Align your Latents: High-Resolution Video Synthesis with Latent Diffusion Models

Andreas Blattmann[1] *,† Robin Rombach[1] *,† Huan Ling[2,3,4] * Tim Dockhorn[2,3,5] *,†
Seung Wook Kim[2,3,4] Sanja Fidler[2,3,4] Karsten Kreis[2]

[1]LMU Munich [2]NVIDIA [3]Vector Institute [4]University of Toronto [5]University of Waterloo

"A horse galloping through van Gogh's 'Starry Night'"

"A dog wearing virtual reality goggles in sunset, 4k, high resolution"

"The Orient Express driving through a fantasy landscape, animated oil on canvas"

Figure 1. **Video LDM samples**. *Top:* Text-to-Video generation. *Bottom:* 512 × 1024 resolution real driving scene video generation.

Abstract

Latent Diffusion Models (LDMs) enable high-quality image synthesis while avoiding excessive compute demands by training a diffusion model in a compressed lower-dimensional latent space. Here, we apply the LDM paradigm to high-resolution video generation, a particularly resource-intensive task. We first pre-train an LDM on images only; then, we turn the image generator into a video generator by introducing a temporal dimension to the latent space diffusion model and fine-tuning on encoded image sequences, i.e., videos. Similarly, we temporally align diffusion model upsamplers, turning them into temporally consistent video super resolution models. We focus on two relevant real-world applications: Simulation of in-the-wild driving data and creative content creation with text-to-video modeling. In particular, we validate our ***Video LDM*** *on real driving videos of resolution* 512 × 1024*, achieving state-of-the-art performance. Furthermore, our approach can easily leverage off-the-shelf pre-trained image LDMs, as we only need to train a temporal alignment model in that case. Doing so, we turn the publicly available, state-of-the-art text-to-image LDM Stable Diffusion into an efficient and expressive text-to-video model with resolution up to* 1280 × 2048*. We show that the temporal layers trained in this way generalize to different fine-tuned text-to-image LDMs. Utilizing this property, we show the first results for personalized text-to-video generation, opening exciting directions for future content creation. Project page:* https://nv-tlabs.github.io/VideoLDM/

*Equal contribution.
†Andreas, Robin and Tim did the work during internships at NVIDIA.

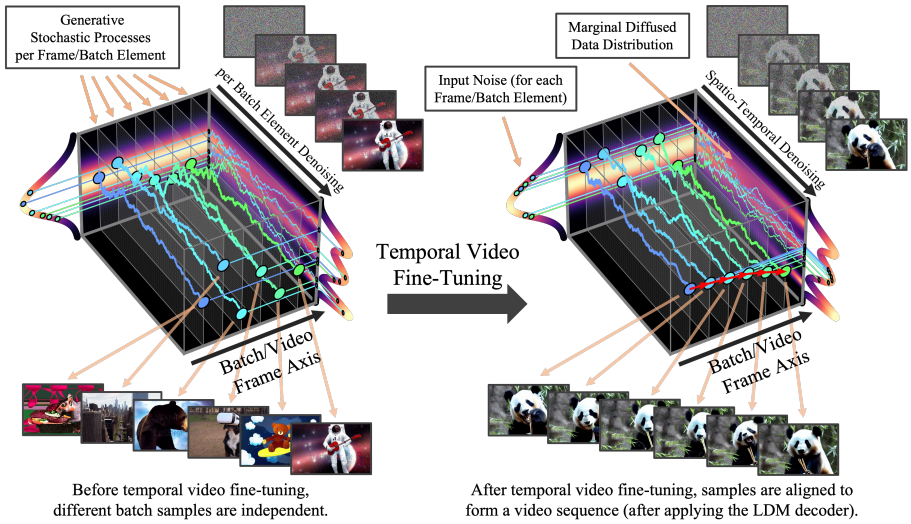

Figure 2. **Temporal Video Fine-Tuning.** We turn pre-trained image diffusion models into temporally consistent video generators. Initially, different samples of a batch synthesized by the model are independent. After temporal video fine-tuning, the samples are temporally aligned and form coherent videos. The stochastic generation process before and after fine-tuning is visualised for a diffusion model of a one-dim. toy distribution. For clarity, the figure corresponds to alignment in pixel space. In practice, we perform alignment in LDM's latent space and obtain videos after applying LDM's decoder (see Fig. 3). We also video fine-tune diffusion model upsamplers in pixel or latent space (Sec. 3.4).

1. Introduction

Generative models of images have received unprecedented attention, owing to recent breakthroughs in the underlying modeling methodology. The most powerful models today are built on generative adversarial networks [21, 38–40, 75], autoregressive transformers [15, 63, 105], and most recently diffusion models [10, 28, 29, 57, 58, 62, 65, 68, 79, 82]. Diffusion models (DMs) in particular have desirable advantages; they offer a robust and scalable training objective and are typically less parameter intensive than their transformer-based counterparts. However, while the image domain has seen great progress, *video* modeling has lagged behind—mainly due to the significant computational cost associated with training on video data, and the lack of large-scale, general, and publicly available video datasets. While there is a rich literature on video synthesis [1, 6, 8, 9, 17, 19, 22, 23, 32, 32, 37, 42, 44, 47, 51, 55, 59, 71, 78, 85, 91, 94, 97–99, 103, 106], most works, including previous video DMs [24, 31, 33, 93, 104], only generate relatively low-resolution, often short, videos. Here, we apply video models to real-world problems and generate high-resolution, long videos. Specifically, we focus on two relevant real-world video generation problems: (i) video synthesis of high-resolution real-word driving data, which has great potential as a simulation engine in the context of autonomous driving, and (ii) text-guided video synthesis for creative content generation; see Fig. 1.

To this end, we build on latent diffusion models (LDMs), which can reduce the heavy computational burden when training on high-resolution images [65]. We propose *Video LDMs* and extend LDMs to high-resolution *video* generation, a particularly compute-intensive task. In contrast to previous work on DMs for video generation [24, 31, 33, 93, 104], we first pre-train our Video LDMs on images only (or use available pre-trained image LDMs), thereby allowing us to leverage large-scale image datasets. We then transform the LDM image generator into a video generator by introducing a temporal dimension into the latent space DM and training only these temporal layers on encoded image sequences, *i.e.*, videos (Fig. 2), while fixing the pre-trained spatial layers. We similarly fine-tune LDM's decoder to achieve temporal consistency in pixel space (Fig. 3). To further enhance the spatial resolution, we also temporally align pixel-space and latent DM upsamplers [29], which are widely used for image super resolution [43, 65, 68, 69], turning them into temporally consistent video super resolution models. Building on LDMs, our method can generate globally coherent and long videos in a computationally and memory efficient manner. For synthesis at very high resolutions, the video upsampler only needs to operate locally, keeping training and computational requirements low. We ablate our method and test on 512×1024 real driving scene videos, achieving state-of-the-art video quality, and synthesize videos of several minutes length. We also video fine-tune a powerful, publicly available text-to-image LDM, *Stable Diffusion* [65], and turn it into an efficient and powerful text-to-video generator with resolution up to 1280×2048. Since we only need to train the temporal alignment layers in that case, we can use a relatively small training set of captioned videos. By transferring the trained temporal layers to differently fine-tuned text-to-image LDMs, we demonstrate personalized text-to-video generation for the first time. We hope our work opens new avenues for efficient digital content creation and autonomous driving simulation.

