# MAGVIT: Masked Generative Video Transformer

Lijun Yu[‡†⋄]*, Yong Cheng[†], Kihyuk Sohn[†], José Lezama[†], Han Zhang[†], Huiwen Chang[†],
Alexander G. Hauptmann[‡], Ming-Hsuan Yang[†], Yuan Hao[†], Irfan Essa[†ƚ], and Lu Jiang[†⋄]
[‡]Carnegie Mellon University, [†]Google Research, [ƚ]Georgia Institute of Technology
[⋄] Correspondence to lijun@cmu.edu, lujiang@google.com

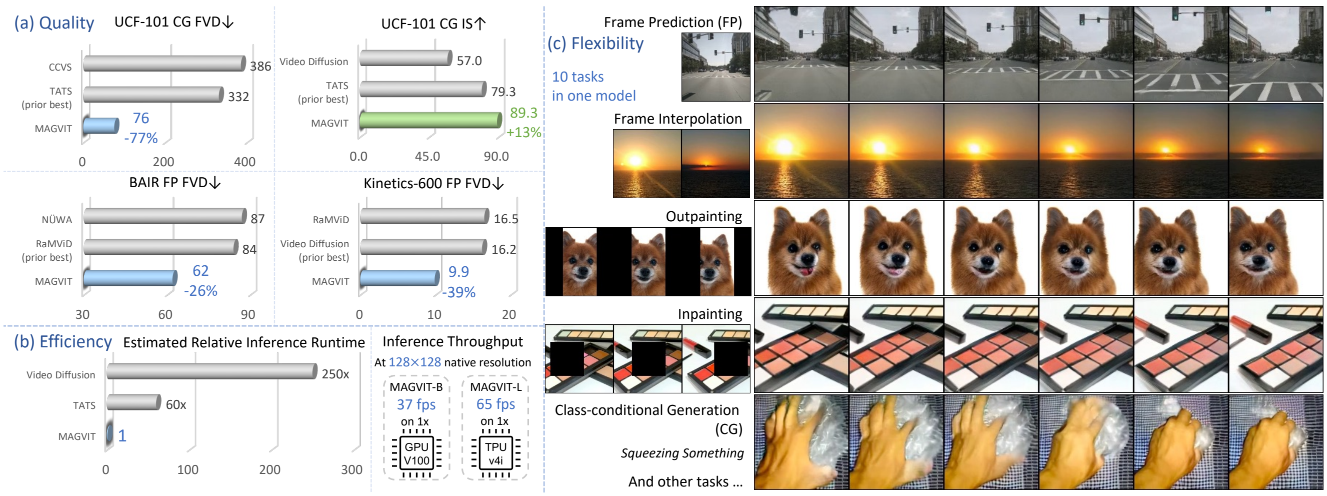



Figure 1. **Overview of the video generation *quality*, *efficiency*, and *flexibility* of the proposed MAGVIT model.** (a) MAGVIT achieves the state-of-the-art FVD [61] and Inception Score (IS) [49] on two video generation tasks and three benchmarks, in comparison with prior best diffusion models (RaMViD [35], Video Diffusion [33]) and autoregressive models (CCVS [41], TATS [21], NÜWA [70]). (b) It is two orders of magnitude faster than diffusion models and 60× faster than autoregressive models. (c) A single MAGVIT model accommodates different generation tasks, ranging from class-conditional generation to dynamic inpainting of a moving object.

## Abstract

*We introduce the MAsked Generative VIdeo Transformer,* MAGVIT*, to tackle various video synthesis tasks with a single model. We introduce a 3D tokenizer to quantize a video into spatial-temporal visual tokens and propose an embedding method for masked video token modeling to facilitate multi-task learning. We conduct extensive experiments to demonstrate the quality, efficiency, and flexibility of MAGVIT. Our experiments show that (i) MAGVIT performs favorably against state-of-the-art approaches and establishes the best-published FVD on three video generation benchmarks, including the challenging Kinetics-600. (ii) MAGVIT outperforms existing methods in inference time by two orders of magnitude against diffusion models and by 60× against autoregressive models. (iii) A single MAGVIT model supports ten diverse generation tasks and generalizes across videos from different visual domains. The source code and trained models will be released to the public at* https://magvit.cs.cmu.edu*.*

*Work partially done during a research internship at Google Research.

## 1. Introduction

Recent years have witnessed significant advances in image and video content creation based on learning frameworks ranging from generative adversarial networks (GANs) [15, 43, 48, 59, 66], diffusion models [25, 33, 35, 47, 65], to vision transformers [44, 45, 69]. Inspired by the recent success of generative image transformers such as DALL·E [46] and other approaches [12, 18, 20, 73], we propose an efficient and effective video generation model by leveraging masked token modeling and multi-task learning.

We introduce the MAsked Generative VIdeo Transformer (*MAGVIT*) for multi-task video generation. Specifically, we build and train a single MAGVIT model to perform a variety of diverse video generation tasks and demonstrate the model's efficiency, effectiveness, and flexibility against state-of-the-art approaches. Fig. 1(a) shows the quality metrics of MAGVIT on a few benchmarks with efficiency comparisons in (b), and generated examples under different task setups such as frame prediction/interpolation, out/in-painting, and class conditional generation in (c).

MAGVIT models a video as a sequence of visual tokens in the latent space and learns to predict masked tokens with BERT [17]. There are two main modules in the proposed framework. First, we design a 3D quantization model to tokenize a video, with high fidelity, into a low-dimensional spatial-temporal manifold [21, 71]. Second, we propose an effective *masked token modeling* (MTM) scheme for multi-task video generation. Unlike conventional MTM in image understanding [67] or image/video synthesis [12, 26, 28], we present an embedding method to model a video condition using a multivariate mask and show its efficacy in training.

We conduct extensive experiments to demonstrate the quality, efficiency, and flexibility of MAGVIT against state-of-the-art approaches. Specifically, we show that MAGVIT performs favorably on two video generation tasks across three benchmark datasets, including UCF-101 [55], BAIR Robot Pushing [19, 61], and Kinetics-600 [10]. For the class-conditional generation task on UCF-101, MAGVIT reduces state-of-the-art FVD [61] from 332 [21] to 76 ($\downarrow$77%). For the frame prediction task, MAGVIT performs best in terms of FVD on BAIR (84 [35] $\rightarrow$ 62, $\downarrow$ 26%) and Kinetics-600 (16 [33] $\rightarrow$ 9.9, $\downarrow$ 38%).

Aside from the visual quality, MAGVIT's video synthesis is highly efficient. For instance, MAGVIT generates a 16-frame $128 \times 128$ video clip in 12 steps, which takes 0.25 seconds on a single TPUv4i [36] device. On a V100 GPU, a base variant of MAGVIT runs at 37 frame-per-second (fps) at $128 \times 128$ resolution. When compared at the same resolution, MAGVIT is two orders of magnitude faster than the video diffusion model [33]. In addition, MAGVIT is 60 times faster than the autoregressive video transformer [21] and 4-16 times more efficient than the contemporary non-autoregressive video transformer [26].

