16918*, 2023.

[68] 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.

[69] Amit Raj, Srinivas Kaza, Ben Poole, Michael Niemeyer, Nataniel Ruiz, Ben Mildenhall, Shiran Zada, Kfir Aberman, Michael Rubinstein, Jonathan T. Barron, Yuanzhen Li, and Varun Jampani. Dreambooth3d: Subject-driven text-to-3d generation. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2349–2359, 2023.

[70] Jiawei Ren, Liang Pan, Jiaxiang Tang, Chi Zhang, Ang Cao, Gang Zeng, and Ziwei Liu. Dreamgaussian4d: Generative 4d gaussian splatting. *arXiv preprint arXiv:2312.17142*, 2023.

[71] Weiming Ren, Harry Yang, Ge Zhang, Cong Wei, Xinrun Du, Stephen Huang, and Wenhu Chen. Consisti2v: Enhancing visual consistency for image-to-video generation. *arXiv preprint arXiv:2402.04324*, 2024.

[72] Yixuan Ren, Yang Zhou, Jimei Yang, Jing Shi, Difan Liu, Feng Liu, Mingi Kwon, and Abhinav Shrivastava. Customize-a-video: One-shot motion customization of text-to-video diffusion models, 2024.

[73] Alex Rogozhnikov. Einops: Clear and reliable tensor manipulations with einstein-like notation. In *International Conference on Learning Representations*, 2022.

[74] 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 (CVPR)*, pages 10684–10695, 2022.

[75] 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.

[76] RunwayML. Stable diffusion v1.5 model card. `https://huggingface.co/runwayml/stable-diffusion-v1-5`, 2022.

[77] Masaki Saito, Shunta Saito, Masanori Koyama, and Sosuke Kobayashi. Train sparsely, generate densely: Memory-efficient unsupervised training of high-resolution temporal gan. *International Journal of Computer Vision*, 128(10):2586–2606, 2020.

[78] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022.

[79] Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model, 2023.

[80] Yichun Shi, Peng Wang, Jianglong Ye, Long Mai, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv:2308.16512*, 2023.

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

[82] Uriel Singer, Shelly Sheynin, Adam Polyak, Oron Ashual, Iurii Makarov, Filippos Kokkinos, Naman Goyal, Andrea Vedaldi, Devi Parikh, Justin Johnson, et al. Text-to-4d dynamic scene generation. In *International Conference on Machine Learning*, pages 31915–31929. PMLR, 2023.

[83] Stability. Stable diffusion v2 model card. `https://huggingface.co/stabilityai/stable-diffusion-2-depth`, 2022. stable-diffusion2-depth.

[84] Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. *arXiv preprint arXiv:2402.05054*, 2024.

https://doi.org/10.52202/079017-1987

[85] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *ArXiv*, abs/2309.16653, 2023.

[86] Haithem Turki, Jason Y. Zhang, Francesco Ferroni, and Deva Ramanan. Suds: Scalable urban dynamic scenes. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12375–12385, 2023.

[87] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Gelly. FVD: A new metric for video generation, 2019.

[88] Aaron van den Oord, Oriol Vinyals, and koray kavukcuoglu. Neural discrete representation learning. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, 2017.

[89] Vikram Voleti, Chun-Han Yao, Mark Boss, Adam Letts, David Pankratz, Dmitry Tochilkin, Christian Laforte, Robin Rombach, and Varun Jampani. Sv3d: Novel multi-view synthesis and 3d generation from a single image using latent video diffusion, 2024.

[90] Fu-Yun Wang, Wenshuo Chen, Guanglu Song, Han-Jia Ye, Yu Liu, and Hongsheng Li. Gen-l-video: Multi-text to long video generation via temporal co-denoising. *arXiv preprint arXiv:2305.18264*, 2023.

[91] Fu-Yun Wang, Zhaoyang Huang, Xiaoyu Shi, Weikang Bian, Guanglu Song, Yu Liu, and Hongsheng Li. Animatelcm: Accelerating the animation of personalized diffusion models and adapters with decoupled consistency learning. *arXiv preprint arXiv:2402.00769*, 2024.

[92] Jiuniu Wang, Hangjie Yuan, Dayou Chen, Yingya Zhang, Xiang Wang, and Shiwei Zhang. Modelscope text-to-video technical report. *arXiv preprint arXiv:2308.06571*, 2023.

[93] Peng Wang and Yichun Shi. Imagedream: Image-prompt multi-view diffusion for 3d generation. *arXiv preprint arXiv:2312.02201*, 2023.

[94] Qian Wang, Weiqi Li, Chong Mou, Xinhua Cheng, and Jian Zhang. 360dvd: Controllable panorama video generation with 360-degree video diffusion model. *arXiv preprint arXiv:2401.06578*, 2024.

[95] Xiang Wang, Hangjie Yuan, Shiwei Zhang, Dayou Chen, Jiuniu Wang, Yingya Zhang, Yujun Shen, Deli Zhao, and Jingren Zhou. Videocomposer: Compositional video synthesis with motion controllability, 2023.

[96] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *Advances in Neural Information Processing Systems*, 36, 2024.

