Long Video Generation with Time-Agnostic VQGAN and Time-Sensitive Transformer

Songwei Ge[1⋆], Thomas Hayes[2], Harry Yang[2], Xi Yin[2], Guan Pang[2]
David Jacobs[1], Jia-Bin Huang[1,2], and Devi Parikh[2,3]

[1]University of Maryland [2]Meta AI [3]Georgia Tech

Abstract. Videos are created to express emotion, exchange information, and share experiences. Video synthesis has intrigued researchers for a long time. Despite the rapid progress driven by advances in visual synthesis, most existing studies focus on improving the frames' quality and the transitions between them, while little progress has been made in generating longer videos. In this paper, we present a method that builds on 3D-VQGAN and transformers to generate videos with thousands of frames. Our evaluation shows that our model trained on 16-frame video clips from standard benchmarks such as UCF-101, Sky Time-lapse, and Taichi-HD datasets can generate diverse, coherent, and high-quality long videos. We also showcase conditional extensions of our approach for generating meaningful long videos by incorporating temporal information with text and audio. Videos and code can be found at https://songweige.github.io/projects/tats.

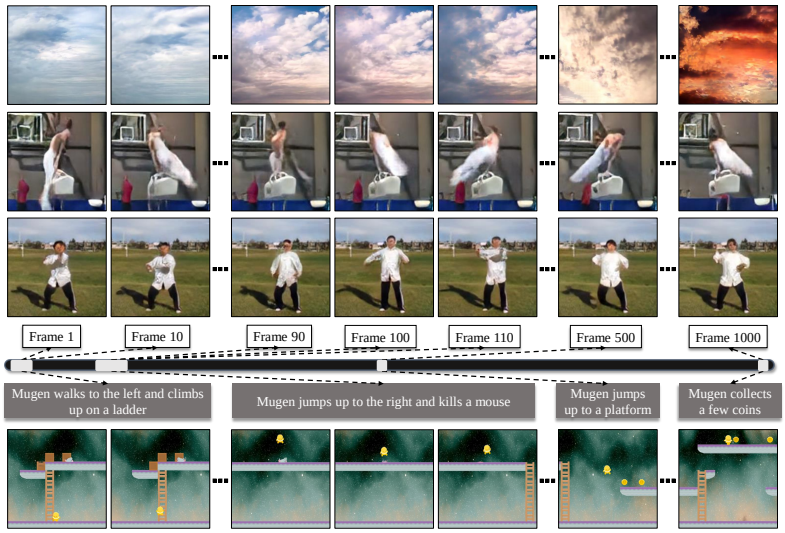


Fig. 1: **Long videos generated by our model TATS with** 1024 **frames.**

⋆ Work done primarily during an internship at Meta AI.

1 Introduction

From conveying emotions that break language and cultural barriers to being the most popular medium on social platforms, videos are arguably the most informative, expressive, diverse, and entertaining visual form. Video synthesis has been an exciting yet long-standing problem. The challenges include not only achieving high visual quality in each frame and a natural transitions between frames, but a consistent theme, and even a meaningful storyline throughout the video. The former has been the focus of many existing studies [49,36,9,45,57] working with tens of frames. The latter is largely unexplored and requires the ability to model *long-range temporal dependence* in videos with many more frames.

Prior works and their limitations. *GAN-based methods* [49,36,9,26] can generate plausible short videos, but extending them to longer videos requires prohibitively high memory and time cost of training and inference.[1] *Autoregressive methods* alleviate the training cost constraints through *sequential prediction*. For example, RNNs and LSTMs generate temporal noise vectors for an image generator [36,47,9,37,27,45]; transformers either directly generate pixel values [20,52] or indirectly predict latent tokens [29,10,33,54,25,57]. These approaches circumvent training on the long videos directly by unrolling the RNN states or using a sliding window during inference. Recent works use *implicit neural representations* to reduce cost by directly mapping temporal positions to either pixels [58] or StyleGAN [23] feature map values [41]. However, the visual quality of the generated frames deteriorates quickly when performing generation beyond the training video length for all such methods as shown in Figure 2.

Our work. We tackle the problem of long video generation. Building upon the recent advances of VQGAN [10] for high-resolution *image generation*, we first develop a baseline by extending the 2D-VQGAN to 3D (2D space and 1D time) for modeling videos. This naively extended method, however, fails to produce high-quality, coherent long videos. Our work investigates the model design and identifies simple changes that significantly improve the capability to generate long videos of thousands of frames without quality degradation when conditioning on no or weak information. Our core insights lie in 1) removing the undesired dependence on time from VQGAN and 2) enabling the transformer to capture long-range temporal dependence. Below we outline these two key ideas.

Time-agnostic VQGAN. Our model is trained on short video clips, e.g., 16 frames, like the previous methods [49,36,47,9,37,45]. At inference time, we use a sliding window approach [5] on the transformer to sample tokens for a longer length. The sliding window repeatedly appends the most recently generated tokens to the partial sequence and drops the earliest tokens to maintain a fixed sequence length. However, applying a sliding attention window to 3D-VQVAE (e.g. VideoGPT [57]) or 3D-VQGAN fails to preserve the video quality *beyond the training length*, as shown in Figure 2. The reason turns out to be that the zero

[1] Training DVD-GAN [9] or DVD-GAN-FP [26] on 16-frame videos requires 32-512 TPU replicas and 12-96 hours.

Fig. 2: Video generation results with a vanilla video VQGAN and a time-agnostic VQGAN, within and beyond the training length using sliding window attention.

paddings used in these models *corrupts* the latent tokens and results in token sequences at the inference time that are drastically different from those observed during training when using sliding window. The amount of corruption depends on the temporal position of the token. [2] We address this issue by using *replicate padding* that mitigates the corruption by better approximating the real frames and brings no computational overhead. As a result, the transformer trained on the tokens encoded by our time-agnostic VQGAN effectively preserves the visual quality *beyond the training video length*.