Contributions. *(i)* We present an efficient approach for training high-resolution, long-term consistent video generation models based on LDMs. Our key insight is to leverage pre-trained image DMs and turn them into video generators by inserting temporal layers that learn to align images in a temporally consistent manner (Figs. 2 and 3). *(ii)* We further temporally fine-tune super resolution DMs, which are ubiquitous in the literature. *(iii)* We achieve state-of-the-art high-resolution video synthesis performance on real driving scene videos, and we can generate multiple minute long

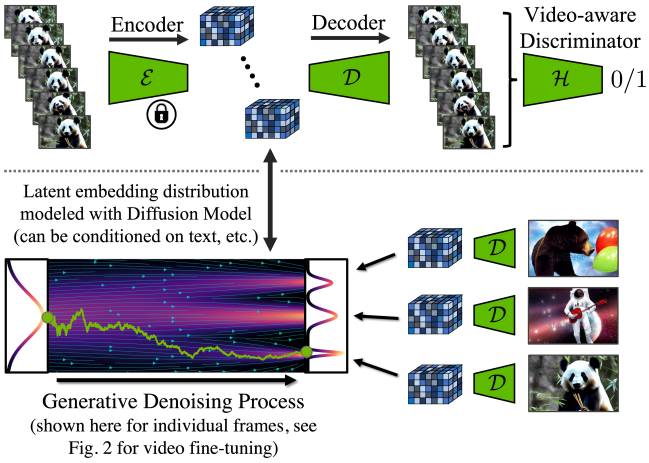

Figure 3. *Top:* During temporal decoder fine-tuning, we process video sequences with a frozen encoder, which processes frames independently, and enforce temporally coherent reconstructions across frames. We additionally employ a video-aware discriminator. *Bottom:* in LDMs, a diffusion model is trained in latent space. It synthesizes latent features, which are then transformed through the decoder into images. Note that the bottom visualization is for individual frames; see Fig. 2 for the video fine-tuning framework that generates temporally consistent frame sequences.

videos. *(iv)* We transform the publicly available *Stable Diffusion* text-to-image LDM into a powerful and expressive text-to-video LDM, and *(v)* show that the learned temporal layers can be combined with different image model checkpoints (*e.g.*, *DreamBooth* [66]).

2. Background

DMs [28, 79, 82] learn to model a data distribution $p_{\text{data}}(\mathbf{x})$ via *iterative denoising* and are trained with *denoising score matching* [28,34,50,79,81,82,92]: Given samples $\mathbf{x} \sim p_{\text{data}}$, *diffused* inputs $\mathbf{x}_\tau = \alpha_\tau \mathbf{x} + \sigma_\tau \boldsymbol{\epsilon},\ \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ are constructed; α_τ and σ_τ define a *noise schedule*, parameterized via a diffusion-time τ, such that the logarithmic signal-to-noise ratio $\lambda_\tau = \log(\alpha_\tau^2/\sigma_\tau^2)$ monotonically decreases. A denoiser model $\mathbf{f}_\theta$ (parameterized with learnable parameters θ) receives the diffused $\mathbf{x}_\tau$ as input and is optimized minimizing the denoising score matching objective

$$\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}, \tau\sim p_\tau, \boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\boldsymbol{I})} \left[\|\mathbf{y} - \mathbf{f}_\theta(\mathbf{x}_\tau; \mathbf{c}, \tau)\|_2^2 \right], \quad (1)$$

where $\mathbf{c}$ is optional conditioning information, such as a text prompt, and the target vector $\mathbf{y}$ is either the random noise $\boldsymbol{\epsilon}$ or $\mathbf{v} = \alpha_\tau \boldsymbol{\epsilon} - \sigma_\tau \mathbf{x}$. The latter objective (often referred to as $\mathbf{v}$-*prediction*) has been introduced in the context of progressive distillation [73] and empirically often yields faster convergence of the model (here, we use both objectives). Furthermore, p_τ is a uniform distribution over the diffusion time τ. The forward diffusion as well as the reverse generation process in diffusion models can be described via stochastic differential equations in a continuous-time framework [82] (see Figs. 2 and 3), but in practice a fixed discretization can be used [28]. The maximum diffusion time is generally chosen such that the input data is entirely perturbed into Gaussian random noise and an iterative generative denoising process that employs the learned denoiser $\mathbf{f}_\theta$ can be initialized from such Gaussian noise to synthesize novel data. Here, we use $p_\tau \sim \mathcal{U}\{0, 1000\}$ and rely on a *variance-preserving* noise schedule [82], for which $\sigma_\tau^2 = 1 - \alpha_\tau^2$ (see Appendices F and H for details).

Latent Diffusion Models (LDMs) [65] improve in computational and memory efficiency over pixel-space DMs by first training a compression model to transform input images $\mathbf{x}\sim p_{\text{data}}$ into a spatially lower-dimensional latent space of reduced complexity, from which the original data can be reconstructed at high fidelity. In practice, this approach is implemented with a regularized autoencoder, which reconstructs inputs $\mathbf{x}$ via an encoder module $\mathcal{E}$ and a decoder $\mathcal{D}$, such that the reconstruction $\hat{\mathbf{x}}{=}\mathcal{D}(\mathcal{E}(\mathbf{x}))\approx\mathbf{x}$ (Fig. 3). To ensure photorealistic reconstructions, an adversarial objective can be added to the autoencoder training [65], which is implemented using a patch-based discriminator [35]. A DM can then be trained in the compressed latent space and $\mathbf{x}$ in Eq. (1) is replaced by its latent representation $\mathbf{z}{=}\mathcal{E}(\mathbf{x})$. This latent space DM can be typically smaller in terms of parameter count and memory consumption compared to corresponding pixel-space DMs of similar performance.

3. Latent Video Diffusion Models

Here we describe how we *video fine-tune* pre-trained image LDMs (and DM upsamplers) for high-resolution video synthesis. We assume access to a dataset p_{data} of videos, such that $\mathbf{x} \in \mathbb{R}^{T\times 3\times \tilde{H}\times \tilde{W}}$, $\mathbf{x} \sim p_{\text{data}}$ is a sequence of T RGB frames, with height and width $\tilde{H}$ and $\tilde{W}$.