[97] Zhengyi Wang, Yikai Wang, Yifei Chen, Chendong Xiang, Shuo Chen, Dajiang Yu, Chongxuan Li, Hang Su, and Jun Zhu. Crm: Single image to 3d textured mesh with convolutional reconstruction model. *arXiv preprint arXiv:2403.05034*, 2024.

[98] 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.

[99] Wenming Weng, Ruoyu Feng, Yanhui Wang, Qi Dai, Chunyu Wang, Dacheng Yin, Zhiyuan Zhao, Kai Qiu, Jianmin Bao, Yuhui Yuan, Chong Luo, Yueyi Zhang, and Zhiwei Xiong. Art•v: Auto-regressive text-to-video generation with diffusion models. *arXiv preprint arXiv:2311.18834*, 2023.

[100] Sangmin Woo, Byeongjun Park, Hyojun Go, Jin-Young Kim, and Changick Kim. Harmonyview: Harmonizing consistency and diversity in one-image-to-3d, 2023.

[101] Chao-Yuan Wu and Philipp Krahenbuhl. Towards long-form video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1884–1894, 2021.

[102] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7623–7633, 2023.

[103] Ruiqi Wu, Liangyu Chen, Tong Yang, Chunle Guo, Chongyi Li, and Xiangyu Zhang. Lamp: Learn a motion pattern for few-shot-based video generation. *arXiv preprint arXiv:2310.10769*, 2023.

[104] Wenpeng Xiao, Wentao Liu, Yitong Wang, Bernard Ghanem, and Bing Li. Automatic animation of hair blowing in still portrait photos. In *ICCV*, 2023.

[105] Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Wangbo Yu, Hanyuan Liu, Xintao Wang, Tien-Tsin Wong, and Ying Shan. Dynamicrafter: Animating open-domain images with video diffusion priors. *arXiv preprint arXiv:2310.12190*, 2023.

[106] Zhen Xing, Qi Dai, Han Hu, Zuxuan Wu, and Yu-Gang Jiang. Simda: Simple diffusion adapter for efficient video generation, 2023.

[107] Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*, 2021.

[108] Jiayu Yang, Ziang Cheng, Yunfei Duan, Pan Ji, and Hongdong Li. Consistnet: Enforcing 3d consistency for multi-view images diffusion. *arXiv*, 2023.

[109] 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.

[110] Shuai Yang, Yifan Zhou, Ziwei Liu, , and Chen Change Loy. Rerender a video: Zero-shot text-guided video-to-video translation. In *ACM SIGGRAPH Asia Conference Proceedings*, 2023.

[111] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023.

[112] Jianglong Ye, Peng Wang, Kejie Li, Yichun Shi, and Heng Wang. Consistent-1-to-3: Consistent image to 3d view synthesis via geometry-aware diffusion models, 2023.

[113] Yuyang Yin, Dejia Xu, Zhangyang Wang, Yao Zhao, and Yunchao Wei. 4dgen: Grounded 4d content generation with spatial-temporal consistency. *arXiv preprint arXiv:2312.17225*, 2023.

[114] Haiyu Zhang, Xinyuan Chen, Yaohui Wang, Xihui Liu, Yunhong Wang, and Yu Qiao. 4diffusion: Multi-view video diffusion model for 4d generation. *arXiv preprint arXiv:2405.20674*, 2024.

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

[116] Mingyuan Zhang, Daisheng Jin, Chenyang Gu, Fangzhou Hong, Zhongang Cai, Jingfang Huang, Chongzhi Zhang, Xinying Guo, Lei Yang, Ying He, et al. Large motion model for unified multi-modal motion generation. *arXiv preprint arXiv:2404.01284*, 2024.

[117] 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.

[118] 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.

[119] Yiming Zhang, Zhening Xing, Yanhong Zeng, Youqing Fang, and Kai Chen. Pia: Your personalized image animator via plug-and-play modules in text-to-image models. *arXiv preprint arXiv:2312.13964*, 2023.

[120] Chuanxia Zheng and Andrea Vedaldi. Free3d: Consistent novel view synthesis without 3d representation. *arXiv*, 2023.

[121] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models. *arXiv preprint arXiv:2211.11018*, 2022.

Vivid-ZOO: Multi-View Video Generation with Diffusion Model

— Supplementary Material —

In this appendix, we provide additional content to complement the main manuscript:

- Appendix A: Discussions about limitations of the current method and possible future improvements.
- Appendix B: Preliminaries about diffusion and latent diffusion models.
- Appendix C: Evaluation metrics for our multi-view video generation.
- Appendix D: Details about the multi-view captioned video dataset we construct.
- Appendix E: More details of our proposed model, including the spatial and temporal module.
- Appendix F: Additional qualitative visualized results.
- Appendix G: Societal impact, ethic concerns, dataset copyrights, and our safeguard policies.

A Limitations and future works

While our method takes a step forward in T2MVid generation, our method can be improved in a few aspects.

A.1 Qualitative quality

For example, the visual quality of generated videos is not as high as that of multi-view image diffusion models due to the complexity of modeling multi-view videos. The spatial module of our method can be replaced with more advanced multi-view image diffusion models [34], to improve the performance of multi-view consistency. A large dataset of multi-view videos can be constructed, which further improves our method.