Time-sensitive transformer. While removing the temporal dependence in VQGAN is desirable, long video generation certainly needs temporal information! This is necessary to model long-range dependence through the video and follow a sequence of events a storyline might suggest. While transformers can generate arbitrarily long sequences, errors tend to accumulate, leading to quality degradation for long video generation. To mitigate this, we introduce a hierarchical architecture where an *autoregressive transformer* first generates a set of sparse latent frames, providing a more global structure. Then an *interpolation transformer* fills in the skipped frames autoregressively while attending to the generated sparse frames on both ends. With these modifications, our transformer models long videos more effectively and efficiently. Together with our proposed 3D-VQGAN, we call our model Time-Agnostic VQGAN and Time-Sensitive Transformer (TATS). We highlight the capability of TATS by showing generated video samples of 1024 frames in Figure 1.

Our results. We evaluate our model on several video generation benchmarks. We first consider a standard short video generation setting. Then, we carefully analyze its effectiveness for long video generation, comparing it against several recent models. Given that the evaluation of long video generation has not been well studied, we generalize several popular metrics for this task considering important evaluation axes for video generation models, including long term quality

[2] The large spatial span in image synthesis [10] disguises the issue. When applying sliding window to border tokens, the problem resurfaces in supp. mat. Figure 12.

and coherence. Our model achieves state-of-the-art short and long video generation results on the UCF-101 [42], Sky Time-lapse [55], Taichi-HD [38], and AudioSet-Drum [12] datasets. We further demonstrate the effectiveness of our model by conditioning on temporal information such as text and audio.

Our contributions.

- We identify the undesired temporal dependence introduced by the zero paddings in VQGAN as a cause of the ineffectiveness of applying a sliding window for long video generation. We propose a simple yet effective fix.
- We propose a hierarchical transformer that can model longer time dependence and delay the quality degradation, and show that our model can generate meaningful videos according to the story flow provided by text or audio.
- To our knowledge, we are the first to generate long videos and analyze their quality. We do so by generalizing several popular metrics to a longer video span and showing that our model can generate more diverse, coherent, and higher-quality long videos.

2 Methodology

In this section, we first briefly recap the VQGAN framework and describe its extension to video generation. Next, we present our time-agnostic VQGAN and time-sensitive transformer models for long video generation (Figure 3).

2.1 Extending the VQGAN framework for video generation

Vector Quantised Variational AutoEncoder (VQVAE) [29] uses a discrete bottleneck as the latent space for reconstruction. An autoregressive model such as a transformer is then used to model the prior distribution of the latent space. VQGAN [10] is a variant of VQVAE that uses perceptual and GAN losses to achieve better reconstruction quality when increasing the bottleneck compression rates.

Vanilla video VQGAN. We adapt the VQGAN architecture for video generation by replacing its 2D convolution operations with 3D convolutions. Given a video $\mathbf{x} \in \mathbb{R}^{T\times H\times W\times 3}$, the VQVAE consists of an encoder $f_{\mathcal{E}}$ and a decoder $f_{\mathcal{G}}$. The discrete latent tokens $\mathbf{z} = \mathbf{q}(f_{\mathcal{E}}(\mathbf{x})) \in \mathbb{Z}^{t\times h\times w}$ with embeddings $\mathbf{c}_z \in \mathbb{R}^{t\times h\times w\times c}$ are computed using a quantization operation $\mathbf{q}$ which applies nearest neighbor search using a trainable codebook $\mathcal{C} = \{\mathbf{c}_i\}_{i=1}^{K}$. The embeddings of the tokens are then fed into the decoder to reconstruct the input $\hat{\mathbf{x}} = f_{\mathcal{G}}(\mathbf{c}_z)$. The VQVAE is trained with the following loss:

$$\mathcal{L}_{\text{vqvae}} = \underbrace{\|\mathbf{x} - \hat{\mathbf{x}}\|_1}_{\mathcal{L}_{\text{rec}}} + \underbrace{\|\text{sg}[f_{\mathcal{E}}(\mathbf{x})] - \mathbf{c}_z\|_2^2}_{\mathcal{L}_{\text{codebook}}} + \underbrace{\beta\|\text{sg}\,[\mathbf{c}_z] - f_{\mathcal{E}}(\mathbf{x})\|_2^2}_{\mathcal{L}_{\text{commit}}},$$

where sg is a stop-gradient operation and we use $\beta = 0.25$ following the VQGAN paper [10]. We optimize $\mathcal{L}_{\text{codebook}}$ using an EMA update and circumvent the non-differentiable quantization step $\mathbf{q}$ with a straight-through gradient estimator [29].

VQGAN additionally adopts a perceptual loss [18,59] and a discriminator $f_{\mathcal{D}}$ to improve the reconstruction quality. Similar to other GAN-based video generation models [47,9], we use two types of discriminators in our model - a spatial discriminator $f_{\mathcal{D}_s}$ that takes in random reconstructed frames $\hat{\mathbf{x}}_i \in \mathbb{R}^{H\times W\times 3}$ to encourage frame quality and a temporal discriminator $f_{\mathcal{D}_t}$ that takes in the entire reconstruction $\hat{\mathbf{x}} \in \mathbb{R}^{T\times H\times W\times 3}$ to penalize implausible motions:

$$\mathcal{L}_{\text{disc}} = \log f_{\mathcal{D}_{s/t}}(\mathbf{x}) + \log(1 - f_{\mathcal{D}_{s/t}}(\hat{\mathbf{x}}))$$

We also use feature matching losses [51,50] to stabilize the GAN training:

$$\mathcal{L}_{\text{match}} = \sum_i p_i \left\| f^{(i)}_{\mathcal{D}_{s/t}/\text{VGG}}(\hat{\mathbf{x}}) - f^{(i)}_{\mathcal{D}_{s/t}/\text{VGG}}(\mathbf{x}) \right\|_1,$$

where $f^{(i)}_{\mathcal{D}_{s/t}/\text{VGG}}$ denotes the i^{th} layer of either a trained VGG network [39] or discriminators with a scaling factor p_i. When using a VGG network, this loss is known as the perceptual loss [59]. p_i is a learned constant for VGG and the reciprocal of the number of elements in the layer for discriminators. Our overall VQGAN training objective is as follows:

$$\min_{f_{\mathcal{E}}, f_{\mathcal{G}}, \mathcal{C}} \left(\max_{f_{\mathcal{D}_s}, f_{\mathcal{D}_t}} (\lambda_{\text{disc}} \mathcal{L}_{\text{disc}}) \right) + \\ \min_{f_{\mathcal{E}}, f_{\mathcal{G}}, \mathcal{C}} (\lambda_{\text{match}} \mathcal{L}_{\text{match}} + \lambda_{\text{rec}} \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{codebook}} + \beta \mathcal{L}_{\text{commit}})$$

Directly applying GAN losses to VideoGPT [57] or 3D VQGAN [10] leads to training stability issues. In addition to the feature matching loss, we discuss other necessary architecture choices and training heuristics in supp. mat. A.1.

