StyleInV: A Temporal Style Modulated Inversion Network for Unconditional Video Generation

Yuhan Wang, Liming Jiang, Chen Change Loy
S-Lab, Nanyang Technological University
{yuhan004, liming002, ccloy}@ntu.edu.sg

Abstract

Unconditional video generation is a challenging task that involves synthesizing high-quality videos that are both coherent and of extended duration. To address this challenge, researchers have used pretrained StyleGAN image generators for high-quality frame synthesis and focused on motion generator design. The motion generator is trained in an autoregressive manner using heavy 3D convolutional discriminators to ensure motion coherence during video generation. In this paper, we introduce a novel motion generator design that uses a learning-based inversion network for GAN. The encoder in our method captures rich and smooth priors from encoding images to latents, and given the latent of an initially generated frame as guidance, our method can generate smooth future latent by modulating the inversion encoder temporally. Our method enjoys the advantage of sparse training and naturally constrains the generation space of our motion generator with the inversion network guided by the initial frame, eliminating the need for heavy discriminators. Moreover, our method supports style transfer with simple fine-tuning when the encoder is paired with a pretrained StyleGAN generator. Extensive experiments conducted on various benchmarks demonstrate the superiority of our method in generating long and high-resolution videos with decent single-frame quality and temporal consistency. Code is available at https://github.com/johannwyh/StyleInV.

1. Introduction

Unconditional video generation aims at learning a generative model to create novel videos from latent vectors. Despite extensive studies [42, 32, 33, 38, 11, 50] in addressing this problem, it remains challenging to generate high-resolution videos with both favorable quality and motion coherence over a long-term duration. The core difficulties in this task lie in modeling consistent motion and managing the high memory consumption introduced by the addition of the temporal dimension.

To ensure high single-frame resolution and quality, many existing studies, such as MoCoGAN-HD [36], employ a powerful image generator such as StyleGAN [22] as a backbone to serve as a strong generative prior. This approach shifts the focus towards developing a robust motion generator that can capture temporally coherent motion. Most of these methods model motion in an auto-regressive manner, where the next latent is sampled conditioned on the previous one (see Fig. 1). However, this design has two main drawbacks. First, while good performance requires seeing

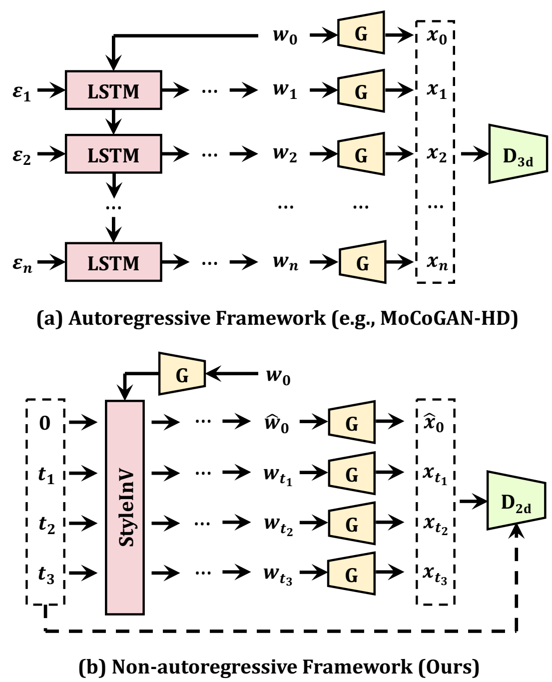


Figure 1: A comparison between autoregressive and non-autoregressive pipeline: (a) Previous autoregressive motion generators require generating the whole clip for a 3D-convolution-based discriminator. (b) Our non-autoregressive motion generator, **StyleInV**, is an inversion network modulated by temporal style (as a random function of t), which enjoys sparse training using a 2D-convolution-based discriminator.

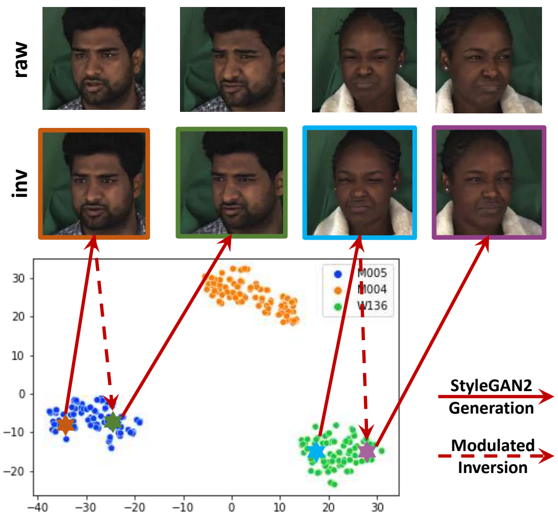


Figure 2: Inverted latent space visualization and modulated inversion process: When the StyleGAN generator is trained with video frame data, $\mathcal{W}$ space is well clustered by human identities and provides promising inversion results. Thus, the modulated inversion process can easily find the target latent corresponding to the same identity (shifted to the next motion) as the source one.

a long sequence of images, the use of heavy 3D discriminators limits its ability to be trained with longer videos. Second, the autoregressive motion generator can lead to motion collapse when extrapolating to generate longer videos.

In this study, we present an effective framework for non-autoregressive motion generation that is capable of generating long and high-resolution videos. Our approach leverages learning-based Generative Adversarial Network (GAN) inversion, which learns the inverse mapping of GANs via an inversion network that consists of an encoder and a decoder[1]. To generate long and coherent videos, we exploit the unique characteristic of the inversion encoder, which captures a rich and smooth manifold between the mapping of images and latent. As illustrated in Fig. 1, to generate a sequence of smooth motion latents, we just need to provide the initial latent code and modulate the inversion encoder with temporal style codes, which are encodings of timestamps with randomness. The motion latents can then be mapped by a StyleGAN decoder to generate a video.

The proposed framework offers several advantages in a single unified framework. **First**, the use of an inversion network naturally constrains the generation space to stay consistent with the desired appearance, which is defined by the initial latent code. As demonstrated in Fig. 2, this leads to a significant benefit. **Second**, thanks to the flexibility of the inversion network in accepting temporal styles of arbitrary timestamps, the framework allows non-autoregressive generation and sparse training [52, 35]. These merits help alleviate the need for heavy discriminators to ensure temporal consistency, as is required in existing approaches. In our implementation, we only need to use a 2D convolutional discriminator instead of a 3D discriminator like MoCoGAN-HD. **Third**, Unlike existing state-of-the-art methods [52, 35, 7] that couple content and motion decoding in one synthesis network, our framework can naturally support content decoder fine-tuning on different image datasets. Specifically, after fine-tuning the decoder (*e.g.*, StyleGAN2) on another image dataset with the mapping layers and low-resolution synthesis layers fixed, given the same sequence of synthesized motion latents, the generated video can possess the new style of the fine-tuning dataset while preserving the motion patterns of the video generated by the parent content decoder.