A.2 Lighting

For lighting and rendering, we followed the settings of [80] to ensure fair comparisons. Since they used point light sources, our learned model also generates multi-view videos under the assumption of point light sources, which may result in different exposures across the viewpoints in the videos, as shown in Fig. III. Future work could explore generating videos that simulate more complex ambient lighting settings and even achieve controllable lighting for different viewpoints in the generation process.

A.3 Topic of generation

Currently, the proposed Vivid-ZOO mainly supports generating multi-view videos for dynamic creatures with natural motions. Though we can also generate some categories like humans (*astronaut and horse*, Fig. II) and common objects (*waving flag*, Fig. IV), we believe our research can further inspire the community to develop more T2MVid techniques for the generation of more diverse and complex topics, similar to image diffusion model counterparts. For example, more specialized models that generate man-made artifacts like *moving cars* or *articulated furniture*, more powerful models that generate *multiple dynamic objects with complex motions*, and more diverse models like Vivid-Scene that generate dynamic scenes like *a stormy sea, an erupting volcano*.

B Background

Diffusion model. Diffusion models [27] learn to model a data distribution by iteratively recovering original data from noisy one, which comprises forward and backward phases. Given a clean sample x_0 from the data distribution p_{data}, the forward process gradually adds Gaussian noise to the sample, generating random latent variables x_t at each time step $t \in [0, T]$:

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t\mathbf{I}) \quad (7)$$

https://doi.org/10.52202/079017-1987

where β_t is a hyperparameter that determines the noise schedule. With a large time step, x_T is assumed to be perturbed into a standard Gaussian noise. Given x_T, the denoising network is trained to gradually remove the noise and recover the original data:

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{8}$$

where θ denote parameters of the denoising network, μ and Σ are mean and variance, respectively.

Latent diffusion model (LDM). By embedding images into low-dimensional latent codes, latent diffusion models [74] (LDMs) perform the diffusion process in the latent space of latent codes, significantly reducing the computational cost. Typically, LDMs employ a pre-trained autoencoder (*e.g.*, VQ-VAE [88]), which consists of an encoder and decoder, where the encoder transforms images into the latent space and the decoder maps denoised latent codes back to the pixel space.

C Details about evaluation metrics

- *Multi-view Text alignment*: Text-to-2D-video diffusion models (*e.g.*, [23, 36, 102]) adopt CLIP score [68] to measure the alignment between an input text and a corresponding generated 2D video. To evaluate the alignment between the input text and generated multi-view videos, we first measure the CLIP score for each view and then average the CLIP scores for all views.
- *Video quality*: We adopt Frechet Video Distance (FVD) to measure the quality of generated multi-view videos. FVD is the standard metric adopted by many 2D video generation [95, 106] and animation methods [6, 105]. FVD measures the quality of generated videos by measuring the data distribution between generated and training videos, where I3D networks pre-trained on the Kinetics dataset [10] are employed to extract features.
- *Human preference.* We conduct a user study to measure the overall quality of generated multi-view video text alignment, temporal coherence and multi-view consistency.

  We adopt paired comparison. We invite 10 subjects to participate in the user study. For each subject, we display two multi-view video sequences generated by different generation methods as well as the corresponding input text prompt, where the two results are arranged in an up-and-down order, and a resulting multi-view video sequence is displayed in a single row. We told each subject that the task is to generate four orthogonal views of a dynamic 3D object. Then the subject was asked to choose a multi-view video sequence whose overall quality is better in text alignment, temporal coherence and multi-view consistency. For comparison with MVdream + IP-AnimateDiff, we use 10 input prompts to generate multi-view videos for the user study. For the ablation study, there are five methods in total, leading to more combinations of paired comparisons. We hence use 5 input prompts in the use study for the ablation study.

D More details about our multi-view video dataset

2D video diffusion models [6] have pointed out that data curation is essential to improve the generation performance of diffusion models. The training of 2D Video diffusion models can be degraded if the training dataset contains many static 2D videos [6]. Hence, we developed an animated 3D object selection tool that automatically discards 4D objects that are static or close to static. In particular, given a 4D object, we first render a video from a single viewpoint. For efficiency, instead of using advanced optical flow algorithms, we calculate the pixel difference between different frames in each video, in order to identify whether 4D objects are static.

With the selected 4D objects, we employ Cycles [3] as the rendering engine to render multi-view videos. Given a 4D object, we render 24-view videos with the resolution of 512 × 512. We first uniformly distribute the camera poses around the normalized 4D object and then add subtle disturbances. The radius of a camera to a 4D object is in the range of [2.2 2.6], and a camera's height is in [0.8 1.2]. The background of a multi-view video sequence is randomly filled in gray color. The frame number of multi-view video sequences is diverse, depending on its 4D objects.