We show that MAGVIT is flexible and robust for multiple video generation tasks with a single trained model, including frame interpolation, class-conditional frame prediction, inpainting, and outpainting, etc. In addition, MAGVIT learns to synthesize videos with complex scenes and motion contents from diverse and distinct visual domains, including actions with objects [23], autonomous driving [9], and object-centric videos from multiple views [2].

The main contributions of this work are:

- To the best of our knowledge, we present the first masked multi-task transformer for efficient video generation and manipulation. We show that a trained model can perform ten different tasks at inference time.
- We introduce a spatial-temporal video quantization model design with high reconstruction fidelity.
- We propose an effective embedding method with diverse masks for numerous video generation tasks.
- We show that MAGVIT achieves the best-published fidelity on three widely-used benchmarks, including UCF-101, BAIR Robot Pushing, and Kinetics-600 datasets.

## 2. Preliminaries: Masked Image Synthesis

The proposed video generation framework is based on a two-stage image synthesis process [20, 46] with non-autoregressive transformers [12, 42]. In the first stage, an image is quantized and flattened into a sequence of discrete tokens by a Vector-Quantized (VQ) auto-encoder [20, 63, 72]. In the second stage, masked token modeling (MTM) is used to train a transformer model [12, 25] on the tokens. Let $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$ be an image and $\mathbf{z} \in \mathbb{Z}^N$ denote the corresponding token sequence of length $N$.

We take MaskGIT [12] as an example. In the second stage, it applies a binary mask $\mathrm{m}_i \in \{x \rightarrow x, x \rightarrow \texttt{[MASK]}\}$ to each token to build a corrupted sequence $\overline{\mathbf{z}} = \mathbf{m}(\mathbf{z})$. Condition inputs, such as class labels, are incorporated as the prefix tokens $\mathbf{c}$. A BERT [17] parameterized by $\theta$ is learned to predict the masked tokens in the input sequence $[\mathbf{c}, \overline{\mathbf{z}}]$, where $[\cdot, \cdot]$ concatenates the sequences. The objective is to minimize the cross-entropy between the predicted and the ground-truth token at each masked position:

$$\mathcal{L}_{\text{mask}}(\mathbf{z}; \theta) = \mathop{\mathbb{E}}_{\mathbf{m} \sim p_{\mathcal{U}}} \Big[ \sum_{\overline{\mathrm{z}}_i = \texttt{[MASK]}} -\log p_\theta(\mathrm{z}_i \mid [\mathbf{c}, \overline{\mathbf{z}}]) \Big] \tag{1}$$

During training, MaskGIT randomly samples $\mathbf{m}$ from a prior distribution $p_{\mathcal{U}}$ where the mask ratio follows a cosine scheduling function $\gamma(\cdot)$ [12]. Specifically, it first uniformly samples a per-token mask score $\mathrm{s}_i \sim \mathcal{U}(0, 1)$ to form a sequence denoted as $\mathbf{s}$. Then it samples $r \sim \mathcal{U}(0, 1)$ and computes a cut-off threshold $s^*$ as the $\lceil \gamma(r) N \rceil$-*th* smallest element in $\mathbf{s}$. Finally, a mask $\mathbf{m}$ is created such that $\mathrm{m}_i(x) = \texttt{[MASK]}$ if $\mathrm{s}_i \leq s^*$ and $\mathrm{m}_i(x) = x$ otherwise.

For inference, the non-autoregressive decoding method [22, 24, 40] is used to synthesize an image [12, 42, 76]. For example, MaskGIT generates an image in $K = 12$ steps [12] from a blank canvas with all visual tokens masked out. At each step, it predicts all tokens in parallel while retaining tokens with the highest prediction scores. The remaining tokens are masked and predicted in the next iteration until all tokens are generated. Similar to the training stage, the mask ratio is computed by the schedule function $\gamma$, but with a deterministic input as $\gamma(\frac{t}{K})$, where $t$ is the current step.

## 3. Masked Generative Video Transformer

Our goal is to design a multi-task video generation model with high quality and inference efficiency. We propose MAsked Generative VIdeo Transformer (MAGVIT), a vision transformer framework that leverages masked token modeling and multi-task learning. MAGVIT generates a video from task-specific condition inputs, such as a frame, a partially-observed video volume, or a class identifier.

The framework consists of two stages. First, we learn a 3D vector-quantized (VQ) autoencoder to quantize a video into discrete tokens. In the second stage, we learn a video transformer by multi-task masked token modeling.

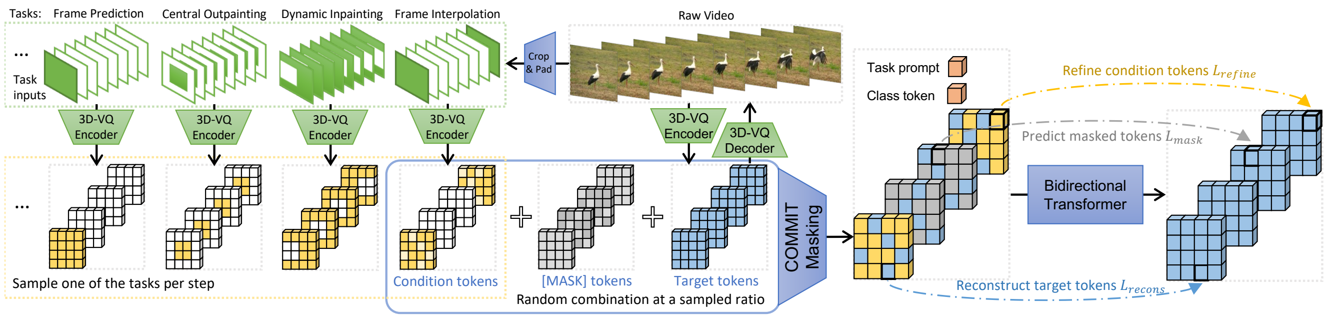



Figure 2. **MAGVIT pipeline overview.** The 3D-VQ encoder quantizes a video into discrete tokens, while the 3D-VQ decoder maps them back to the pixel space. We sample one of the tasks at each training step and build its condition inputs by cropping and padding the raw video, where green denotes valid pixels and white is padding. We quantize the condition inputs with the 3D-VQ encoder and select the non-padding part as condition tokens. The masked token sequence combines condition tokens, `[MASK]` tokens, and the target tokens, with a task prompt and a class token as the prefix. The bidirectional transformer learns to predict the target tokens through three objectives: refining condition tokens, predicting masked tokens, and reconstructing target tokens.

Fig. 2 illustrates the training in the second stage. At each training step, we sample one of the tasks with its prompt token, obtain a task-specific conditional mask, and optimize the transformer to predict all target tokens given masked inputs. During inference, we adapt the non-autoregressive decoding method to generate tokens conditionally on the task-specific inputs, which will be detailed in Algorithm 1.

### 3.1. Spatial-Temporal Tokenization

Our video VQ autoencoder is built upon the image VQ-GAN [20]. Let $\mathbf{V} \in \mathbb{R}^{T\times H\times W\times 3}$ be a video clip of $T$ frames. The VQ encoder tokenizes the video as $f_{\mathcal{T}} : \mathbf{V} \rightarrow \mathbf{z} \in \mathbb{Z}^N$, where $\mathbb{Z}$ is the codebook. The decoder $f_{\mathcal{T}}^{-1}$ maps the latent tokens back to video pixels.

