Generative Image Dynamics

Zhengqi Li Richard Tucker Noah Snavely Aleksander Holynski
Google Research

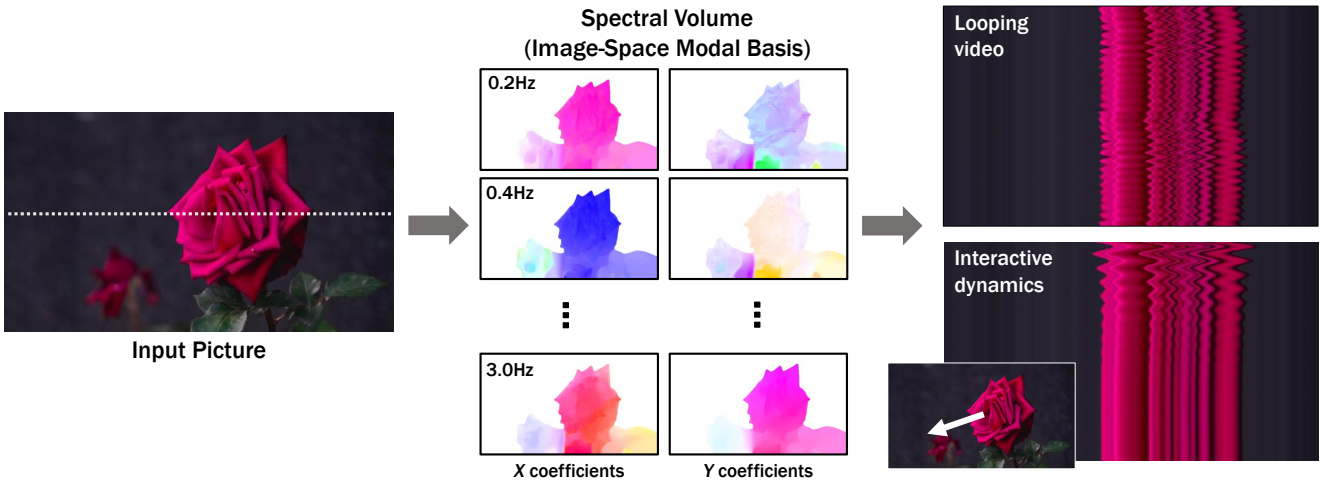


Figure 1. We model a generative image-space prior on scene motion: from a single RGB image, our method generates a *spectral volume* [23], a motion representation that models dense, long-term pixel trajectories in the Fourier domain. Our learned motion priors enable applications such as turning a single picture into a seamlessly looping video, or—by interpreting a predicted spectral volume as an image-space modal basis [22]—taking an image and creating an interactive dynamic simulation that responds to inputs like dragging and releasing points. On the right, we visualize output videos using space-time X-t slices through 10 seconds of video (along the scanline shown in the input picture).

Abstract

We present an approach to modeling an image-space prior on scene motion. Our prior is learned from a collection of motion trajectories extracted from real video sequences depicting natural, oscillatory dynamics such as trees, flowers, candles, and clothes swaying in the wind. We model this dense, long-term motion prior in the Fourier domain: given a single image, our trained model uses a frequency-coordinated diffusion sampling process to predict a spectral volume, which can be converted into a motion texture that spans an entire video. Along with an image-based rendering module, these trajectories can be used for a number of downstream applications, such as turning still images into seamlessly looping videos, or allowing users to realistically interact with objects in real pictures by interpreting the spectral volumes as image-space modal bases, which approximate object dynamics. See our project page for more results: generative-dynamics.github.io.

1. Introduction

The natural world is always in motion, with even seemingly static scenes containing subtle oscillations as a result of factors such as wind, water currents, respiration, or other natural rhythms. Motion is one of the most salient visual signals, and humans are particularly sensitive to it: captured imagery without motion (or even with slightly unrealistic motion) can often seem uncanny or unreal.

While it is easy for humans to interpret or imagine motion in scenes, training a model to learn realistic scene motion is far from trivial. The motion we observe in the world is the result of a scene's underlying physical dynamics, i.e., forces applied to objects that respond according to their unique physical properties — their mass, elasticity, etc. These properties and forces are hard to measure and capture at scale, but fortunately, in many cases measuring them is unnecessary: the necessary signals for producing plausible motion can often be extracted from observed 2D motion [23]. While real-world observed motion is multi-modal and grounded in

complex physical effects, it is nevertheless often predictable: candles will flicker in certain ways, trees will sway, and their leaves will rustle. This predictability is ingrained in our human perception of real scenes: by viewing a still image, we can imagine plausible motions that might have been ongoing as the picture was captured — or, since there might have been many possible such motions, a *distribution* of natural motions conditioned on that image. Given the facility with which humans are able to imagine these possible motions, a natural research problem is to model this same distribution computationally.

Recent advances in generative models, in particular conditional diffusion models [43, 83, 85], have enabled us to model rich distributions, including distributions of real images conditioned on text [71–73]. This capability has enabled a number of previously impossible applications, such as text-conditioned generation of diverse and realistic image content. Following the success of these image models, recent work has extended these models to other domains, such as videos [7, 42] and 3D geometry [75, 98, 99, 101].

In this paper, we explore modeling a generative prior for *image-space scene motion*, i.e., the motion of all pixels in a single image. This model is trained on motion trajectories automatically extracted from a large collection of real video sequences. In particular, from each training video we compute motion in the form of a *spectral volume* [22, 23], a frequency-domain representation of dense, long-range pixel trajectories. This motion representation is well-suited to scenes that exhibit oscillatory dynamics such as trees and flowers moving in the wind. We find that this representation is also highly efficient and effective as an output of a diffusion model for modeling scene motions. We train a generative model that, conditioned on a single image, can sample spectral volumes from its learned distribution. A predicted spectral volume can then be directly transformed into a motion texture—a set of per-pixel, long-range pixel motion trajectories—that can be used to animate the image. Further, the spectral volume can be interpreted as an *image-space modal basis* that can be used to simulate interactive dynamics, using the modal analysis technique of Davis *et al*. [22]. In this paper, we refer to the underlying frequency-space representation as either a spectral volume or an image-space modal basis, depending on whether it is used to encode a specific motion texture, or used to simulate dynamics.

We predict spectral volumes from input images using a diffusion model that generates coefficients one frequency at a time, but coordinates these predictions across frequency bands through a shared attention module. The predicted motions can be used to synthesize future frames (via an image-based rendering model)—turning still images into realistic animations, as illustrated in Fig. 1.

Compared with priors over raw RGB pixels, priors over motion capture more fundamental, lower-dimensional underlying structure that efficiently explains long-range variations in pixel values. Hence, generating intermediate motion leads to more coherent long-term generation and more fine-grained control over animations when compared with methods that perform image animation via synthesis of raw video frames. We demonstrate the use of our trained model in several downstream applications, such as creating seamless looping videos, editing the generated motions, and enabling interactive dynamic images via image-space modal bases, i.e., simulating the response of object dynamics to user-applied forces [22].

2. Related Work

Generative synthesis. Recent advances in generative models have enabled photorealistic synthesis of images conditioned on text prompts [16, 17, 24, 71–73]. These generative text-to-image models can be augmented to synthesize video sequences by extending the generated image tensors along a time dimension [7, 9, 42, 61, 82, 104, 104, 109]. While these methods are effective at producing plausible video sequences that capture the spatiotemporal statistics of real footage, the resulting videos often suffer from artifacts such as incoherent motion, unrealistic temporal variation in textures, and violations of physical constraints like preservation of mass.

Animating images. Instead of generating videos entirely from text, other techniques take as input a still picture and animate it. Many recent deep learning methods adopt a 3D-Unet architecture to produce video volumes directly from an input image [27, 35, 39, 46, 52, 91]. Because these models are effectively the same video generation models (but conditioned on image information instead of text), they exhibit similar artifacts to those mentioned above. One way to overcome these limitations is to not directly generate the video content itself, but instead animate an input source image through explicit or implicit image-based rendering, i.e., moving the image content around according to motion derived from external sources such as a driving video [50, 78–80, 97], motion or 3D geometry priors [8, 29, 45, 62, 63, 65, 88, 95, 99, 100, 102, 107], or user annotations [6, 18, 20, 32, 37, 96, 103, 106]. Animating images according to motion fields yields greater temporal coherence and realism, but these prior methods require additional guidance signals or user input, or otherwise utilize limited motion representations (e.g., optical flow fields, as opposed to full-video dense motion trajectories).