3.1. Turning Latent Image into Video Generators

Our key insight for efficiently training a video generation model is to re-use a pre-trained, fixed image generation model; an LDM parameterized by parameters θ. Formally, let us denote the neural network layers that comprise the image LDM and process inputs over the pixel dimensions as *spatial* layers l_θ^i, with layer index i. However, although such a model is able to synthesize individual frames at high quality, using it directly to render a video of T consecutive frames will fail, as the model has no temporal awareness. We thus introduce additional *temporal* neural network layers l_ϕ^i, which are interleaved with the existing *spatial* layers l_θ^i and learn to align individual frames in a temporally consistent manner. These L additional temporal layers $\{l_\phi^i\}_{i=1}^L$ define the *video-aware* temporal backbone of our model, and the full model $\mathbf{f}_{\theta,\phi}$ is thus the combination of the spatial and temporal layers; see Fig. 4 for a visualization.

We start from a frame-wise encoded input video $\mathcal{E}(\mathbf{x}) = \mathbf{z} \in \mathbb{R}^{T\times C\times H\times W}$, where C is the number of latent channels

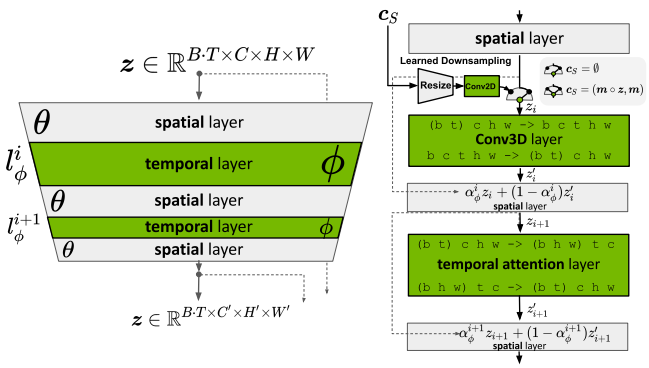

Figure 4. *Left:* We turn a pre-trained LDM into a video generator by inserting *temporal* layers that learn to align frames into temporally consistent sequences. During optimization, the image backbone θ remains fixed and only the parameters ϕ of the temporal layers l^i_ϕ are trained, *cf*. Eq. (2). *Right:* During training, the base model θ interprets the input sequence of length T as a batch of images. For the temporal layers l^i_ϕ, these batches are reshaped into video format. Their output $\mathbf{z}'$ is combined with the spatial output $\mathbf{z}$, using a learned merge parameter α. During inference, skipping the temporal layers (α^i_ϕ=1) yields the original image model. For illustration purposes, only a single U-Net Block is shown. B denotes batch size, T sequence length, C input channels and H and W the spatial dimensions of the input. $\mathbf{c}_S$ is optional context frame conditioning, when training prediction models (Sec. 3.2).

and H and W are the spatial latent dimensions. The spatial layers interpret the video as a batch of independent images (by shifting the temporal axis into the batch dimension), and for each *temporal mixing layer* l^i_ϕ, we reshape back to video dimensions as follows (using `einops` [64] notation):

$$\mathbf{z}' \leftarrow \texttt{rearrange}(\mathbf{z},\ \texttt{(b t) c h w} \rightarrow \texttt{b c t h w})$$
$$\mathbf{z}' \leftarrow l^i_\phi(\mathbf{z}', \mathbf{c})$$
$$\mathbf{z}' \leftarrow \texttt{rearrange}(\mathbf{z}',\ \texttt{b c t h w} \rightarrow \texttt{(b t) c h w}),$$

where we added the batch dimension `b` for clarity. In other words, the spatial layers treat all $B{\cdot}T$ encoded video frames independently in the batch dimension `b`, while the temporal layers $l^i_\phi(\mathbf{z}', \mathbf{c})$ process entire videos in a new temporal dimension `t`. Furthermore, $\mathbf{c}$ is (optional) conditioning information such as a text prompt. After each temporal layer, the output $\mathbf{z}'$ is combined with $\mathbf{z}$ as $\alpha^i_\phi \mathbf{z} + (1-\alpha^i_\phi)\mathbf{z}'$; $\alpha^i_\phi \in [0, 1]$ denotes a (learnable) parameter (also Appendix D).

In practice, we implement two different kinds of temporal mixing layers: (i) temporal attention and (ii) residual blocks based on 3D convolutions, *cf*. Fig. 4. We use sinusoidal embeddings [28, 89] to provide the model with a positional encoding for time.

Our video-aware temporal backbone is then trained using the same noise schedule as the underlying image model, and, importantly, we fix the spatial layers l^i_θ and *only* optimize the temporal layers l^i_ϕ via

$$\arg\min_\phi \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}, \tau\sim p_\tau, \boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\boldsymbol{I})} \left[\|\mathbf{y} - \mathbf{f}_{\theta,\phi}(\mathbf{z}_\tau; \mathbf{c}, \tau)\|_2^2\right], \quad (2)$$

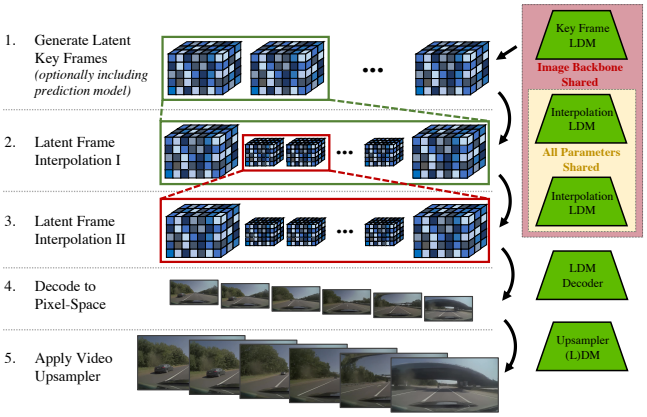

Figure 5. **Video LDM Stack.** We first generate sparse key frames. Then we temporally interpolate in two steps with the same interpolation model to achieve high frame rates. These operations are all based on latent diffusion models (LDMs) that share the same image backbone. Finally, the latent video is decoded to pixel space and optionally a video upsampler diffusion model is applied.

where $\mathbf{z}_\tau$ denotes diffused encodings $\mathbf{z} = \mathcal{E}(\mathbf{x})$. This way, we retain the native image generation capabilities by simply skipping the temporal blocks, *e.g.* by setting $\alpha^i_\phi = 1$ for each layer. A crucial advantage of our strategy is that huge image datasets can be used to pre-train the spatial layers, while the video data, which is often less widely available, can be utilized for focused training of the temporal layers.

3.1.1 Temporal Autoencoder Finetuning

Our video models build on pre-trained image LDMs. While this increases efficiency, the autoencoder of the LDM is trained on images only, causing flickering artifacts when encoding and decoding a temporally coherent sequence of images. To counteract this, we introduce additional temporal layers for the autoencoder's decoder, which we finetune on video data with a (patch-wise) temporal discriminator built from 3D convolutions, *cf*. Fig. 3. Note that the encoder remains unchanged from image training such that the image DM that operates in latent space on encoded video frames can be re-used. As demonstrated by computing reconstruction FVD [87] scores in Table 3, this step is critical for achieving good results.

