This WACV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

FB-4D: Spatial-Temporal Coherent Dynamic 3D Content Generation with Feature Banks

Jinwei Li[*1,2], Huan-ang Gao[*1,2], Wenyi Li[1], Haohan Chi[1,2], Chenyu Liu[1,2], Chenxi Du[2], Yiqian Liu[1,2] Mingju Gao[1], Guiyu Zhang[1], Zongzheng Zhang[1], Li Yi[3], Yao Yao[4], Jingwei Zhao[5], Hongyang Li[6], Yikai Wang[7], Hao Zhao[†1]

[1]AIR, Tsinghua University [2]DCST, Tsinghua University [3]IIIS, Tsinghua University
[4]Nanjing University [5]Xiaomi Corporation [6]The University of Hong Kong
[7]School of Artificial Intelligence, Beijing Normal University

Project Page: https://FB-4D.c7w.tech/

Abstract

With the rapid advancements in diffusion models and 3D generation techniques, dynamic 3D content generation has become a crucial research area. However, achieving high-fidelity 4D (dynamic 3D) generation with strong spatial-temporal consistency remains a challenging task. Inspired by recent findings that pretrained diffusion features capture rich correspondences, we propose FB-4D, a novel 4D generation framework that integrates a Feature Bank mechanism to enhance both spatial and temporal consistency in generated frames. In FB-4D, we store features extracted from previous frames and fuse them into the process of generating subsequent frames, ensuring consistent characteristics across both time and multiple views. To ensure a compact representation, the Feature Bank is updated by a proposed dynamic merging mechanism. Leveraging this Feature Bank, we demonstrate for the first time that generating additional reference sequences through multiple autoregressive iterations can continuously improve generation performance. Experimental results show that FB-4D significantly outperforms existing methods in terms of rendering quality, spatial-temporal consistency, and robustness. It surpasses all multi-view generation tuning-free approaches by a large margin and achieves performance on par with training-based methods. Our code and data will be publicly available to support future research.

1. Introduction

Dynamic 3D content generation, commonly known as 4D generation, involves not only creating the 3D geometry and appearance of objects but also capturing their motion over time within 3D space. Recently, interest in 4D content generation has surged [1, 10, 23, 24, 30, 53, 55, 57, 66, 74, 80, 84, 86, 91], driven by applications in autonomous driving, gaming, digital avatars, and immersive media.

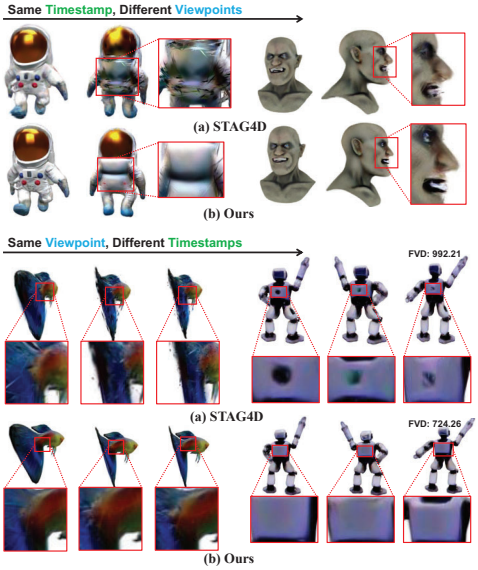


Figure 1. Compared to STAG4D [86], our method enhances spatial consistency (top row, better texture alignment across different viewpoints) and temporal consistency (bottom row, with fewer floaters and smoother motion transitions) by leveraging implicit correspondences in diffusion features. This translates to a superior FVD score than STAG4D (724.26 v.s. 992.21).

The primary challenge in dynamic 3D generation lies in jointly modeling spatial-temporal correspondences. To tackle this, borrowing from static 3D [29, 39, 41, 50], current approaches typically employ a two-stage process. The first stage involves generating multiple anchor views of sequences, while the second stage uses these anchor views to train a 4D representation with score distillation [40]. For the first stage, there are two primary approaches. **(I) Training-**

based methods, such as SV4D [74] and 4Diffusion [88], solve this task in a data-driven manner by modeling the joint latent space of views and frames, and conducting view and frame attention interleaved. However, the length of videos and the number of views that can be processed with these methods are constrained by GPU memory limitations, such as the need for 80GB GPUs, which are not accessible to content creators using typical workstations. **(II) Training-free methods** disentangles the problem by relying on priors from pre-trained video and multi-view generative models [52]. A notable example is STAG4D [86], which takes a single-view video as input, generates multi-view sequences conditioned on the first frame, and then produces 4D content via Gaussian Splatting. However, as shown in Fig. 1(a), generated results of these approaches often suffer from inconsistencies in both the spatial dimension (e.g., textures misaligned with frontal views) and the temporal dimension (e.g., floaters disrupting motion continuity), as they model object motion and novel view synthesis separately.

No Joint Modeling. Re-use Past Features for the Present. Recent studies have shown that diffusion features [42, 59], captured during denoising, contain rich correspondences across images. Building on this insight, we propose FB-4D, a novel 4D generation framework that *employs a spatial-temporal feature bank* to address the correspondence learning challenge by leveraging pretrained representations from diffusion-based multi-view generation models. To ensure the feature bank functions effectively, simply storing intermediate features from past frames or using a sliding window approach yields suboptimal results. Instead, we propose a dynamic mechanism for managing the feature bank's contents by continuously merging redundant information from new and stored features. This approach enables the preservation of the most representative features while maintaining a consistent bank size over time.

The more you see in 2D, the more you perceive in 3D. Enhancing 3D generation by introducing additional 2D anchors has proven effective [16]. A straightforward approach to this is through autoregressive generation, which produces additional reference images from a generated view to optimize the neural field. However, for 4D generation, this method alone is problematic, as inconsistencies introduced at each generation step can accumulate across iterations. With the integration of our proposed feature bank mechanism, however, we demonstrate for the first time that *generating additional reference sequences through multiple autoregressive iterations* can reliably enhance downstream performance. And this approach can be further strengthened by progressively selecting reference views.

FB-4D requires *neither training nor fine-tuning* of any multi-view or video generation model, yet it effectively addresses the challenge of correspondence matching across multiple frames and views, as illustrated in Fig. 1(b). Through extensive qualitative and quantitative experiments, we show that FB-4D achieves state-of-the-art results on the well-established dynamic 3D content generation benchmark, Consistent4D [23], outperforming all training-free methods by a substantial margin and matching the performance of the training-based SV4D [74]. Furthermore, we conduct in-depth experiments to explore how, why, and where the proposed feature bank mechanism is most effective. In summary, our contributions are as follows:

- We propose FB-4D, a novel framework that leverages the feature bank mechanism to enable training-free, efficient correspondence modeling across multiple frames and views for dynamic 3D content generation tasks.
- We conduct extensive experiments and analyses to provide in-depth insights into the feature bank mechanism, detailing how, why, and where it operates effectively.
- Our method achieves state-of-the-art results on the 4D generation benchmark, significantly outperforming all training-free methods and matching the performance of training-based approaches like SV4D [74].

2. Related Works

3D Generation and Video Generation. The domain of static 3D generation has drawed significant attention due to the advancements in 3D representation learning methods [13, 26, 36, 43, 73, 75, 79, 85]. Despite that various research endeavors aim to enhance score distillation loss [9, 14, 31, 34, 49, 52, 56, 58, 67, 70, 82, 92] and facilitate generation in a feed-forward manner [20, 22, 60, 65, 69, 93], our focus is on approaches that generate dense multi-view images with sufficient 3D consistency, subsequently reconstructing 3D content from these images [25, 28, 37–41, 50, 51, 62, 64, 81]. Our research adheres to this paradigm, but extends it by generating consistent multi-view videos (instead of images) and subsequently reconstructing the 4D object. For video generation, thanks to the powerful generative modeling ability of diffusion [5, 8, 12, 27, 33, 44, 48, 76, 87, 89], video diffusion models have shown exceptional performance with consistent geometry and realistic motions [3, 5, 6, 15, 17–19, 54, 61, 68]. Their robust generalization stemming from training on extensive image and video datasets prompts us to leverage them for text-to-video or image-to-video generation, creating the initial reference sequence.