To further improve the quality of our dataset, we manually discard low-quality data from our dataset. We found that many multi-view videos contain distorted shapes or motions. We remove these low-quality data to avoid their negative effect on the training generation models. On the other hand, we

[3] https://www.cycles-renderer.org/

https://doi.org/10.52202/079017-1987

also remove multi-view videos that contain large translation motions. Due to the large translation motions, objects disappear in a few frames, which leads to these frames having only a background. In addition, many 4D objects are textureless in Objaverse [17]. We keep 10% of these textureless objects. For caption generation, we adopt a caption method *i.e.*, Cap3D which is designed to caption multi-view images of a 3D object. The Cap3D is used to describe a multi-view frame sequence sampled from a multi-view video sequence.

E More implementation details

Training settings. Table II provides detailed information on the hyperparameter settings and hardware configuration used for model training. During training, four orthogonal views are randomly chosen, leading to four-view videos. For a video from a viewpoint, the starting frame is randomly selected, and then we extract one frame every 3 frames. The frame size is 256 × 256 and the frame number is set to 16 (see Table I).

Table I: Training dataset settings

Name	Parameter value
view number	4
sample size	256 × 256
sample stride	3
frame number	16

Table II: Training settings

Name	Parameter value
noise scheduler type	DDIMScheduler
noise scheduler timesteps number	1000
noise scheduler start beta	0.00085
noise scheduler end beta	0.012
noise scheduler beta schedule	linear
noise scheduler steps offset	1
noise scheduler clip sample	false
optimizer	AdamW
learning rate	0.0001
train step number	100000
batch size	16
CPU memory size in total	320G
GPU type	NVIDIA A100
GPU number	8

Inference settings. Table III lists the hyperparameters and hardware configurations in the inference stage. The resolution and number of frames in the generated video are the same as those in the training settings.

Table III: Inference settings

Name	Parameter value
sample step number	50
CFG weight	7.5
CPU memory	30G
GPU type	NVIDIA A100
GPU number	1

https://doi.org/10.52202/079017-1987

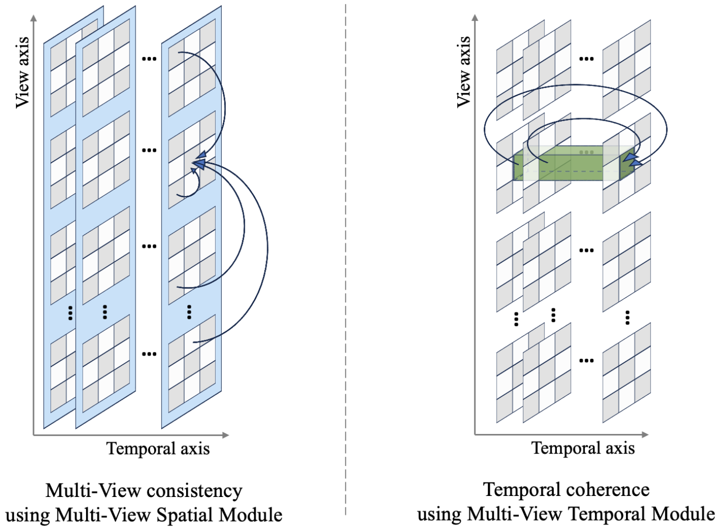


Figure I: The multi-view spatial module of our method ensures multi-view consistency of generated multi-view videos via capturing correlations across different views. The multi-view temporal module enforces temporal coherence via capturing temporal correlations among frames in a video of a viewpoint.

E.1 More details about multi-view spatial module

Our multi-view spatial module adapts Stable Diffusion to handle multi-view videos, and reuses the pre-trained weight of MVDream[80]. In this main paper, we have elaborated how to adapt Stable Diffusion's self-attention layers to handle multi-view videos' 6D feature tensors. With the adaption, the self-atention layers model multi-view consistency among views, as shown in Fig. I. For other layers, we first reshape the features of multi-view videos using the rearrange operation: rearrange($\mathbf{F}$, $b\ K\ N\ h\ w\ d \rightarrow (b\ N)K\ h\ w\ d$). When the features are fed to the multi-view temporal module, we transform the dimensions of the output feature $\mathbf{F}'$ back with rearrange operation: $(\mathbf{F}',\ (b\ N)K\ h\ w\ d \rightarrow b\ K\ N\ h\ w\ d)$.

E.2 More details of multi-view temporal module

Fig. I shows how our multi-view temporal module leverages 2D temporal layers to caption temporal correlations among frames in each view video.

Both a **3D-2D Alignment** layer and **2D-3D Alignment** layer are implemented using a Linear MLP layer. We experimented with 2-layer/3-layer setups, but there was no improvement in performance. Therefore, we use a simple single layer for implementation. In this paper, we reuse AnimateDiff in the multi-view temporal module, where the 2D-in-layer refers to the “project_in" layer and 2D-out-layer refers to “_out" layer in the AnimateDiff.

F Additional results

F.1 Additional text to multi-view video examples

Some additional experimental results are presented from Fig. II to Fig. VIII. We strongly recommend the readers to watch the corresponding videos on our anonymous website to get a better feel for the movement of the objects in the picture.