3.2. Prediction Models for Long-Term Generation

Although the approach described in Sec. 3.1 is efficient for generating short video sequences, it reaches its limits when it comes to synthesizing very long videos. Therefore, we also train models as *prediction models* given a number of (first) S context frames. We implement this by introducing a temporal binary mask $\mathbf{m}_S$ which masks the $T - S$ frames the model has to predict, where T is the total sequence length as in Sec. 3.1. We feed this mask and the masked encoded video frames into the model for condition-

Figure 6. 1280×2048 resolution samples from our Stable Diffusion-based text-to-video LDM, including video fine-tuned upsampler. Prompts: "An astronaut flying in space, 4k, high resolution" and "Milk dripping into a cup of coffee, high definition, 4k".

ing. Specifically, the frames are encoded with LDM's image encoder $\mathcal{E}$, multiplied by the mask, and then fed (channel-wise concatenated with the masks) into the temporal layers l_ϕ^i after being processed with a learned downsampling operation, see Fig. 4. Let $\mathbf{c}_S = (\mathbf{m}_S \circ \mathbf{z}, \mathbf{m}_S)$ denote the concatenated spatial conditioning of masks and masked (encoded) images. Then, the objective from Eq. (2) reads

$$\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}, \mathbf{m}_S \sim p_S, \tau \sim p_\tau, \boldsymbol{\epsilon}} \left[\| \mathbf{y} - \mathbf{f}_{\theta,\phi}(\mathbf{z}_\tau; \mathbf{c}_S, \mathbf{c}, \tau) \|_2^2 \right], \quad (3)$$

where p_S represents the (categorical) mask sampling distribution. In practice, we learn prediction models that condition either on 0, 1 or 2 context frames, allowing for classifier-free guidance as discussed below.

During inference, for generating long videos, we can apply the sampling process iteratively, re-using the latest predictions as new context. The first initial sequence is generated by synthesizing a single context frame from the base image model and generating a sequence based on that; afterwards, we condition on two context frames to encode movement (details in Appendix). To stabilize this process, we found it beneficial to use *classifier-free diffusion guidance* [30], where we guide the model during sampling via

$$\mathbf{f}'_{\theta,\phi}(\mathbf{z}_\tau; \mathbf{c}_S) = \mathbf{f}_{\theta,\phi}(\mathbf{z}_\tau) + s \cdot (\mathbf{f}_{\theta,\phi}(\mathbf{z}_\tau; \mathbf{c}_S) - \mathbf{f}_{\theta,\phi}(\mathbf{z}_\tau)) \quad (4)$$

where $s \geq 1$ denotes the guidance scale and we dropped the explicit conditioning on τ and other information $\mathbf{c}$ for readability. We refer to this guidance as *context guidance*.

3.3. Temporal Interpolation for High Frame Rates

High-resolution video is characterized not only by high spatial resolution, but also by high temporal resolution, *i.e.*, a high frame rate. To achieve this, we divide the synthesis process for high-resolution video into two parts: The first is the process described in Sec. 3.1 and Sec. 3.2, which can generate *key frames* with large semantic changes, but (due to memory constraints) only at a relatively low frame rate. For the second part, we introduce an additional model whose task is to interpolate between given key frames. To implement this, we use the masking-conditioning mechanism introduced in Sec. 3.2. However, unlike the prediction task, we now mask the frames to be interpolated—otherwise, the mechanism remains the same, *i.e.*, the image model is refined into a video interpolation model. In our experiments, we predict three frames between two given key frames, thereby training a $T \rightarrow 4T$ interpolation model. To achieve even larger frame rates, we train the model simultaneously in the $T \rightarrow 4T$ and $4T \rightarrow 16T$ regimes (using videos with different fps), specified by binary conditioning.

Our training approach for prediction and interpolation models is inspired by recent works [24, 33, 93] that use similar masking techniques (also see Appendix C).

3.4. Temporal Fine-tuning of SR Models

Although the LDM mechanism already provides a good native resolution we aim to push this towards the megapixel range. We take inspiration from cascaded DMs [29] and use a DM to further scale up the Video LDM outputs by $4\times$. For our driving video synthesis experiments, we use a pixel-space DM [29] (Sec. 4.1) and scale to 512×1024; for our text-to-video models, we use an LDM upsampler [65] (Sec. 4.2) and scale to 1280×2048. We use noise augmentation with noise level conditioning [29, 68] and train the super resolution (SR) model $\mathbf{g}_{\theta,\phi}$ (on images or latents) via

$$\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}, (\tau,\tau_\gamma)\sim p_\tau, \boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\boldsymbol{I})} \left[\| \mathbf{y} - \mathbf{g}_{\theta,\phi}(\mathbf{x}_\tau; \mathbf{c}_{\tau_\gamma}, \tau_\gamma, \tau) \|_2^2 \right] \quad (5)$$

where $\mathbf{c}_{\tau_\gamma} = \alpha_{\tau_\gamma}\mathbf{x} + \sigma_{\tau_\gamma}\boldsymbol{\epsilon}$, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$, denotes a noisy low-resolution image given to the model via concatenation, and τ_γ the amount of noise added to the low-resolution image following the noise schedule α_τ, σ_τ.

Since upsampling video frames independently would result in poor temporal consistency, we also make this SR model video-aware. We follow the mechanism introduced in Sec. 3.1 with spatial layers l_θ^i and temporal layers l_ϕ^i and similarly video fine-tune the upscaler, conditioning on a low-resolution sequence of length T and concatenating low-resolution video images frame-by-frame. Since the upscaler operates locally, we conduct all upscaler training efficiently on patches only and later apply the model convolutionally.

Figure 7. 512×1024 resolution video modeling of real-world driving scenes with our Video LDM and video upsampler. *Top:* (Night time) **Driving Video Generation**. *Middle:* **Multimodal Driving Scenario Prediction**: We simulate two different scenarios given the same initial frame (red). *Bottom:* **Specific Driving Scenario Simulation**: We synthesize a scenario based on a manually designed, initial scene generated with a bounding box-conditioned Image LDM (yellow). More examples in the Appendix I.3.

Overall, we believe that the combination of an LDM with an upsampler DM is ideal for efficient high-resolution video synthesis. On the one hand, the main LDM component of our Video LDM leverages a computationally efficient, compressed latent space to perform all video modeling. This allows us to use large batch sizes and jointly encode more video frames, which benefits long-term video modeling, without excessive memory demands, as all video predictions and interpolations are carried out in latent space. On the other hand, the upsampler can be trained in an efficient patch-wise manner, therefore similarly saving computational resources and reducing memory consumption, and it also does not need to capture long-term temporal correlations due to the low-resolution conditioning. Therefore, no prediction and interpolation framework is required for this component. A model overview, bringing together all components from Sec. 3.1 to Sec. 3.4, is depicted in Fig. 5.

A discussion of related work can be found in Appendix C.