Autoregressive prior model. After training the video VQGAN, each video can be encoded into its discrete representation $\mathbf{z} = \mathbf{q}(f_{\mathcal{E}}(\mathbf{x}))$. Following VQGAN [10], we unroll these tokens into a 1D sequence using the row-major order frame by frame. We then train a transformer $f_{\mathcal{T}}$ to model the prior categorical distribution of $\mathbf{z}$ in the dataset autoregressively:

$$p(\mathbf{z}) = p(\mathbf{z}_0) \prod_{i=0}^{t\times h\times w-1} p(\mathbf{z}_{i+1}|\mathbf{z}_{0:i}),$$

where $p(\mathbf{z}_{i+1}|\mathbf{z}_{0:i}) = f_{\mathcal{T}}(\mathbf{z}_{0:i})$ and $\mathbf{z}_0$ is given as the start of sequence token. We train the transformer to minimize the negative log-likelihood over training samples:

$$\mathcal{L}_{\text{transformer}} = \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z}_{\text{data}})}[-\log p(\mathbf{z})]$$

At inference time, we randomly sample video tokens from the predicted categorical distribution $p(\mathbf{z}_{i+1}|\mathbf{z}_{0:i})$ in sequence and feed them into the decoder to generate the videos $\hat{\mathbf{x}} = f_{\mathcal{G}}(\hat{\mathbf{c}}_z)$. To synthesize videos longer than the training length, we generalize sliding attention window for our use [5] . A similar idea has been used in 2D to generate images of higher resolution [10]. We denote the

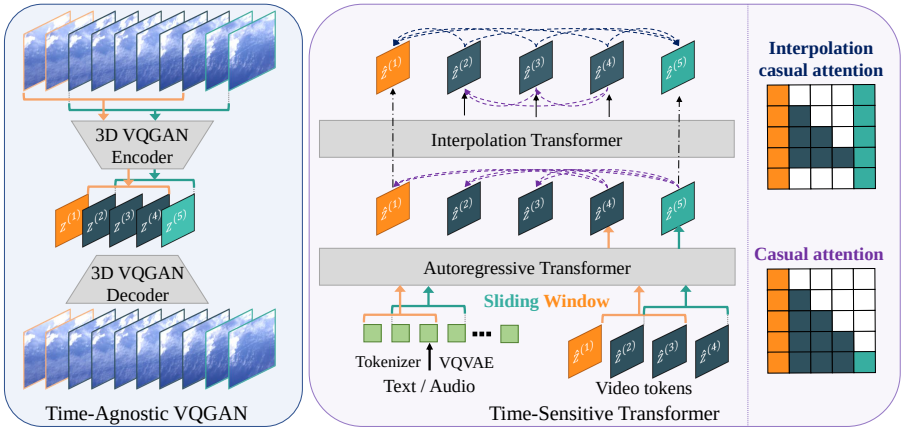


Fig. 3: **Overview of the proposed framework.** Our model contains two modules: time-agnostic VQGAN and time-sensitive transformer. The former compresses the videos both temporally and spatially into discrete tokens without injecting any dependence on the relative temporal position, which allows the usage of a sliding window during inference for longer video generation. The latter uses a hierarchical transformer for capturing longer temporal dependence.

j^{th} temporal slice of $\mathbf{z}$ to be $\mathbf{z}^{(j)} \in \mathbb{Z}^{h \times w}$, where $0 \leq j \leq t-1$. For instance, to generate $\mathbf{z}^{(t)}$ that is beyond the training length, we condition on the $t-1$ slices before it to match the transformer sequence length $p(\mathbf{z}^{(t)}|\mathbf{z}^{(1:t-1)}) = f_{\mathcal{T}}(\mathbf{z}^{(1:t-1)})$. However, as shown in Figure 2, when paired with sliding window attention, the vanilla video VQGAN and transformer cannot generate longer videos without quality degradation. Next, we discuss the reason and a simple yet effective fix.

2.2 Time-Agnostic VQGAN

When the Markov property holds, a transformer with a sliding window can generate arbitrarily long sequences as demonstrated in long article generation [5]. However, a crucial premise that has been overlooked is that the transformer needs to see sequences that start with tokens similar to $\mathbf{z}^{(1:t-1)}$ during training to predict token $\mathbf{z}^{t}$. This premise breaks down in VQGAN. We provide some intuitions about the reason and defer a detailed discussion to supp. mat. A.2.

Different from natural language modeling where the tokens come from realistic data, VQGAN tokens are produced by an encoder $f_{\mathcal{E}}$ which by default adopts zero paddings for the desired output shape. When a short video clip is encoded, the zero paddings in the temporal dimension also get encoded and affect the output tokens, causing an unbalanced effects to tokens at different temporal position [17,24,56,2]. The tokens closer to the temporal boundary will be affected more significantly. As a result, for real data to match $\mathbf{z}^{(1:t-1)}$, they have to contain these zero-frames, which is not the case in practice. Therefore,

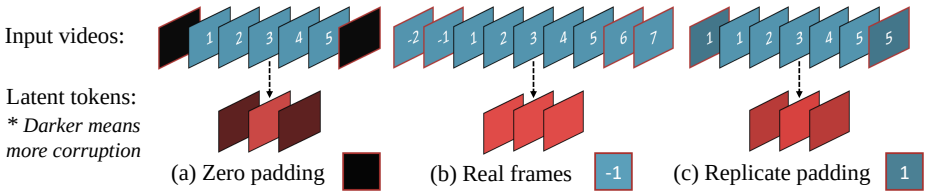


Fig. 4: **Illustration of the temporal effects induced by different paddings**. Real frame padding makes the encoder temporally shift-equivariant[3] but introduces extra computations. Replicate padding makes decent approximation to the real frames while bringing no computational overhead.

removing those paddings in the temporal dimension is crucial for making the encoder *time-agnostic* and enabling sliding window.