4D Generation. The generation of 4D content, or dynamic 3D content, is enabled by the advancements in 4D representation learning [5, 63]. Given the challenges of applying score distillation sampling across multiple frames to directly distill motion knowledge from video diffusion models [74], existing text-to-4D [2, 11, 35, 55] and image-to-4D methods [90] often exhibit suboptimal appearance quality. Consequently, many approaches have shifted towards video-to-4D generation [23, 45, 46, 82]. These meth-

ods either models joint latent of view and time [74, 88] or utilize contemporary multiview diffusion models [39, 40, 50] to compute the SDS loss [23, 46, 72] on generated per-frame multi-view images [79, 86] as a supervisory signal [23, 86].

3. Preliminaries and Our Motivation

Overview of STAG4D. As discussed in the introduction, we abandon jointly modeling the latents of frames and views, opting instead to use past features for the generation of the current frame. To understand this approach, we first delve into STAG4D, a framework designed for high-fidelity 4D generation by integrating pre-trained diffusion models with dynamic 3D Gaussian splatting. The framework operates in two stages. Given **Single-view video frames** are used as input, **stage 1** employs a multi-view diffusion model, such as Zero123++ [50], to generate multi-view frame sequences from the reference frame sequence.

For convenience, we define the input viewpoint as $v = 0$ and the output viewpoints as $v = 1 : 6$. Let the input frame sequence contain T frames, denoted as $F^{[v=0,t=1:T]}$. After passing through the multi-view diffusion model, the output frame sequence can be represented as $F^{[v=1:6,t=1:T]}$. The entire multi-view diffusion process can be formulated as:

$$F^{[v=1:6,t=1:T]} = \text{Generator}\left(F^{[v=0,t=1:T]}\right), \quad (1)$$

where the Generator denotes the multi-view diffusion process. **Stage 2** utilizes score distillation sampling to refine and optimize the 4D Gaussians [86], given the reference frame sequence and multi-view generated sequences as input. For each rendering viewpoint, the closest viewpoint from the multi-view generated sequences is determined based on proximity, denoted as i, where i ranges from 1 to 6. The multi-view score distillation loss is formulated as:

$$L_{\text{MVSDS}} = \lambda_1 L^i_{\text{SDS}} + \lambda_2 L^0_{\text{SDS}}, \quad (2)$$

where λ_1 and λ_2 are weighting factors.

In Stage 1, STAG4D builds upon Zero123++, which incorporates the reference image into U-Net self-attention blocks through a reference branch and a noised latent branch. Given an input $F^{[v=0,t=i]}$, the self-attention keys, queries, and values from the reference branch are denoted as $\{\mathbf{Q_i}^{\text{ref}}, \mathbf{K_i}^{\text{ref}}, \mathbf{V_i}^{\text{ref}}\}$, while those from the noised latent branch are denoted as $\{\mathbf{Q_i}, \mathbf{K_i}, \mathbf{V_i}\}$, producing an output denoted as $\mathbf{O_i}$ (see Fig. 2 (a)). While this design enhances spatial consistency, it processes each frame independently, leading to temporal inconsistencies. To address this, STAG4D extends keys and values in self-attention layers with those of the first input frame $F^{[v=0,t=1]}$, ensuring greater temporal coherence (see Fig. 2 (b)).

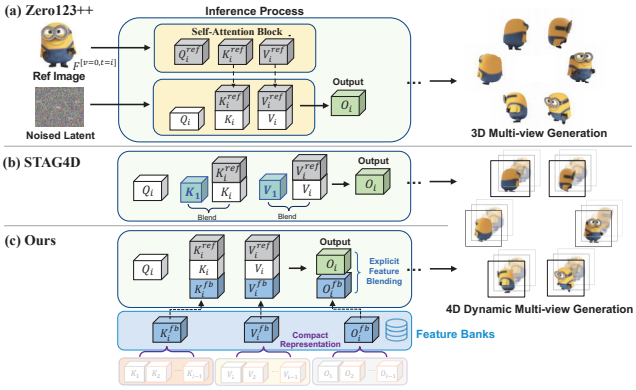


Figure 2. **Paradigm comparison of our method with previous works.** (a) Zero123++ uses a **dual-branch self-attention mechanism**, one branch for the reference image and the other for noised latent, with the reference image enhancing spatial consistency. (b) STAG4D adds **key-value information from the first frame** to improve frame consistency. (c) Our method introduces a feature bank that stores a **compact representation** deduced from the previous $i-1$ frames, with **the same size** as a *single* frame but contains richer feature temporally. This bank is used to generate $F^{[v=0,t=i]}$. After obtaining the output $\mathbf{O_i}$ from the self-attention block, we blend it with the stored $\mathbf{O_i^{fb}}$ from the feature bank to enhance temporal consistency (detailed in Sec. 4.2).

However, as shown in Fig. 1 (b), temporal inconsistency issues remain. We believe that considering only the first input frame $F^{[v=1:6,t=1]}$ as a condition during the generation of the subsequent i-th frame $F^{[v=1:6,t=i]}$ is insufficient for achieving better temporal consistency. (Take the rear view $v = 3, 4$ for example. Only conditioned on the first frame cannot ensure a smooth transition through time.) Instead, the conditions from all previous time, $F^{[v=1:6,t=1:(i-1)]}$, should be incorporated during the generation of $F^{[v=1:6,t=i]}$. Therefore, we extended the design of the attention blocks to incorporate all the relevant features from previously generated frames (see Fig. 2(c)).

4. Method

4.1. Overview

The overall pipeline, as shown in Fig. 3, illustrates the process of video-to-4D generation using FB-4D method. To address the spatial-temporal inconsistencies, we integrate the multi-view generation model [50] with a feature bank, implicitly modeling correspondences across views and timestamps. In Sec. 4.2, we explore this design by explaining how the feature bank is updated and fetched, detailing the compact representation and feature blending in Fig. 2. In Sec. 4.3, we introduce a novel approach that combines auto-regressive generation with our feature bank to generate more multi-view image sequences. For iteration index j, one of the output sequences from the previous