Motion models and motion priors. In computer graphics, natural, oscillatory 3D motion (e.g., water rippling or trees waving in the wind) has long been modeled with noise that is shaped in the Fourier domain and then converted via an inverse Fourier transform to time-domain motion fields [77, 86]. Some of these methods rely on a modal analysis of the

underlying dynamics of the systems being simulated [22, 25, 87]. These spectral techniques were adapted to animate plants, water, and clouds from single 2D pictures by Chuang *et al*. [20] with additional user annotations. Our work is particularly inspired by that of Davis [23], who showed how to connect modal analysis of a scene with the motions observed in a video of that scene, and how to use this analysis to simulate interactive dynamics from a video. We adopt the frequency-space motion representation of the *spectral volume* from Davis *et al*., extract this representation from a large set of training videos, and show that this representation is suitable for predicting motion from single images with diffusion models.

Other methods have also used various motion representations in *prediction* tasks — where an image or video is used to inform a deterministic future motion estimate [33, 69], or a more rich *distribution* of possible motions (which can be modeled explicitly or by predicting the pixel values that would be induced by some implicit motion estimate) [92, 94, 102]. However, many of these methods predict an optical flow motion estimate (i.e., the instantaneous motion of each pixel), not full per-pixel motion trajectories. In addition, much of this prior work is focused on tasks like activity recognition, not on synthesis tasks. More recent work has demonstrated the advantages of modeling and predicting motion using generative models in a number of closed-domain settings such as humans and animals [2, 19, 28, 70, 89, 105].

Videos as textures. Certain moving scenes can be thought of as a kind of texture—termed *dynamic textures* by Doretto *et al*. [26]—that model videos as space-time samples of a stochastic process. Dynamic textures can represent smooth, natural motions such as waves, flames, or moving trees, and have been widely used for video classification, segmentation or encoding [12–15, 74]. A related kind of texture, called a *video texture*, represents a moving scene as a set of input video frames along with transition probabilities between any pair of frames [64, 76]. A large body of work exists for estimating and producing dynamic or video textures through analysis of scene motion and pixel statistics, with the aim of generating seamlessly looping or infinitely varying output videos [1, 21, 31, 57, 58, 76]. In contrast to much of this previous work, our method learns priors in advance that can then be applied to single images.

3. Overview

Given a single picture I_0, our goal is to generate a video $\{\hat{I}_1, \hat{I}_2., ..., \hat{I}_T\}$ of length T featuring oscillation dynamics such as those of trees, flowers, or candle flames moving in the breeze. Our system consists of two modules, a motion prediction module and an image-based rendering module. Our pipeline begins by using a latent diffusion model (LDM) to predict a spectral volume $\mathcal{S} = \left(S_{f_0}, S_{f_1}, ..., S_{f_{K-1}}\right)$ for the input image I_0. The predicted spectral volume is then transformed to a sequence of motion displacement fields (a motion texture) $\mathcal{F} = (F_1, F_2, ..., F_T)$ using an inverse discrete Fourier transform. This motion determines the position of each input pixel at each future time step.

Given a predicted motion texture, our rendering module animates the input RGB image using an image-based rendering technique that splats encoded features from the input image and decodes these splatted features into an output frame with an image synthesis network (Sec. 5). We explore applications of this method, including producing seamless looping animations and simulating interactive dynamics, in Sec. 6.

4. Predicting motion

4.1. Motion representation

Formally, a motion texture is a sequence of time-varying 2D displacement maps $\mathcal{F} = \{F_t | t = 1, ..., T\}$, where the 2D displacement vector $F_t(\mathbf{p})$ at each pixel coordinate $\mathbf{p}$ from input image I_0 defines the position of that pixel at a future time t [20]. To generate a future frame at time t, one can splat pixels from I_0 using the corresponding displacement map D_t, resulting in a forward-warped image I'_t:

$$I'_t(\mathbf{p} + F_t(\mathbf{p})) = I_0(\mathbf{p}). \quad (1)$$

If our goal is to produce a video via a motion texture, then one choice would be to predict a time-domain motion texture directly from an input image. However, the size of the motion texture would need to scale with the length of the video: generating T output frames implies predicting T displacement fields. To avoid predicting such a large output representation for long output videos, many prior animation methods either generate video frames autoregressively [7, 29, 56, 59, 91], or predict each future output frame independently via an extra time embedding [4]. However, neither strategy ensures long-term temporal consistency of generated video frames.

Fortunately, many natural motions, can be described as a superposition of a small number of harmonic oscillators represented with different frequencies, amplitude and phases [20, 23, 25, 49, 67]. Because the underlying motions are quasi-periodic, it is natural to model them in the frequency domain, from which it is convenient to generate a video of arbitrary length.

Hence, we adopt an efficient frequency space representation of motion in a video from Davis *et al*. [23] called a *spectral volume*, visualized in Figure 1. A spectral volume is the temporal Fourier transform of pixel trajectories extracted from a video, organized into images called *modal images*. Davis *et al*. further shows that, under certain assumptions, the spectral volume, evaluated at certain frequencies, forms an *image-space modal basis* that is a projection of the vibration modes of the underlying scene (or, more generally,

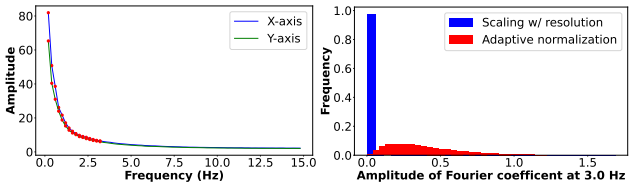


Figure 2. **Left:** We visualize the average motion power spectrum for the X and Y motion components extracted from a dataset of real videos, shown as the blue and green curves. Natural oscillation motions are composed primarily of low-frequency components, and so we use the first $K = 16$ terms as marked by red dots. **Right:** we show a histogram of the amplitude of Fourier terms at 3Hz ($K = 16$) after (1) scaling amplitude by image width and height (blue), or (2) frequency adaptive normalization (red). Our adaptive normalization prevents the coefficients from concentrating at extreme values.

captures spatial correlations in motion) [22]. We use the term *spectral volume* to refer to a frequency-space encoding of a specific motion texture (with high frequencies removed). Later, we also refer to a spectral volume as an "image-space modal basis" when it is used for simulation.

Given this motion representation, we formulate the motion prediction problem as a multi-modal image-to-image translation task: from an input image to an output spectral motion volume. We adopt latent diffusion models (LDMs) to generate spectral volumes comprised of a $4K$-channel 2D motion spectrum map, where $K << T$ is the number of frequencies modeled, and where at each frequency we need four scalars to represent the complex Fourier coefficients for the x and y dimensions. Note that the motion trajectory of a pixel at future time steps $\mathcal{F}(\mathbf{p}) = \{F_t(\mathbf{p})|t = 1, 2, ...T\}$ and its representation as a spectral volume $\mathcal{S}(\mathbf{p}) = \{S_{f_k}(\mathbf{p})|k = 0, 1, ..\frac{T}{2} - 1\}$ are related by the Fast Fourier transform (FFT):

$$\mathcal{S}(\mathbf{p}) = \text{FFT}(\mathcal{F}(\mathbf{p})). \quad (2)$$

How should we select the K output frequencies? Prior work in real-time animation has observed that most natural oscillation motions are composed primarily of low-frequency components [25, 67]. To validate this observation, we computed the average power spectrum of the motion extracted from 1,000 randomly sampled 5-second real video clips. As shown in the left plot of Fig. 2, the power spectrum of the motion decreases exponentially with increasing frequency. This suggests that most natural oscillation motions can indeed be well represented by low-frequency terms. In practice, we found that the first $K = 16$ Fourier coefficients are sufficient to realistically reproduce the original natural motion in a range of real videos and scenes.