4. Experiments

Datasets. Since we focus on driving scene video generation as well as text-to-video, we use two corresponding datasets/models: *(i)* An in-house dataset of real driving scene (RDS) videos. The dataset consists of 683,060 videos of 8 seconds each at resolution 512×1024 ($H \times W$) and frame rate up to 30 fps. Furthermore, the videos have binary night/day labels, annotations for the number of cars in a scene ("crowdedness"), and a subset of the data also has car bounding boxes. *(ii)* We use the WebVid-10M [2] dataset to turn the publicly available *Stable Diffusion* Image LDM [65] into a Video LDM. WebVid-10M consists of 10.7M video-caption pairs with a total of 52K video hours. We resize the videos into resolution 320×512. *(iii)* Moreover, in Appendix I.2, we show experiments on the Mountain Biking dataset by Brooks et al. [6].

Evaluation Metrics. To evaluate our models, we use frame-wise Fréchet Inception Distance (FID) [26] as well as Fréchet Video Distance (FVD) [87]. Since FVD can be unreliable (discussed, for instance, by Brooks et al. [6]), we additionally perform human evaluation. For our text-to-video experiments, we also evaluate CLIP similarity (CLIP-SIM) [98] and (video) inception score (IS) (Appendix G).

Model Architectures and Sampling. Our Image LDMs are based on Rombach et al. [65]. They use convolutional encoders and decoders, and their latent space DM architecture build on the U-Net by Dhariwal et al. [10]. Our pixel-space upsampler DMs use the same Image DM backbone [10]. DM sampling is performed using DDIM [80] in all experiments.

Further architecture, training, evaluation, sampling and dataset details can be found in the Appendix.

4.1. High-Resolution Driving Video Synthesis

We train our Video LDM pipeline, including a $4\times$ pixel-space video upsampler, on the real driving scene (RDS) data. We condition on day/night labels and crowdedness, and randomly drop these labels during training to allow for classifier-free guidance and unconditional synthesis (we do not condition on bounding boxes here). Following the proposed training strategy above, we first train the image backbone LDM (spatial layers) on video frames independently, before we then train the temporal layers on videos. We also

Table 1. *Left:* Comparison with LVG on RDS; *Right:* Ablations.

Method	FVD	FID
LVG [6]	478	53.5
Ours	389	**31.6**
Ours (cond.)	**356**	51.9

Method	FVD	FID
Pixel-baseline	639,56	59.70
End-to-end LDM	1155.10	71.26
Attention-only	704.41	50.01
Ours	534.17	**48.26**
Ours (context-guided)	**508.82**	54.16

Table 2. User study on Driving Video Synthesis on RDS.

Method	Pref. A	Pref. B	Equal
Ours (cond.) v.s *Ours* (uncond.)	**49.33**	42.67	8.0
Ours (uncond.) v.s LVG	**54.02**	40.23	5.74
Ours (cond.) v.s LVG	**62.03**	31.65	6.33

Table 3. *Left:* Evaluating temporal fine-tuning for diffusion upsamplers on RDS data; *Right:* Video fine-tuning of the first stage decoder network leads to significantly improved consistency.

Method	FVD	FID
Ours Image Upsampler	165.98	**19.71**
Ours Video Upsampler	**45.39**	19.85

Decoder	*image-only*	*finetuned*
FVD	390.88	**32.94**
FID	**7.61**	9.17

train Long Video GAN (LVG) [6], the previous state-of-the-art in long-term high-resolution video synthesis, on the RDS data to serve as main baseline. Table 1 (left) shows our main results for the Video LDM at 128×256 resolution, without upsampler. We show both performance of our model with and without conditioning on crowdedness and day/night. Our Video LDM generally outperforms LVG and adding conditioning further reduces FVD. Table 2 shows our human evaluation: Our samples are generally preferred over LVG in terms of realism, and samples from our conditional model are also preferred over unconditional samples.

Next, we compare our video fine-tuned pixel-space upsampler with independent frame-wise image upsampling (Table 3), using 128×256 30 fps ground truth videos for conditioning. We find that temporal alignment of the upsampler is crucial for high performance. FVD degrades significantly, if the video frames are upsampled independently, indicating loss of temporal consistency. As expected, FID is essentially unaffected, because the individual frames are still of high quality when upsampled independently.

In Fig. 1 (bottom) and Fig. 7 (top), we show conditional samples from the combined Video LDM and video upsampler model. We observe high-quality videos. Moreover, using our prediction approach, we find that we can generate very long, temporally coherent high-resolution driving videos of multiple minutes. We validated this for up to 5 minutes; see Appendix and supplementary video for results.

4.1.1 Ablation Studies

To show the efficacy of our design choices (Sec. 3), we compare a smaller version of our Video LDM with various baselines on the RDS dataset and present the results in Table 1 (right) (for evaluation details, see Appendix G). First, using the exact same architecture as for our Video LDM, we apply our temporal finetuning strategy to a pre-trained pixel-space image diffusion model, which is clearly outperformed by ours. Further, we train an End-to-End LDM, whose entire set of parameters $\{\theta, \phi\}$ is learned on RDS videos without image pre-training of θ, leading to heavy degradations both in FID and FVD, when compared with our Video LDM. Another important architectural choice is the introduction of 3D convolutional temporal layers, since they allow us to feed the context frames $\boldsymbol{c}_S$ to the network spatially. This model achieves both lower FVD and FID scores than an attention-only temporal model, which uses the same set of spatial layers θ and has the same number of trainable parameters. Finally, we see that we can further lower FVD scores by applying *context guidance* while sacrificing a bit of visual quality indicated by increased FID scores.

Moreover, we provide an analysis on the effects of video fine-tuning the decoder of the compression model (*cf*. Sec. 3.1.1) which encompasses the LDM framework [65]. We apply our fine-tuning strategy to decoders of these compression models on the RDS dataset and compare both the obtained FVD/FID scores of reconstructed videos/image frames with those of their non-video-finetuned counterparts. Video fine-tuning leads to improvements by orders of magnitudes, as can be seen in Table 3.

4.1.2 Driving Scenario Simulation

A high-resolution video generator trained on in-the-wild driving scenes can potentially serve as a powerful simulation engine. We qualitatively explore this in Fig. 7. Given an initial frame, our video model can generate several different plausible future predictions. Furthermore, we also trained a separate, bounding box-conditioned image LDM on our data (only for image synthesis). A user can now manually create a scene composition of interest by specifying the bounding boxes of different cars, generate a corresponding image, and then use this image as initialization for our Video LDM, which can then predict different scenarios in a multimodal fashion (bottom in Fig. 7).