The VQ autoencoder is a crucial module as it not only sets a quality bound for the generation but also determines the token sequence length, hence affecting generation efficiency. Existing methods apply VQ encoders either on each frame independently (2D-VQ) [26, 41] or on a super-voxel (3D-VQ) [21, 71]. We propose different designs that facilitate MAGVIT to perform favorably against other VQ models for video (see Tab. 7).

**3D architecture.** We design a 3D-VQ network architecture to model the temporal dynamics as follows. The encoder and decoder of VQGAN consist of cascaded residual blocks [29] interleaved by downsampling (average pooling) and upsampling (resizing plus convolution) layers. We expand all 2D convolutions to 3D convolutions with a temporal axis. As the overall downsampling rate is usually different between temporal and spatial dimensions, we use both 3D and 2D downsampling layers, where the 3D ones appear in the shallower layers of the encoder. The decoder mirrors the encoder with 2D upsampling layers in the first few blocks, followed by 3D ones. Appendix A.1 illustrates the detailed architecture. Note that a token is not only correlated to its corresponding supervoxel but depends on other patches due to the non-local receptive field.

**Inflation and padding.** We initialize our 3D-VQ with weights from a 2D-VQ in a matching architecture to transfer learned spatial relationships [11], known as 3D inflation. We use inflation on small datasets such as UCF-101 [55]. We use a central inflation method for the convolution layers, where the corresponding 2D kernel fills in the temporally central slice of a zero-filled 3D kernel. The parameters of the other layers are directly copied. To improve token consistency for the same content at different locations [21], we replace the `same` (zero) padding in the convolution layers with `reflect` padding, which pads with non-zero values.

**Training.** We apply the image perceptual loss [20] on each frame. The LeCam regularization [58] is added to the GAN loss to improve the training stability. We adopt the discriminator architecture from StyleGAN [38] and inflate it to 3D. With these components, unlike VQGAN, our model is trained stably with GAN loss from the beginning.

### 3.2. Multi-Task Masked Token Modeling

In MAGVIT, we adopt various masking schemes to facilitate learning for video generation tasks with different conditions. The conditions can be a spatial region for inpainting/outpainting or a few frames for frame prediction/interpolation. We refer to these partially-observed video conditions as *interior conditions*.

We argue that it is suboptimal to directly unmask the tokens corresponding to the region of the interior condition [12]. As discussed in Section 3.1, the non-local receptive field of the tokenizer can leak the ground-truth information into the unmasked tokens, leading to problematic non-causal masking and poor generalization.

We propose a method, COnditional Masked Modeling by Interior Tokens (or *COMMIT* for short), to embed interior conditions inside the corrupted visual tokens.

**Training.** Each training example includes a video $\mathbf{V}$ and the optional class annotation $\mathbf{c}$. The target visual tokens come from the 3D-VQ as $\mathbf{z} = f_{\mathcal{T}}(\mathbf{V})$. At each step, we sample a task prompt $\rho$, obtain the task-specific interior condition pixels, pad it into $\tilde{\mathbf{V}}$ with the same shape as $\mathbf{V}$, and get the condition tokens $\tilde{\mathbf{z}} = f_{\mathcal{T}}(\tilde{\mathbf{V}})$. Appendix B.1 lists the padding functions for each task.

At a sampled mark ratio, we randomly replace target tokens $\mathrm{z}_i$, with either 1) the condition token $\tilde{\mathrm{z}}_i$, if the corresponding supervoxel of $\mathrm{z}_i$ contains condition pixels; or 2) the special `[MASK]` token, otherwise. Formally, we compute the *multivariate* conditional mask $\mathbf{m}(\cdot \mid \tilde{\mathbf{z}})$ as

$$\mathrm{m}(\mathrm{z}_i \mid \tilde{\mathrm{z}}_i) = \begin{cases} \tilde{\mathrm{z}}_i & \text{if } \mathrm{s}_i \leq s^* \wedge \neg\,\mathrm{ispad}(\tilde{\mathrm{z}}_i) \\ \texttt{[MASK]} & \text{if } \mathrm{s}_i \leq s^* \wedge \mathrm{ispad}(\tilde{\mathrm{z}}_i) \\ \mathrm{z}_i & \text{if } \mathrm{s}_i > s^* \end{cases} \tag{2}$$

where $\mathrm{s}_i$ and $s^*$ are the per-token mask score and the cut-off score introduced in Section 2. $\mathrm{ispad}(\tilde{\mathrm{z}}_i)$ returns whether the corresponding supervoxel of $\tilde{\mathrm{z}}_i$ in $\tilde{\mathbf{V}}$ only contains padding.

Eq. (2) indicates that COMMIT embeds interior conditions as corrupted visual tokens into the multivariate mask $\mathbf{m}$, which follows a new distribution $p_{\mathcal{M}}$ instead of the prior $p_{\mathcal{U}}$ for binary masks. With the corrupted token sequence $\overline{\mathbf{z}} = \mathbf{m}(\mathbf{z} \mid \tilde{\mathbf{z}})$ as input, the *multi-task* training objective is

$$\mathcal{L}(\mathbf{V};\theta) = \mathop{\mathbb{E}}_{\rho,\hat{\mathbf{V}}}\ \mathop{\mathbb{E}}_{\mathbf{m}\sim p_{\mathcal{M}}} \Big[\sum_i -\log p_\theta(\mathrm{z}_i \mid [\rho, \mathbf{c}, \overline{\mathbf{z}}])\Big] \tag{3}$$

We can decompose the loss in Eq. (3) into three parts according to Eq. (2): $\mathcal{L}_{\text{refine}}$ refines the task-specific condition tokens, $\mathcal{L}_{\text{mask}}$ predicts masked tokens , and $\mathcal{L}_{\text{recons}}$ reconstructs target tokens. Let $\overline{\mathbf{c}} = [\rho, \mathbf{c}, \overline{\mathbf{z}}]$ for simplicity,

$$\begin{aligned} \sum_{i=1}^{N} -\log p_\theta(\mathrm{z}_i \mid [\rho, \mathbf{c}, \overline{\mathbf{z}}]) = & \underbrace{\sum_{\overline{\mathrm{z}}_i = \tilde{\mathrm{z}}_i} -\log p_\theta(\mathrm{z}_i \mid \overline{\mathbf{c}})}_{\text{Refine condition tokens } \mathcal{L}_{\text{refine}}} \\ + \underbrace{\sum_{\overline{\mathrm{z}}_i = \texttt{[MASK]}} -\log p_\theta(\mathrm{z}_i \mid \overline{\mathbf{c}})}_{\text{Predict masked tokens } \mathcal{L}_{\text{mask}}} & + \underbrace{\sum_{\overline{\mathrm{z}}_i = \mathrm{z}_i} -\log p_\theta(\mathrm{z}_i \mid \overline{\mathbf{c}})}_{\text{Reconstruct target tokens } \mathcal{L}_{\text{recons}}} \end{aligned} \tag{4}$$

While $\mathcal{L}_{\text{mask}}$ is the same as the MTM loss in Eq. (1) and $\mathcal{L}_{\text{recons}}$ sometimes is used as a regularizer (*e.g.*, in NLP tasks), $\mathcal{L}_{\text{refine}}$ is a new component introduced by COMMIT.