4.2. Predicting motion with a diffusion model

We choose a latent diffusion model (LDM) [72] as the backbone for our motion prediction module, as LDMs are more computationally efficient than pixel-space diffusion models, while preserving generation quality. A standard LDM consists of two main modules: (1) a Variational Autoencoder (VAE) that compresses the input image to a latent space through an encoder $z = E(I)$, then reconstructs the input from the latent features via a decoder $I = D(z)$, and (2) a U-Net based diffusion model that learns to iteratively denoise latent features starting from Gaussian random noise. Our training applies this not to input images but to motion spectra from real video sequences, which are encoded and then diffused for n steps with a pre-defined variance schedule to produce noisy latents z^n. The 2D U-Nets are trained to denoise the noisy latents by iteratively estimating the noise $\epsilon_\theta(z^n; n, c)$ used to update the latent feature at each step $n \in (1, 2, ..., N)$. The training loss for the LDM is written as

$$\mathcal{L}_{\text{LDM}} = \mathbb{E}_{n\in\mathcal{U}[1,N],\epsilon^n\in\mathcal{N}(0,1)} \left[||\epsilon^n - \epsilon_\theta(z^n; n, c)||^2\right] \quad (3)$$

where c is the embedding of any conditional signal, such as text, semantic labels, or, in our case, the first frame of the training video sequence, I_0. The clean latent features z^0 are then passed through the decoder to recover the spectral volume.

Frequency adaptive normalization. One issue we observed is that motion textures have particular distribution characteristics across frequencies. As visualized in the left plot of Fig. 2, the amplitude of our motion textures spans a range of 0 to 100 and decays approximately exponentially with increasing frequency. As diffusion models require that output values lie between 0 and 1 for stable training and denoising, we must normalize the coefficients of $\mathcal{S}$ extracted from real videos before using them for training. If we scale the magnitudes of $\mathcal{S}$ coefficients to [0,1] based on image width and height as in prior work [29, 75], almost all the coefficients at higher frequencies will end up close to zero, as shown in Fig. 2 (right-hand side). Models trained on such data can produce inaccurate motions, since during inference, even small prediction errors can lead to large relative errors after denormalization when the magnitude of the normalized $\mathcal{S}$ coefficients are very close to zero.

To address this issue, we employ a simple but effective frequency adaptive normalization technique. In particular, we first independently normalize Fourier coefficients at each frequency based on statistics computed from the training set. Namely, at each individual frequency f_j, we compute the 97^{th} percentile of the Fourier coefficient magnitudes over all input samples and use that value as a per-frequency scaling factor s_{f_j}. Furthermore, we apply a power transformation to each scaled Fourier coefficient to pull it away from extremely small or large values. In practice, we found that a square root transform performs better than other transformations, such as log or reciprocal. In summary, the final coefficient values of spectral volume $\mathcal{S}(\mathbf{p})$ at frequency f_j (used for training

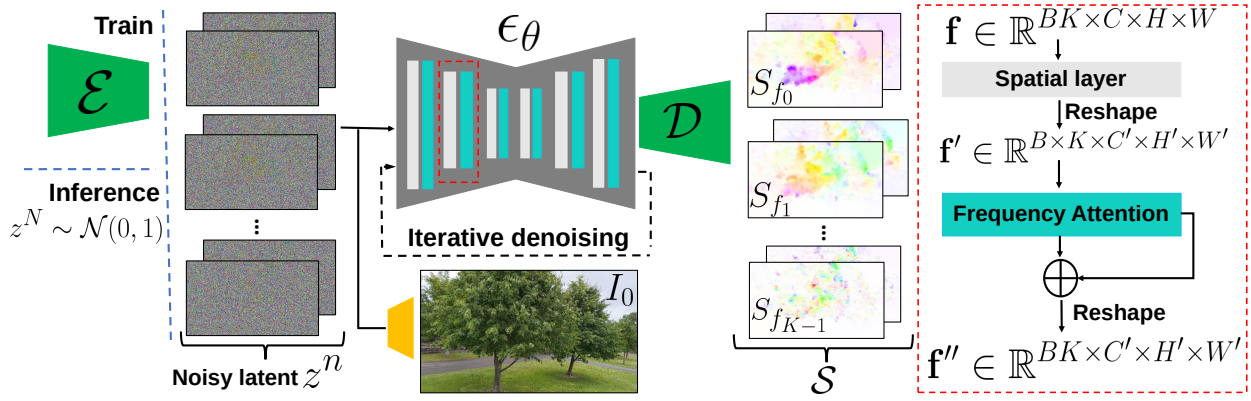


Figure 3. **Motion prediction module.** We predict a spectral volume $\mathcal{S}$ through a frequency-coordinated denoising model. Each block of the diffusion network ϵ_θ interleaves 2D spatial layers with frequency cross-attention layers (**red box, right**), and iteratively denoises latent features z^n. The denoised features are fed to a VAE decoder $\mathcal{D}$ to produce $\mathcal{S}$. During training, we concatenate the downsampled input I_0 with noisy latent features encoded from a real motion texture via a VAE encoder $\mathcal{E}$, and replace the noisy features with Gaussian noise z^N during inference (**left**).

our LDM) are computed as

$$S'_{f_j}(\mathbf{p}) = \text{sign}(S_{f_j}) \sqrt{\left| \frac{S_{f_j}(\mathbf{p})}{s_{f_j}} \right|}. \quad (4)$$

As shown on the right plot of Fig. 2, after applying frequency adaptive normalization the spectral volume coefficients no longer concentrate in a range of extremely small values.

Frequency-coordinated denoising. The straightforward way to to predict a spectral volume $\mathcal{S}$ with K frequency bands is to output a tensor of $4K$ channels from a standard diffusion U-Net. However, as in prior work [7], we observe that training a model to produce such a large number of channels tends to produce over-smoothed and inaccurate output. An alternative would be to independently predict a motion spectrum map at each individual frequency by injecting an extra frequency embedding to the LDM, but this results in uncorrelated predictions in the frequency domain, leading to unrealistic motion.

Therefore, we propose a frequency-coordinated denoising strategy as shown in Fig. 3. In particular, given an input image I_0, we first train an LDM ϵ_θ to predict a spectral volume texture map S_{f_j} with four channels to represent each individual frequency f_j, where we inject extra frequency embedding along with time-step embedding to the LDM network. We then freeze the parameters of this LDM model ϵ_θ and introduce attention layers and interleave them with 2D spatial layers of ϵ_θ across K frequency bands. Specifically, for a batch size B of input images, the 2D spatial layers of ϵ_θ treat the corresponding $B \cdot K$ noisy latent features of channel size C as independent samples with shape $\mathcal{R}^{(B\cdot K)\times C\times H\times W}$. The cross-attention layer then interprets these as consecutive features spanning the frequency axis, and we reshape the latent features from previous 2D spatial layers to $\mathcal{R}^{B\times K\times C\times H\times W}$ before feeding them to the attention layers. In other words, the frequency attention layers are used to coordinate the pre-trained motion latent features across all frequency channels in order to produce coherent spectral volumes. In our experiments, we observed that the average VAE reconstruction error improves from 0.024 to 0.018 when we switch from a standard 2D U-Net to a frequency-coordinated denoising module, suggesting an improved upper bound on LDM prediction accuracy; in our ablation study in Sec. 7.6, we also demonstrate that this design choice improves video generation quality compared with simpler configurations mentioned above.

5. Image-based rendering

We now describe how we take a spectral volume $\mathcal{S}$ predicted for a given input image I_0 and render a future frame $\hat{I}_t$ at time t. We first derive motion trajectory fields in the time domain using the inverse temporal FFT applied at each pixel $\mathcal{F}(\mathbf{p}) = \text{FFT}^{-1}(\mathcal{S}(\mathbf{p}))$. The motion trajectory fields determine the position of every input pixel at every future time step. To produce a future frame $\hat{I}_t$, we adopt a deep image-based rendering technique and perform splatting with the predicted motion field F_t to forward warp the encoded I_0, as shown in Fig. 4. Since forward warping can lead to holes, and multiple source pixels can map to the same output 2D location, we adopt the feature pyramid softmax splatting strategy proposed in prior work on frame interpolation [66].