The main contribution of this work is a novel motion generator that modulates a GAN inversion network. This is the first attempt to build such a generator, and it offers several advantages in a unified framework over existing approaches. These advantages include consistent generation, sparse training, and flexibility in supporting style transfer with simple fine-tuning. We additionally contribute a reformulation to the conventional sparse training, through first-frame-aware acyclic positional encoding (FFA-APE) and first-frame-aware sparse training (FFA-ST), to ensure that our motion generator can faithfully reconstruct the initial frame and that the generated video is smooth and continuous. Extensive experiments on DeeperForensics [17], FaceForensics [31], SkyTimelapse [47] and Tai-Chi-HD [34] datasets show that our model is comparable to or even better than state-of-the-art unconditional video generation methods [36, 52, 35] both qualitatively and quantitatively.

2. Related Work

GAN inversion. The goal of GAN inversion is to find the corresponding vector in the latent space of a pretrained GAN [22, 23] to reconstruct the input image. Existing methods can be classified into three categories [46]: (1) learning-based methods [8, 37, 49, 3, 2, 44, 29], which leverage an encoder network to directly map an image into a latent vector; (2) optimization-based methods [43, 1, 45, 48, 56, 57], which iteratively find the latent vector that best reconstructs the input image using gradient descent; and (3) hybrid models [6, 5, 9, 55], which initialize the iteration process with the result of an encoder network. The design of our motion generator follows the learning-based approach. Therefore, our method is trainable, efficient for single-image inference, and suitable for hierarchical mod-

[1]In many contexts, the decoder is a StyleGAN, and the encoder learns to encode a given image to meaningful latent vectors in the StyleGAN space. There is a variety of image manipulation applications [46, 29, 4] developed based upon such an inversion framework.

ulation. We devise the motion generator on the $\mathcal{W}$ space and use the StyleGAN generated latent as the initial content code to guide the modulated inversion process (see Fig. 2).

Unconditional video generation. Unconditional video generation aims to model the distribution of real videos in a training dataset and generate videos from sampled noise vectors. Many recent studies on this topic are inspired by the success of GANs in image generation. VGAN [42] applies 3D convolutions in both the generator and discriminator, while TGAN [32] optimizes this design by decomposing the generator into an *image generator*, which is shared by the generation of each frame, and a *motion generator*[2]. This framework has been followed by most subsequent studies, such as MoCoGAN [38], which applies a content-motion decomposition. Some approaches [33, 11, 19] have focused on reducing the computational cost of the video discriminator, but the cost is still proportional to the video duration and resolution. Some recent methods have applied more advanced generative frameworks and techniques to unconditional video generation. For example, VideoGPT [50] uses VQ-VAE [28] and GPT [10] to formulate a non-GAN-based video generation approach. Recent studies have also explored unconditional video generation with higher resolution and longer duration. For example, Long-Video-GAN [7] develops a two-phase model that focuses on improving the long-term temporal dynamics of video generation. MoCoGAN-HD [36] and StyleVideoGAN [13] study the generation of latent trajectories in the latent space of a pretrained StyleGAN2 generator. Our approach is inspired by these studies, but differs in the design of the motion generator. Our motion generator is non-autoregressive, thus alleviating the use of heavy discriminators, and it is unique since it obtains the motion latent via modulating a GAN inversion network. This design allows us to attain better motion consistency and semantics.

Recent works [52, 35] explored neural representation-based generators and trained them sparsely as an image GAN. StyleSV [53] improves this framework by introducing StyleGAN3 [21] architecture and several temporal designs. In our work, we extend the idea of sparse training to first-frame-aware sparse training, allowing it to be applied to a generation pipeline conditioned on the initial latent.

Diffusion-based video generation. The diffusion models [15, 30], a new paradigm for image generation tasks, have also achieved significant progress in the task of unconditional video generation [16, 24, 41, 51]. Despite their success, temporal consistency is still an open problem for diffusion models, and GAN-based models exhibit a clear advantage in terms of inference speed.

[2]In the original paper of TGAN [32], the authors called this module *temporal generator*, which is equivalent to the *motion generator* used in subsequent studies [38, 36, 52] and in our paper.

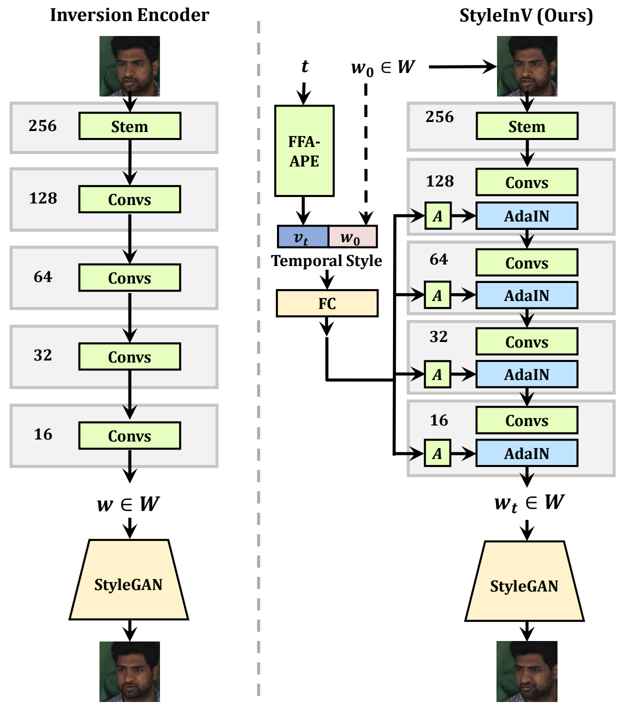


Figure 3: From a typical inversion encoder to StyleInV: We add `AdaIN` layers at the end of each residual block to inject the temporal style, which is a combination of time positional encoding and the first frame latent code. Here "A" stands for a learned affine transform [22].

3. Methodology

3.1. Preliminaries of Inversion Encoder

An inversion encoder maps an input image to a vector in the $\mathcal{W}$ or $\mathcal{W}+$ latent space of a pretrained StyleGAN2 generator. The generated image that corresponds to this vector should faithfully reconstruct the details of the input image. Therefore, when based on $\mathcal{W}$ latent space, given an input image $\mathbf{x}$, the reconstruction process can be defined on top of the inversion network Inv as:

$$\hat{\mathbf{x}} := G(\mathrm{Inv}(\mathbf{x})) := G(E(\mathbf{x}) + \overline{\mathbf{w}}). \quad (1)$$

Here E and G denote the inversion encoder and StyleGAN generator, respectively. $\overline{\mathbf{w}} \in \mathbb{R}^{512}$ denotes the average latent vector of the generator in the $\mathcal{W}$ latent space. In our implementation, the encoder E is a convolutional network backbone that outputs a 512-dimensional vector from the last layer embedding, as shown in Fig. 3(left). We build the encoder on the $\mathcal{W}$ latent space, which eases the design of temporal modulation.