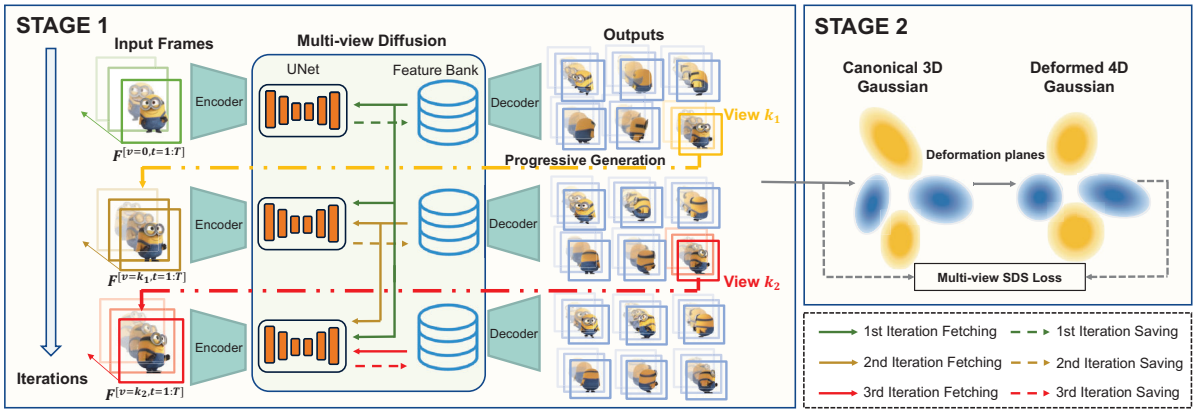


Figure 3. **Overall pipeline of video-to-4D generation in FB-4D.** Given a single-view video input, we integrate a multi-view generative diffusion model [50] with a feature bank to enhance spatial and temporal consistency, implicitly modeling the correspondence across views and timestamps (detailed in Sec. 4.2). We perform multiple iterations of generation, where each generated view is used as input for the next iteration. For example, in the first iteration, we use $F^{[v=0,t=1:T]}$ to generate multi-view sequences $F^{[v=1:6,t=1:T]}$ and in the second iteration, we select the input view progressively (detailed in Sec. 4.3). The input for this iteration becomes $F^{[v=k_1,t=1:T]}$, where $k_1 \in [1,6]$. This iterative process enables interaction across feature banks, refining the multi-view sequences over time. The generated sequences maintain high consistency in both spatial and temporal dimensions. After several iterations, we finally train a deformable 3D Gaussian to represent the 4D model.

iterations, $F^{[v=1:(6j-6),t=1:T]}$, serves as the input for the next iteration. Specifically, $F^{[v=k,t=1:T]}$ is selected, where k belongs to the range $[1, 6j-6]$. Furthermore, we propose a new progressive viewpoint selection strategy to enhance the generation process. After generating multi-view, multi-frame images, these are used to train a deformable 3D Gaussian field, enabling novel view rendering at continuous spatial positions and timestamps.

4.2. Implicit Correspondence Modeling with Feature Banks

Hooking Self-Attention Layers with Feature Bank: Overview. To store features from past frames, we propose integrating a newly designed feature bank module in the self-attention layers of multi-view generation models. For clarity, when processing the i-th input frame $F^{[v=0,t=i]}$, we denote the current feature bank set as $\mathbf{S}_\mathbf{i}^{\text{fb}}$, where $\mathbf{S} \in \{\mathbf{K}, \mathbf{V}, \mathbf{O}\}$. The shape of an element in the set, e.g., $\mathbf{K_i}$, is $\frac{hw}{s^2} \times D$, where $\frac{hw}{s^2}$ is the token count and D is the feature dimension. During the denoising process, the feature bank utilizes the feature set $\mathbf{S_i}$ from the noised latent branch, i.e., $(\mathbf{K_i}, \mathbf{V_i}, \mathbf{O_i})$, to update itself (Note that the notation t represents the index of the frame being generated. To enhance adaptability and performance, we employ distinct feature banks for different inference steps in the diffusion process and omit the notation for it).

The feature bank mechanism is integrated into all self-attention blocks. And we provide a detailed explanation of the processes for fetching and updating the feature bank, during which we implemented compact representation and feature blending. (see Fig. 2 (c))

Updating the Feature Bank: A Compact Representation. A straightforward update method operates in a queue-like fashion, where the maximum number of past frames is fixed. When the feature bank reaches its capacity, new frame features are added, and the oldest are removed. While maintaining a fixed size, this limits current frame's access to earlier features. Increasing the window size to include more frames requires more memory and may introduce redundancy, diminishing its effectiveness (see Table 4).

To address this, we propose a dynamic update strategy for the feature bank that retains essential features from past frames while efficiently controlling its size. We employ an efficient greedy strategy to merge the features in $\mathbf{S_i}$ and $\mathbf{S}_\mathbf{i}^{\text{fb}}$ [4, 7, 32, 77]. Specifically, the feature bank update procedure is outlined as follows: (illustrated in Fig. 4)

1. We merge the current frame's $\mathbf{S_i}$ with $\mathbf{S}_\mathbf{i}^{\text{fb}}$ from the feature bank in the token dimension to form $\mathbf{S}_\mathbf{i}^{\text{merge}}$, and then randomly split it into two subsets, src and dst, to maintain diversity without clustering's high cost.
2. For each feature src_p in src, we find the most similar feature dst_q in dst using cosine similarity.
3. We then fuse features by averaging those matched to dst_q, resulting in the updated feature set $\mathbf{S}_\mathbf{i}^{\text{fb}'}$.

Through this approach, we effectively control the size of the feature bank while integrating all past features. As a result, $\mathbf{S}_\mathbf{i}^{\text{fb}}$ serves as a compact representation of the previous $i-1$ frames $F^{[v=0,t=1:(i-1)]}$.

For initialization of the feature bank, we utilize a warm-up procedure before performing the inference. Specifically, we use the first five frames to initialize the feature bank. This strategy enables the feature bank to accumulate a more diverse and representative set of features.

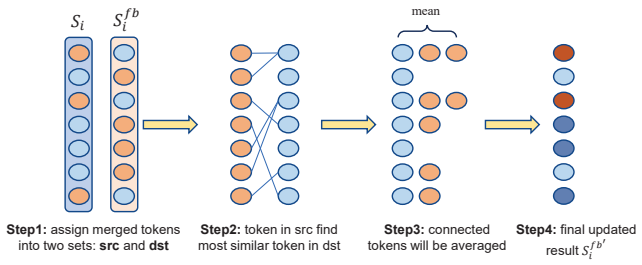


Figure 4. **Illustration of the compact feature fusion in the updating process**. The tokens in the feature bank are continuously updated through the aforementioned four steps, effectively integrating past features while ensuring that the size of the feature bank remains constant.

Fetching From the Feature Bank. *(I) Fetching Key and Value.* During self-attention computation for frame $F^{[v=0,t=i]}$, the model retrieves the key $\mathbf{K}_{\text{fb}}$ and value $\mathbf{V}_{\text{fb}}$ from the feature set $\mathbf{S}_{\mathbf{i}}^{\text{fb}}$, efficiently conditioning on past frames. To utilize these features, we extend reference attention [50] (Fig. 2), formulated as follows (see Fig. 5):

$$\text{SelfAttn}(\mathbf{F}^{[v=0,t=i]}) = \text{softmax}\left(\frac{\mathbf{Q_i}[\mathbf{K_i}, \mathbf{K}_{\text{ref}}, \mathbf{K}_{\text{fb}}]^T}{\sqrt{d}}\right)[\mathbf{V_i}, \mathbf{V}_{\text{ref}}, \mathbf{V}_{\text{fb}}], \quad (3)$$

$$\mathbf{O_i} = \text{MLP}_{\text{to_out}}(\text{SelfAttn}(\mathbf{F}^{[v=0,t=i]})), \quad (4)$$

(II) Fetching Output. In addition to the key and value, the output $\mathbf{O}_{\text{fb}}$ is also used in a similarity fusion manner for the current generation (see feature blending in Fig. 2 (c)). For each token α in $\mathbf{O}_i$, we use cosine similarity to find the most similar token β in $\mathbf{O}_{\text{fb}}$. The final output element $\mathbf{O}'_i(\alpha)$ can then be expressed as,

$$\mathbf{O}'_i(\alpha) = (1-\lambda)\cdot\mathbf{O}_i(\alpha) + \lambda\cdot\mathbf{O}_{\text{fb}}(\beta), \quad (5)$$

where λ is a hyperparameter used to control the strength of the fusion. We validate the necessity of fetching key-values and outputs separately in Tab. 3. Additionally, to prevent excessive fusion with past frames, a threshold τ is set to generate a mask, ensuring feature fusion occurs only when the cosine similarity is high. This preserves important details and prevents the current frame from becoming overly similar to past frames, avoiding reduced motion magnitude.