Specifically, we encode I_0 through a feature extractor network to produce a multi-scale feature map $\mathcal{M} = \{M_j | j = 0, ..., J\}$. For each individual feature map M_j at scale j, we resize and scale the predicted 2D motion field F_t according to the resolution of M_j. As in Davis *et al.* [22], we use predicted flow magnitude, as a proxy for depth, to determine

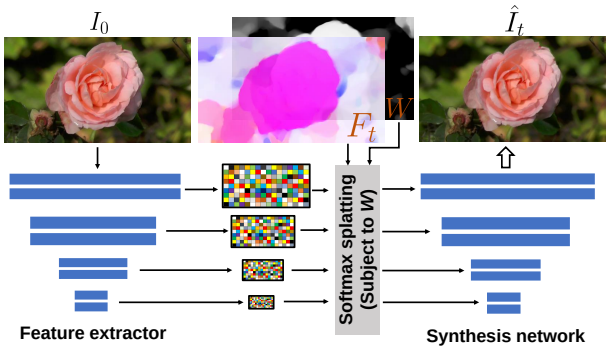


Figure 4. **Rendering module.** We fill in missing content and refine the warped input image using a motion-aware deep image-based rendering module, where multi-scale features are extracted from the input image I_0. Softmax splatting is then applied over the features with a motion field F_t from time 0 to t (subject to the weights W derived from motion). The warped features are fed to an image synthesis network to produce the refined image $\hat{I}_t$.

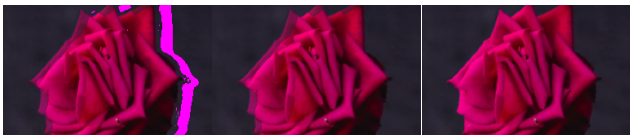
the contributing weight of each source pixel mapped to its destination location. In particular, we compute a per-pixel weight, $W(\mathbf{p}) = \frac{1}{T}\sum_t ||F_t(\mathbf{p})||_2$ as the average magnitude of the predicted motion trajectory fields. In other words, we assume large motions correspond to moving foreground objects, and small or zero motions correspond to background objects. We use motion-derived weights instead of learnable ones because we observe that in the single-view case, learnable weights are not effective for addressing disocclusion ambiguities, as shown in the second column of Fig. 5.

With the motion field F_t and weights W, we apply softmax splatting to warp feature map at each scale to produce a warped feature $M'_{j,t} = \mathcal{W}_{\text{softmax}}(M_j, F_t, W)$, where $\mathcal{W}_{\text{softmax}}$ is the softmax splatting operation. The warped features $M'_{j,t}$ are then injected into intermediate blocks of an image synthesis decoder network to produce a final rendered image $\hat{I}_t$.

We jointly train the feature extractor and synthesis networks with start and target frames (I_0, I_t) randomly sampled from real videos, using the estimated flow field from I_0 to I_t to warp encoded features from I_0, and supervising predictions $\hat{I}_t$ against I_t with a VGG perceptual loss [48]. As shown in Fig. 5, compared to direct average splatting and a baseline deep warping method [45], our motion-aware feature splatting produces a frame without holes or artifacts around disocclusions.

6. Applications

We demonstrate applications that add dynamics to single still images using our proposed motion representations and animation pipeline.



(a) Average-splat (b) Baseline-splat (c) Ours

Figure 5. From left to right, we show a rendered future frame with (a) average splatting in RGB pixel space, (b) softmax splatting with learnable weights [45], and (c) motion-aware feature splatting.

6.1. Image-to-video

Our system enables the animation of a single still picture by first predicting a spectral volume from the input image and generating an animation by applying our image-based rendering module to the motion displacement fields derived from the spectral volume. Since we explicitly model scene motion, this allows us to produce slow-motion videos by linear interpolating the motion displacement fields and to magnify (or minify) animated motions by adjusting the amplitude of predicted spectral volume coefficients.

6.2. Seamless looping

It is sometimes useful to generate videos with motion that loops seamlessly, meaning that there is no discontinuity in appearance or motion between the start and end of the video. Unfortunately, it is hard to find a large collection of seamlessly looping videos for training diffusion models. Instead, we devise a method to use our motion diffusion model, trained on regular non-looping video clips, to produce seamless looping video. Inspired by recent work on guidance for image editing [3, 30], our method is a *motion self-guidance* technique that guides the motion denoising sampling processing using explicit looping constraints. In particular, at each iterative denoising step during the inference stage, we incorporate an additional motion guidance signal alongside standard classifier-free guidance [44], where we enforce each pixel's position and velocity at the start and end frames to be as similar as possible:

$$\hat{\epsilon}^n = (1+w)\epsilon_\theta(z^n; n, c) - w\epsilon_\theta(z^n; n, \emptyset) + u\sigma^n \nabla_{z^n}\mathcal{L}_g^n$$
$$\mathcal{L}_g^n = ||F_T^n - F_1^n||_1 + ||\nabla F_T^n - \nabla F_1^n||_1 \quad (5)$$

where F_t^n is the predicted 2D motion displacement field at time t and denoising step n. w is the classifier-free guidance weight, and u is the motion self-guidance weight.

6.3. Interactive dynamics from a single image

As shown in Davis *et al*. [22], the image-space motion spectrum from an observed video of an oscillating object, under certain assumptions, is proportional to the projections of vibration mode shapes of that object, and thus a spectral volume can be interpreted as an *image-space modal basis*. The modal shapes capture underlying oscillation dynamics

Method	Image Synthesis FID	KID	Video Synthesis FVD	FVD_{32}	DTFVD	$DTFVD_{32}$
TATS [34]	65.8	1.67	265.6	419.6	22.6	40.7
Stochastic I2V [27]	68.3	3.12	253.5	320.9	16.7	41.7
MCVD [91]	63.4	2.97	208.6	270.4	19.5	53.9
LFDM [65]	47.6	1.70	187.5	254.3	13.0	45.6
DMVFN [47]	37.9	1.09	206.5	316.3	11.2	54.5
Endo *et al*. [29]	10.4	0.19	166.0	231.6	5.35	65.1
Holynski *et al*. [45]	11.2	0.20	179.0	253.7	7.23	46.8
Ours	**4.03**	**0.08**	**47.1**	**62.9**	**2.53**	**6.75**

Table 1. **Quantitative comparisons on the test set.** We report both image synthesis and video synthesis quality. Here, KID is scaled by 100. Lower is better for all error. See Sec. 7.4 for descriptions of baselines and error metrics.

of the object at different frequencies, and hence can be used to simulate the object's response to a user-defined force such as poking or pulling. Therefore, we adopt the modal analysis technique from prior work [22, 68], which assumes that the motion of an object can be explained by the superposition of a set of harmonic oscillators. This allows us to write the image-space 2D motion displacement field for the object's physical response as a weighted sum of Fourier spectrum coefficients S_{f_j} modulated by the state of complex modal coordinates $\mathbf{q}_{f_j,t}$ at each simulated time step t:

$$F_t(\mathbf{p}) = \sum_{f_j} S_{f_j}(\mathbf{p}) \mathbf{q}_{f_j,t} \tag{6}$$

We simulate the state of the modal coordinates $\mathbf{q}_{f_j,t}$ via a forward Euler method applied to the equations of motion for a decoupled mass-spring-damper system (in modal space) [22, 23, 68]. We refer readers to the original work for a full derivation. Note that our method produces an interactive scene from a *single picture*, whereas these prior methods required a video as input.