3.2. Temporal Style Modulated Inversion Encoder

We observe that the latent space of a StyleGAN trained on a video dataset is typically well-clustered by its content subject. Figure 2 shows an example of human face videos, where we depict the results of inverting video clips of different identities into the $\mathcal{W}$ space and visualizing them with t-SNE [40]. It can be observed that the latent space is grouped by human identities. We also observe the same property in video datasets that follow other distributions. This phenomenon suggests that the inversion network inherits some important temporal priors that we could leverage to maintain motion consistency in generated videos.

Motivated by this observation, we propose **StyleInV**, in which the motion latent is generated by modulating a GAN inversion network with temporal styles. Figure 3(right) illustrates the pipeline of our framework. The temporal style $\boldsymbol{s}_t$ of a timestamp t consists of two parts: the motion code $\boldsymbol{v}_t$ and the latent code of the initial frame $\boldsymbol{w}_0$. Inspired by [35], we use an acyclic positional encoding module to compute a dynamic embedding of the timestamp t. However, unlike [35], we make the embedding of the zero timestamp fixed, so this module becomes first-frame-aware. We provide more details in Section 3.3. The latent code $\boldsymbol{w}_0$ of the initial frame is concatenated with the motion code for content-adaptive affine transform.

The temporal style is injected into the inversion encoder through `AdaIN` layers at the end of each convolution block. With this design, the encoder E of StyleInV becomes a function of the initial latent code w_0 and timestamp t. The modulated inversion process can be defined as:

$$\hat{\mathbf{x}}_t := G(\text{StyleInV}(\boldsymbol{w}_0, t)) := G(E(G(\boldsymbol{w}_0), \boldsymbol{s}_t) + \boldsymbol{w}_0). \quad (2)$$

Notably, the output of E serves as the residual w.r.t. w_0, instead of $\overline{\mathbf{w}}$. This modification provides more explicit content information guidance for the inversion encoder.

During training, we first train a raw inversion encoder following Eq. (1) on all video frames. Then, we use this network to initialize the weights of all convolution layers in the StyleInV encoder. Other parameters (*e.g.*, FFA-APE and Affine Transforms) are randomly initialized. Finally, the entire StyleInV encoder is trained end-to-end.

3.3. FFA-APE

The original implementation of acyclic positional encoding (APE) [35] samples a series of noise vectors $\boldsymbol{z}^{\mathrm{m}}_{t_0}, \cdots, \boldsymbol{z}^{\mathrm{m}}_{t_n} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ where $t_i = i \cdot \delta^z$. We call these temporal points *anchor points*. Here δ^z is a set constant distance between adjacent anchor points. Then, the noise vectors are mapped to tokens $\boldsymbol{u}_{t_0}, \cdots, \boldsymbol{u}_{t_n}$ by a padding-less `conv1d`-based motion mapping network. The computation of the acyclic positional encoding $\boldsymbol{v}_t$ of arbitrary timestamp t is achieved by a scalable and learnable interpolation between the tokens of two adjacent *anchor points* that cover t. The computation pipeline is shown in Fig. 4.

In our non-autoregressive generation pipeline, the modulated inversion encoder needs to faithfully reconstruct the

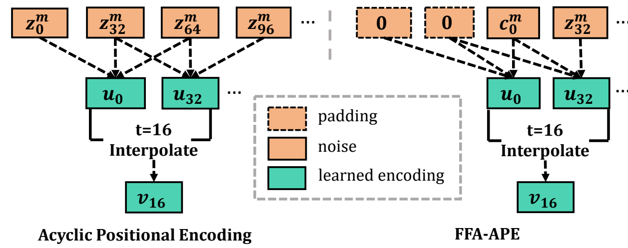


Figure 4: FFA-APE: A simplified case to compute v_{16} for demonstrating the original acyclic positional encoding and our FFA-APE when $\delta^z = 32$ and `conv1d` kernel size is 3. In FFA-APE, the encoding of zero timestamp (u_0) only depends on constant paddings and a constant noise vector, thus is fixed for any sampled noise vector sequence.

initial frame when the input timestamp is zero, making it necessary to fix the computation of the APE for the zero timestamp $\boldsymbol{v}_0$. The original APE computation for $\boldsymbol{v}_0$ is dynamic and depends on randomly sampled noise vectors, which can lead to dynamic output that is not desired. To address this, we devise a first-frame-aware acyclic positional encoding (FFA-APE) method that fixes $\boldsymbol{v}_0$ while maintaining the smoothness of APE (see Fig. 4). We achieve this by replacing the noise vector for the first anchor point with a learnable constant vector $\boldsymbol{c}^{\mathrm{m}}_0$, and using left-sided `conv1d` layers with constant padding instead of the padding-less `conv1d` layers. This way, the value of $\boldsymbol{v}_0$ only depends on the constant vector $\boldsymbol{c}^{\mathrm{m}}_0$ and the left-padded vectors, which are also constant. As a result, $\boldsymbol{v}_0$ is naturally fixed without affecting the continuity of positional encoding.

3.4. FFA-ST

In this section, we introduce the first-frame-aware sparse training specially designed for our framework. Recent non-autoregressive video generation approaches [52, 35] use a discriminator design that only considers k frames $\boldsymbol{x}_{t_1}, \cdots, \boldsymbol{x}_{t_k}$ for each video, distinguishing the realness of the input conditioned on the time difference of input frames $\delta_i = t_{i+1} - t_i$. This training scheme is called sparse training. StyleGAN-V [35] has analyzed the choice of k and found that $k = 3$ is ideal for most datasets. The discriminator is defined as $D(\boldsymbol{x}_{t_{1,2,3}}, \delta_{1,2})$.

We follow this training scheme to make full use of our non-autoregressive framework. Nonetheless, using only three randomly sampled timestamps to train the generator and discriminator can result in sharp transitions at the beginning of the generated video, where the generated $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$ usually diverge too much, and sometimes even switch to another identity and never return. This happens because although we define the generation process of a video as a modulated inversion process of the start frame, the discriminator is unaware of it. The discriminator only focuses on the smoothness of generated latent trajectories, failing to ensure the motion generator produces frames that share the identity with the start frame.