4.3. Progressive Generation in Multiple Iterations

We compared three multi-iteration methods, distinguished by colors: **brown**, **green** and **purple**.

The More You See in 2D, the More You Perceive in 3D? SAP3D [16] showed that increasing input images from diverse viewpoints during deformable 3D Gaussian training enhances generation quality. Naturally, we formulate an auto-regressive generation process, using generated viewpoints as new inputs for Zero123++ [50], and generate more

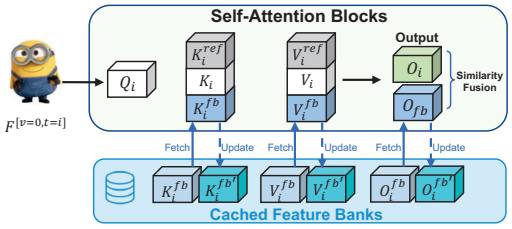


Figure 5. **Illustration of feature bank fetching and updating.** In the multi-view inference process of frame $F^{[v=0,t=i]}$, we first retrieve the corresponding feature set for computation and then utilize the keys, values, and outputs ($\mathbf{K_i}$, $\mathbf{V_i}$, $\mathbf{O_i}$) from the self-attention blocks to update the feature bank.

Method	viewpoint	F-B	Iter 1	Iter 2	Iter 3
(a)	random	×	881.17	902.28 +21.11	930.06 +48.89
(b)	random	✓	784.36	774.71 -9.65	772.05 -12.31
(c)	progressive	✓	784.36	756.90 -27.46	728.86 -55.50

Table 1. **FVD comparison in multi-iteration settings.** F-B indicates feature bank usage. The **brown line (a)** represents random selection of previous outputs. The **green line (b)** improves quality with feature banks. The **purple line (c)** further enhances stability and clarity through progressive generation.

anchor sequences. However, this yielded suboptimal results. More iterations and additional inputs greatly increase FVD in 4D generation for the baselines (Tab. 1(a)).

We argue that the issue persists in maintaining multi-view and frame-to-frame consistency. Specifically, as the number of iterations increases, spatial consistency tends to degrade. This accumulated error of inconsistencies across iterations introduces more confusion in the training process of the 3D deformable Gaussian model.

Utilizing Our Feature Banks in Multiple Iterations. For each iteration, due to the different input viewpoints, we utilize distinct feature banks to focus on more diverse features. For clarity, we define the feature set in the feature bank of the j-th iteration as $\mathbf{S}_{\text{fb}}^{j}$, where, for example, $\mathbf{K}_{\text{fb}}$ becomes $\mathbf{K}_{\text{fb}}^{j}$. To facilitate discussion, we define the input viewpoints for the second and third iterations as k_1 and k_2 respectively, where $k_1 \in [1, 6]$ and $k_2 \in [1, 12]$ and $k_1 \neq k_2$. For convenience, the first iteration input viewpoint, previously denoted as 0, is also represented as k_0. For three iterations, the whole process is formulated as (where Generator denotes the multi-view diffusion process):

$$\begin{aligned} F^{[v=1:6,t=1:T]} &= \text{Generator}(F^{[v=k_0,t=1:T]}), \\ F^{[v=7:12,t=1:T]} &= \text{Generator}(F^{[v=k_1,t=1:T]}), \\ F^{[v=13:18,t=1:T]} &= \text{Generator}(F^{[v=k_2,t=1:T]}), \end{aligned} \quad (6)$$

Determinating the weights of feature banks in previous iteration. To address the accumulated inconsistency, we leverage the feature bank from previous iterations. By as-

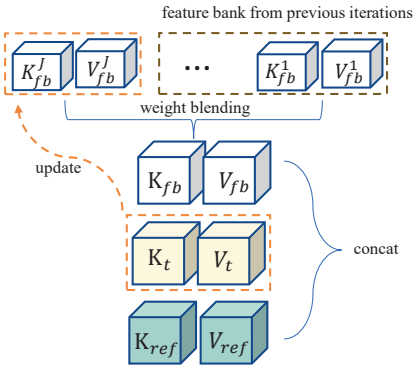


Figure 6. **Utilization of feature bank for generation in multiple iterations.** For fetching, keys and values from multiple feature banks are blended with weighted sum (Eq.7). For updating, newly predicted features are only updated into current feature banks.

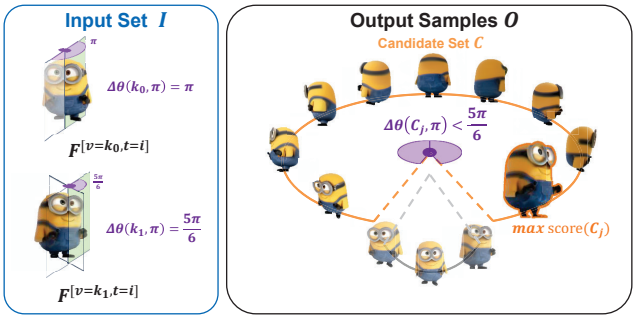


Figure 7. **Visualization of our progressive iteration**. For the current iteration $i = 3$, we compute the difference between the previous inputs $F^{[v=k_0,t=i]}$ and $F^{[v=k_1,t=i]}$ with π, as shown in the left part. Among all the generated viewpoints shown in the right part, we select the one that is closer to the rear viewpoint (π) as the candidate set for the next iteration. And finally, we choose the highest-scoring viewpoint based on Eq. 9 as the final input.

signing corresponding weights to feature banks from different iterations, we perform key-value weight blending, as illustrated in Fig. 6, formulated as:

$$\mathbf{K}_{\text{fb}} = \sum_{j=1}^{J} w_j \cdot \mathbf{K}_{\text{fb}}^{j}, \quad \mathbf{V}_{\text{fb}} = \sum_{j=1}^{J} w_j \cdot \mathbf{V}_{\text{fb}}^{j}. \tag{7}$$

where J denotes the current iteration. But how should these weights be determined? Intuitively, the greater the difference in viewpoints between the current iteration J and the previous iteration j, denoted as $\Delta\theta(k_{J-1}, k_{j-1}) = |\text{azimuth}(k_{J-1}) - \text{azimuth}(k_{j-1})|$, the lower the weight assigned to the previous iteration. Therefore, we define the weight for past feature banks as follows: $w_j = \frac{\pi - \Delta\theta(k_{J-1}, k_{j-1})}{2\pi \cdot (J-1)}$. And the weight assigned to the current feature bank is then given by $w_J = 1 - \sum_{j=1}^{J-1} w_j$.

In addition, in each iteration, we update only the feature bank of the current iteration to avoid excessive informational confusion across views also shown in Fig. 6.

As shown in Tab. 1**(b)**, during multiple iterations, leveraging the previous feature bank results in a significant reduction in FVD compared to not utilizing it. This demonstrates, for the first time, that generating additional reference sequences through multiple auto-regressive iterations can continuously improve downstream performance.

Upon further investigation, we identified two issues with the multi-view image sequences generated after several iterations: (i) Randomly selecting viewpoints as new inputs may cause significant differences between the previous input viewpoints and the current ones, leading to a reduced utilization of knowledge in the feature bank; (ii) The quality of generated images from rear or side viewpoints is generally lower. To address these issues, we introduce a progressive generation mechanism.