4.2. Text-to-Video with Stable Diffusion

Instead of first training our own Image LDM backbone, our Video LDM approach can also leverage existing Image LDMs and turn them into video generators. To demonstrate this, we turn the publicly available text-to-image LDM *Stable Diffusion* into a text-to-video generator. Specifically, using the WebVid-10M text-captioned video dataset, we train a temporally aligned version of Stable Diffusion for text-conditioned video synthesis. We briefly fine-tune Stable Diffusion's spatial layers on frames from WebVid, and then insert the temporal alignment layers and train them (at resolution 320×512). We also add text-conditioning in those alignment layers. Moreover, we further video fine-tune the publicly available latent *Stable Diffusion upsampler*, which

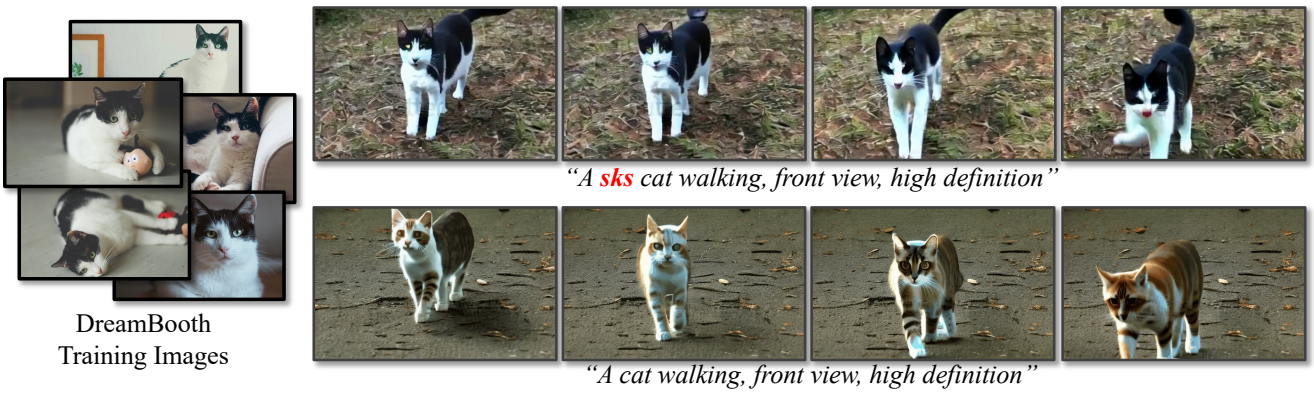

Figure 8. *Left:* DreamBooth Training Images. *Top row:* Video generated by our Video LDM with DreamBooth Image LDM backbone. *Bottom row:* Video generated without DreamBooth Image backbone. We see that the DreamBooth model preserves subject identity well.

Table 4. UCF-101 text-to-video generation.

Method	Zero-Shot	IS (↑)	FVD (↓)
CogVideo (Chinese) [32]	Yes	23.55	751.34
CogVideo (English) [32]	Yes	25.27	701.59
MagicVideo [109]	Yes	-	699.00
Make-A-Video [76]	Yes	33.00	367.23
Video LDM *(Ours)*	Yes	29.49	656.49

Table 5. MSR-VTT text-to-video generation performance.

Method	Zero-Shot	CLIPSIM (↑)
GODIVA [98]	No	0.2402
NÜWA [99]	No	0.2439
CogVideo (Chinese) [32]	Yes	0.2614
CogVideo (English) [32]	Yes	0.2631
Make-A-Video [76]	Yes	0.3049
Video LDM *(Ours)*	Yes	0.2848

enables $4\times$ upscaling and allows us to generate videos at resolution 1280×2048. We generate videos of 4.27 (30 fps) seconds length. Samples from the trained models are shown in Figs. 1 and 6. While WebVid-10M consists of photo-quality real-life videos, we are able to generate highly expressive and artistic videos beyond the video training data. This demonstrates that the general image generation capabilities of the Image LDM backbone readily translate to video generation, even though the video dataset we trained on is much smaller and limited in diversity and style. The Video LDM effectively combines the styles and expressions from the image model with the movements and temporal consistency learnt from the WebVid videos.

We evaluate zero-shot text-to-video generation on UCF-101 [83] and MSR-VTT [101] (Tabs. 4 & 5). Evaluation details in Appendix G. We outperform all baselines except Make-A-Video [76]. However, Make-A-Video is concurrent work, focuses entirely on text-to-video and trains with more video data than we do. We use only WebVid-10M; Make-A-Video also uses HD-VILA-100M [102].

In Appendix D, we show how we can apply our model "convolutional in time" and "convolutional in space", enabling longer and spatially-extended generation without upsampler and prediction models. More video samples shown in Appendix I.1. Experiment details in Appendix H.2.

4.2.1 Personalized Text-to-Video with Dreambooth

Since we have separate spatial and temporal layers in our Video LDM, the question arises whether the temporal layers trained on one Image LDM backbone transfer to other model checkpoints (*e.g.* fine-tuned). We test this for personalized text-to-video generation: Using DreamBooth [66], we fine-tune our Stable Diffusion spatial backbone on small sets of images of certain objects, tying their identity to a rare text token ("*sks*"). We then insert the temporal layers from the previously video-tuned Stable Diffusion (without DreamBooth) into the new DreamBooth version of the original Stable Diffusion model and generate videos using the token tied to the training images for DreamBooth (see Fig. 8 and examples in Appendix I.1.3). We find that we can generate personalized coherent videos that correctly capture the identity of the Dreambooth training images. This validates that our temporal layers generalize to other Image LDMs. To the best of our knowledge, we are the first to demonstrate personalized text-to-video generation.

Additional results and experiments in Appendix I.

5. Conclusions

We presented *Video Latent Diffusion Models* for efficient high-resolution video generation. Our key design choice is to build on pre-trained image diffusion models and to turn them into video generators by temporally video fine-tuning them with temporal alignment layers. To maintain computational efficiency, we leverage LDMs, optionally combined with a super resolution DM, which we also temporally align. Our Video LDM can synthesize high-resolution and temporally coherent driving scene videos of many minutes. We also turn the publicly available *Stable Diffusion* text-to-image LDM into an efficient text-to-video LDM and show that the learned temporal layers transfer to different model checkpoints. We leverage this for personalized text-to-video generation. We hope that our work can benefit simulators in the context of autonomous driving research and help democratize high quality video content creation (see Appendix B for broader impact and limitations).

References

[1] Mohammad Babaeizadeh, Chelsea Finn, Dumitru Erhan, Roy H. Campbell, and Sergey Levine. Stochastic variational video prediction. In *International Conference on Learning Representations*, 2018. 2, 15

[2] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *IEEE International Conference on Computer Vision*, 2021. 6, 17, 24

[3] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, Tero Karras, and Ming-Yu Liu. ediff-i: Text-to-image diffusion models with ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 15

[4] Fan Bao, Chongxuan Li, Jun Zhu, and Bo Zhang. Analytic-dpm: an analytic estimate of the optimal reverse variance in diffusion probabilistic models. In *International Conference on Learning Representations*, 2022. 15

[5] Andreas Blattmann, Timo Milbich, Michael Dorkenwald, and Björn Ommer. ipoke: Poking a still image for controlled stochastic video synthesis. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, 2021. 15

[6] Tim Brooks, Janne Hellsten, Miika Aittala, Ting-Chun Wang, Timo Aila, Jaakko Lehtinen, Ming-Yu Liu, Alexei A Efros, and Tero Karras. Generating long videos of dynamic scenes. *arXiv:2206.03429*, 2022. 2, 6, 7, 15, 17, 20, 21, 24, 25, 26