After removing all the padding, one needs to pad real frames to both ends of the input videos to obtain the desired output size [21,41]. Note that the zeros are also padded in the intermediate layers when applied, and importantly, they need not be computed. But, if we want to pad with realistic values, the input needs to be padded long enough to cover the entire receptive field, and all these extra values in the intermediate layers need to be computed. The number of needed real frames can be as large as $\mathcal{O}(Ld)$, where L is the number of layers and d is the compression rate in the temporal direction of the video. Although padding with real frames makes the encoder fully time agnostic, it can be expensive with a large compression rate or a deep network. In addition, frames near both ends of the videos need to be discarded for not having enough real frames to pad, and consequently some short videos in the dataset may be completely dropped.

Instead of padding with real values, we propose to approximate real frames with the values close to them. An assumption, which is often referred to as the "boring videos" assumption in the previous literature [6], is to assume that the videos are frozen beyond the given length. Following the assumption, the last boundary slices can be used for padding without computation. More importantly, this can be readily implemented by the replicate padding mode using a standard machine learning library. An illustration of the effects induced by different paddings can be found in Figure 4. Other reasonable assumptions can also be adopted and may correspond to different padding modes. For instance, the reflected padding mode can be used if videos are assumed to play in reverse beyond their length, which introduces more realistic motions than the frozen frames. In supp. mat. A.3, we provide a careful analysis of the time dependence when applying different padding types and numbers of padded real frames. We find that the replicate padding alone resolves the issue well in practice and inherits the low computational cost merit of zero paddings. Therefore, in the following experiments, we use the replicate paddings and no real frames padded.

[3] The expressions *temporally shift-equivariant* and *time-agnostic* will be used interchangeably hereinafter.

2.3 Time-Sensitive Transformer

The time-agnostic property of the VQGAN makes it feasible to generate long videos using a transformer with sliding window attention. However, long video generation does need temporal information! To maintain a consistent theme running from the beginning of the video to the end requires the capacity to model long-range dependence. And besides, the spirit of a long video is in its underlying story, which requires both predicting the motions in the next few frames and the ability to plan on how the events in the video proceed. This section discusses the time-sensitive transformer for improved long video generation.

Due to the probabilistic nature of transformers, errors can be introduced when sampling from a categorical distribution, which then accumulate over time. One common strategy to improve the long-term capacity is to use a hierarchical model [11,7] to reduce the chance of drifting away from the target theme. Specifically, we propose to condition one interpolation transformer on the tokens generated by the other autoregressive transformer that outputs more sparsely sampled tokens. The autoregressive transformer is trained in a standard way but on the sampled frames with larger intervals. For the interpolation transformer, it fills in the missing frames between any two adjacent frames generated by the autoregressive transformer. To do so, we propose interpolation attention as shown in Figure 3, where the predicted tokens attend to both the previously generated tokens in a causal way and the tokens from the sparsely sampled frames at both ends. A more detailed description can be found in supp. mat. A.4. We consider an autoregressive transformer that generates $4\times$ more sparse video tokens. Generalizing this model to even more extreme cases such as video generation based on key-frames would be an interesting future work.

Another simple yet effective way to improve the long period coherence is to provide the underlying "storyline" of the desired video directly. The VQVAE framework has been shown effective in conditional image generation [10,34]. We hence consider several kinds of conditional information that provide additional temporal information such as audio [8], text [53], and so on. To utilize conditional information for long video generation, we use either a tokenizer or an additional VQVAE to discretize the text or audio and prepend the obtained tokens to the video tokens and remove the start of the sequence token $\mathbf{z}_0$. At inference time, we extend the sliding window by simultaneously applying it to the conditioned tokens and video tokens.

3 Experiments

In this section, we evaluate the proposed method on several benchmark datasets for video generation with an emphasis on long video generation.

3.1 Experimental Setups

Datasets and evaluation. We show results on UCF-101 [42], Sky Time-lapse [55], and Taichi-HD [38] for unconditional or class-conditioned video generation with

Table 1: Quantitative results of standard video generation on different datasets. We report FVD and KVD on the Taichi-HD and Sky Time-lapse datasets, IS and FVD on the UCF-101 dataset, SSIM and PSNR at the 45$^{\text{th}}$ frame on the AudioSet-Drum dataset. * denotes training on the entire UCF-101 dataset instead of the train split. The class column indicates whether the class labels are used as conditional information.

(a) Sky Time-lapse

Method	FVD (↓)	KVD (↓)
MoCoGAN-HD	183.6$_{\pm 5.2}$	13.9$_{\pm 0.7}$
DIGAN	**114.6**$_{\pm 4.9}$	6.8$_{\pm 0.5}$
TATS-base	132.56$_{\pm 2.6}$	**5.7**$_{\pm 0.3}$

(b) TaiChi-HD

Method	FVD (↓)	KVD (↓)
MoCoGAN-HD	144.7$_{\pm 6.0}$	25.4$_{\pm 1.9}$
DIGAN	128.1$_{\pm 4.9}$	20.6$_{\pm 1.1}$
TATS-base	**94.60**$_{\pm 2.7}$	**9.8**$_{\pm 1.0}$

(c) AudioSet-Drum

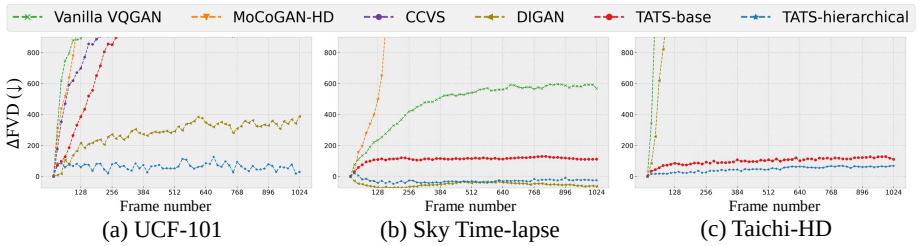
Method	SSIM (↑)	PSNR (↑)
SVG-LP	0.510$_{\pm 0.008}$	13.5$_{\pm 0.1}$
Vougioukas *et al.*	0.896$_{\pm 0.015}$	23.3$_{\pm 0.3}$
Sound2Sight	0.947$_{\pm 0.007}$	27.0$_{\pm 0.3}$
CCVS	0.945$_{\pm 0.008}$	27.3$_{\pm 0.5}$
TATS-base	**0.964**$_{\pm 0.005}$	**27.7**$_{\pm 0.4}$