Progressive Generation in Multiple Iterations. To address the aforementioned issues, we designed a progressive generation process. In this process, by the time of the J-th iteration, the difference between the input azimuth angle k_{J-1} and the rear viewpoint (π degrees) (denoted as $\Delta\theta(k_{J-1}, \pi)$) is gradually reduced over successive iterations, enabling smoother transitions between viewpoints and improving the consistency of generated images across different angles. This approach ensures that side and rear views are generated progressively, leading to higher consistency and better overall image quality.

To explain further, the input viewpoints from the previous $J-1$ iterations form a set I, formulated as:

$$I = \left\{k_0, k_1, \ldots, k_{(J-2)}\right\},$$

representing all the input viewpoints before the current iteration. Here, $k_{(J-2)}$ is the viewpoint closest to the rear (π degrees) in this set. We also define the set of all viewpoints of output image sequences as O, which contains v ranging from 1 to $6J-6$. Next, we define a set of candidate viewpoints, C, which consists of viewpoints from the remaining image sequences that are closer to the rear viewpoint than $k_{(J-2)}$. The next input viewpoint for the current iteration is selected from C. This set can be expressed as:

$$C = \{k \in O \setminus I \mid \Delta\theta(k, \pi) < \Delta\theta(k_{J-2}, \pi)\} \tag{8}$$

After that, we calculate the similarity of each element in the candidate set $C_j \in C$ to the previous input viewpoints and select the viewpoint with the highest score for the next iteration input (as shown in Fig. 7), written as,

$$\text{score}(C_j) = \sum_{p=0}^{J-2} \sum_{q=1}^{T} W_p \cdot \text{CLIP}(F^{[v=k_p,t=q]}, F^{[v=C_j,t=q]}) \tag{9}$$

where W_p is the weight determined by the angular difference between the viewpoints, $W_p = (\pi - \Delta\theta(k_p, C_j))/2\pi$, with larger differences resulting in smaller weights.

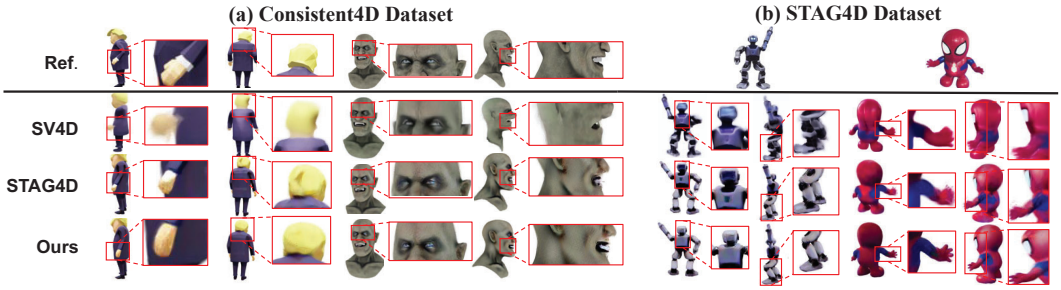


Figure 8. **Comparison of 4D generation quality on (a) Consistent4D dataset and (b) STAG4D dataset across different methods.**

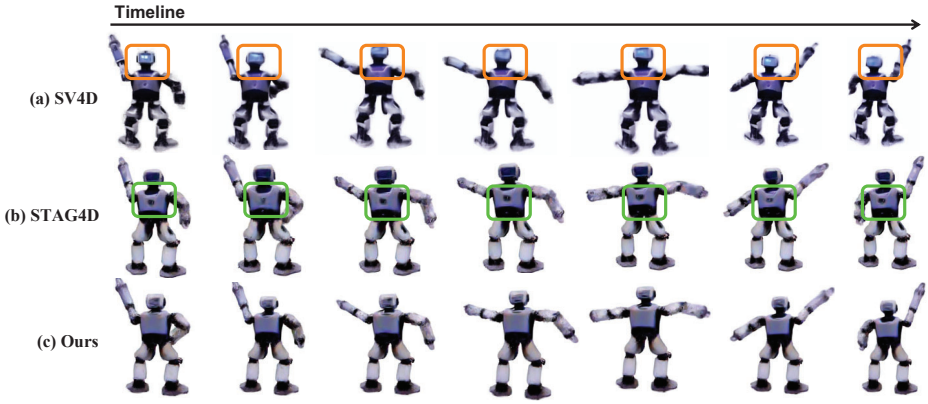


Figure 9. **4D Generation Quality Comparison (Back View).** Both (a) SV4D and (b) STAG4D show rear view inconsistencies.

The progressive generation approach maximizes feature bank usage for consistent multi-view image sequences and produces more high-quality sequences from different viewpoints. With this technique introduced, we are able to increase both the quantity and diversity of the generated images while explicitly ensuring their quality. As shown in the Tab. 1(c), our progressively generated method has achieved favorable results against **(a)** and **(b)**.

5. Experiments

5.1. Setup

Dataset and Metrics. We conduct experiments using the Consistent4D dataset [81], which includes multi-view videos of seven dynamic objects. We evaluate using three metrics: CLIP, LPIPS, and FVD [81]. CLIP and LPIPS assess image-level semantic and perceptual similarity, while FVD evaluates both frame quality and temporal coherence, making it ideal for video generation tasks. Additionally, we perform qualitative evaluation on the in-the-wild STAG4D dataset [86], generating 4D content for 28 more videos.

Implementation Details. Our feature bank is implemented using Zero123++ v1.2 [50], and we perform $K = 3$ iterations of generation. Our method does not require additional training; instead, we perform 75 inference steps per image. In the experiment, we set $\tau = 0.98$ and $\lambda = 0.8$ to ensure that fusion occurs only when local output features exhibit high similarity with feature bank features. This approach maintains consistency with previous frames while preserving information from the current frame.

Method	T-F	FVD (↓)	CLIP (↑)	LPIPS (↓)
SV4D [74]	×	732.40	0.920	0.118
4Diffusion [88]+	×	1551.63	0.873	0.228
L4GM [47]+	×	898.06	0.916	0.150
DS4D-GA [78]+	×	799.94	0.921	0.131
DS4D-DA [78]+	×	784.02	0.923	0.131
Consistent4D [23]	✓	1133.93	0.870	0.160
4DGen [83]	✓	-	0.894	0.130
STAG4D [86]	✓	992.21	0.909	0.126
SC4D [71]+	✓	852.98	0.912	0.137
MVTokenFlow [21]	✓	846.32	**0.948**	**0.122**
FB-4D (Ours)	✓	**724.26**	0.913	0.125

Table 2. **Comparison with state-of-the-art methods.** FB-4D achieves comparable performance to the training-based SV4D. T-F denotes training-free in Stage 1 (+ from [78]).

5.2. Experimental Results

In this section, we conduct extensive and comprehensive comparisons with previously mentioned models, including training-based SV4D [74], and training-free STAG4D [86], Consistent4D [23] and 4DGen [83]. Our results demonstrate the effectiveness of our design in improving spatiotemporal consistency and 4D generation quality.

Quantitative and Qualitative Results on Video-to-4D.

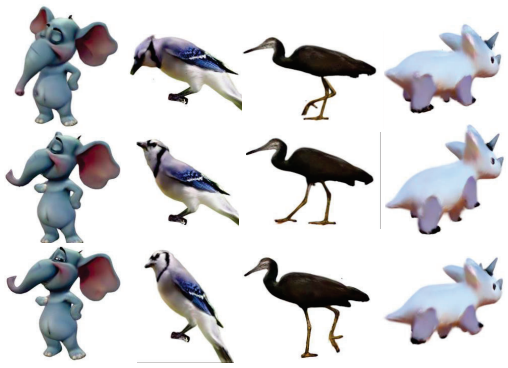


Figure 10. **More 4D results across timesteps and viewpoints.**

FB (K & V)	FB (Hidden States)	FVD (↓)	CLIP (↑)	LPIPS (↓)
×	×	881.17	0.909	0.129
✓	×	832.16	**0.911**	0.127
×	✓	844.39	0.909	0.128
✓	✓	**784.36**	0.909	**0.125**

Table 3. **Validation of the effectiveness of the feature bank.** FB means Feature Bank (only 1 iteration).