The COMMIT method facilitates multi-task video generation in three aspects. First, it provides a correct causal masking for all interior conditions. Second, it produces a fixed-length sequence for different conditions of arbitrary regional volume, improving training and memory efficiency since no padding tokens are needed. Third, it achieves state-of-the-art multi-task video generation results (see Tab. 5).

**Video generation tasks.** We consider *ten* tasks for multi-task video generation where each task has a different interior condition and mask: Frame Prediction (FP), Frame Interpolation (FI), Central Outpainting (OPC), Vertical Outpainting (OPV), Horizontal Outpainting (OPH), Dynamic Outpainting (OPD), Central Inpainting (IPC), and Dynamic Inpainting (IPD), Class-conditional Generation (CG), Class-conditional Frame Prediction (CFP). We provide the detailed definitions in Appendix B.1.

**Algorithm 1** Non-autoregressive Decoding by COMMIT

**Input**: prefix $\rho$ and $\mathbf{c}$, condition $\tilde{\mathbf{z}}$, steps $K$, temperature $T$
**Output**: predicted visual tokens $\hat{\mathbf{z}}$

```
s = 0, s* = 1, ẑ = 0^N
for t ← 0, 1, ··· , K − 1 do
    z̄ ← m(ẑ | z̃; s, s*)
    ẑ_i ∼ p_θ(z_i | [ρ, c, z̄]), ∀i where s_i ≤ s*
    s_i ← p_θ(ẑ_i | [ρ, c, z̄]), ∀i where s_i ≤ s*
    s_i ← s_i + T(1 − (t+1)/K) Gumbel(0, 1), ∀i where s_i < 1
    s* ← The ⌈γ((t+1)/K)N⌉-th smallest value of s
    s_i ← 1, ∀i where s_i > s*
end for
return ẑ = [ẑ_1, ẑ_2, ··· , ẑ_N]
```

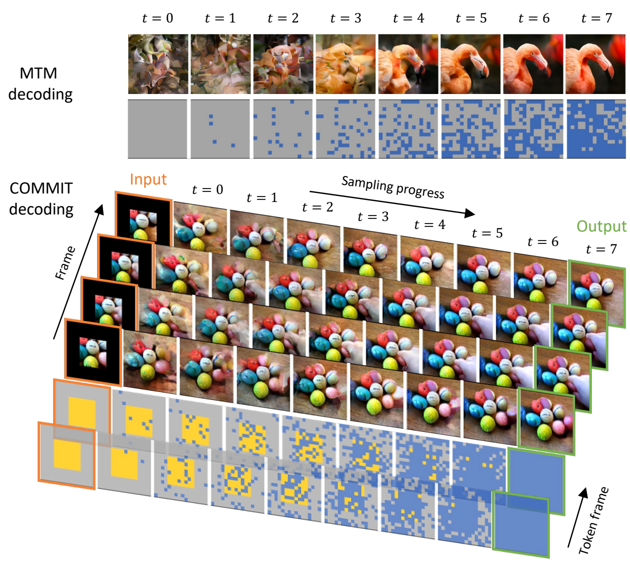



Figure 3. **Comparison between MTM decoding for image [12] and COMMIT decoding for video**. We show the output tokens and image/video at each decoding step $t$, with a central outpainting example for COMMIT. Unlike the MTM denoising decoding from all `[MASK]`, COMMIT performs a conditional generation process toward the output tokens while gradually replacing the interior condition tokens. Videos and tokens are temporally downsampled and stacked for visualization.

**Inference.** We use a non-autoregressive decoding method to generate video tokens from input conditions in $K$ steps (*e.g.*, 12). Each decoding step follows the COMMIT masking in Eq. (2) with a gradually reduced mask ratio. Algorithm 1 outlines the inference procedure.

Fig. 3 compares the non-autoregressive image decoding [12] and our video decoding procedure. Different from the MTM decoding in [12] which performs denoising from all `[MASK]`, COMMIT decoding starts from a *multivariate* mask that embeds the interior conditions. Guided by this mask, Algorithm 1 performs a conditional transition process toward the output tokens by replacing a portion of newly generated tokens at each step. In the end, all tokens are predicted where the interior condition tokens get refined.

| Method | Extra Video | Class | FVD↓ | IS↑ |
|---|---|---|---|---|
| RaMViD [35] | | | - | 21.71±0.21 |
| StyleGAN-V* [51] | | | - | 23.94±0.73 |
| DIGAN [74] | | | 577±21 | 32.70±0.35 |
| DVD-GAN [15] | | ✓ | - | 32.97±1.70 |
| Video Diffusion* [33] | | | - | 57.00±0.62 |
| TATS [21] | | | 420±18 | 57.63±0.24 |
| CCVS+StyleGAN [41] | | | 386±15 | 24.47±0.13 |
| Make-A-Video* [50] | | ✓ | 367 | 33.00 |
| TATS [21] | | ✓ | 332±18 | 79.28±0.38 |
| CogVideo* [34] | ✓ | ✓ | 626 | 50.46 |
| Make-A-Video* [50] | ✓ | ✓ | 81 | 82.55 |
| *MAGVIT*-B-CG (ours) | | ✓ | 159±2 | 83.55±0.14 |
| *MAGVIT*-L-CG (ours) | | ✓ | **76**±2 | **89.27**±0.15 |

Table 1. **Generation performance on the UCF-101 dataset.** Methods in gray are pretrained on additional large video data. Methods with ✓ in the Class column are class-conditional, while the others are unconditional. Methods marked with * use custom resolutions, while the others are at 128×128. See Appendix C for more comparisons with earlier works.

# 4. Experimental Results

We conduct extensive experiments to demonstrate the video generation quality (Section 4.2), efficiency (Section 4.3), and flexibility for multi-task generation (Section 4.4). We show a few generation results here, and refer to the web page[1] for more examples.

## 4.1. Experimental Setups

**Datasets.** We evaluate the single-task video generation performance of MAGVIT on three standard benchmarks, *i.e.*, class-conditional generation on UCF-101 [55] and frame prediction on BAIR Robot Pushing [19, 61] (1-frame condition) and Kinetics-600 [10] (5-frame condition). For multi-task video generation, we quantitatively evaluate MAGVIT on BAIR and SSv2 [23] on 8-10 tasks. Furthermore, to evaluate model generalizability, we train models with the same learning recipe on three additional video datasets: nuScenes [9], Objectron [2], and 12M Web videos. We show their generated videos in the main paper and quantitative performance in Appendix C.

**Evaluation metrics.** We use FVD [61] as our primary evaluation metric. Similar to [21, 33], FVD features are extracted with an I3D model trained on Kinetics-400 [11]. We also report the Inception Score (IS) [49] calculated with a C3D [57] model on UCF-101, and PSNR, SSIM [68], and LPIPS [75] on BAIR. We report the mean and standard deviation for each metric calculated over four runs.