(d) UCF-101

Method	Class	IS (↑)	FVD (↓)
VGAN	✓	8.31$_{\pm .09}$	-
TGAN	✗	11.85$_{\pm .07}$	-
TGAN	✓	15.83$_{\pm .18}$	-
MoCoGAN	✓	12.42$_{\pm .07}$	-
ProgressiveVGAN	✓	14.56$_{\pm .05}$	-
LDVD-GAN	✗	22.91$_{\pm .19}$	-
VideoGPT	✗	24.69$_{\pm .30}$	-
TGANv2	✓	28.87$_{\pm .67}$	1209$_{\pm 28}$
DVD-GAN*	✓	27.38$_{\pm .53}$	-
MoCoGAN-HD*	✗	32.36	838
DIGAN	✗	29.71$_{\pm .53}$	655$_{\pm 22}$
DIGAN*	✗	32.70$_{\pm .35}$	577$_{\pm 21}$
CCVS*+Real frame	✗	41.37$_{\pm .39}$	389$_{\pm 14}$
CCVS*+StyleGAN	✗	24.47$_{\pm .13}$	386$_{\pm 15}$
StyleGAN-V	✗	23.94$_{\pm .73}$	-
CogVideo	✓	50.46	626
Video Diffusion	✗	57.00$_{\pm .62}$	**295**$_{\pm 3}$
Real data	-	90.52	-
TATS-base	✗	57.63$_{\pm .24}$	420$_{\pm 18}$
TATS-base	✓	**79.28**$_{\pm .38}$	332$_{\pm 18}$

128 × 128 resolution following [58], AudioSet-Drum [12] for audio-conditioned video generation with 64 × 64 resolution following [25], and MUGEN [14] with 256 × 256 resolution for text-conditioned video generation. We follow the previous methods [45,58] to use Fréchet Video Distance (FVD) [48] and Kernel Video Distance (KVD) [48] as the evaluation metrics on UCF-101, Sky Time-lapse, and Taichi-HD datasets. In addition, we follow the methods evaluated on UCF-101 [9,25] to report the Inception Score (IS) [37] calculated by a trained C3D model [46]. For audio-conditioned generation evaluation, we measure the SSIM and PSNR at the 45$^{\text{th}}$ frame which is the longest videos considered in the previous methods [8,25]. See supp. mat. B.1 for more details about the datasets.

Training details. To compare our methods with previous ones, we train our VQGAN on videos with 16 frames. We adopt a compression rate $d = 4$ in temporal dimension and $d = 8$ in spatial dimensions. For transformer models, we train a decoder-only transformer with size between GPT-1 [31] and GPT-2 [32]



(a) UCF-101 (b) Sky Time-lapse (c) Taichi-HD

Fig. 5: **Quality degradation.** FVD changes of non-overlapping 16-frame clips from the long videos generated by models trained on the UCF-101, Sky Time-lapse, and Taichi-HD datasets. The lower value indicates the slower degradation.

for a consideration of computational cost. We refer to our model with a single autoregressive transformer as **TATS-base** and the proposed hierarchical transformer as **TATS-hierarchical**. For audio-conditioned model, we train another VQGAN to compress the Short-Time Fourier Transform (STFT) data into a discrete space. For text-conditioned model, we use a BPE tokenizer pretrained by CLIP [30]. See supp. mat. B.2 for more details about the training and inference, and supp. mat. B.3. for the comparison of computational costs.

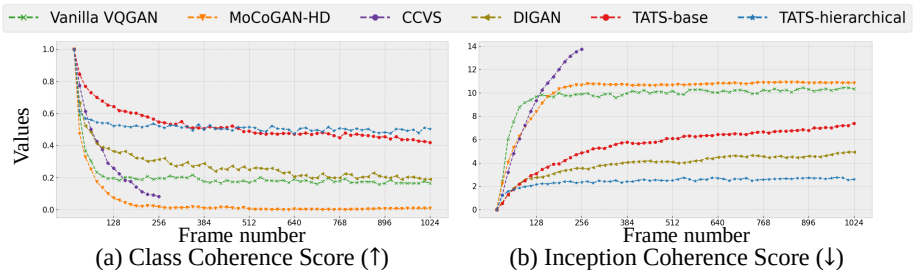
3.2 Quantitative Evaluation on Short Video Generation

In this section, we demonstrate the effectiveness of our TATS-base model under a standard short video generation setting, where only 16 frames are generated for each video. The quantitative results are shown in Table 1.

Our model achieves state-of-art FVD and KVD on the UCF-101 and Taichi-HD datasets, and state-of-art KVD on the Sky Time-lapse dataset for unconditional video generation, and improved the quality on the AudioSet-Drum for audio-conditioned video generation. TATS-base improves on IS by 76.2% over previous method with synthetic initial frames [25] and is competitive against concurrent works [41,16,15]. On the UCF-101 dataset, we also explore generation with class labels as conditional information. Following the previous method [36], we sample labels from the prior distribution as the input to the generation. We find that conditioning on the labels significantly eases the transformer modeling task and boosts the generation quality, improving IS from 57.63 to 79.28, which is close to the upper bound shown in the "Real data" row [1,19]. This has been extensively observed in image generation [3,10] but not quite yet revealed in the video generation. TATS-base significantly advances the IS over the previous methods, demonstrating its power in modeling diverse video datasets.

3.3 Quantitative Evaluation on Long Video Generation

Quantitatively evaluating long video generation results has been under explored. In this section, we propose several metrics by generalizing existing metrics to



(a) Class Coherence Score (↑) (b) Inception Coherence Score (↓)

Fig. 6: **Video coherency.** CCS and ICS values of every 16-frame clip extracted from long videos generated by different models trained on the UCF-101 dataset.

evaluate the crucial aspects of the long videos. We generate 512 videos with 1024 frames on each of the Sky Time-lapse, Taichi-HD, and UCF-101 datasets. We compare our proposed methods, TATS-base and TATS-hierarchical, with the baseline Vanilla VQGAN and the state-of-the-art models including MoCoGAN-HD [45], DIGAN [58], and CCVS [25] by unrolling RNN states, directly sampling, and sliding attention window using their official model checkpoints.