To solve this problem, we introduce the initial frame into the discriminator to enhance content consistency and motion smoothness. The adversarial loss for the first-frame-aware discriminator (FFA-D) can be written as:

$$\begin{aligned} \boldsymbol{y}_{t_{0,1,2,3}} &= G(\text{StyleInV}(w_0, t_{0,1,2,3})), \\ \mathcal{L}_{adv} &= \mathbb{E}_{\boldsymbol{x}\sim p_v}\left[\log D(\boldsymbol{x}_{t_{0,1,2,3}}, \delta_{0,1,2})\right] \\ &\quad + \mathbb{E}_{w_0\sim p_{\mathcal{W}}}\left[\log(1 - D(\boldsymbol{y}_{t_{0,1,2,3}}, \delta_{0,1,2})\right], \end{aligned} \tag{3}$$

where we specify $t_0 = 0$. Here, p_v and $p_{\mathcal{W}}$ denote the real data distribution and $\mathcal{W}$ latent space distribution, respectively. To explicitly enforce initial frame reconstruction, we use a L_2 loss for the generated $\boldsymbol{y}_{t_0}$:

$$\mathcal{L}_{L_2} = ||G(w_0) - G(\text{StyleInV}(w_0, 0))||_2. \tag{4}$$

Finally, we apply latent regularization [29, 26] to the encoder's output, so as to enhance content consistency:

$$\mathcal{L}_{reg} = \sum\nolimits_{i=0}^{3} ||E(G(w_0), t_i)||_2. \tag{5}$$

The overall loss function for training our motion generator and the discriminator is defined as:

$$\min_{E} \max_{D} \mathcal{L}_{adv} + \min_{E}(\lambda_{L_2}\mathcal{L}_{L_2} + \lambda_{reg}\mathcal{L}_{reg}). \tag{6}$$

Here λ_{L_2} and λ_{reg} are the loss hyperparameters. We also apply discriminator adaptive augmentation [20, 35] and $r1$ regularization [22, 35] to further improve the training stability and generation quality.

3.5. Finetuning-based Style Transfer

Our 'inversion encoder+decoder' framework can naturally take a pretrained StyleGAN model as the generator. And such a configuration allows the generator to be fine-tuned for different styles, and yet still able to use the motion generator for generating new video with styles. The capability is not possible with existing non-autoregressive video generation methods [52, 35] because they cannot be finetuned under an image GAN training scheme.

To achieve style transfer, as illustrated in Fig. 5, we fine-tune the pre-trained StyleGAN model using an image dataset, such as MegaCartoon [27], while keeping the mapping network and low-resolution ($\leqslant 32^2$, coarse and middle layers in [22]) synthesis blocks fixed. This configuration maintains the distribution of the $\mathcal{W}$ space during fine-tuning. To improve identity preservation and reduce artifacts, we apply both a perceptual loss [18] and an identity loss [12] between the images generated by the original and fine-tuned StyleGAN. We show some visual results in Fig. 8. The style-transferred video maintains the same motion pattern as the video generated by the parent model, while adopting a new style from the fine-tuning image dataset. It is noteworthy that the finetuning process is independent of the video generation training. It means that the finetuning-based style transfer is "plug-and-play" as the fine-tuned image generator can be used on any StyleInV models. There is no additional computational cost in the inference either.

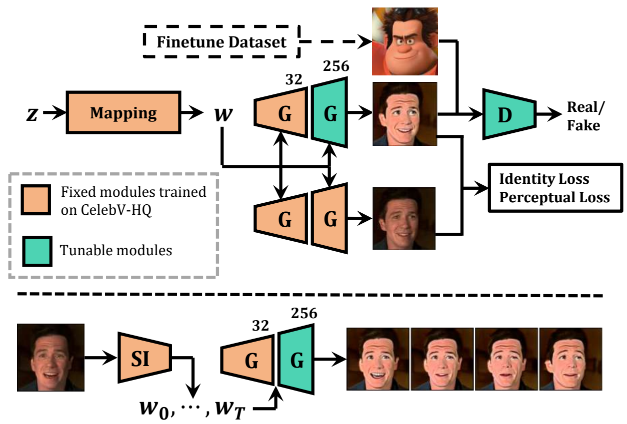


Figure 5: Finetuning-based style transfer: our framework allows easy fine-tuning of decoder (a pretrained StyleGAN generator) to a new domain by freezing mapping and low-resolution ($\leqslant 32^2$) layers. Standard identity loss and perceptual loss are applied to improve identity preservation and reduce artifacts. The style-transferred videos can then be generated by incorporating the StyleInV motion generator.

4. Experiments

Datasets. We use four video datasets in our main experiments: DeeperForensics 256^2 [17], FaceForensics 256^2 [31], SkyTimelapse 256^2 [47] and TaiChi 256^2 [34]. The cropping strategy for DeeperForensics [17] and FaceForensics [31] is different. For DeeperForensics, we use a stabilized FFHQ [22] cropping strategy [25], while we follow the strategy firstly adopted by TGAN-V2 [33] for FaceForensics. Please refer to the *supplementary material* for a detailed discussion.

Baselines. We explore four state-of-the-art methods for comparison: MoCoGAN-HD [36], DIGAN [52], StyleGAN-V [35] and Long-Video-GAN [7]. Among these methods, MoCoGAN-HD and DIGAN require an explicit setting of the training clip length. We follow the default setting of their paper to set the clip length as 16 for both methods. This setting is identical to StyleGAN-V [35].

In addition, on DeeperForensics, we explore an optimized setting on DIGAN and MoCoGAN-HD for a more fair comparison. For DIGAN, we directly increase the clip length to 128 frames. For MoCoGAN-HD, we apply the first-frame-aware sparse training to train its motion generator, so as to avoid using a heavy 3D discriminator, allowing it to be trained with 128-frame clips as other methods.

Evaluation. We use Fréchet Inception Distance (FID) [14] and Fréchet Video Distance (FVD) [39] to evaluate all models quantitatively. In practice, we follow the metric calculation framework provided by StyleGAN-V [35] to first generate a fake video dataset with 2,048 synthesized clips, each of 128 frames. For FID, we sample 50k frames from real and fake video datasets to compute the result. For FVD, we compute FVD_{16} and FVD_{128} with the first 16 frames and all 128 frames of each clip, respectively. We use FID results to show the single-frame image quality of each method.

To ensure a fair comparison, we re-benchmark the quantitative results of every method on every dataset. We retrain all the baselines using the official paper setting, except for MoCoGAN-HD on SkyTimelapse, where an officially released checkpoint is available. For more implementation details, please refer to the supplementary material.

4.1. Main Results

Quantitative results. Table 1 summarizes the quantitative results of our method compared to other baselines. Our method achieves competitive quantitative results on all the benchmarks. Notably, although MoCoGAN-HD and DIGAN are trained with clips of 16 frames, we still outperform them in terms of FVD_{16} metrics on all four datasets.