7. Experiments

7.1. Implementation details

We use an LDM [72] as the backbone for predicting spectral volumes, for which we use a variational auto-encoder (VAE) with a continuous latent space of dimension 4. We train the VAE with an L_1 reconstruction loss, a multi-scale gradient consistency loss [53–55], and a KL-divergence loss with respective weights of $1, 0.2, 10^{-6}$. We train the same 2D U-Net used in the original LDM work to perform iterative denosing with a simple MSE loss [43], and adopt the attention layers from [40] for frequency-coordinated denoising. For quantitative evaluation, we train the VAE and LDM on images of size 256×160, which takes around 6 days to converge using 16 Nvidia A100 GPUs. For our main quantitative and qualitative results, we run the motion diffusion model with DDIM [84] for 250 steps. We also show generated videos of

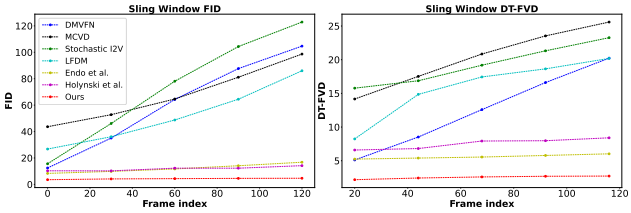


Figure 6. **Sliding Window FID and DT-FVD.** We show sliding window FID of window size 30 frames, and DT-FVD of size 16 frames, for videos generated by different methods.

up to a resolution of 512×288, created by fine-tuning our models on corresponding higher resolution data.

We adopt ResNet-34 [38] as feature extractor in our IBR module. Our image synthesis network is based on a co-modulation StyleGAN architecture, which is a prior conditional image generation and inpainting model [56, 108]. Our rendering module runs in real-time at 25FPS on a Nvidia V100 GPU during inference. Additionally, we adopt the universal guidance [3] to produce seamless looping videos, where we set weights $w = 1.75, u = 200$, and 500 DDIM steps with 2 self-recurrence iterations.

7.2. Data and baselines

Data. Since our focus is on natural scenes exhibiting oscillatory motion such as trees, flowers, and candles moving in the wind, we collect and process a set of 3,015 videos of such phenomena from online sources as well as from our own captures, where we withhold 10% of the videos for testing and use the remainder for training. To generate ground truth spectral volumes for training our motion prediction module, we found the choice of optical flow method to be crucial. In particular, we observed that deep-learning based flow estimators tend to produce over-smoothed flow fields. Instead, we apply a coarse-to-fine image pyramid-based optical flow algorithm [10, 60] between selected starting image and every future frame within a video sequence. We treat every 10th frame from each training video as a starting image and generate corresponding ground truth spectral volumes using the following 149 frames. We filter out samples with incorrect motion estimates or significant camera motion by removing examples with an average flow motion magnitude >8 pixels, or where all pixels have an average motion magnitude larger than one pixel. In total, our data consists of more than 150K samples of image-motion pairs.

7.3. Metrics

Baselines. We compare our approach to several recent single-image animation and video prediction methods. Both Endo *et al*. [29] and DMVFN [47] predict instantaneous 2D motion fields and future frames in an auto-regressive manner. We also compare with Holynsky *et al*. [45] which animate

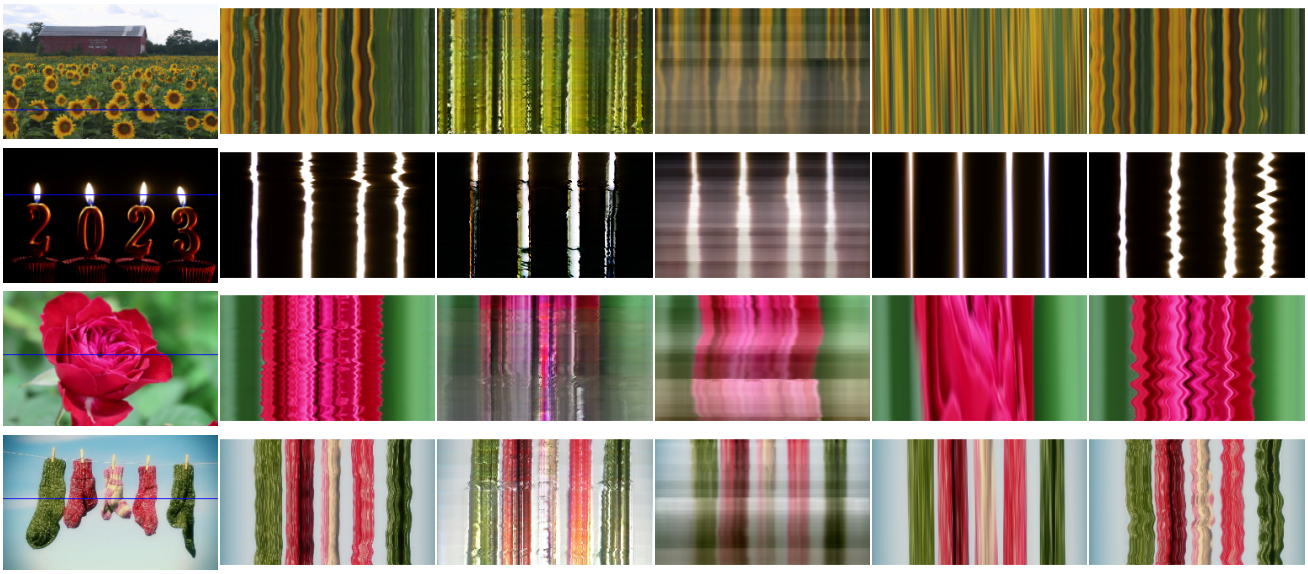


Figure 7. X-t **slices of videos generated by different approaches.** From left to right: input image and corresponding X-t video slices from the ground truth video, from videos generated by three baselines [27, 29, 91], and finally videos generated by our approach.

a picture through predicted Eulerian Motion Fields. Other recent work such as Stochastic Image-to-Video (Stochastic-I2V) [27], TATS [34], and MCVD [91] adopt either VAEs, temporal transformers, or diffusion models to directly predict video frames. LFDM [65] predicts flow fields in latent space with a diffusion model, then uses those flow fields to warp the encoded input image, generating future frames via a decoder. For methods that predict videos of short length, we apply them autoregressively to generate longer videos by taking the last output frame and using it as the input to another round of generation until the video reaches a length of 150 frames. We train all the above methods on our data using their respective open-source implementations[1].

We evaluate the quality of the videos generated by our approach and by prior baselines in two main ways. First, we evaluate the quality of individual synthesized frames using metrics designed for image synthesis tasks. We adopt the Fréchet Inception Distance (FID) [41] and Kernel Inception Distance (KID) [5] to measure the average distance between the distribution of generated frames and the distribution of ground truth frames.

Second, to evaluate the quality and temporal coherence of synthesized videos, we adopt the Fréchet Video Distance [90] with window size 16 (FVD) and 32 (FVD_{32}), based on an I3D model [11] trained on the Human Kinetics datasets [51]. To more faithfully reflect synthesis quality for the natural oscillation motions we seek to generate, we also adopt the Dynamic Texture Frechet Video Distance proposed by Dorkenwald *et al*. [27], which measures the distance from videos of window size 16 (DTFVD) and size 32 (DTFVD_{32}), using a I3D model trained on the Dynamic Textures Database [36], a dataset consisting primarily of natural motion textures.

We further use a sliding window FID of a window size of 30 frames, and a sliding window DTFVD with window size 16 frames, as proposed by [56, 59], to measure how generated video quality degrades over time.

For all the methods, we evaluate each error metric on videos generated without performing temporal interpolation, at 256×128 resolution.

[1]We use open-source reimplementation from Fan *et al*. [81] for the method of Holynsky *et al*. [45].

Method	Image Synthesis FID	KID	Video Synthesis FVD	FVD_{32}	DTFVD	DTFVD_{32}
$K = 4$	3.97	0.08	60.3	78.4	3.12	8.59
$K = 8$	3.95	0.07	52.1	68.7	2.71	7.37
$K = 24$	4.09	0.08	48.2	65.1	2.50	6.94
Scale w/ resolution	4.53	0.09	62.7	80.1	3.16	8.19
Independent pred.	4.00	0.08	52.5	71.3	2.70	7.40
Volume pred.	4.74	0.09	53.7	71.1	2.83	7.79
Average splat	4.52	0.10	51.4	68.9	2.83	7.44
Baseline splat [45]	4.25	0.09	49.5	66.8	2.83	7.27
Full ($K = 16$)	4.03	0.08	47.1	62.9	2.53	6.75

Table 2. **Ablation study.** Please see Sec. 7.6 for the details of the different configurations.