[7] 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. 20

[8] Lluis Castrejon, Nicolas Ballas, and Aaron Courville. Improved conditional vrnns for video prediction. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 2, 15

[9] Emily Denton and Rob Fergus. Stochastic video generation with a learned prior. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, 2018. 2, 15

[10] Prafulla Dhariwal and Alexander Quinn Nichol. Diffusion models beat GANs on image synthesis. In *Advances in Neural Information Processing Systems*, 2021. 2, 6, 15, 17

[11] Tim Dockhorn, Arash Vahdat, and Karsten Kreis. Genie: Higher-order denoising diffusion solvers. In *Advances in Neural Information Processing Systems*, 2022. 15

[12] Tim Dockhorn, Arash Vahdat, and Karsten Kreis. Score-based generative modeling with critically-damped langevin diffusion. In *International Conference on Learning Representations (ICLR)*, 2022. 15

[13] Michael Dorkenwald, Timo Milbich, Andreas Blattmann, Robin Rombach, Konstantinos G. Derpanis, and Björn Ommer. Stochastic image-to-video synthesis using cinns. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, 2021. 15

[14] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. *arXiv preprint arXiv:2302.03011*, 2023. 16

[15] Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis. *arXiv preprint arXiv:2012.09841*, 2020. 2

[16] Gereon Fox, Ayush Tewari, Mohamed Elgharib, and Christian Theobalt. Stylevideogan: A temporal generative model using a pretrained stylegan. In *British Machine Vision Conference (BMVC)*, 2021. 15, 16

[17] Jean-Yves Franceschi, Edouard Delasalles, Mickaël Chen, Sylvain Lamprier, and Patrick Gallinari. Stochastic latent residual video prediction. In *Proceedings of the 37th International Conference on Machine Learning*, 2020. 2, 15

[18] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. 15

[19] 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. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, pages 102–118, Cham, 2022. Springer Nature Switzerland. 2, 15

[20] 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. 20

[21] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 2

[22] Sonam Gupta, Arti Keshari, and Sukhendu Das. Rv-gan: Recurrent gan for unconditional video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 2024–2033, June 2022. 2, 15

[23] Tanmay Gupta, Dustin Schwenk, Ali Farhadi, Derek Hoiem, and Aniruddha Kembhavi. Imagine this! scripts to compositions to videos. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, pages 610–626, Cham, 2018. Springer International Publishing. 2, 16

[24] William Harvey, Saeid Naderiparizi, Vaden Masrani, Christian Weilbach, and Frank Wood. Flexible diffusion modeling of long videos. *arXiv preprint arXiv:2205.11495*, 2022. 2, 5, 16

[25] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 15

[26] 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. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. 6, 20

[27] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P. Kingma, Ben Poole, Mohammad Norouzi, David J. Fleet, and Tim Salimans. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 16, 24

[28] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, 2020. 2, 3, 4, 15, 16, 17

[29] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *arXiv preprint arXiv:2106.15282*, 2021. 2, 5, 15, 21, 23

[30] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 5

[31] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J. Fleet. Video diffusion models. *arXiv preprint arXiv:2204.03458*, 2022. 2, 16

[32] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv:2205.15868*, 2022. 2, 8, 15, 16, 21, 24

[33] Tobias Höppe, Arash Mehrjou, Stefan Bauer, Didrik Nielsen, and Andrea Dittadi. Diffusion models for video prediction and infilling. *arXiv preprint arXiv:2206.07696*, 2022. 2, 5, 16

[34] Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6:695–709, 2005. 3

[35] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 3

[36] Alexia Jolicoeur-Martineau, Ke Li, Rémi Piché-Taillefer, Tal Kachman, and Ioannis Mitliagkas. Gotta go fast when generating data with score-based models. *arXiv:2105.14080*, 2021. 15

[37] Emmanuel Kahembwe and Subramanian Ramamoorthy. Lower dimensional kernels for video discriminators. *Neural Networks*, 132:506–520, 2020. 2, 15

[38] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. In *Proc. NeurIPS*, 2021. 2, 20

[39] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 2

[40] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020. 2

[41] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. *arXiv preprint arXiv:2201.11793*, 2022. 15

[42] Alex X. Lee, Richard Zhang, Frederik Ebert, Pieter Abbeel, Chelsea Finn, and Sergey Levine. Stochastic adversarial video prediction. *arXiv preprint arXiv:1804.01523*, 2018. 2, 15

[43] Haoying Li, Yifan Yang, Meng Chang, Shiqi Chen, Huajun Feng, Zhihai Xu, Qi Li, and Yueting Chen. Srdiff: Single image super-resolution with diffusion probabilistic models. *Neurocomputing*, 479:47–59, 2022. 2, 15

[44] Yitong Li, Martin Renqiang Min, Dinghan Shen, David Carlson, and Lawrence Carin. Video generation from text. *arXiv preprint arXiv:1710.00421*, 2017. 2, 16

[45] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. In *International Conference on Learning Representations*, 2022. 15

[46] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *arXiv:2206.00927*, 2022. 15

[47] Pauline Luc, Aidan Clark, Sander Dieleman, Diego de Las Casas, Yotam Doron, Albin Cassirer, and Karen Simonyan. Transformation-based adversarial video prediction on large-scale data. *ArXiv*, 2020. 2, 15

[48] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. *arXiv preprint arXiv:2201.09865*, 2022. 15

[49] Eric Luhman and Troy Luhman. Knowledge distillation in iterative generative models for improved sampling speed. *arXiv preprint arXiv:2101.02388*, 2021. 15

[50] Siwei Lyu. Interpretation and generalization of score matching. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, page 359–366, Arlington, Virginia, USA, 2009. AUAI Press. 3

[51] Tanya Marwah, Gaurav Mittal, and Vineeth N. Balasubramanian. Attentive semantic video generation using captions. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1435–1443, 2017. 2, 16

[52] Chenlin Meng, Ruiqi Gao, Diederik P. Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. *arXiv preprint arXiv:2210.03142*, 2022. 15

[53] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022. 15

[54] Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. Which training methods for gans do actually converge? In

International Conference on Machine Learning (ICML), 2018. 21

[55] Gaurav Mittal, Tanya Marwah, and Vineeth N. Balasubramanian. Sync-draw: Automatic video generation using deep recurrent attentive architectures. In *Proceedings of the 25th ACM International Conference on Multimedia*, MM '17, page 1096–1104, New York, NY, USA, 2017. Association for Computing Machinery. 2, 16

[56] Eyal Molad, Eliahu Horwitz, Dani Valevski, Alex Rav Acha, Yossi Matias, Yael Pritch, Yaniv Leviathan, and Yedid Hoshen. Dreamix: Video diffusion models are general video editors. *arXiv preprint arXiv:2302.01329*, 2023. 16

[57] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 2, 15

[58] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, 2021. 2, 15

[59] Yingwei Pan, Zhaofan Qiu, Ting Yao, Houqiang Li, and Tao Mei. To create what you tell: Generating videos from captions. *Proceedings of the 25th ACM international conference on Multimedia*, 2017. 2, 16