Qualitative results. Figure 6 shows the qualitative comparison between our method and the baselines on all four datasets. MoCoGAN-HD and DIGAN both suffer from motion collapse, resulting in a degraded generation quality over time. StyleGAN-V shows an impressive visual performance on FaceForensics and SkyTimelapse, but it sometimes fails to maintain the identity and accessories on DeeperForensics and lacks diversity and magnitude of motion over a long time span on TaiChi (the subject gradually fixes at one state). Long-Video-GAN is exceptionally good at SkyTimelapse, but it cannot achieve similar performance on other datasets. It fails to maintain the identity on DeeperForensics, and its single-frame content on TaiChi lacks details and is inferior to other methods. The generated videos by Long-Video-GAN collapse on FaceForensics.

In contrast to existing methods, our method demonstrates stable results on all four datasets, particularly with superior identity preservation on human-face video and long-term generation quality on TaiChi. Although our method outperforms existing methods in terms of content quality, continuity, and quantitative results, the motion semantics of our generated videos on SkyTimelapse are inferior to those on other datasets. This could be one of the limitations of our work and an area for future improvement.

Extended experiments. We present more in-depth comparisons in Table 2 and Fig. 7 by introducing training improvements to baselines. Increasing the clip length generally improves the results of MoCoGAN-HD and DIGAN, but they are still inferior to our method. Notably, training

Table 1: FID, FVD_{16} and FVD_{128} results of video generation methods on (a) DeeperForensics 256^2, (b) FaceForensics 256^2, (c) TaiChi 256^2, and (d) SkyTimelapse 256^2. **Bolds** indicate best and <u>underlines</u> indicate the second best.

(a) DeeperForensics 256^2

Method	FID (↓)	FVD_{16} (↓)	FVD_{128} (↓)
MoCoGAN-HD	135.30	101.07	610.30
DIGAN	191.99	46.69	1060.27
StyleGAN-V	59.59	**39.33**	<u>68.81</u>
Long-Video-GAN	<u>56.54</u>	74.77	169.45
StyleInV (ours)	**54.05**	<u>41.58</u>	**53.93**

(b) FaceForensics 256^2

Method	FID (↓)	FVD_{16} (↓)	FVD_{128} (↓)
MoCoGAN-HD	24.45	112.67	486.69
DIGAN	151.53	146.62	1993.20
StyleGAN-V	**8.64**	<u>52.92</u>	<u>108.86</u>
Long-Video-GAN	40.40	233.26	567.78
StyleInV (ours)	<u>12.06</u>	**47.88**	**103.63**

(c) TaiChi 256^2

Method	FID (↓)	FVD_{16} (↓)	FVD_{128} (↓)
MoCoGAN-HD	73.61	315.03	622.95
DIGAN	67.24	196.77	954.93
StyleGAN-V	**35.68**	254.74	<u>477.78</u>
Long-Video-GAN	43.90	<u>248.55</u>	502.65
StyleInV (ours)	<u>41.55</u>	**185.72**	**328.90**

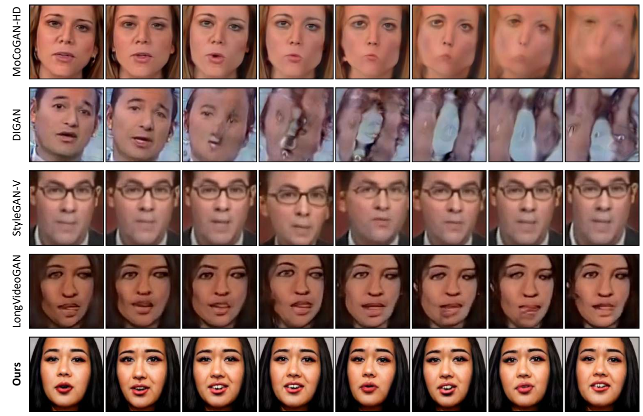
(d) SkyTimelapse 256^2

Method	FID (↓)	FVD_{16} (↓)	FVD_{128} (↓)
MoCoGAN-HD	251.81	696.58	4116.03
DIGAN	32.83	148.08	269.43
StyleGAN-V	<u>16.95</u>	<u>81.32</u>	197.83
Long-Video-GAN	25.41	116.50	**152.70**
StyleInV (ours)	**14.32**	**77.04**	<u>194.25</u>

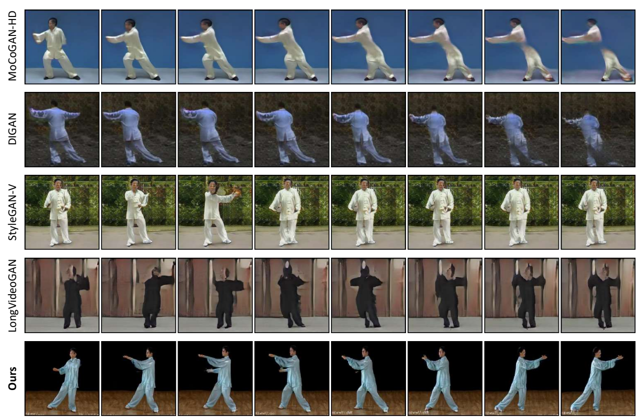
Table 2: FID, FVD_{16} and FVD_{128} results of extended experiments on DeeperForensics 256^2. We apply sparse training to MoCoGAN-HD [36] (#1) and change the preset clip length of DIGAN [52] to 128 (#2). **Bolds** indicate best. (-) indicates a smaller (better) quantitative result, while (+) indicates a larger (worse) one, compared with Table 1a.

#	Method	FID (↓)	FVD_{16} (↓)	FVD_{128} (↓)
1	[36] + Sparse Training	55.84 (-)	54.58 (-)	129.13 (-)
2	[52] + Clip 128	74.80 (-)	87.42 (+)	95.80 (-)
3	StyleInV (ours)	**54.05**	**41.58**	**53.93**

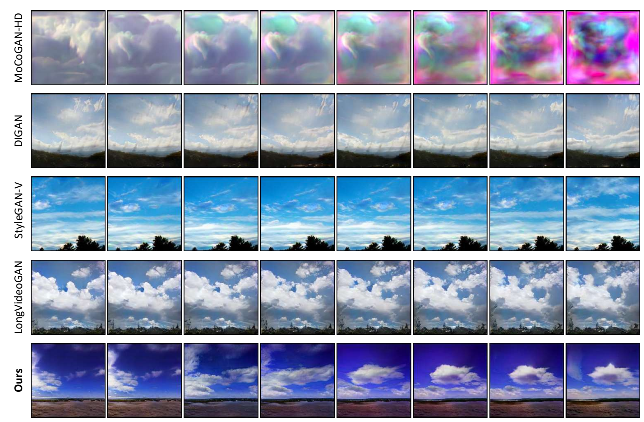
(a) DeeperForensics 256^2



(b) FaceForensics 256^2



(c) TaiChi 256^2



(d) SkyTimelapse 256^2

Figure 6: *Uncurated* samples from the existing methods on DeeperForensics 256^2, FaceForensics 256^2, TaiChi 256^2 and SkyTimelapse 256^2, respectively. We sample a 128-frame video and display every 16 frames, starting from $t = 0$.