Updating Method	FVD (↓)	CLIP (↑)	LPIPS (↓)
queue (1)	818.24	**0.910**	0.128
queue (2)	844.48	0.908	**0.125**
dynamic (Ours)	**784.36**	0.909	**0.125**

Table 4. **Comparison of different feature bank updating implementations** (only 1 iteration).

Table 2 presents a comprehensive comparison between our method and several baselines across various evaluation metrics. Our approach outperforms all previous methods in video quality and smoothness, demonstrating superior realism and temporal consistency. Additionally, we achieve significantly better CLIP and LPIPS scores than Consistent4D, 4DGen, and STAG4D, indicating stronger semantic alignment with ground truth and enhanced realism. While SV4D surpasses our method on image-level metrics, our training-free design achieves comparable generation quality without requiring a 4D dataset.

For qualitative evaluation, Figure 8 (a) visually compares our results with other methods on the Consistent4D dataset [23], while Figure 8 (b) presents renderings from a dataset provided by STAG4D[86], with video sources from online resources. Additional results from online sources are shown in Fig. 10. These comparisons further highlight the effectiveness of our approach in generating high-quality and temporally coherent 4D content.

5.3. Ablation Study

Feature Bank. To validate our feature bank, we ablated key-value and hidden state utilization with one iteration for control. Results in Table 3 show that feature bank improves spatiotemporal consistency and generation quality, with detailed visualizations in the Supplementary Materials.

We also tested different update strategies. Table 4 shows that queue-based updates limit past information and cause temporal inconsistency, while longer queues introduce redundancy. Our method instead consolidates past information to enhance spatiotemporal consistency (see supplementary materials).

Progressive Generation Table 5 shows that integrating past features and selecting nearby views improves smoothness and realism. Table 6 further indicates that three iterations give the best results, with visual examples in the supplementary materials.

P-F	P-I	FVD (↓)	CLIP (↑)	LPIPS (↓)
×	×	902.28	0.909	0.128
✓	×	774.71	0.910	**0.124**
×	✓	805.74	**0.914**	0.125
✓	✓	**758.41**	**0.914**	**0.124**

Table 5. **Validation of integrating past feature banks (P-F) into progressive iterations (P-I, 2 iters).**

Iterations	FVD (↓)	CLIP (↑)	LPIPS (↓)
1	784.36	0.909	0.125
2	758.41	**0.914**	**0.124**
3	**724.26**	0.913	0.125

Table 6. **Comparison across different numbers of iterations.**

Sparse clip score for speeding up To balance quality and speed, CLIP similarity can be computed only on a sparse set of representative frames rather than all frames in Eq.9. This reduces runtime while maintaining guidance effectiveness. Table 7 summarizes the runtime–quality tradeoff.

clip guidance	FVD (↓)	CLIP (↑)	LPIPS (↓)	Time (min)
no clip guidance	785.33	0.910	0.126	**103**
sparse (5 frames)	733.26	0.912	**0.124**	114
all frames	**724.26**	**0.913**	0.125	152

Table 7. **Ablation of CLIP evaluation strategies** (3 iterations).

6. Limitations and Discussion

We introduce FB-4D, a framework for dynamic 3D content generation from monocular videos. By utilizing feature banks, FB-4D improves spatial and temporal consistency and mitigates drifting issues in progressive iterations, showing strong potential for real-world applications. As to the limitations, generation remains frame-by-frame with progressive iterations despite sparse CLIP guidance, leaving room for efficiency improvements in long sequences. Possible directions include (i) parallelizing generation while preserving spatiotemporal consistency, and (ii) using model distillation or lightweight diffusion architectures. Detailed computational cost is provided in supplementary materials.

References

[1] Sherwin Bahmani, Xian Liu, Wang Yifan, Ivan Skorokhodov, Victor Rong, Ziwei Liu, Xihui Liu, Jeong Joon Park, Sergey Tulyakov, Gordon Wetzstein, et al. Tc4d: Trajectory-conditioned text-to-4d generation. In *European Conference on Computer Vision*, pages 53–72. Springer, 2024. 1

[2] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7996–8006, 2024. 2

[3] Sherwin Bahmani, Ivan Skorokhodov, Aliaksandr Siarohin, Willi Menapace, Guocheng Qian, Michael Vasilkovsky, Hsin-Ying Lee, Chaoyang Wang, Jiaxu Zou, Andrea Tagliasacchi, et al. Vd3d: Taming large video diffusion transformers for 3d camera control. *arXiv preprint arXiv:2407.12781*, 2024. 2

[4] Benjamin Biggs, Arjun Seshadri, Yang Zou, Achin Jain, Aditya Golatkar, Yusheng Xie, Alessandro Achille, Ashwin Swaminathan, and Stefano Soatto. Diffusion soup: Model merging for text-to-image diffusion models. *arXiv preprint arXiv:2406.08431*, 2024. 4

[5] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 2

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

[7] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. *arXiv preprint arXiv:2210.09461*, 2022. 4

[8] Mingjin Chen, Junhao Chen, Xiaojun Ye, Huan-ang Gao, Xiaoxue Chen, Zhaoxin Fan, and Hao Zhao. Ultraman: single image 3d human reconstruction with ultra speed and detail. *arXiv preprint arXiv:2403.12028*, 2024. 2

[9] Zilong Chen, Feng Wang, Yikai Wang, and Huaping Liu. Text-to-3d using gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21401–21412, 2024. 2

[10] Wen-Hsuan Chu, Lei Ke, and Katerina Fragkiadaki. Dreamscene4d: Dynamic multi-object scene generation from monocular videos. *Advances in Neural Information Processing Systems*, 37:96181–96206, 2025. 1

[11] Yunze Deng, Haijun Xiong, Bin Feng, Xinggang Wang, and Wenyu Liu. Stp4d: Spatio-temporal-prompt consistent modeling for text-to-4d gaussian splatting. *arXiv preprint arXiv:2504.18318*, 2025. 2

[12] Huan-ang Gao, Mingju Gao, Jiaju Li, Wenyi Li, Rong Zhi, Hao Tang, and Hao Zhao. Scp-diff: Spatial-categorical joint prior for diffusion based semantic image synthesis. In *European Conference on Computer Vision*, pages 37–54. Springer, 2024. 2

[13] Antoine Guédon and Vincent Lepetit. Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5354–5363, 2024. 2

[14] Pengsheng Guo, Hans Hao, Adam Caccavale, Zhongzheng Ren, Edward Zhang, Qi Shan, Aditya Sankar, Alexander G Schwing, Alex Colburn, and Fangchang Ma. Stabledreamer: Taming noisy score distillation sampling for text-to-3d. *arXiv preprint arXiv:2312.02189*, 2023. 2

[15] 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. *arXiv preprint arXiv:2307.04725*, 2023. 2

[16] Xinyang Han, Zelin Gao, Angjoo Kanazawa, Shubham Goel, and Yossi Gandelsman. The more you see in 2d the more you perceive in 3d. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20912–20922, 2024. 2, 5

[17] 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. 2

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

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

[20] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*, 2023. 2

[21] Hanzhuo Huang, Yuan Liu, Ge Zheng, Jiepeng Wang, Zhiyang Dou, and Sibei Yang. Mvtokenflow: High-quality 4d content generation using multiview token flow. *arXiv preprint arXiv:2502.11697*, 2025. 7