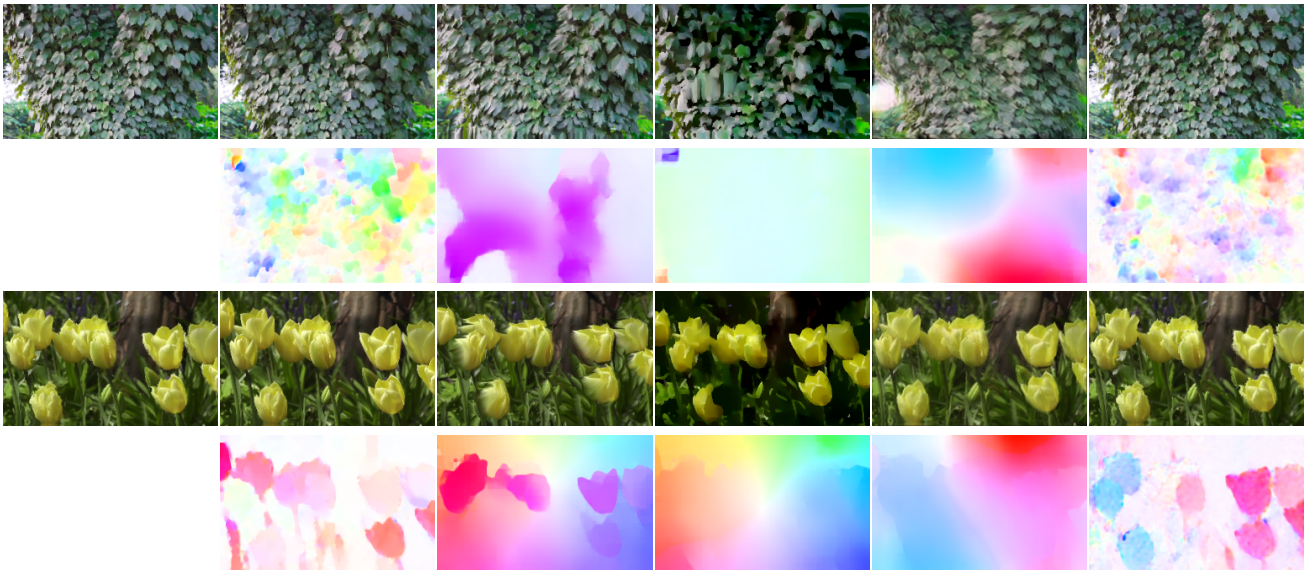


Figure 8. **Visual comparisons of generated future frames and corresponding motion fields.** By inspecting differences with a reference image from the ground truth video, we observe that our approach produces more realistic textures and motions compared with baselines.

7.4. Quantitative results

Table 1 shows quantitative comparisons between our approach and baselines on our test set of unseen video clips. Our approach significantly outperforms prior single-image animation baselines in terms of both image and video synthesis quality. Specifically, our much lower FVD and DT-FVD distances suggest that the videos generated by our approach are more realistic and more temporally coherent. Further, Fig. 6 shows the sliding window FID and sliding window DT-FVD distances of generated videos from different methods. Thanks to the global spectral volume representation, videos generated by our approach are more temporally consistent and do not suffer from drift or degradation over time.

7.5. Qualitative results

We visualize qualitative comparisons between videos generated by our approach and by baselines in two ways. First, we show spatio-temporal X-t slices of the generated videos, a standard way of visualizing small or subtle motions in a video [93]. As shown in Fig. 7, our generated video dynamics more strongly resemble the motion patterns observed in the corresponding real reference videos (second column), compared to other methods. Baselines such as Stochastic I2V [27] and MCVD [91] fail to model both appearance and motion realistically over time. Endo *et al*. [29] produces video frames with fewer artifacts but exhibits over-smooth or non-oscillation motions.

We also qualitatively compare the quality of individual generated frames and motions across different methods by visualizing the predicted image $\hat{I}_t$ and its corresponding motion displacement field at time $t = 128$. Fig. 8 shows that the frames generated by our approach exhibit fewer artifacts and distortions compared to other methods, and our corresponding 2D motion fields most resemble the reference displacement fields estimated from the corresponding real videos. In contrast, the background content generated by other methods tend to drift, as shown in the flow visualizations in the even-numbered rows. Moreover, the video frames generated by other methods exhibit significant color distortion or ghosting artifacts, suggesting that the baselines are less stable when generating videos with long time duration.

7.6. Ablation study

We conduct an ablation study to validate the major design choices in our motion prediction and rendering modules, comparing our full configuration with different variants. Specifically, we evaluate results using different numbers of frequency bands $K = 4$, 8, 16, and 24. We observe that increasing the number of frequency bands improves video prediction quality, but the improvement is marginal when using more than 16 frequencies. Next, we remove adaptive frequency normalization from the ground truth spectral volumes, and instead just scale them based on input image width and height (*Scale w/ resolution*). Additionally, we remove the frequency coordinated-denoising module (*Independent pred.*), or replace it with a simpler module where a tensor volume of $4K$ channel spectral volumes are predicted jointly via a standard 2D U-net diffusion model (*Volume pred.*). Finally, we compare results where we render video

frames using average splatting (*Average splat*), or use a baseline rendering method that applies softmax splatting over single-scale features subject to learnable weights used in Holynski *et al*. [45] (*Baseline splat*). From Table 2, we observe that all simpler or alternative configurations lead to worse performance compared with our full model.

8. Discussion and conclusion

Limitations. Since our approach only predicts spectral volumes at lower frequencies, it might fail to model general non-oscillating motions or high-frequency vibrations such as those of musical instruments. Furthermore, the quality of our generated videos relies on the quality of the motion trajectories. Thus, we observed that animation quality can degrade if the motion in the videos consists of large displacements. Moreover, since our approach is based on image-based rendering from input pixels, the animation quality can also degrade if the generated videos require the creation of large amounts of content unseen in the input frame.

Conclusion. We present a new approach for modeling natural oscillation dynamics from a single still picture. Our image-space motion prior is represented with spectral volumes [23], a frequency representation of per-pixel motion trajectories, which we find to be highly suitable for prediction with diffusion models, and which we learn from collections of real world videos. The spectral volumes are predicted using our frequency-coordinated latent diffusion model and are used to animate future video frames using a neural image-based rendering module. We show that our approach produces photo-realistic animations from a single picture and significantly outperforms prior baseline methods, and that it can enable other downstream applications such as creating interactive animations.

Acknowledgements. We thank Abe Davis, Rick Szeliski, Andrew Liu, Boyang Deng, Qianqian Wang, Xuan Luo, and Lucy Chai for fruitful discussions and helpful comments.

References

[1] Aseem Agarwala, Ke Colin Zheng, Chris Pal, Maneesh Agrawala, Michael Cohen, Brian Curless, David Salesin, and Richard Szeliski. Panoramic video textures. In *ACM SIGGRAPH 2005 Papers*, pages 821–827. 2005.

[2] Hyemin Ahn, Esteve Valls Mascaro, and Dongheui Lee. Can we use diffusion probabilistic models for 3d motion prediction? *arXiv preprint arXiv:2302.14503*, 2023.

[3] Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Soumyadip Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 843–852, 2023.

[4] Hugo Bertiche, Niloy J Mitra, Kuldeep Kulkarni, Chun-Hao P Huang, Tuanfeng Y Wang, Meysam Madadi, Sergio Escalera, and Duygu Ceylan. Blowing in the wind: Cyclenet for human cinemagraphs from still images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 459–468, 2023.

[5] Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. *arXiv preprint arXiv:1801.01401*, 2018.

[6] Andreas Blattmann, Timo Milbich, Michael Dorkenwald, and Björn Ommer. ipoke: Poking a still image for controlled stochastic video synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14707–14717, 2021.

[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] Richard Strong Bowen, Richard Tucker, Ramin Zabih, and Noah Snavely. Dimensions of motion: Monocular prediction through flow subspaces. In *2022 International Conference on 3D Vision (3DV)*, pages 454–464. IEEE, 2022.