F.2 More ablation study results

Fig. IX and Fig. X show the contributions of the 3D-2D and 2D-3D alignment layers respectively.

https://doi.org/10.52202/079017-1987

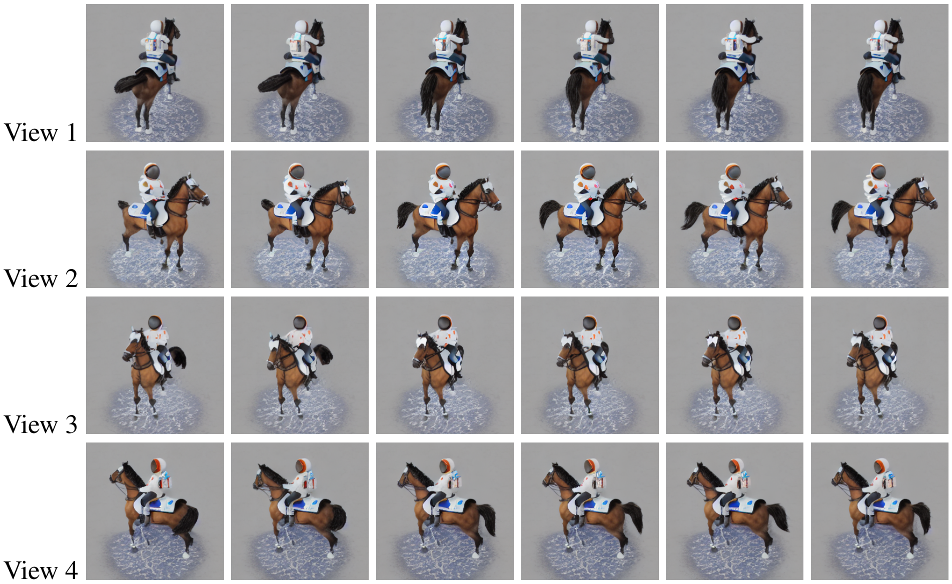


Figure II: Text prompt: *an astronaut riding a horse, 3d asset*

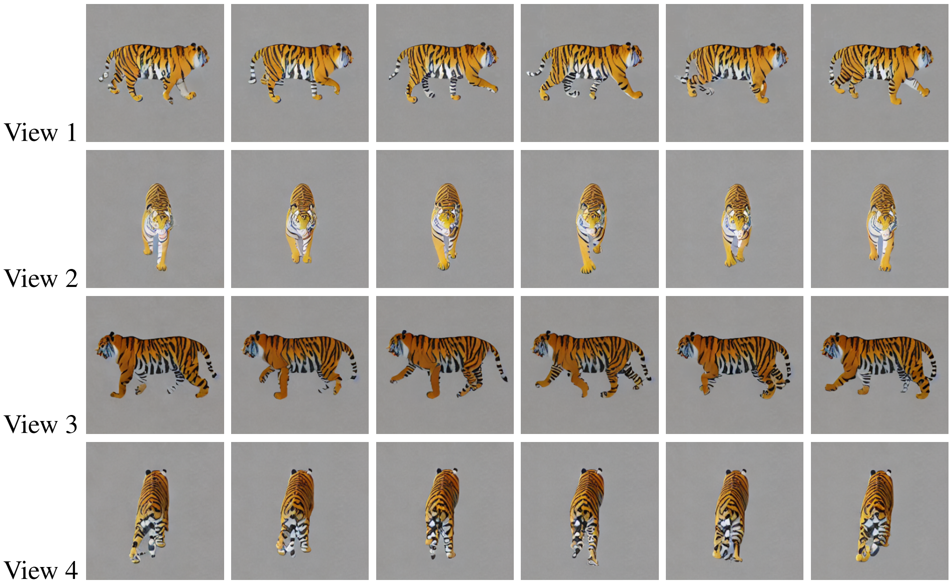


Figure III: Text prompt: *A full-bodied tiger walking, 3d asset*

G Societal impact and ethic concerns

G.1 Positive societal impact

Our method is able to generate vivid multi-view videos for dynamic creatures. Therefore, our method can be directly applied to enhance creativity and entertainment, e.g., creating AR/VR and game assets. Due to the availability of multi-view videos, our method can also be used for art creation and interactive educational content, which could benefit many people, like artists, designers, educators, and film and television creators. Researchers in fields such as biology, ecology, and zoology can benefit from this technology by creating accurate multi-view visualizations of dynamic creatures, which can aid in research, analysis, and presentations.