**Implementation details.** We train MAGVIT to generate 16-frame videos at 128×128 resolution, except for BAIR at 64×64. The proposed 3D-VQ model quantizes a video into 4×16×16 visual tokens, where the visual codebook size is 1024. We use the BERT transformer [17] to model the token sequence, which includes 1 task prompt, 1 class token, and 1024 visual tokens. Two variants of MAGVIT, *i.e.*, base (B) with 128M parameters and large (L) with 464M, are evaluated. We train both stages with the Adam optimizer [39] in JAX/Flax [5, 30] on TPUs. Appendix B.2 details training configurations.

[1] https://magvit.cs.cmu.edu

| Method | K600 FVD↓ | BAIR FVD↓ |
|---|---|---|
| CogVideo [34] | 109.2 | - |
| CCVS [41] | 55.0±1.0 | 99±2 |
| Phenaki [64] | 36.4±0.2 | 97 |
| TrIVD-GAN-FP [43] | 25.7±0.7 | 103 |
| Transframer [44] | 25.4 | 100 |
| MaskViT [26] | - | 94 |
| FitVid [4] | - | 94 |
| MCVD [65] | - | 90 |
| NÜWA [70] | - | 87 |
| RaMViD [35] | 16.5 | 84 |
| Video Diffusion [33] | 16.2±0.3 | - |
| *MAGVIT*-B-FP (ours) | 24.5±0.9 | 76±0.1 (48±0.1) |
| *MAGVIT*-L-FP (ours) | **9.9**±0.3 | **62**±0.1 (31±0.2) |

Table 2. **Frame prediction performance on the BAIR and Kinetics-600 datasets.** - marks that the value is unavailable in their paper or incomparable to others. The FVD in parentheses uses a debiased evaluation protocol on BAIR detailed in Appendix B.3. See Appendix C for more comparisons with earlier works.

| Method | FVD↓ | PSNR↑ | SSIM↑ | LPIPS↓ |
|---|---|---|---|---|
| CCVS [41] | 99 | - | 0.729 | - |
| MCVD [65] | 90 | 16.9 | 0.780 | - |
| *MAGVIT*-L-FP (ours) | **62** | **19.3** | **0.787** | 0.123 |

Table 3. **Image quality metrics on BAIR frame prediction.**

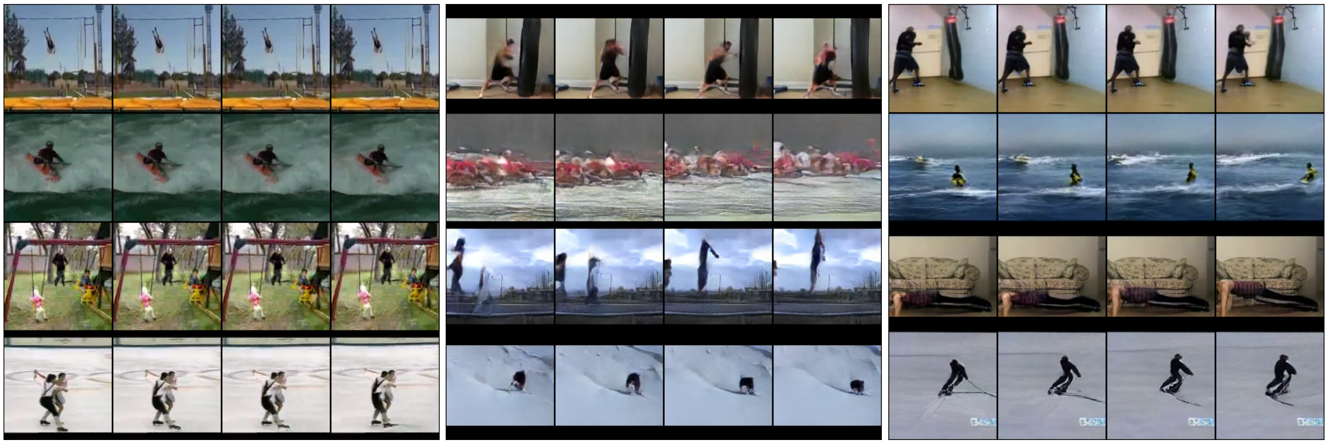
## 4.2. Single-Task Video Generation

**Class-conditional generation.** The model is given a class identifier in this task to generate the full video. Tab. 1 shows that MAGVIT surpasses the previous best-published FVD and IS scores. Notably, it outperforms Make-A-Video [50] which is pretrained on additional 10M videos with a text-image prior. In contrast, MAGVIT is just trained on the 9.5K training videos of UCF-101.

Fig. 4 compares the generated videos to baseline models. We can see that CCVS+StyleGAN [41] gets a decent single-frame quality, but yields little or no motion. TATS [21] generates some motion but with artifacts. In contrast, our model produces higher-quality frames with substantial motion.

**Frame prediction.** The model is given a single or a few frames to generate future frames. In Tab. 2, we compare MAGVIT against highly-competitive baselines. MAGVIT surpasses the previous state-of-the-art FVD on BAIR by a large margin (84 → 62). Inspired by [61], a "debiased" FVD is also reported in the parentheses to overcome the small validation set. See more discussion in Appendix B.3. In Tab. 3, it demonstrates better image quality.

On the large dataset of Kinetics-600, it establishes a new state-of-the-art result, improving the previous best FVD in [33] from 16.2 to 9.9 by a relative 39% improvement.



(a) CCVS+StyleGAN [41] (b) TATS [21] (c) MAGVIT-L-CG (ours)

Figure 4. **Comparison of class-conditional generation samples on UCF-101.** 16-frame videos are generated at 128×128 resolution 25 fps and shown at 6.25 fps. Samples for [21, 41] are obtained from their official release[2]. More comparisons are provided in Appendix D.

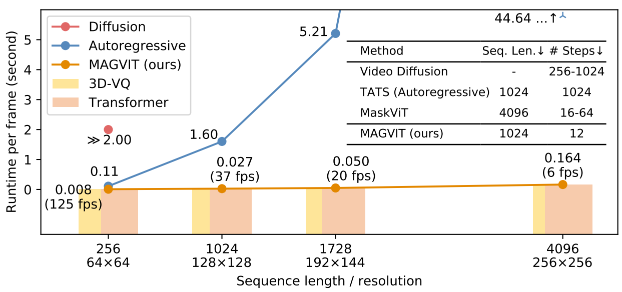



Figure 5. **Inference-time generation efficiency comparison.** The average runtime for generating one frame is measured at different resolutions. The colored bars show the time breakdown between the 3D-VQ and the transformer. The embedded table compares the critical factors of inference efficiency for different methods at 16-frame 128×128, except for Video Diffusion [33] at 64×64.

The above results verify MAGVIT's compelling generation quality, including on the large Kinetics dataset.