[9] Tim Brooks, Janne Hellsten, Miika Aittala, Ting-Chun Wang, Timo Aila, Jaakko Lehtinen, Ming-Yu Liu, Alexei Efros, and Tero Karras. Generating long videos of dynamic scenes. *Advances in Neural Information Processing Systems*, 35:31769–31781, 2022.

[10] Thomas Brox, Andrés Bruhn, Nils Papenberg, and Joachim Weickert. High accuracy optical flow estimation based on a theory for warping. In *Proc. European Conf. on Computer Vision (ECCV)*, pages 25–36. Springer, 2004.

[11] 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.

[12] Dan Casas, Marco Volino, John Collomosse, and Adrian Hilton. 4d video textures for interactive character appearance. In *Computer Graphics Forum*, volume 33, pages 371–380. Wiley Online Library, 2014.

[13] Antoni B Chan and Nuno Vasconcelos. Mixtures of dynamic textures. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 1, pages 641–647. IEEE, 2005.

[14] Antoni B Chan and Nuno Vasconcelos. Classifying video with kernel dynamic textures. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–6. IEEE, 2007.

[15] Antoni B Chan and Nuno Vasconcelos. Modeling, clustering, and segmenting video with mixtures of dynamic textures. *IEEE transactions on pattern analysis and machine intelligence*, 30(5):909–926, 2008.

[16] Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023.

[17] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11315–11325, 2022.

[18] Tsai-Shien Chen, Chieh Hubert Lin, Hung-Yu Tseng, Tsung-Yi Lin, and Ming-Hsuan Yang. Motion-conditioned diffusion model for controllable video synthesis. *arXiv preprint arXiv:2304.14404*, 2023.

[19] Xin Chen, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, and Gang Yu. Executing your commands via motion diffusion in latent space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18000–18010, 2023.

[20] Yung-Yu Chuang, Dan B Goldman, Ke Colin Zheng, Brian Curless, David H Salesin, and Richard Szeliski. Animating pictures with stochastic motion textures. In *ACM SIGGRAPH 2005 Papers*, pages 853–860. 2005.

[21] Vincent C Couture, Michael S Langer, and Sebastien Roy. Omnistereo video textures without ghosting. In *2013 International Conference on 3D Vision-3DV 2013*, pages 64–70. IEEE, 2013.

[22] Abe Davis, Justin G Chen, and Frédo Durand. Image-space modal bases for plausible manipulation of objects in video. *ACM Transactions on Graphics (TOG)*, 34(6):1–7, 2015.

[23] Myers Abraham Davis. *Visual vibration analysis*. PhD thesis, Massachusetts Institute of Technology, 2016.

[24] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

[25] Julien Diener, Mathieu Rodriguez, Lionel Baboud, and Lionel Reveret. Wind projection basis for real-time animation of trees. In *Computer graphics forum*, volume 28, pages 533–540. Wiley Online Library, 2009.

[26] Gianfranco Doretto, Alessandro Chiuso, Ying Nian Wu, and Stefano Soatto. Dynamic textures. *International journal of computer vision*, 51:91–109, 2003.

[27] Michael Dorkenwald, Timo Milbich, Andreas Blattmann, Robin Rombach, Konstantinos G. Derpanis, and Bjorn Ommer. Stochastic image-to-video synthesis using cinns. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3742–3753, June 2021.

[28] Yuming Du, Robin Kips, Albert Pumarola, Sebastian Starke, Ali Thabet, and Artsiom Sanakoyeu. Avatars grow legs: Generating smooth human motion from sparse tracking inputs with diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 481–490, 2023.

[29] Yuki Endo, Yoshihiro Kanamori, and Shigeru Kuriyama. Animating landscape: Self-supervised learning of decoupled motion and appearance for single-image video synthesis. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2019)*, 38(6):175:1–175:19, 2019.

[30] Dave Epstein, Allan Jabri, Ben Poole, Alexei A Efros, and Aleksander Holynski. Diffusion self-guidance for controllable image generation. *arXiv preprint arXiv:2306.00986*, 2023.

[31] Matthew Flagg, Atsushi Nakazawa, Qiushuang Zhang, Sing Bing Kang, Young Kee Ryu, Irfan Essa, and James M Rehg. Human video textures. In *Proceedings of the 2009 symposium on Interactive 3D graphics and games*, pages 199–206, 2009.

[32] Jean-Yves Franceschi, Edouard Delasalles, Mickaël Chen, Sylvain Lamprier, and Patrick Gallinari. Stochastic latent residual video prediction. In *International Conference on Machine Learning*, pages 3233–3246. PMLR, 2020.

[33] Ruohan Gao, Bo Xiong, and Kristen Grauman. Im2Flow: Motion hallucination from static images for action recognition. In *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2018.

[34] Songwei Ge, Thomas Hayes, Harry Yang, Xi Yin, Guan Pang, David Jacobs, Jia-Bin Huang, and Devi Parikh. Long video generation with time-agnostic vqgan and time-sensitive transformer. *arXiv preprint arXiv:2204.03638*, 2022.

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

[36] Isma Hadji and Richard P Wildes. A new large scale dynamic texture dataset with application to convnet understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 320–335, 2018.

[37] Zekun Hao, Xun Huang, and Serge Belongie. Controllable video generation with sparse trajectories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7854–7863, 2018.

[38] 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.

[39] Yingqing He, Menghan Xia, Haoxin Chen, Xiaodong Cun, Yuan Gong, Jinbo Xing, Yong Zhang, Xintao Wang, Chao Weng, Ying Shan, et al. Animate-a-story: Storytelling with retrieval-augmented video generation. *arXiv preprint arXiv:2307.06940*, 2023.

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

[41] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[42] 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.

[43] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[44] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[45] Aleksander Holynski, Brian L Curless, Steven M Seitz, and Richard Szeliski. Animating pictures with Eulerian motion fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5810–5819, 2021.

[46] Tobias Hoppe, Arash Mehrjou, Stefan Bauer, Didrik Nielsen, and Andrea Dittadi. Diffusion models for video prediction

and infilling. *Trans. Mach. Learn. Res.*, 2022, 2022.

[47] Xiaotao Hu, Zhewei Huang, Ailin Huang, Jun Xu, and Shuchang Zhou. A dynamic multi-scale voxel flow network for video prediction. *ArXiv*, abs/2303.09875, 2023.

[48] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 694–711. Springer, 2016.

[49] Hitoshi Kanda and Jun Ohya. Efficient, realistic method for animating dynamic behaviors of 3d botanical trees. In *2003 International Conference on Multimedia and Expo. ICME'03. Proceedings (Cat. No. 03TH8698)*, volume 2, pages II–89. IEEE, 2003.

[50] Johanna Karras, Aleksander Holynski, Ting-Chun Wang, and Ira Kemelmacher-Shlizerman. Dreampose: Fashion image-to-video synthesis via stable diffusion. *arXiv preprint arXiv:2304.06025*, 2023.

[51] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

[52] Alex X Lee, Richard Zhang, Frederik Ebert, Pieter Abbeel, Chelsea Finn, and Sergey Levine. Stochastic adversarial video prediction. *arXiv preprint arXiv:1804.01523*, 2018.

[53] Zhengqi Li, Tali Dekel, Forrester Cole, Richard Tucker, Noah Snavely, Ce Liu, and William T Freeman. Learning the depths of moving people by watching frozen people. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4521–4530, 2019.

[54] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2041–2050, 2018.

[55] Zhengqi Li, Qianqian Wang, Forrester Cole, Richard Tucker, and Noah Snavely. Dynibar: Neural dynamic image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4273–4284, 2023.

[56] Zhengqi Li, Qianqian Wang, Noah Snavely, and Angjoo Kanazawa. Infinitenature-zero: Learning perpetual view generation of natural scenes from single images. In *European Conference on Computer Vision*, pages 515–534. Springer, 2022.

[57] Jing Liao, Mark Finch, and Hugues Hoppe. Fast computation of seamless video loops. *ACM Transactions on Graphics (TOG)*, 34(6):1–10, 2015.