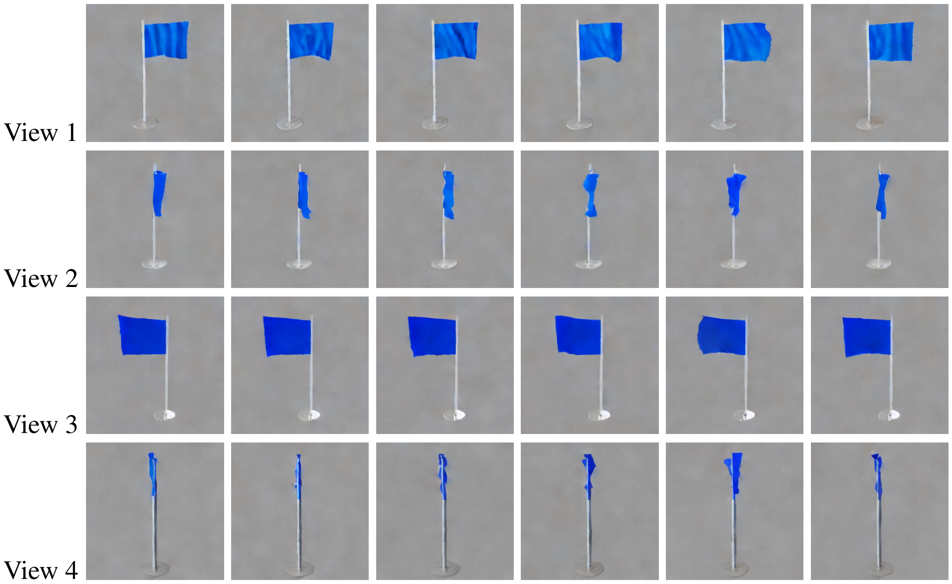


Figure IV: Text prompt: *a blue flag attached to a flagpole, with a smooth curve, 3d asset*

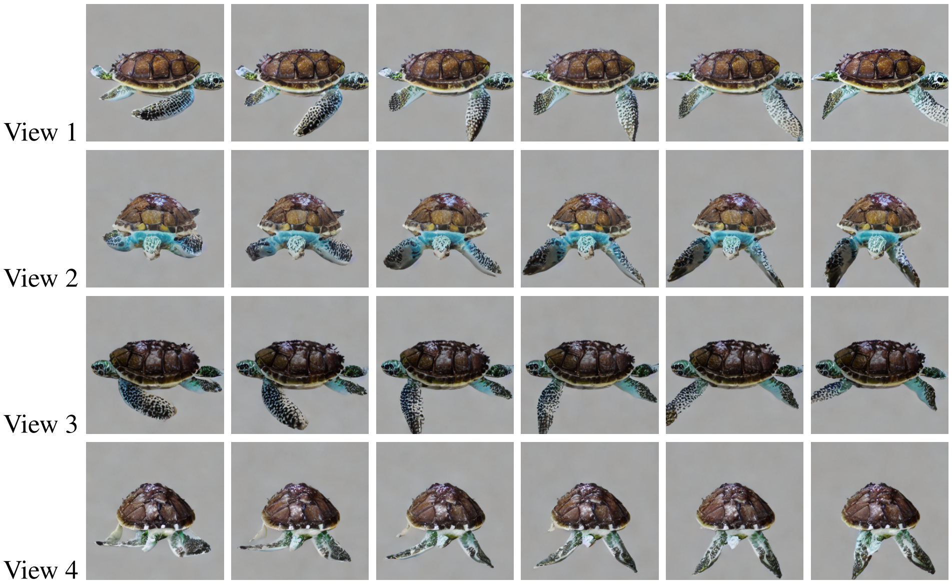


Figure V: Text prompt: *a spiked sea turtle, 3d asset*

G.2 Negative societal impact

Our Text-to-Multi-view-Video diffusion method is based on one existing text-to-multi-view image model and one text-to-video model. Therefore, its internal representation may inherit some bias from these two base models. Our multi-view video generation could be exploited to create highly realistic deepfakes. These fake videos can be used for malicious purposes such as spreading disinformation, manipulating public opinion, or creating fake profiles for fraudulent activities. If users input provocative prompts or maliciously fine-tune the model parameters, our model could potentially generate harmful videos, such as those containing vulgarity, gore, or violence. However, since our model is fine-tuned on the dynamic creature dataset, we believe the risk of such content is significantly lower compared to previous open-domain generative models like MV-Dream [80] and SVD [6]. Additionally, we will implement gated access and usage guidelines for our model and continuously monitor community usage and feedback to prevent such harmful content as much as possible.