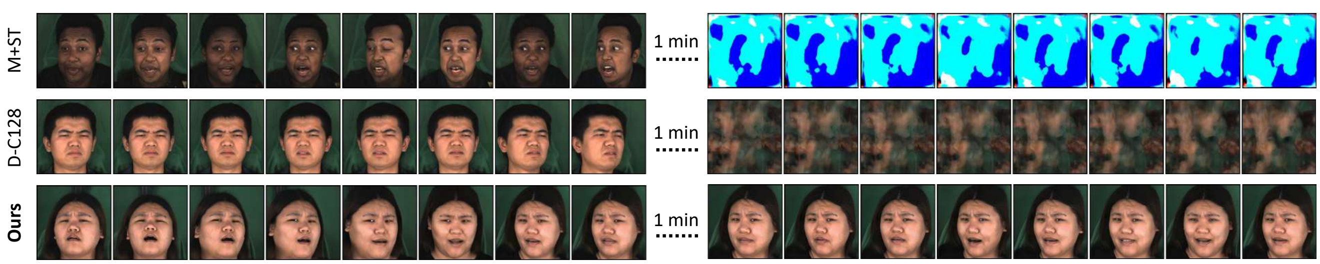


Figure 7: Qualitative comparison of extended experiments. "M+ST" and "D-C128" correspond to Table 2 (#1) and (#2), respectively. Each row shows the first and last 128 frames of a 2056-frame (68.5s) video, displayed every 16 frames.

with longer clips harms the short-term FVD_{16} result of DIGAN, which indicates its tradeoff between duration length and local temporal quality. Qualitatively, for both methods, the generated content is evidently improved within 128 frames, although the sparsely trained MoCoGAN-HD exhibits issues with identity switching. Motion collapse is still observed when MoCoGAN-HD and DIGAN generates long videos. In contrast, our method can stably generate extremely long videos without motion collapse. Our method outperforms the sparsely trained MoCoGAN-HD, demonstrating the superiority of our motion generator design.

4.2. Properties

As discussed in Section 3.5, our method has the unique advantage over state-of-the-art methods, such as StyleGAN-V and Long-Video-GAN, on its high compati-

Figure 8: Finetuning-based style transfer result. The 1st row is generated by the parent model trained on CelebV-HQ [54]. The 2nd, 3rd, and 4th row uses the StyleGAN generator fintuned on Cartoon, Arcane, and MetFace, respectively.

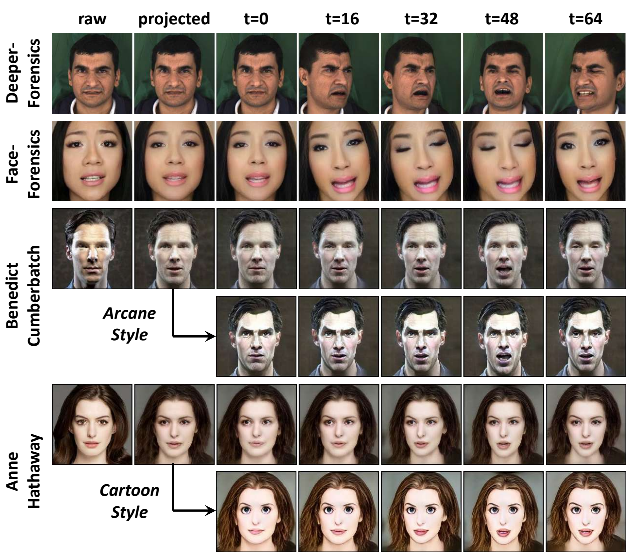


Figure 9: Initial-frame conditioned generation and style transferred results.

Table 3: Ablation result on the DeeperForensics dataset.

#	Method	FID (↓)	FVD_{16} (↓)	FVD_{128} (↓)
1	w/o inversion encoder	54.35	59.49	152.82
2	w/o FFA-APE	55.26	88.98	144.52
3	w/o Eq.(4) & Eq.(3)	52.55	67.43	58.88
4	w/o Eq.(3)	53.95	86.32	59.76
5	Ours	54.05	41.58	53.93

bility with StyleGAN-based downstream techniques.

Finetuning-based style transfer. We train the parent model (motion generator and StyleGAN) on CelebV-HQ [54], as its rich identity makes it more suitable for transfer learning. To perform style transfer, we fine-tune the StyleGAN on the Cartoon [27], MetFace [20], and Arcane datasets following the procedure outlined in Section 3.5. In Fig. 8, we show examples where the same StyleInV-generated latent sequence is decoded by different but aligned StyleGAN generators. Our method achieves satisfactory results in terms of smooth video style transfer with well-aligned face structure, identity, and expression, demonstrating its desirable properties and potential for various applications. More results can be found in the supplementary video.

Initial-frame conditioned generation. Our network supports generating a series of content given a real-world image as the initial frame. We first inverse the image into the StyleGAN2 latent space with a pSp [29] encoder, which is trained to initialize the weights of StyleInV. We treat it as the 512-dimensional initial frame latent w_0, then use it to generate a video with our StyleInV. The generated latent sequence can be also applied to a finetuned image generator to synthesize a style-transferred animation video. Through this pipeline, the real image is reconstructed twice, the first time is during the inversion process, while the second time is when synthesizing $G(\text{StyleInV}(w_0, 0))$.

When the real images are sampled from the training dataset (see Fig. 9 first two rows), $G(\text{StyleInV}(w_0, 0))$ can faithfully reconstruct the raw image and generate high-quality videos. We then test the generation quality for real images sampled out of the training set (see Fig. 9 last two rows, where we select Benedict Cumberbatch and Anne Hathaway). We use the StyleGAN2 generator and StyleInV model trained on CelebV-HQ [54] dataset as it is richer in its identities. The results show that our StyleInV network can still generate meaningful videos while reconstructing the initial frame decently, and the style transfer results are smooth and well-aligned. Please refer to the supplementary material for more results.