Quality. We measure video quality with respect to the duration by evaluating every 16 frames extracted side-by-side from the generated videos. Ideally, every set of 16-frame videos should come from the same distribution as the training set. Therefore, we report the FVD changes of these generated clips compared with the first generated 16 frames in Figure 5, to measure the degradation of the video quality. The figure shows that our TATS-base model successfully delays the quality degradation compared with the vanilla VQGAN baseline, MoCoGAN-HD, and CCVS models. In addition, the TATS-hierarchical model further improves the long-term quality of the TATS-base model. The concurrent work DIGAN [58] also claims the ability of extrapolation, while we show that the generation still degrades severely after certain number frames on the UCF-101 and Taichi-HD datasets. We conjecture the unusual decrease of the FVD w.r.t. the duration of DIGAN and TATS-hierarchical on Sky Time-lapse can be explained by that the I3D model [6] used to calculate FVD is trained on Kinetics-400 dataset, and the sky videos can be outliers of the training data and lead to weak activation in the logit layers and therefore such unusual behaviors. We further perform qualitative and human evaluations to compare our method with DIGAN in supp. mat. C. The results confirm that the sky videos generated by TATS have better quality.

Coherence. The generated long videos should follow a consistent topical theme. We evaluate the coherence of the videos on the UCF-101 dataset since it has multiple themes (classes). We expect the generated long videos to be classified as the same class all the time. We adopt the same trained C3D model for IS calculated [37], and propose two metrics, Class Coherence Score (CCS) and Inception Coherence Score (ICS) at time step t, measuring the theme similarity between the non-overlapped 16 frames w.r.t. the first 16 frames, defined as below:

$$\text{CCS}_t = \sum_i \frac{\mathbf{1}(\arg\max p_{C3D}(y|x_i^{(0:15)}), \arg\max p_{C3D}(y|x_i^{(t:t+15)})}{N}$$

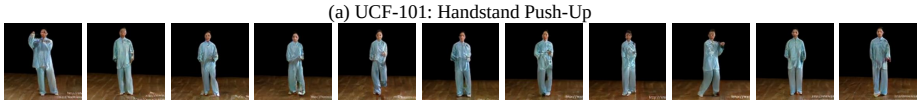
$$\text{ICS}_t = \sum_i p_{C3D}(y|x_i^{(t:t+15)}) \log \frac{p_{C3D}(y|x_i^{(t:t+15)})}{p_{C3D}(y|x_i^{(0:15)})}$$

The ICS captures class shift more accurately than the CCS, which only looks at the most probable class. On the other hand, CCS is more intuitive and allows us to define single metrics such as the area under the curve. CCS also doesn't have the asymmetry issue (unlike KL divergence used in ICS). We show the scores of TATS models and several baselines in Figure 6. TATS-base achieves decent coherence as opposite to its quality degradation shown previously. Such difference can be explained by its failure on a small portion of generated videos as shown in supp. mat. Figure 20, which dominates the FVD. This also shows that CCS and ICS measure coherence on individual videos that complementary to FVD changes. Furthermore, TATS-hierarchical outperforms the baselines on both metrics. For example, more than half of the videos are still classified consistently in the last 16 frames of the entire 1024 frames.

3.4 Qualitative Evaluation on Long Video Generation

This section shows qualitative results of videos with 1024 frames and discusses their common properties. 1024 frames per video approaches the upper bound in the training data as shown in supp. mat. Figure 15, which means the generated videos are probably different from the training videos, at least in duration.
Recurrent actions. We find that some video themes contain repeated events such as the *Handstand Push-Up* class in the UCF-101 dataset and videos in the Taichi-HD dataset. As shown in Figure 7, our model generalizes to long video generation by producing realistic and recurrent actions. However, we find that all the 1024 generated frames in these videos are unique and motions are not identical, which shows that our model is not simply copying the short loops.

(a) UCF-101: Handstand Push-Up



(b) Taichi-HD

Fig. 7: **Videos with recurrent events.** Every 100^{th} frame is extracted from the generated videos with 1024 frames on the UCF-101 and Taichi-HD datasets.

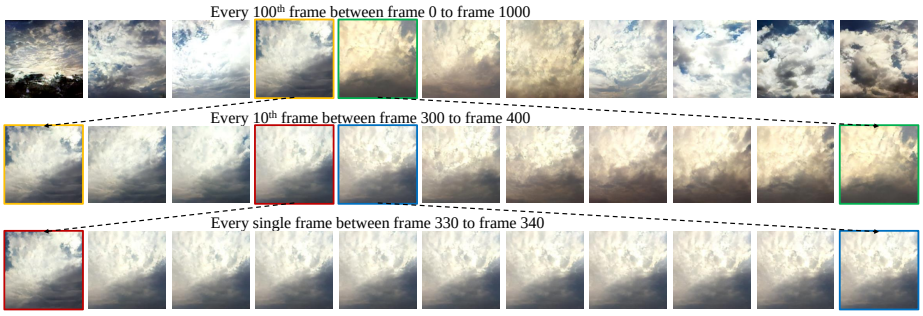


Fig. 8: **Videos with homomorphisms.** Every 100^{th}, 10^{th}, and consecutive frames are extracted from a generated sky video with 1024 frames.

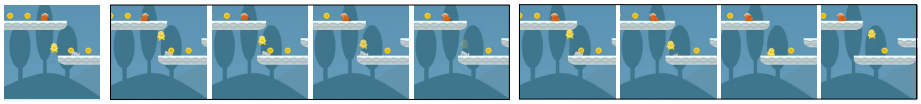


Fig. 9: **Videos with meanings.** Video manipulation by modifying the texts.

Smooth transitions. Videos with the same theme often share visual features such as scenes and motions. We show that with enough training data available for a single theme, our model learns to "stitch" the long videos through generating smooth transitions between them. For example in Figure 8, we show that our model generates a long sky video containing different weather and timing while the transitions between these conditions are still natural and realistic. To show that this is not the case in the training data, we compute the LPIPS score [59] and color histrogram correlation between the 1^{st} and the 1024^{th} frames and report the mean and standard deviation on the 500 generated and 216 real long videos in Table 2. It shows that such transitions are much more prominent on the generated videos than the training videos. Similar examples of the UCF-101 and Taichi-HD videos can be found in supp. mat. Figure 19. A separation of content and motion latent features [47,58,45] may better leverage such transitions, which we leave as future work.