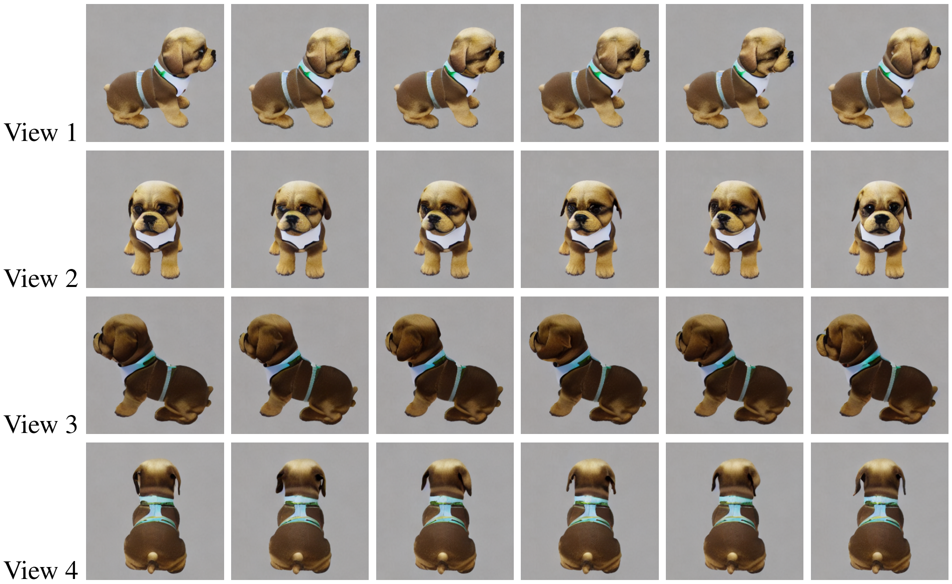


Figure VI: Text prompt: *a dog wearing a outfit, 3d asset*

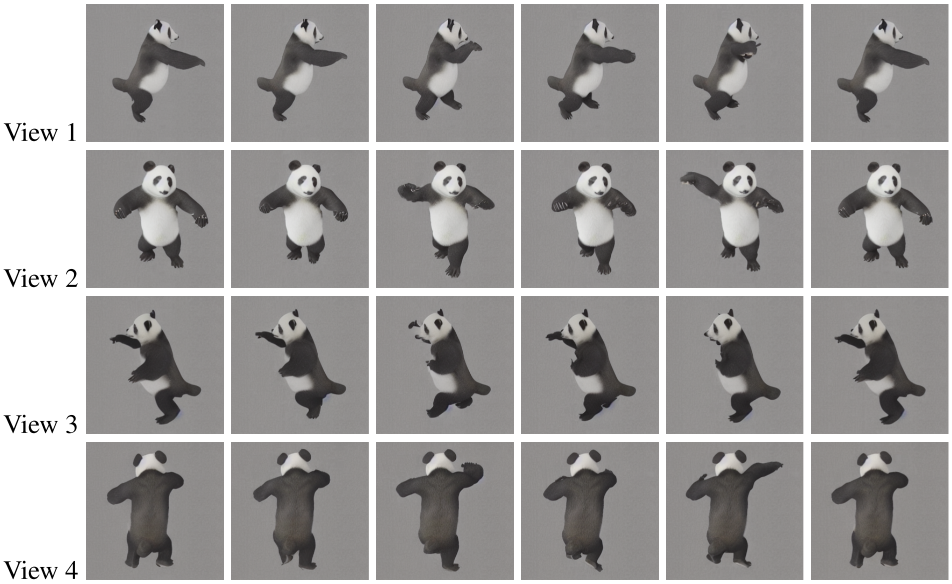


Figure VII: Text prompt: *a panda is dancing*

G.3 Copyright

Our dynamic dataset is directly sourced from the already open-sourced and published Objaverse [17] dataset and is used solely for scientific research purposes. Therefore, it does not infringe on the legal rights and copyrights of the original 3D/4D model creators and collectors.

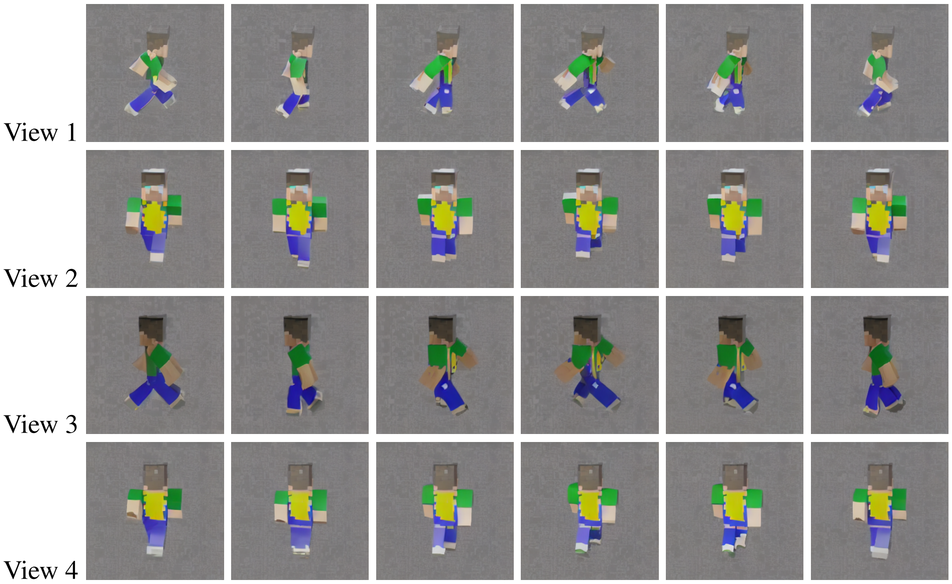
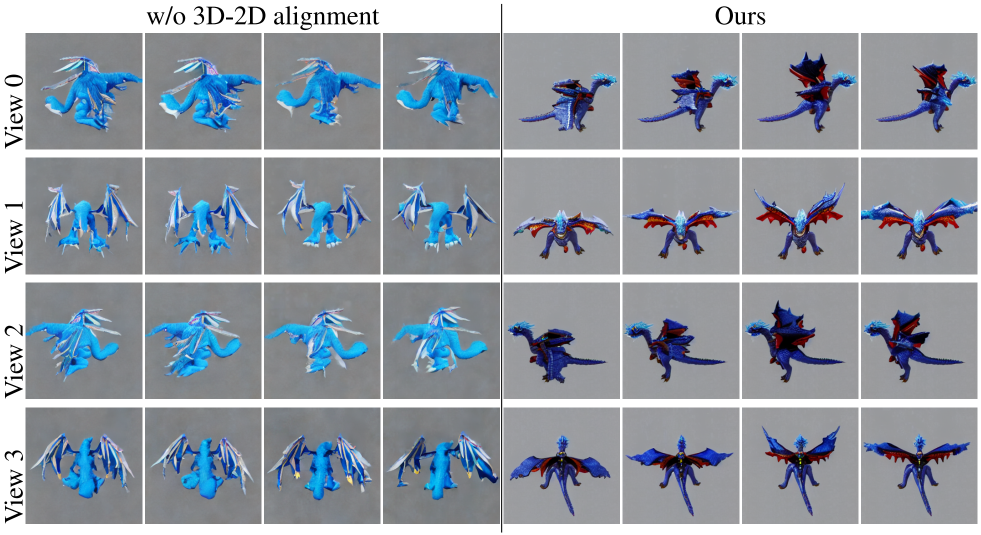