[22] Hanwen Jiang, Zhenyu Jiang, Yue Zhao, and Qixing Huang. Leap: Liberate sparse-view 3d modeling from camera poses. *arXiv preprint arXiv:2310.01410*, 2023. 2

[23] Yanqin Jiang, Li Zhang, Jin Gao, Weimin Hu, and Yao Yao. Consistent4d: Consistent 360 {\deg} dynamic object generation from monocular video. *arXiv preprint arXiv:2311.02848*, 2023. 1, 2, 3, 7, 8

[24] Yanqin Jiang, Chaohui Yu, Chenjie Cao, Fan Wang, Weiming Hu, and Jin Gao. Animate3d: Animating any 3d model with multi-view video diffusion. *Advances in Neural Information Processing Systems*, 37:125879–125906, 2025. 1

[25] Animesh Karnewar, Niloy J Mitra, Andrea Vedaldi, and David Novotny. Holofusion: Towards photo-realistic 3d generative modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22976–22985, 2023. 2

[26] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkuehler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics (TOG)*, 42(4):139:1 – 139:14, 2023. 2

[27] Cheng Li, Keyuan Zhou, Tong Liu, Yu Wang, Mingqiao Zhuang, Huan-ang Gao, Bu Jin, and Hao Zhao. Avd2: Accident video diffusion for accident video description. *arXiv preprint arXiv:2502.14801*, 2025. 2

[28] Jiahao Li, Hao Tan, Kai Zhang, Zexiang Xu, Fujun Luan, Yinghao Xu, Yicong Hong, Kalyan Sunkavalli, Greg Shakhnarovich, and Sai Bi. Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model. *arXiv preprint arXiv:2311.06214*, 2023. 2

[29] Peng Li, Yuan Liu, Xiaoxiao Long, Feihu Zhang, Cheng Lin, Mengfei Li, Xingqun Qi, Shanghang Zhang, Wei Xue, Wenhan Luo, et al. Era3d: high-resolution multiview diffusion using efficient row-wise attention. *Advances in Neural Information Processing Systems*, 37:55975–56000, 2024. 1

[30] Renjie Li, Panwang Pan, Bangbang Yang, Dejia Xu, Shijie Zhou, Xuanyang Zhang, Zeming Li, Achuta Kadambi, Zhangyang Wang, Zhengzhong Tu, et al. 4k4dgen: Panoramic 4d generation at 4k resolution. *arXiv preprint arXiv:2406.13527*, 2024. 1

[31] Weiyu Li, Rui Chen, Xuelin Chen, and Ping Tan. Sweetdreamer: Aligning geometric priors in 2d diffusion for consistent text-to-3d. *arXiv preprint arXiv:2310.02596*, 2023. 2

[32] Wenyi Li, Huan-ang Gao, Mingju Gao, Beiwen Tian, Rong Zhi, and Hao Zhao. Training-free model merging for multi-target domain adaptation. *arXiv preprint arXiv:2407.13771*, 2024. 4

[33] Wenyi Li, Haoran Xu, Guiyu Zhang, Huan-ang Gao, Mingju Gao, Mengyu Wang, and Hao Zhao. Fairdiff: Fair segmentation with point-image diffusion. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 617–628. Springer, 2024. 2

[34] Yixun Liang, Xin Yang, Jiantao Lin, Haodong Li, Xiaogang Xu, and Yingcong Chen. Luciddreamer: Towards high-fidelity text-to-3d generation via interval score matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6517–6526, 2024. 2

[35] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8576–8588, 2024. 2

[36] Junchen Liu, Wenbo Hu, Zhuo Yang, Jianteng Chen, Guoliang Wang, Xiaoxue Chen, Yantong Cai, Huan-ang Gao, and Hao Zhao. Rip-nerf: Anti-aliasing radiance fields with ripmap-encoded platonic solids. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 2

[37] Minghua Liu, Ruoxi Shi, Linghao Chen, Zhuoyang Zhang, Chao Xu, Xinyue Wei, Hansheng Chen, Chong Zeng, Jiayuan Gu, and Hao Su. One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10072–10083, 2024. 2

[38] 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.

[39] 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. 1, 3

[40] 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. 1, 3

[41] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9970–9980, 2024. 1, 2

[42] Grace Luo, Lisa Dunlap, Dong Huk Park, Aleksander Holynski, and Trevor Darrell. Diffusion hyperfeatures: Searching through time and space for semantic correspondence. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[43] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)*, 41(4):1–15, 2022. 2

[44] Yuyan Ni, Shikun Feng, Haohan Chi, Bowen Zheng, Huan-ang Gao, Wei-Ying Ma, Zhi-Ming Ma, and Yanyan Lan. Straight-line diffusion model for efficient 3d molecular generation. *arXiv preprint arXiv:2503.02918*, 2025. 2

[45] Jangho Park, Taesung Kwon, and Jong Chul Ye. Zero4d: Training-free 4d video generation from single video using off-the-shelf video diffusion. *arXiv preprint arXiv:2503.22622*, 2025. 2

[46] 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. 2, 3

[47] Jiawei Ren, Cheng Xie, Ashkan Mirzaei, Karsten Kreis, Ziwei Liu, Antonio Torralba, Sanja Fidler, Seung Wook Kim, Huan Ling, et al. L4gm: Large 4d gaussian reconstruction model. *Advances in Neural Information Processing Systems*, 37:56828–56858, 2024. 7

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

[49] Kyle Sargent, Zizhang Li, Tanmay Shah, Charles Herrmann, Hong-Xing Yu, Yunzhi Zhang, Eric Ryan Chan, Dmitry La-

gun, Li Fei-Fei, Deqing Sun, et al. Zeronvs: Zero-shot 360-degree view synthesis from a single real image. *arXiv preprint arXiv:2310.17994*, 2023. 2

[50] 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. *arXiv preprint arXiv:2310.15110*, 2023. 1, 2, 3, 4, 5, 7

[51] Yukai Shi, Jianan Wang, He Cao, Boshi Tang, Xianbiao Qi, Tianyu Yang, Yukun Huang, Shilong Liu, Lei Zhang, and Heung-Yeung Shum. Toss: High-quality text-guided novel view synthesis from a single image. *arXiv preprint arXiv:2310.10644*, 2023. 2

[52] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*, 2023. 2

[53] Yahao Shi, Yang Liu, Yanmin Wu, Xing Liu, Chen Zhao, Jie Luo, and Bin Zhou. Drive any mesh: 4d latent diffusion for mesh deformation from video. *arXiv preprint arXiv:2506.07489*, 2025. 1

[54] 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. 2

[55] 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. *arXiv preprint arXiv:2301.11280*, 2023. 1, 2

[56] Jingxiang Sun, Bo Zhang, Ruizhi Shao, Lizhen Wang, Wen Liu, Zhenda Xie, and Yebin Liu. Dreamcraft3d: Hierarchical 3d generation with bootstrapped diffusion prior. *arXiv preprint arXiv:2310.16818*, 2023. 2

[57] Qi Sun, Zhiyang Guo, Ziyu Wan, Jing Nathan Yan, Shengming Yin, Wengang Zhou, Jing Liao, and Houqiang Li. Eg4d: Explicit generation of 4d object without score distillation. *arXiv preprint arXiv:2405.18132*, 2024. 1

[58] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *arXiv preprint arXiv:2309.16653*, 2023. 2

[59] Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. *Advances in Neural Information Processing Systems*, 36:1363–1389, 2023. 2