4.3. Ablation Studies

Motion generator design. We explore two alternative motion generator designs. The first is the autoregressive MoCoGAN-HD design, which has been discussed in Sec-

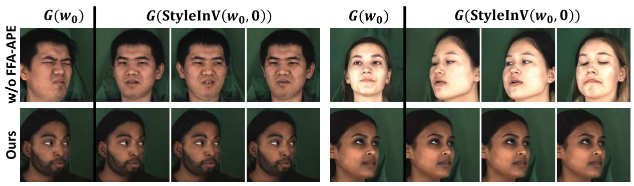


Figure 10: Ablating FFA-APE. Generate the first frame with a fixed w_0 but different temporal noise sequences. The model without FFA-APE fails to reconstruct the initial frame and generates the first frame with randomness.

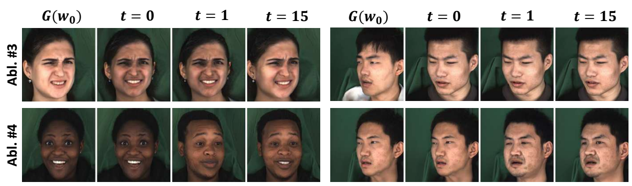


Figure 11: Ablating FFA-ST. The L_2 loss ensures the initial frame reconstruction. But without the initial frame included in the discriminator, we still cannot preserve the identity.

tion 4.1. For the second design, we remove all `Convs`, `AdaIN` and affine transform layers in Fig. 3 and let the mapped temporal style be the output of inversion encoder, *i.e.*, the residual w.r.t. initial frame latent w_0. It largely harms identity preservation. Both FVD_{16} and FVD_{128} degrade significantly as is shown in Table 3(#1).

FFA-APE. We evaluate the importance of our first-frame-aware acyclic positional encoding (FFA-APE) module by replacing it with the original design proposed by [35]. As shown in Fig. 10, the dynamic embedding of the zero timestamp prevents the network from faithfully reconstructing the initial frame. In contrast, our full method can stably realize reconstruction. In addition, the L2 loss in Eq. (4) fails to converge for the ablation method, and its gradient further harms the learning of the positional encoding module, leading to a much worse quantitative result shown in Table 3 (#2).

FFA-ST. We conduct two ablation experiments for the FFA-ST modules. In the first experiment, we remove the initial frame from the discriminator and remove the reconstruction loss (Eq.(4)) (Table 3(#3)). In the second experiment, we only remove the initial frame from the discriminator while keeping the reconstruction loss (Table 3(#4)). As shown in Fig. 11, without the reconstruction loss, our model cannot reconstruct the initial frame accurately. With the L_2 loss, the initial frame is reconstructed, but there is a sudden transition between the first two frames, and sometimes, the identity also changes, leading to a worse FVD_{16} result. These experiments demonstrate the importance of our first-frame-aware discriminator (FFA-D).

5. Conclusion

We have presented a novel approach for unconditional video generation by employing a pretrained StyleGAN image generator. The proposed StyleInV motion generator generates latents in the StyleGAN2 latent space by modulating a learning-based inversion network, and thus capable of inheriting its informative priors of the initial latent. Our network features non-autoregressive training and uniquely supports fine-tuning based style transfer. Extensive experiments demonstrate the superiority of our method in generating long and high-resolution videos, outperforming state-of-the-art baselines. Here we also briefly discuss our limitations and broader impacts.

5.1. Limitations

Inferior motion semantics on SkyTimelapse. Our motion semantics on SkyTimelapse [47] are inferior to those on other datasets. This could be due to different dataset characteristics, as videos in SkyTimelapse are not subject-centric and typically driven by global motions, which does not align perfectly with our model nature.

The impact of dataset identity richness. When the scale of facial identities in the video dataset is too small, the effects of inversion, editing, and style transfer are constrained.

Image generation quality. The generation quality of StyleGAN determines the performance upper bound of our method. The images generated by the StyleGAN2 models have artifacts in the background on SkyTimelapse [47], and lack fine details and a sense of structure on TaiChi [34].

Model training. Our approach is two-stage, requiring 7.5 and 9 GPU days each, which is more than the 8 GPU days of StyleGAN-V [35]. Despite this, StyleInV is as efficient when finetuning the hyperparameters of the video generator, since the image generator only needs to be trained once.

5.2. Broader Impacts

We believe that the potential of StyleInV can be further exploited. Our method can provide a natural solution towards mega-pixel level video generation and StyleGAN-based editing, and it might in return promote the research of learning-based GAN inversion methods.

As for the negative side, StyleInV may ease the synthesis of better-quality fake videos with threats. We believe that it can be alleviated by developing more advanced falsified media detection methods or contributing larger-scale and higher-quality forgery detection datasets.

Acknowledgement. This work is supported by the National Research Foundation, Singapore under its AI Singapore Programme (AISG Award No: AISG2-PhD-2022-01-030). It is also supported under the RIE2020 Industry Alignment Fund Industry Collaboration Projects (IAF-ICP) Funding Initiative, as well as cash and in-kind contribution from the industry partner(s). We thank Shuai Yang for his help in this work.

References

[1] Rameen Abdal, Peihao Zhu, Niloy J Mitra, and Peter Wonka. StyleFlow: Attribute-conditioned exploration of StyleGAN-generated images using conditional continuous normalizing flows. *ACM TOG*, 40:1–21, 2021. 2

[2] Yuval Alaluf, Or Patashnik, and Daniel Cohen-Or. Only a matter of style: Age transformation using a style-based regression model. *ACM TOG*, 40:1–12, 2021. 2

[3] Yuval Alaluf, Or Patashnik, and Daniel Cohen-Or. ReStyle: A residual-based StyleGAN encoder via iterative refinement. In *ICCV*, 2021. 2

[4] Yuval Alaluf, Omer Tov, Ron Mokady, Rinon Gal, and Amit H Bermano. HyperStyle: StyleGAN inversion with hypernetworks for real image editing. In *CVPR*, 2022. 2

[5] David Bau, Hendrik Strobelt, William Peebles, Jonas Wulff, Bolei Zhou, Jun-Yan Zhu, and Antonio Torralba. Semantic photo manipulation with a generative image prior. *ACM TOG*, 38:1–11, 2019. 2

[6] David Bau, Jun-Yan Zhu, Jonas Wulff, William Peebles, Hendrik Strobelt, Bolei Zhou, and Antonio Torralba. Seeing what a GAN cannot generate. In *ICCV*, 2019. 2

[7] Tim Brooks, Janne Hellsten, Miika Aittala, Ting-Chun Wang, Timo Aila, Jaakko Lehtinen, Ming-Yu Liu, Alexei A Efros, and Tero Karras. Generating long videos of dynamic scenes. In *NeurIPS*, 2022. 2, 3, 5

[8] Lucy Chai, Jonas Wulff, and Phillip Isola. Using latent space regression to analyze and leverage compositionality in GANs. In *ICLR*, 2021. 2