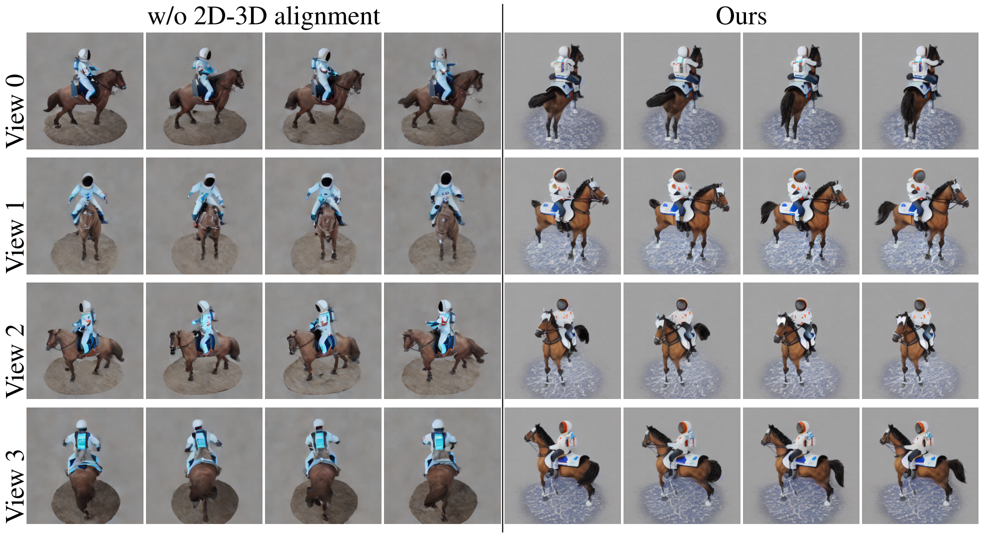
Figure VIII: Text prompt: *a pixelated Minecraft character walking, 3d asset*



Text prompt: a blue-winged dragon, also depicted as a flying monster, 3d asset

Figure IX: Visual comparison of the contributions of our 3D-2D alignment layers

https://doi.org/10.52202/079017-1987



Text prompt: an astronaut riding a horse, 3d asset

Figure X: Visual comparison of the contributions of our 2D-3D alignment layers

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The main claims accurately reflect the paper's contributions.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We describe the limitations of our work in Limitations (Sec. A).

   Guidelines:

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

3. **Theory Assumptions and Proofs**

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

   Answer: [NA]

https://doi.org/10.52202/079017-1987

Justification: The paper does not include theoretical results.

Guidelines:

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

4. **Experimental Result Reproducibility**

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

Answer: [Yes]

Justification: The paper contains the overall structure of the model in Sec. 3, and Sec. E of the appendix contains detailed model parameters.

Guidelines:

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

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

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

https://doi.org/10.52202/079017-1987

Answer: [No]

Justification: We will release the code and pretrained models upon acceptance.

Guidelines:

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

6. **Experimental Setting/Details**

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

   Answer: [Yes]

   Justification: We specify these details in Sec. E. We will make all the code and pretrained models publicly available upon acceptance.

   Guidelines:

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

7. **Experiment Statistical Significance**

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

   Answer: [Yes]

   Justification: We report error bars in Tab. 1. Yet. Our method is for the multi-view video generation task, and there are no standard qualitative metrics available.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

https://doi.org/10.52202/079017-1987

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

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

   Answer: [Yes]

   Justification: We provide this information in Sec. 4.2 and Sec. E.

   Guidelines:

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

9. **Code Of Ethics**

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

   Answer: [Yes]

   Justification: The research conducted in the paper conforms with the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader Impacts**

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

    Answer: [Yes]

    Justification: We have discussed in Sec. G.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

https://doi.org/10.52202/079017-1987

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [Yes]

    Justification: We have discussed the safeguards in Sec. G.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: All assets used in the paper are cited and credited. The license and terms of use are mentioned and respected.

    Guidelines:

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

https://doi.org/10.52202/079017-1987

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

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

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

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

14. **Crowdsourcing and Research with Human Subjects**

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

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

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

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

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

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

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