Videos	LPIPS metric	Color Similarity
Real	0.1839 ± 0.0683	0.7330 ± 0.2401
Fake	0.3461 ± 0.1184	0.0797 ± 0.1445

Table 2: LPIPS and color histogram correlation between the first and the last frames of the sky videos.

Meaningful synthesis. By conditioning on the temporal information, we can achieve more controllable generation, which allows us to directly create or modify videos based our own will. For example, in Figure 9, we show that it is possible to manipulate the videos by changing the underlying storyline - by replacing "killed by" with "kills" we completely change the destiny of Mugen!

4 Related Work

In this section, we discuss the related work in video generation using different models. We focus on the different strategies adopted by these methods to handle temporal dynamics and their potential for and challenges associated with long video generation. Also see supp. mat. D for other relevant models and tasks.

GAN-based video generator. Adapting GANs [22,23,13] to video synthesis requires modeling the temporal dimension. Both 3D deconvolutionals [49] and additional RNN or LSTM [36,47,47,9,37,27,45] have been used. By unrolling the steps taken by the RNNs, videos of longer duration can be generated. However, as shown in our experiments, the quality of these videos degrades quickly. In addition, without further modifications, the length of videos generated by these models is limited by the GPU memory (e.g., at most 140 frames can be generated on 32GB V100 GPU by HVG [7]).

AR-based video generator. Autoregressive models have become a ubiquitous generative model for video synthesis [35,43,20,52]. A common challenge faced by AR models is their slow inference speed. This issue is mitigated by training on the compressed tokens with VQVAEs [33,54,25,57]. Our model falls into this line of video generators. We show that such a VQVAE-based AR model is promising to generate long videos with long-range dependence.

INR-based video generator. Implicit Neural Representations [40,44] represent continuous signals such as videos by mapping the coordinate space to RGB space [58,41]. The advantage of these models is their ability to generate arbitrarily long videos non-autoregressively. However, their generated videos still suffer from the quality degradation or contain periodic artifacts due to the positional encoding and struggle at synthesizing new content.

Concurrent works on long video generation. In parallel to our work, [15] proposes a gradient conditioning method for sampling longer videos with a diffusion model, [28] and [16] explore a hierarchical model with frame-level VQVAE embeddings or RGB frames. [4] uses a low-resolution long video generator and short-video super-resolution network to generate videos of dynamic scenes.

5 Conclusion

We propose TATS, a time-agnostic VQGAN and time-sensitive transformer model, that is only trained on clips with tens of frames and can generate thousands of frames using a sliding window during inference time. Our model generates meaningful videos when conditioned on text and audio. Our paper is a small step but we hope it can encourage future works on more interesting forms of video synthesis with a realistic number of frames, perhaps even movies.

Acknowledgements We thank Oran Gafni, Sasha Sheng, and Isabelle Hu for helpful discussion and feedback; Patrick Esser and Robin Rombach for sharing additional insights for training VQGAN models; Anoop Cherian and Moitreya Chatterjee for sharing the pre-processing code for the AudioSet dataset.

References

1. Acharya, D., Huang, Z., Paudel, D.P., Van Gool, L.: Towards high resolution video generation with progressive growing of sliced wasserstein gans. arXiv preprint arXiv:1810.02419 (2018) 10
2. Alsallakh, B., Kokhlikyan, N., Miglani, V., Yuan, J., Reblitz-Richardson, O.: Mind the pad – CNNs can develop blind spots. In: ICLR (2021) 6
3. Brock, A., Donahue, J., Simonyan, K.: Large scale gan training for high fidelity natural image synthesis. In: ICLR (2018) 10
4. Brooks, T., Hellsten, J., Aittala, M., Wang, T.C., Aila, T., Lehtinen, J., Liu, M.Y., Efros, A.A., Karras, T.: Generating long videos of dynamic scenes. arXiv preprint arXiv:2206.03429 (2022) 14
5. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. NeurIPS (2020) 2, 5, 6
6. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: CVPR (2017) 7, 11
7. Castrejon, L., Ballas, N., Courville, A.: Hierarchical video generation for complex data. arXiv preprint arXiv:2106.02719 (2021) 8, 14
8. Chatterjee, M., Cherian, A.: Sound2sight: Generating visual dynamics from sound and context. In: ECCV (2020) 8, 9
9. Clark, A., Donahue, J., Simonyan, K.: Adversarial video generation on complex datasets. arXiv preprint arXiv:1907.06571 (2019) 2, 5, 9, 14
10. Esser, P., Rombach, R., Ommer, B.: Taming transformers for high-resolution image synthesis. In: CVPR (2021) 2, 3, 4, 5, 8, 10
11. Fan, A., Lewis, M., Dauphin, Y.: Hierarchical neural story generation (2018) 8
12. Gemmeke, J.F., Ellis, D.P., Freedman, D., Jansen, A., Lawrence, W., Moore, R.C., Plakal, M., Ritter, M.: Audio set: An ontology and human-labeled dataset for audio events. In: ICASSP (2017) 4, 9
13. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. NeurIPS (2014) 14
14. Hayes, T., Zhang, S., Yin, X., Pang, G., Sheng, S., Yang, H., Ge, S., Hu, Q., Parikh, D.: Mugen: A playground for video-audio-text multimodal understanding and generation. arXiv preprint arXiv:2204.08058 (2022) 9
15. Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., Fleet, D.J.: Video diffusion models. arXiv preprint arXiv:2204.03458 (2022) 10, 14
16. Hong, W., Ding, M., Zheng, W., Liu, X., Tang, J.: Cogvideo: Large-scale pretraining for text-to-video generation via transformers. arXiv preprint arXiv:2205.15868 (2022) 10, 14
17. Islam, M.A., Jia, S., Bruce, N.D.: How much position information do convolutional neural networks encode? In: ICLR (2019) 6
18. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: ECCV (2016) 5
19. Kahembwe, E., Ramamoorthy, S.: Lower dimensional kernels for video discriminators. Neural Networks (2020) 10
20. Kalchbrenner, N., Oord, A., Simonyan, K., Danihelka, I., Vinyals, O., Graves, A., Kavukcuoglu, K.: Video pixel networks. In: ICML (2017) 2, 14
21. Karras, T., Aittala, M., Laine, S., Härkönen, E., Hellsten, J., Lehtinen, J., Aila, T.: Alias-free generative adversarial networks. NeurIPS (2021) 7

22. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: CVPR (2019) 14
23. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and improving the image quality of stylegan. In: CVPR (2020) 2, 14
24. Kayhan, O.S., Gemert, J.C.v.: On translation invariance in cnns: Convolutional layers can exploit absolute spatial location. In: CVPR (2020) 6
25. Le Moing, G., Ponce, J., Schmid, C.: Ccvs: Context-aware controllable video synthesis. NeurIPS (2021) 2, 9, 10, 11, 14
26. Luc, P., Clark, A., Dieleman, S., Casas, D.d.L., Doron, Y., Cassirer, A., Simonyan, K.: Transformation-based adversarial video prediction on large-scale data. arXiv preprint arXiv:2003.04035 (2020) 2
27. Munoz, A., Zolfaghari, M., Argus, M., Brox, T.: Temporal shift gan for large scale video generation. In: WACV (2021) 2, 14
28. Nash, C., Carreira, J., Walker, J., Barr, I., Jaegle, A., Malinowski, M., Battaglia, P.: Transframer: Arbitrary frame prediction with generative models. arXiv preprint arXiv:2203.09494 (2022) 14
29. van den Oord, A., Vinyals, O., Kavukcuoglu, K.: Neural discrete representation learning. In: NeurIPS (2017) 2, 4
30. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML (2021) 10
31. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving language understanding by generative pre-training 9
32. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language models are unsupervised multitask learners (2019) 9
33. Rakhimov, R., Volkhonskiy, D., Artemov, A., Zorin, D., Burnaev, E.: Latent video transformer. arXiv preprint arXiv:2006.10704 (2020) 2, 14
34. Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., Sutskever, I.: Zero-shot text-to-image generation. arXiv preprint arXiv:2102.12092 (2021) 8
35. Ranzato, M., Szlam, A., Bruna, J., Mathieu, M., Collobert, R., Chopra, S.: Video (language) modeling: a baseline for generative models of natural videos. arXiv preprint arXiv:1412.6604 (2014) 14
36. Saito, M., Matsumoto, E., Saito, S.: Temporal generative adversarial nets with singular value clipping. In: ICCV (2017) 2, 10, 14
37. Saito, M., Saito, S., Koyama, M., Kobayashi, S.: Train sparsely, generate densely: Memory-efficient unsupervised training of high-resolution temporal gan. IJCV (2020) 2, 9, 11, 14
38. Siarohin, A., Lathuilière, S., Tulyakov, S., Ricci, E., Sebe, N.: First order motion model for image animation. NeurIPS (2019) 4, 8
39. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014) 5
40. Sitzmann, V., Martel, J., Bergman, A., Lindell, D., Wetzstein, G.: Implicit neural representations with periodic activation functions. NeurIPS (2020) 14
41. Skorokhodov, I., Tulyakov, S., Elhoseiny, M.: Stylegan-v: A continuous video generator with the price, image quality and perks of stylegan2. arXiv preprint arXiv:2112.14683 (2021) 2, 7, 10, 14
42. Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012) 4, 8
43. Srivastava, N., Mansimov, E., Salakhudinov, R.: Unsupervised learning of video representations using lstms. In: ICML (2015) 14

44. Tancik, M., Srinivasan, P., Mildenhall, B., Fridovich-Keil, S., Raghavan, N., Singhal, U., Ramamoorthi, R., Barron, J., Ng, R.: Fourier features let networks learn high frequency functions in low dimensional domains. NeurIPS (2020) 14
45. Tian, Y., Ren, J., Chai, M., Olszewski, K., Peng, X., Metaxas, D.N., Tulyakov, S.: A good image generator is what you need for high-resolution video synthesis. In: ICLR (2021) 2, 9, 11, 13, 14
46. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: ICCV (2015) 9
47. Tulyakov, S., Liu, M.Y., Yang, X., Kautz, J.: Mocogan: Decomposing motion and content for video generation. In: CVPR (June 2018) 2, 5, 13, 14
48. Unterthiner, T., van Steenkiste, S., Kurach, K., Marinier, R., Michalski, M., Gelly, S.: Towards accurate generative models of video: A new metric & challenges. ICLR (2019) 9
49. Vondrick, C., Pirsiavash, H., Torralba, A.: Generating videos with scene dynamics. NeurIPS (2016) 2, 14
50. Wang, T.C., Liu, M.Y., Zhu, J.Y., Liu, G., Tao, A., Kautz, J., Catanzaro, B.: Video-to-video synthesis. In: NeurIPS (2018) 5
51. Wang, T.C., Liu, M.Y., Zhu, J.Y., Tao, A., Kautz, J., Catanzaro, B.: High-resolution image synthesis and semantic manipulation with conditional gans. In: CVPR (2018) 5
52. Weissenborn, D., Täckström, O., Uszkoreit, J.: Scaling autoregressive video models. In: ICLR (2020) 2, 14
53. Wu, C., Huang, L., Zhang, Q., Li, B., Ji, L., Yang, F., Sapiro, G., Duan, N.: Godiva: Generating open-domain videos from natural descriptions. arXiv preprint arXiv:2104.14806 (2021) 8
54. Wu, C., Liang, J., Ji, L., Yang, F., Fang, Y., Jiang, D., Duan, N.: N\" uwa: Visual synthesis pre-training for neural visual world creation. arXiv preprint arXiv:2111.12417 (2021) 2, 14
55. Xiong, W., Luo, W., Ma, L., Liu, W., Luo, J.: Learning to generate time-lapse videos using multi-stage dynamic generative adversarial networks. In: CVPR (2018) 4, 8
56. Xu, R., Wang, X., Chen, K., Zhou, B., Loy, C.C.: Positional encoding as spatial inductive bias in gans. In: CVPR (2021) 6
57. Yan, W., Zhang, Y., Abbeel, P., Srinivas, A.: Videogpt: Video generation using vq-vae and transformers. arXiv preprint arXiv:2104.10157 (2021) 2, 5, 14
58. Yu, S., Tack, J., Mo, S., Kim, H., Kim, J., Ha, J.W., Shin, J.: Generating videos with dynamics-aware implicit generative adversarial networks. In: ICLR (2021) 2, 9, 11, 13, 14
59. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: CVPR (2018) 5, 13