[9] Lucy Chai, Jun-Yan Zhu, Eli Shechtman, Phillip Isola, and Richard Zhang. Ensembling with deep generative views. In *CVPR*, 2021. 2

[10] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *ICML*, 2020. 3

[11] Aidan Clark, Jeff Donahue, and Karen Simonyan. Adversarial video generation on complex datasets. *arXiv preprint*, arXiv:1907.06571, 2019. 1, 3

[12] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. ArcFace: Additive angular margin loss for deep face recognition. In *CVPR*, 2019. 5

[13] Gereon Fox, Ayush Tewari, Mohamed Elgharib, and Christian Theobalt. StyleVideoGAN: A temporal generative model using a pretrained StyleGAN. In *BMVC*, 2021. 3

[14] 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 *NeurIPS*, 2017. 6

[15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 3

[16] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In *NeurIPS*, 2022. 3

[17] Liming Jiang, Ren Li, Wayne Wu, Chen Qian, and Chen Change Loy. DeeperForensics-1.0: A large-scale dataset for real-world face forgery detection. In *CVPR*, 2020. 2, 5

[18] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016. 5

[19] Emmanuel Kahembwe and Subramanian Ramamoorthy. Lower dimensional kernels for video discriminators. *Neural Networks*, 132:506–520, 2020. 3

[20] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In *NeurIPS*, 2020. 5, 8

[21] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. In *NeurIPS*, 2021. 3

[22] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019. 1, 2, 3, 5

[23] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *CVPR*, 2020. 2

[24] 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 *CVPR*, 2023. 3

[25] Jacek Naruniec, Leonhard Helminger, Christopher Schroers, and Romann M Weber. High-resolution neural face swapping for visual effects. In *CGF*, 2020. 5

[26] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. StyleCLIP: Text-driven manipulation of StyleGAN imagery. In *ICCV*, 2021. 5

[27] Justin NM Pinkney and Doron Adler. Resolution dependent GAN interpolation for controllable image synthesis between domains. *arXiv preprint*, arXiv:2010.05334, 2020. 5, 8

[28] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with VQ-VAE-2. In *NeurIPS*, 2019. 3

[29] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a StyleGAN encoder for image-to-image translation. In *CVPR*, 2021. 2, 5, 8

[30] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 3

[31] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. FaceForensics: A large-scale video dataset for forgery detection in human faces. *arXiv preprint*, arXiv:1803.09179, 2018. 2, 5

[32] Masaki Saito, Eiichi Matsumoto, and Shunta Saito. Temporal generative adversarial nets with singular value clipping. In *ICCV*, 2017. 1, 3

[33] Masaki Saito, Shunta Saito, Masanori Koyama, and Sosuke Kobayashi. Train sparsely, generate densely: Memory-efficient unsupervised training of high-resolution temporal GAN. *IJCV*, 128:2586–2606, 2020. 1, 3, 5

[34] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. In *NeurIPS*, 2019. 2, 5, 9

[35] Ivan Skorokhodov, Sergey Tulyakov, and Mohamed Elhoseiny. StyleGAN-V: A continuous video generator with the

price, image quality and perks of StyleGAN2. In *CVPR*, 2022. 2, 3, 4, 5, 6, 9

[36] 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 *ICLR*, 2021. 1, 2, 3, 5, 6

[37] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for StyleGAN image manipulation. *ACM TOG*, 40:1–14, 2021. 2

[38] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. MoCoGAN: Decomposing motion and content for video generation. In *CVPR*, 2018. 1, 3

[39] 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. 6

[40] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *JMLR*, 9(11), 2008. 4

[41] Vikram Voleti, Alexia Jolicoeur-Martineau, and Chris Pal. MCVD: Masked conditional video diffusion for prediction, generation, and interpolation. 2022. 3

[42] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *NeurIPS*, 2016. 1, 3

[43] Hui-Po Wang, Ning Yu, and Mario Fritz. Hijack-GAN: Unintended-use of pretrained, black-box GANs. In *CVPR*, 2021. 2

[44] Tianyi Wei, Dongdong Chen, Wenbo Zhou, Jing Liao, Weiming Zhang, Lu Yuan, Gang Hua, and Nenghai Yu. E2Style: Improve the efficiency and effectiveness of StyleGAN inversion. *TIP*, 31:3267–3280, 2022. 2

[45] Zongze Wu, Dani Lischinski, and Eli Shechtman. StyleSpace analysis: Disentangled controls for StyleGAN image generation. In *CVPR*, 2021. 2

[46] Weihao Xia, Yulun Zhang, Yujiu Yang, Jing-Hao Xue, Bolei Zhou, and Ming-Hsuan Yang. GAN inversion: A survey. *TPAMI*, 45:3121–3138, 2022. 2

[47] Wei Xiong, Wenhan Luo, Lin Ma, Wei Liu, and Jiebo Luo. Learning to generate time-lapse videos using multi-stage dynamic generative adversarial networks. In *CVPR*, 2018. 2, 5, 9

[48] Yangyang Xu, Yong Du, Wenpeng Xiao, Xuemiao Xu, and Shengfeng He. From continuity to editability: Inverting GANs with consecutive images. In *ICCV*, pages 13910–13918, 2021. 2

[49] Yinghao Xu, Yujun Shen, Jiapeng Zhu, Ceyuan Yang, and Bolei Zhou. Generative hierarchical features from synthesizing images. In *CVPR*, 2021. 2

[50] Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. VideoGPT: Video generation using VQ-VAE and Transformers. *arXiv preprint*, arXiv:2104.10157, 2021. 1, 3

[51] Siyuan Yang, Lu Zhang, Yu Liu, Zhizhuo Jiang, and You He. Video diffusion models with local-global context guidance. In *IJCAI*, 2023. 3

[52] 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 *ICLR*, 2022. 2, 3, 4, 5, 6

[53] Qihang Zhang, Ceyuan Yang, Yujun Shen, Yinghao Xu, and Bolei Zhou. Towards smooth video composition. In *ICLR*, 2023. 3

[54] Hao Zhu, Wayne Wu, Wentao Zhu, Liming Jiang, Siwei Tang, Li Zhang, Ziwei Liu, and Chen Change Loy. CelebV-HQ: A large-scale video facial attributes dataset. In *ECCV*, 2022. 8

[55] Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. In-domain GAN inversion for real image editing. In *ECCV*, 2020. 2

[56] Peihao Zhu, Rameen Abdal, Yipeng Qin, John Femiani, and Peter Wonka. Improved StyleGAN embedding: Where are the good latents? *arXiv preprint*, arXiv:2012.09036, 2020. 2

[57] Peiye Zhuang, Oluwasanmi Koyejo, and Alexander Schwing. Enjoy your editing: Controllable GANs for image editing via latent space navigation. In *ICLR*, 2021. 2