[60] Konpat Preechakul, Nattanat Chatthee, Suttisak Wizadwongsa, and Supasorn Suwajanakorn. Diffusion autoencoders: Toward a meaningful and decodable representation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 15

[61] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 21

[62] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 2, 15, 16

[63] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8821–8831. PMLR, 18–24 Jul 2021. 2

[64] Alex Rogozhnikov. Einops: Clear and reliable tensor manipulations with einstein-like notation. In *International Conference on Learning Representations*, 2022. 4

[65] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *arXiv preprint arXiv:2112.10752*, 2021. 2, 3, 5, 6, 7, 15, 16, 17, 21

[66] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image dissusion models for subject-driven generation. *arXiv preprint arXiv:2208.12242*, 2022. 3, 8, 15, 24

[67] Chitwan Saharia, William Chan, Huiwen Chang, Chris A. Lee, Jonathan Ho, Tim Salimans, David J. Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. *arXiv preprint arXiv:2111.05826*, 2021. 15

[68] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 2, 5, 15, 16, 21, 23

[69] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *arXiv preprint arXiv:2104.07636*, 2021. 2, 15

[70] Masaki Saito, Eiichi Matsumoto, and Shunta Saito. Temporal generative adversarial nets with singular value clipping. In *ICCV*, 2017. 15

[71] Masaki Saito, Shunta Saito, Masanori Koyama, and Sosuke Kobayashi. Train sparsely, generate densely: Memory-efficient unsupervised training of high-resolution temporal gan. *International Journal of Computer Vision*, May 2020. 2, 15, 21

[72] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, 2016. 21

[73] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations (ICLR)*, 2022. 3, 15

[74] Hiroshi Sasaki, Chris G. Willcocks, and Toby P. Breckon. UNIT-DDPM: Unpaired image translation with denoising diffusion probabilistic models. *arXiv preprint arXiv:2104.05358*, 2021. 15

[75] Axel Sauer, Katja Schwarz, and Andreas Geiger. Stylegan-xl: Scaling stylegan to large diverse datasets. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–10, 2022. 2

[76] 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:2209.14792*, 2022. 8, 16, 21, 24

[77] Abhishek Sinha, Jiaming Song, Chenlin Meng, and Stefano Ermon. D2c: Diffusion-denoising models for few-shot conditional generation. In *Advances in Neural Information Processing Systems*, 2021. 15

[78] 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 (CVPR)*, pages 3626–3636, June 2022. 2, 15

[79] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using

nonequilibrium thermodynamics. In *International Conference on Machine Learning*, 2015. 2, 3, 15

[80] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 6, 15, 17

[81] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Proceedings of the 33rd Annual Conference on Neural Information Processing Systems*, 2019. 3

[82] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. 2, 3, 15

[83] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 8

[84] Xuan Su, Jiaming Song, Chenlin Meng, and Stefano Ermon. Dual diffusion implicit bridges for image-to-image translation. *arXiv preprint arXiv:2203.08382*, 2022. 15

[85] Yu Tian, Jian Ren, Menglei Chai, Kyle Olszewski, Xi Peng, Dimitris N. Metaxas, and Sergey Tulyakov. A good image generator is what you need for high-resolution video synthesis. In *International Conference on Learning Representations*, 2021. 2, 15, 16

[86] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015. 21

[87] 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:1812.01717*, 2018. 4, 6, 20

[88] Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. In *Advances in Neural Information Processing Systems*, 2021. 15

[89] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017. 4

[90] Ruben Villegas, Mohammad Babaeizadeh, Pieter-Jan Kindermans, Hernan Moraldo, Han Zhang, Mohammad Taghi Saffar, Santiago Castro, Julius Kunze, and Dumitru Erhan. Phenaki: Variable length video generation from open domain textual description. *arXiv:2210.02399*, 2022. 16

[91] Ruben Villegas, Jimei Yang, Seunghoon Hong, Xunyu Lin, and Honglak Lee. Decomposing motion and content for natural video sequence prediction. *ICLR*, 2017. 2, 15

[92] Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011. 3

[93] Vikram Voleti, Alexia Jolicoeur-Martineau, and Christopher Pal. Mcvd: Masked conditional video diffusion for prediction, generation, and interpolation. *arXiv preprint arXiv:2205.09853*, 2022. 2, 5, 16

[94] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 2016. 2, 15

[95] Yaohui Wang, Piotr Bilinski, Francois Bremond, and Antitza Dantcheva. G3an: Disentangling appearance and motion for video generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 15

[96] Daniel Watson, William Chan, Jonathan Ho, and Mohammad Norouzi. Learning fast samplers for diffusion models by differentiating through sample quality. In *International Conference on Learning Representations*, 2022. 15

[97] Dirk Weissenborn, Oscar Täckström, and Jakob Uszkoreit. Scaling autoregressive video models. In *International Conference on Learning Representations*, 2020. 2, 15

[98] Chenfei Wu, Lun Huang, Qianxi Zhang, Binyang Li, Lei Ji, Fan Yang, Guillermo Sapiro, and Nan Duan. Godiva: Generating open-domain videos from natural descriptions. *arXiv:2104.14806*, 2021. 2, 6, 8, 15, 16, 21

[99] Chenfei Wu, Jian Liang, Lei Ji, Fan Yang, Yuejian Fang, Daxin Jiang, and Nan Duan. Nüwa: Visual synthesis pre-training for neural visual world creation. In *European Conference on Computer Vision*, pages 720–736. Springer, 2022. 2, 8, 15, 16

[100] Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. Tackling the generative learning trilemma with denoising diffusion GANs. In *International Conference on Learning Representations (ICLR)*, 2022. 15

[101] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 8

[102] Hongwei Xue, Tiankai Hang, Yanhong Zeng, Yuchong Sun, Bei Liu, Huan Yang, Jianlong Fu, and Baining Guo. Advancing high-resolution video-language representation with large-scale video transcriptions. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 8

[103] Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. Videogpt: Video generation using vq-vae and transformers, 2021. 2, 15

[104] Ruihan Yang, Prakhar Srivastava, and Stephan Mandt. Diffusion probabilistic modeling for video generation. *arXiv preprint arXiv:2203.09481*, 2022. 2, 16

[105] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, Ben Hutchinson, Wei Han, Zarana Parekh, Xin Li, Han Zhang, Jason Baldridge, and Yonghui Wu. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2022. 2

[106] Sihyun Yu, Jihoon Tack, Sangwoo Mo, Hyunsu Kim, Junho Kim, Jung-Woo Ha, and Jinwoo Shin. Generating videos with dynamics-aware implicit generative adversarial networks. In *International Conference on Learning Representations*, 2022. 2, 15

[107] Xiaohui Zeng, Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, and Karsten Kreis. Lion: Latent point diffusion models for 3d shape generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 15

[108] Qinsheng Zhang and Yongxin Chen. Fast sampling of diffusion models with exponential integrator. *arXiv:2204.13902*, 2022. 15

[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. 8, 16