## 4.3. Inference-Time Generation Efficiency

Video generation efficiency is an important metric in many applications. We conduct experiments to validate that MAGVIT offers top speed in video generation. Fig. 5 shows the processing time for each frame on a single V100 GPU at different resolutions. We compare MAGVIT-B with an autoregressive transformer of the same size and a diffusion-based model [33]. At 128×128 resolution, MAGVIT-B runs at 37 frames-per-second (fps). When running on a single TPUv4i [36], MAGVIT-B runs at 190 fps and MAGVIT-L runs at 65 fps.

Fig. 5 compares the sequence lengths and inference steps of these models. Diffusion models [33] typically require 256-1000 diffusion steps with a 3D U-Net [14]. Autoregressive models, such as TATS [21], decode visual tokens sequentially, which runs 60 times slower than MAGVIT at 128×128. Compared to the recent non-autoregressive model MaskViT [26], MAGVIT is 4 to 16 times faster due to more efficient decoding on shorter sequences.

## 4.4. Multi-task Video Generation

To demonstrate the flexibility in multi-task video synthesis, we train a single MAGVIT model to perform eight tasks on BAIR or ten tasks on SSv2. We do not intend to compare with dedicated models trained on these tasks but to demonstrate a generic model for video synthesis.

**Eight tasks on BAIR.** We perform a multi-task evaluation on BAIR with eight self-supervised tasks. Tab. 4 lists the "debiased" FVD for each task, where the third column computes the average. We compare the multi-task models (MT) with two single-task baselines trained on unconditional generation (UNC) and frame prediction (FP).

As shown in Tab. 4, the multi-task models achieve better fidelity across all tasks. Single-task models perform considerably worse on the tasks unseen in training (gray values in Tab. 4), especially on the tasks that differ more from the training task. Compared to the single-task models in their training task, MT performs better with a small gain on FP with the same model size.

**Ten tasks on SSv2.** We evaluate on the large-scale SSv2 dataset, where MAGVIT needs to synthesize 174 basic actions with everyday objects. We evaluate a total of ten tasks, with two of them using class labels (CG and CFP), as shown on the right side of Tab. 4. We observe a pattern consistent with BAIR: multi-task models achieve better average FVD across all tasks. The above results substantiate model generalization trained with the proposed multi-task objective.

## 4.5. Ablation Study

**Conditional MTM.** We demonstrate the efficacy of COMMIT by comparing it with conventional MTM meth-

[2] https://songweige.github.io/projects/tats/

| Method | Task | BAIR-MT8↓ | FP | FI | OPC | OPV | OPH | OPD | IPC | IPD | SSV2-MT10↓ | CG | CFP |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MAGVIT-B-UNC | Single | 150.6 | 74.0 | 71.4 | 119.0 | 46.7 | 55.9 | 389.3 | 145.0 | 303.2 | 258.8 | 107.7 | 279.0 |
| MAGVIT-B-FP | Single | 201.1 | 47.7 | 56.2 | 247.1 | 118.5 | 142.7 | 366.3 | 357.3 | 272.7 | 402.9 | 1780.0 | 59.3 |
| MAGVIT-B-*MT* | Multi | 32.8 | 47.2 | 36.0 | 28.1 | 29.0 | 27.8 | 32.1 | 31.1 | 31.0 | 43.4 | 94.7 | 59.3 |
| MAGVIT-L-*MT* | Multi | **22.8** | 31.4 | 26.4 | 21.3 | 21.2 | 19.5 | 20.9 | 21.3 | 20.3 | **27.3** | 79.1 | 28.5 |

Table 4. **Multi-task generation performance on BAIR and SSV2 evaluated by FVD.** Gray values denote unseen tasks during training. We list per-task FVD for all eight tasks on BAIR and the two extra tasks on SSV2 here, and leave the details for SSV2 in Appendix C.

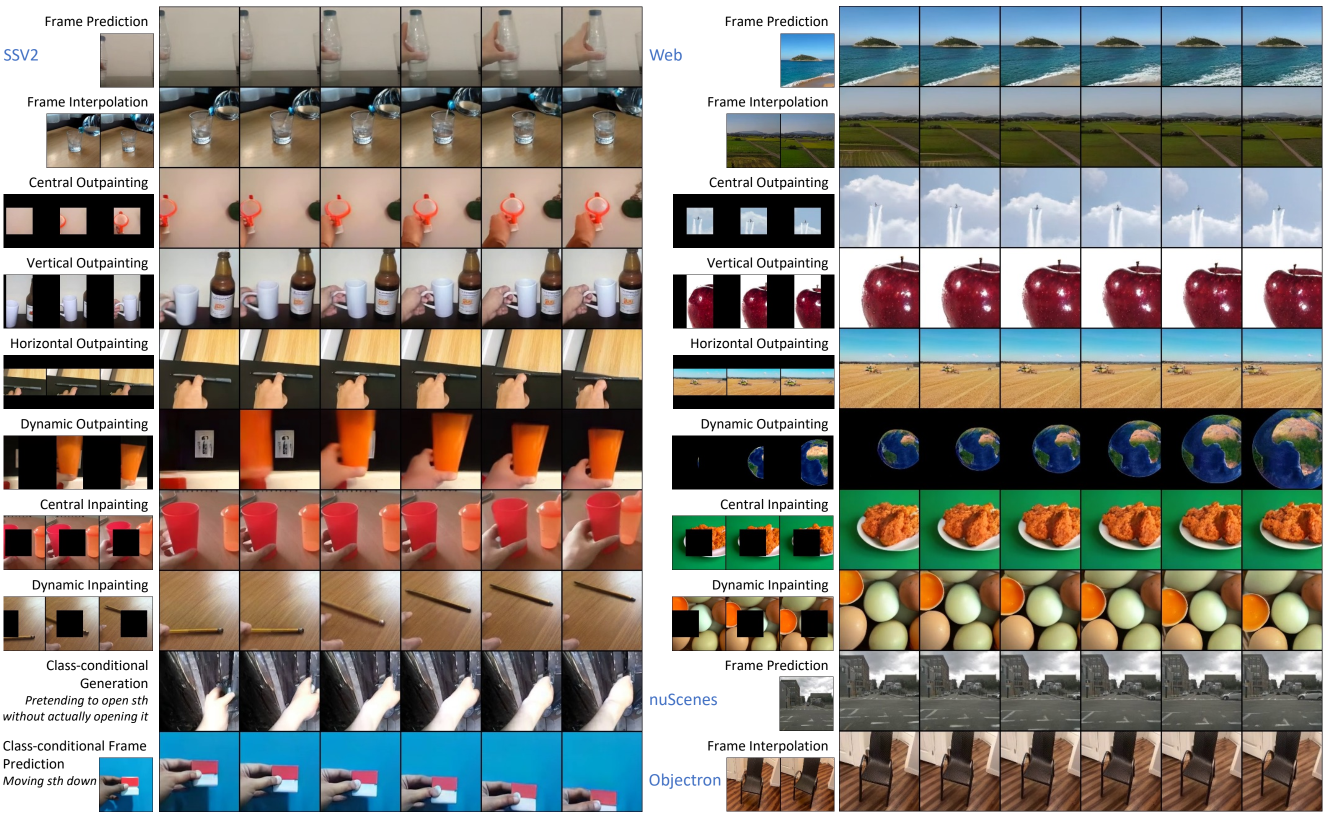



Figure 6. **Multi-task generation samples on four datasets: SSv2 [23], nuScenes [9], Objectron [2], and Web videos.** The left column is from a single ten-task model on SSv2, while the top eight rows on the right are from a single eight-task model on Web data.