[58] Zicheng Liao, Neel Joshi, and Hugues Hoppe. Automated video looping with progressive dynamism. *ACM Transactions on Graphics (TOG)*, 32(4):1–10, 2013.

[59] Andrew Liu, Richard Tucker, Varun Jampani, Ameesh Makadia, Noah Snavely, and Angjoo Kanazawa. Infinite nature: Perpetual view generation of natural scenes from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14458–14467, 2021.

[60] Ce Liu. *Beyond pixels: exploring new representations and applications for motion analysis*. PhD thesis, Massachusetts Institute of Technology, 2009.

[61] Zhengxiong Luo, Dayou Chen, Yingya Zhang, Yan Huang, Liang Wang, Yujun Shen, Deli Zhao, Jingren Zhou, and Tieniu Tan. Videofusion: Decomposed diffusion models for high-quality video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10209–10218, 2023.

[62] Aniruddha Mahapatra and Kuldeep Kulkarni. Controllable animation of fluid elements in still images. In *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2022.

[63] Arun Mallya, Ting-Chun Wang, and Ming-Yu Liu. Implicit warping for animation with image sets. *Advances in Neural Information Processing Systems*, 35:22438–22450, 2022.

[64] Medhini Narasimhan, Shiry Ginosar, Andrew Owens, Alexei A Efros, and Trevor Darrell. Strumming to the beat: Audio-conditioned contrastive video textures. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3761–3770, 2022.

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

[66] Simon Niklaus and Feng Liu. Softmax splatting for video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5437–5446, 2020.

[67] Shin Ota, Machiko Tamura, Kunihiko Fujita, T Fujimoto, K Muraoka, and Norishige Chiba. 1/f/sup/spl beta//noise-based real-time animation of trees swaying in wind fields. In *Proceedings Computer Graphics International 2003*, pages 52–59. IEEE, 2003.

[68] Automne Petitjean, Yohan Poirier-Ginter, Ayush Tewari, Guillaume Cordonnier, and George Drettakis. Modalnerf: Neural modal analysis and synthesis for free-viewpoint navigation in dynamically vibrating scenes. In *Computer Graphics Forum*, volume 42, 2023.

[69] Silvia L. Pintea, Jan C. van Gemert, and Arnold W. M. Smeulders. Déjà vu: Motion prediction in static images. In *Proc. European Conf. on Computer Vision (ECCV)*, 2014.

[70] Sigal Raab, Inbal Leibovitch, Guy Tevet, Moab Arar, Amit H Bermano, and Daniel Cohen-Or. Single motion diffusion. *arXiv preprint arXiv:2302.05905*, 2023.

[71] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

[72] 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.

[73] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022.

[74] Payam Saisan, Gianfranco Doretto, Ying Nian Wu, and Stefano Soatto. Dynamic texture recognition. In *Proceedings*

of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, volume 2, pages II–II. IEEE, 2001.

[75] Saurabh Saxena, Charles Herrmann, Junhwa Hur, Abhishek Kar, Mohammad Norouzi, Deqing Sun, and David J. Fleet. The surprising effectiveness of diffusion models for optical flow and monocular depth estimation, 2023.

[76] Arno Schödl, Richard Szeliski, David H Salesin, and Irfan Essa. Video textures. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 489–498, 2000.

[77] Mikio Shinya and Alain Fournier. Stochastic motion—motion under the influence of wind. *Computer Graphics Forum*, 11(3), 1992.

[78] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. Animating arbitrary objects via deep motion transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2377–2386, 2019.

[79] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. *Advances in neural information processing systems*, 32, 2019.

[80] Aliaksandr Siarohin, Oliver J Woodford, Jian Ren, Menglei Chai, and Sergey Tulyakov. Motion representations for articulated animation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13653–13662, 2021.

[81] Chen Qian Kwan-Yee Lin Hongsheng Li Siming Fan, Jingtan Piao. Simulating fluids in real-world still images. *arXiv preprint*, arXiv:2204.11335, 2022.

[82] Ivan Skorokhodov, Sergey Tulyakov, and Mohamed Elhoseiny. Stylegan-v: A continuous video generator with the price, image quality and perks of stylegan2. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3626–3636, 2022.

[83] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015.

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

[85] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[86] Jos Stam. *Multi-scale stochastic modelling of complex natural phenomena*. PhD thesis, 1995.

[87] Jos Stam. Stochastic dynamics: Simulating the effects of turbulence on flexible structures. *Computer Graphics Forum*, 16(3), 1997.

[88] Ryusuke Sugimoto, Mingming He, Jing Liao, and Pedro V Sander. Water simulation and rendering from a still photograph. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–9, 2022.

[89] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022.

[90] Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018.

[91] Vikram Voleti, Alexia Jolicoeur-Martineau, and Christopher Pal. Mcvd: Masked conditional video diffusion for prediction, generation, and interpolation. In *(NeurIPS) Advances in Neural Information Processing Systems*, 2022.

[92] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *Neural Information Processing Systems*, 2016.

[93] Neal Wadhwa, Michael Rubinstein, Frédo Durand, and William T Freeman. Phase-based video motion processing. *ACM Transactions on Graphics (ToG)*, 32(4):1–10, 2013.

[94] Jacob Walker, Carl Doersch, Abhinav Gupta, and Martial Hebert. An uncertain future: Forecasting from static images using variational autoencoders. In *Proc. European Conf. on Computer Vision (ECCV)*, 2016.

[95] Jacob Walker, Abhinav Gupta, and Martial Hebert. Dense optical flow prediction from a static image. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2443–2451, 2015.

[96] 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. *arXiv preprint arXiv:2306.02018*, 2023.

[97] Yaohui Wang, Di Yang, Francois Bremond, and Antitza Dantcheva. Latent image animator: Learning to animate images via latent space navigation. *arXiv preprint arXiv:2203.09043*, 2022.

[98] Frederik Warburg, Ethan Weber, Matthew Tancik, Aleksander Holynski, and Angjoo Kanazawa. Nerfbusters: Removing ghostly artifacts from casually captured nerfs. *arXiv preprint arXiv:2304.10532*, 2023.

[99] Daniel Watson, William Chan, Ricardo Martin-Brualla, Jonathan Ho, Andrea Tagliasacchi, and Mohammad Norouzi. Novel view synthesis with diffusion models. *arXiv preprint arXiv:2210.04628*, 2022.

[100] Chung-Yi Weng, Brian Curless, and Ira Kemelmacher-Shlizerman. Photo wake-up: 3d character animation from a single photo. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5908–5917, 2019.

[101] Jamie Wynn and Daniyar Turmukhambetov. DiffusioNeRF: Regularizing Neural Radiance Fields with Denoising Diffusion Models. In *CVPR*, 2023.

[102] Tianfan Xue, Jiajun Wu, Katherine L Bouman, and William T Freeman. Visual dynamics: Stochastic future generation via layered cross convolutional networks. *Trans. Pattern Analysis and Machine Intelligence*, 41(9):2236–2250, 2019.

[103] Shengming Yin, Chenfei Wu, Jian Liang, Jie Shi, Houqiang Li, Gong Ming, and Nan Duan. Dragnuwa: Fine-grained control in video generation by integrating text, image, and trajectory. *arXiv preprint arXiv:2308.08089*, 2023.

[104] Sihyun Yu, Kihyuk Sohn, Subin Kim, and Jinwoo Shin. Video probabilistic diffusion models in projected latent

space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18456–18466, 2023.

[105] Mingyuan Zhang, Xinying Guo, Liang Pan, Zhongang Cai, Fangzhou Hong, Huirong Li, Lei Yang, and Ziwei Liu. Remodiffuse: Retrieval-augmented motion diffusion model. *arXiv preprint arXiv:2304.01116*, 2023.

[106] 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.

[107] Jian Zhao and Hui Zhang. Thin-plate spline motion model for image animation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3657–3666, 2022.

[108] Shengyu Zhao, Jonathan Cui, Yilun Sheng, Yue Dong, Xiao Liang, Eric I Chang, and Yan Xu. Large scale image completion via co-modulated generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2021.

[109] 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.