[60] Dmitry Tochilkin, David Pankratz, Zexiang Liu, Zixuan Huang, Adam Letts, Yangguang Li, Ding Liang, Christian Laforte, Varun Jampani, and Yan-Pei Cao. Triposr: Fast 3d object reconstruction from a single image. *arXiv preprint arXiv:2403.02151*, 2024. 2

[61] Vikram Voleti, Alexia Jolicoeur-Martineau, and Chris Pal. Mcvd-masked conditional video diffusion for prediction, generation, and interpolation. *Advances in neural information processing systems*, 35:23371–23385, 2022. 2

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

[63] 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. 2

[64] Peng Wang and Yichun Shi. Imagedream: Image-prompt multi-view diffusion for 3d generation. *arXiv preprint arXiv:2312.02201*, 2023. 2

[65] Peng Wang, Hao Tan, Sai Bi, Yinghao Xu, Fujun Luan, Kalyan Sunkavalli, Wenping Wang, Zexiang Xu, and Kai Zhang. Pf-lrm: Pose-free large reconstruction model for joint pose and shape prediction. *arXiv preprint arXiv:2311.12024*, 2023. 2

[66] Xinzhou Wang, Kai Sun, Xudong Zhang, Fuchun Sun, Ling Wang, and Bin He. Sync-4d: Monocular 4d reconstruction and generation with synchronized canonical distillation. *Journal of Visual Communication and Image Representation*, page 104483, 2025. 1

[67] 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. 2

[68] Zhouxia Wang, Ziyang Yuan, Xintao Wang, Yaowei Li, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. Motionctrl: A unified and flexible motion controller for video generation. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 2

[69] Xinyue Wei, Kai Zhang, Sai Bi, Hao Tan, Fujun Luan, Valentin Deschaintre, Kalyan Sunkavalli, Hao Su, and Zexiang Xu. Meshlrm: Large reconstruction model for high-quality mesh. *arXiv preprint arXiv:2404.12385*, 2024. 2

[70] Haohan Weng, Tianyu Yang, Jianan Wang, Yu Li, Tong Zhang, CL Chen, and Lei Zhang. Consistent123: Improve consistency for one image to 3d object synthesis. *arXiv preprint arXiv:2310.08092*, 2023. 2

[71] Zijie Wu, Chaohui Yu, Yanqin Jiang, Chenjie Cao, Fan Wang, and Xiang Bai. Sc4d: Sparse-controlled video-to-4d generation and motion transfer. In *European Conference on Computer Vision*, pages 361–379. Springer, 2024. 7

[72] Zijie Wu, Chaohui Yu, Yanqin Jiang, Chenjie Cao, Fan Wang, and Xiang Bai. Sc4d: Sparse-controlled video-to-4d generation and motion transfer. In *European Conference on Computer Vision*, pages 361–379. Springer, 2025. 3

[73] Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. Physgaussian: Physics-integrated 3d gaussians for generative dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4389–4398, 2024. 2

[74] Yiming Xie, Chun-Han Yao, Vikram Voleti, Huaizu Jiang, and Varun Jampani. Sv4d: Dynamic 3d content generation with multi-frame and multi-view consistency. *arXiv preprint arXiv:2407.17470*, 2024. 1, 2, 3, 7

[75] Qiangeng Xu, Zexiang Xu, Julien Philip, Sai Bi, Zhixin Shu, Kalyan Sunkavalli, and Ulrich Neumann. Point-nerf: Point-based neural radiance fields. In *Proceedings of*

the IEEE/CVF conference on computer vision and pattern recognition, pages 5438–5448, 2022. 2

[76] Zhiyuan Xu, Yinhe Chen, Huan-ang Gao, Weiyan Zhao, Guiyu Zhang, and Hao Zhao. Diffusion-based visual anagram as multi-task learning. *arXiv preprint arXiv:2412.02693*, 2024. 2

[77] Enneng Yang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. Model merging in llms, mllms, and beyond: Methods, theories, applications and opportunities. *arXiv preprint arXiv:2408.07666*, 2024. 4

[78] Liying Yang, Chen Liu, Zhenwei Zhu, Ajian Liu, Hui Ma, Jian Nong, and Yanyan Liang. Not all frame features are equal: Video-to-4d generation via decoupling dynamic-static features. *arXiv preprint arXiv:2502.08377*, 2025. 7

[79] Zeyu Yang, Zijie Pan, Chun Gu, and Li Zhang. Diffusion^2: Dynamic 3d content generation via score composition of orthogonal diffusion models. *arXiv preprint arXiv:2404.02148*, 2024. 2, 3

[80] Chun-Han Yao, Yiming Xie, Vikram Voleti, Huaizu Jiang, and Varun Jampani. Sv4d 2.0: Enhancing spatio-temporal consistency in multi-view video diffusion for high-quality 4d generation. *arXiv preprint arXiv:2503.16396*, 2025. 1

[81] 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. In *2024 International Conference on 3D Vision (3DV)*, pages 664–674. IEEE, 2024. 2, 7

[82] Taoran Yi, Jiemin Fang, Guanjun Wu, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Qi Tian, and Xinggang Wang. Gaussiandreamer: Fast generation from text to 3d gaussian splatting with point cloud priors. *arXiv preprint arXiv:2310.08529*, 2023. 2

[83] 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. 7

[84] Heng Yu, Chaoyang Wang, Peiye Zhuang, Willi Menapace, Aliaksandr Siarohin, Junli Cao, László Jeni, Sergey Tulyakov, and Hsin-Ying Lee. 4real: Towards photorealistic 4d scene generation via video diffusion models. *Advances in Neural Information Processing Systems*, 37:45256–45280, 2024. 1

[85] Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19447–19456, 2024. 2

[86] Yifei Zeng, Yanqin Jiang, Siyu Zhu, Yuanxun Lu, Youtian Lin, Hao Zhu, Weiming Hu, Xun Cao, and Yao Yao. Stag4d: Spatial-temporal anchored generative 4d gaussians. In *European Conference on Computer Vision*, pages 163–179. Springer, 2025. 1, 2, 3, 7, 8

[87] Guiyu Zhang, Huan-ang Gao, Zijian Jiang, Hao Zhao, and Zhedong Zheng. Ctrl-u: Robust conditional image generation via uncertainty-aware reward modeling. *arXiv preprint arXiv:2410.11236*, 2024. 2

[88] Haiyu Zhang, Xinyuan Chen, Yaohui Wang, Xihui Liu, Yunhong Wang, and Yu Qiao. 4diffusion: Multi-view video diffusion model for 4d generation. *Advances in Neural Information Processing Systems*, 37:15272–15295, 2024. 2, 3, 7

[89] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3836–3847, 2023. 2

[90] Yuyang Zhao, Zhiwen Yan, Enze Xie, Lanqing Hong, Zhenguo Li, and Gim Hee Lee. Animate124: Animating one image to 4d dynamic scene. *arXiv preprint arXiv:2311.14603*, 2023. 2

[91] Yuyang Zhao, Chung-Ching Lin, Kevin Lin, Zhiwen Yan, Linjie Li, Zhengyuan Yang, Jianfeng Wang, Gim Hee Lee, and Lijuan Wang. Genxd: Generating any 3d and 4d scenes. *arXiv preprint arXiv:2411.02319*, 2024. 1

[92] Linqi Zhou, Andy Shih, Chenlin Meng, and Stefano Ermon. Dreampropeller: Supercharge text-to-3d generation with parallel sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4610–4619, 2024. 2

[93] Zi-Xin Zou, Zhipeng Yu, Yuan-Chen Guo, Yangguang Li, Ding Liang, Yan-Pei Cao, and Song-Hai Zhang. Triplane meets gaussian splatting: Fast and generalizable single-view 3d reconstruction with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10324–10335, 2024. 2