| Method | | Seq. Length | FP FVD↓ | MT8 FVD↓ |
|---|---|---|---|---|
| Latent masking in MaskGIT [12] | | 1024 | 74 | 151 |
| Prefix condition | | 1024-1792 | 55 | - |
| *COMMIT* (ours) | $\mathcal{L}_{\text{mask}}$ | | 388 | 143 |
| | $\mathcal{L}_{\text{mask}} + \mathcal{L}_{\text{recons}}$ | 1024 | 51 | 53 |
| | $\mathcal{L}_{\text{mask}} + \mathcal{L}_{\text{recons}} + \mathcal{L}_{\text{refine}}$ | | **48** | **33** |

Table 5. **Comparison of conditional masked token modeling** on BAIR frame prediction (FP) and eight-task (MT8) benchmarks. - indicates we were not able to train to convergence.

ods, including the latent masking in MaskGIT for image synthesis [12] and the commonly-used prefix condition that prepends cropped condition tokens to the input sequence.

Tab. 5 compares these methods on the BAIR dataset where the same 3D-VQ tokenizer is used in all approaches. As discussed in Section 3.2, latent masking in [12], which directly unmasks tokens of the condition region at inference time, leads to poor generalization, especially for the multi-task setup. Prefix condition produces a long sequence of variable length, making it less tractable for multi-task learning. In contrast, COMMIT yields a fixed-length sequence and better generalizability for both single- and multi-task setups.

**Training losses.** The bottom section of Tab. 5 shows the contribution of the training loss components in Eq. (4).

**Decoding methods.** Tab. 6 compares Algorithm 1 with existing autoregressive (AR) and non-autoregressive (NAR) decoding methods. We consider two NAR baselines, *i.e.*, MaskGIT [12] for image and MaskViT [26] for video synthesis. We use the same 3D-VQ tokenizer for MaskGIT, AR, and MAGVIT. As shown, the proposed decoding algo-

| Decoding Method | Tokenizer | Type | Param. | Seq. Len.↓ | # Steps↓ | FVD↓ |
|---|---|---|---|---|---|---|
| MaskGIT [12] | 2D-VQ | NAR | 53M+87M | 4096 | 12 | 222 (177) |
| | 3D-VQ | NAR | 41M+87M | 1024 | 12 | 122 (74) |
| MaskViT [26] | 2D-VQ | NAR | 53M+189M | 4096 | 18 | 94* |
| AR | 3D-VQ | AR | 41M+87M | 1024 | 1024 | 91 (56) |
| *MAGVIT* (ours) | 3D-VQ | NAR | 41M+87M | 1024 | 12 | **76** (**48**) |

Table 6. **Comparison of decoding methods** on BAIR frame prediction benchmark. The number of parameters is broken down as VQ + Transformer. NAR is non-autoregressive and AR is autoregressive. FVD and debiased FVD (in parentheses) are reported. * marks the quoted number from their paper.

| Tokenizer | From Scratch | | ImageNet [16] Initialization | | | |
|---|---|---|---|---|---|---|
| | FVD↓ | IS↑ | FVD↓ | IS↑ | FVD↓ | IS↑ |
| MaskGIT [12] 2D-VQ | 240 | 80.9 | 216 | 82.6 | - | |
| TATS [21] 3D-VQ | 162 | 80.6 | - | | - | |
| | | | Average | | Central | |
| *MAGVIT* 3D-VQ-B (ours) | 127 | 82.1 | 103 | 84.8 | 58 | 87.0 |
| *MAGVIT* 3D-VQ-L (ours) | 45 | 87.1 | 35 | 88.3 | **25** | **88.9** |

Table 7. **Comparison of tokenizer architectures and initialization methods** on UCF-101 training set reconstruction results. The 2D-VQ compresses by 8×8 spatially and the 3D-VQ compresses by 4×8×8 spatial-temporally.

rithm produces the best quality with the 3D-VQ and has a 4× shorter sequence than the 2D-VQ. While the AR transformer obtains a reasonable FVD, it takes over 85× more steps at inference time.

**VQ architecture and training techniques.** We evaluate the design options of our 3D-VQ model in MAGVIT. Tab. 7 lists the reconstruction FVD and IS metrics on the UCF-101 training set, which are different from the generation metrics as they measure the intermediate quantization. Nevertheless, reconstruction quality bounds the generation quality.

Tab. 7 compares the proposed 3D architecture with existing 2D [12] and 3D [21] VQ architectures. We train the MaskGIT [12] 2D-VQ and our 3D-VQ with the same protocol and evaluate the official TATS [21] 3D-VQ model. We compare two inflation methods for our 3D-VQ model, *i.e.*, average [11] and central inflation.

The results show the following. First, 3D-VQ models, despite producing a higher compression rate, show better video reconstruction quality than 2D-VQ, even with fewer parameters. Second, the proposed VQ performs favorably against baseline architectures with a similar size and gets much better with a larger model. Third, ImageNet [16] initialization boosts the performance for 2D and 3D models, where the central inflation outperforms the average inflation. The results demonstrate the excellent reconstruction fidelity of our tokenizer design.

## 5. Related Work

**GAN-based approaches.** Early success in video synthesis has been made by GAN models [1,6,7,15,27,37,48,51, 56,59,66,74]. Training instability and lack of generation diversity [12] are known issues of GAN models.

**Autoregressive transformers.** Inspired by the success of GPT [8], autoregressive transformers have been adapted for image [13,18,20,46,73] and video generation [4,34,69,70]. A focus for video is autoregressive modeling of visual dynamics. Studies have switched from modeling the raw pixels [13,62] to the discrete codes in a latent space [45,71]. The state-of-the-art model TATS [21] uses two hierarchical transformers to reduce the computation for long video generation, with tokens learned by a 3D-VQGAN [20]. Unlike prior works, we introduce a non-autoregressive transformer with higher efficiency and flexibility.

**Non-autoregressive transformers.** Concurrently, a few methods use non-autoregressive transformers for image synthesis [12,42,53,76]. Section 2 reviews a state-of-the-art model called MaskGIT [12]. Compared with these approaches [26,28], we present an embedding mask to model multi-task video conditions with better quality.

**Diffusion models.** Diffusion models have recently received much attention for image synthesis. For example, the state-of-the-art video diffusion model [33] extends the image denoising diffusion model [3,32,52,54,60] by incorporating 3D U-Net [14] architectures and joint training on both images and videos. Despite its high-quality, sampling speed is a bottleneck hindering the application of diffusion models in video synthesis. We show a different solution to train a highly-efficient model that offers compelling quality.

**Multi-task video synthesis.** Multi-task video synthesis [28,44,70] is yet to be well-studied. Transframer [44] is the closest to our work, which adopts an image-level representation for autoregressive modeling of tasks based on frame prediction. We present an efficient non-autoregressive multi-task transformer, and verify the quality and efficiency on ten video generation tasks.

**Text-to-video.** All of our models are trained only on public benchmarks, except the Web video model. We leave the text-to-video task as future work. As shown in recent works [31,50,64], training such models requires large, and sometimes non-public, datasets of paired texts and images.

## 6. Conclusion

In this paper, we propose *MAGVIT*, a generic and efficient mask-based video generation model. We introduce a high-quality 3D-VQ tokenizer to quantize a video and design *COMMIT* for multi-task conditional masked token modeling. We conduct extensive experiments to demonstrate the video generation quality, efficiency, and flexibility for multi-task generation. Notably, MAGVIT establishes a new state-of-the-art quality for class conditional generation on UCF-101 and frame prediction on BAIR Robot Pushing and Kinetics-600 datasets.

## References


[1] Dinesh Acharya, Zhiwu Huang, Danda Pani Paudel, and Luc Van Gool. Towards high resolution video generation with progressive growing of sliced wasserstein gans. *arXiv:1810.02419*, 2018. 8

[2] Adel Ahmadyan, Liangkai Zhang, Artsiom Ablavatski, Jianing Wei, and Matthias Grundmann. Objectron: A large scale dataset of object-centric videos in the wild with pose annotations. In *CVPR*, 2021. 2, 5, 7

[3] Jacob Austin, Daniel Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In *NeurIPS*, 2021. 8

[4] Mohammad Babaeizadeh, Mohammad Taghi Saffar, Suraj Nair, Sergey Levine, Chelsea Finn, and Dumitru Erhan. Fitvid: Overfitting in pixel-level video prediction. *arXiv:2106.13195*, 2021. 5, 8

[5] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. 5

[6] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. In *ICLR*, 2018. 8

[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. *arXiv:2206.03429*, 2022. 8

[8] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *NeurIPS*, 2020. 8

[9] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A multimodal dataset for autonomous driving. In *CVPR*, 2020. 2, 5, 7

[10] Joao Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman. A short note about Kinetics-600. *arXiv:1808.01340*, 2018. 2, 5

[11] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the Kinetics dataset. In *CVPR*, 2017. 3, 5, 8

[12] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. MaskGIT: Masked generative image transformer. In *CVPR*, 2022. 1, 2, 3, 4, 7, 8

[13] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *ICML*, 2020. 8

[14] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3D U-Net: learning dense volumetric segmentation from sparse annotation. In *MICCAI*, 2016. 6, 8

[15] Aidan Clark, Jeff Donahue, and Karen Simonyan. Adversarial video generation on complex datasets. *arXiv:1907.06571*, 2019. 1, 5, 8

[16] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009. 8

[17] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019. 2, 5

[18] Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, et al. CogView: Mastering text-to-image generation via transformers. In *NeurIPS*, 2021. 1, 8

[19] Frederik Ebert, Chelsea Finn, Alex X Lee, and Sergey Levine. Self-supervised visual planning with temporal skip connections. In *CoRL*, 2017. 2, 5

[20] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *CVPR*, 2021. 1, 2, 3, 8

[21] Songwei Ge, Thomas Hayes, Harry Yang, Xi Yin, Guan Pang, David Jacobs, Jia-Bin Huang, and Devi Parikh. Long video generation with time-agnostic VQGAN and time-sensitive transformer. In *ECCV*, 2022. 1, 2, 3, 5, 6, 8

[22] Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. Mask-Predict: Parallel decoding of conditional masked language models. In *EMNLP-IJCNLP*, 2019. 2

[23] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The "something something" video database for learning and evaluating visual common sense. In *ICCV*, 2017. 2, 5, 7

[24] Jiatao Gu and Xiang Kong. Fully non-autoregressive neural machine translation: Tricks of the trade. In *ACL-IJCNLP Findings*, 2021. 2

[25] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In *CVPR*, 2022. 1, 2

[26] Agrim Gupta, Stephen Tian, Yunzhi Zhang, Jiajun Wu, Roberto Martín-Martín, and Li Fei-Fei. MaskViT: Masked visual pre-training for video prediction. *arXiv:2206.11894*, 2022. 2, 3, 5, 6, 7, 8

[27] Sonam Gupta, Arti Keshari, and Sukhendu Das. RV-GAN: Recurrent GAN for unconditional video generation. In *CVPRW*, 2022. 8

[28] Ligong Han, Jian Ren, Hsin-Ying Lee, Francesco Barbieri, Kyle Olszewski, Shervin Minaee, Dimitris Metaxas, and Sergey Tulyakov. Show me what and tell me how: Video synthesis via multimodal conditioning. In *CVPR*, 2022. 2, 8

[29] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3

[30] Jonathan Heek, Anselm Levskaya, Avital Oliver, Marvin Ritter, Bertrand Rondepierre, Andreas Steiner, and Marc van Zee. Flax: A neural network library and ecosystem for JAX, 2020. 5

[31] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben

Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv:2210.02303*, 2022. 8

[32] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 8

[33] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In *ICLR Workshops*, 2022. 1, 2, 5, 6, 8

[34] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. CogVideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv:2205.15868*, 2022. 5, 8

[35] Tobias Höppe, Arash Mehrjou, Stefan Bauer, Didrik Nielsen, and Andrea Dittadi. Diffusion models for video prediction and infilling. *arXiv:2206.07696*, 2022. 1, 2, 5

[36] Norman P Jouppi, Doe Hyun Yoon, Matthew Ashcraft, Mark Gottscho, Thomas B Jablin, George Kurian, James Laudon, Sheng Li, Peter Ma, Xiaoyu Ma, et al. Ten lessons from three generations shaped google's TPUv4i. In *ISCA*, 2021. 2, 6

[37] Emmanuel Kahembwe and Subramanian Ramamoorthy. Lower dimensional kernels for video discriminators. *Neural Networks*, 132:506–520, 2020. 8

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

[39] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv:1412.6980*, 2014. 5

[40] Xiang Kong, Lu Jiang, Huiwen Chang, Han Zhang, Yuan Hao, Haifeng Gong, and Irfan Essa. BLT: Bidirectional layout transformer for controllable layout generation. In *ECCV*, 2022. 2

[41] Guillaume Le Moing, Jean Ponce, and Cordelia Schmid. CCVS: Context-aware controllable video synthesis. In *NeurIPS*, 2021. 1, 3, 5, 6

[42] José Lezama, Huiwen Chang, Lu Jiang, and Irfan Essa. Improved masked image generation with Token-Critic. In *ECCV*, 2022. 2, 8

[43] Pauline Luc, Aidan Clark, Sander Dieleman, Diego de Las Casas, Yotam Doron, Albin Cassirer, and Karen Simonyan. Transformation-based adversarial video prediction on large-scale data. *arXiv:2003.04035*, 2020. 1, 5

[44] Charlie Nash, João Carreira, Jacob Walker, Iain Barr, Andrew Jaegle, Mateusz Malinowski, and Peter Battaglia. Transframer: Arbitrary frame prediction with generative models. *arXiv:2203.09494*, 2022. 1, 5, 8

[45] Ruslan Rakhimov, Denis Volkhonskiy, Alexey Artemov, Denis Zorin, and Evgeny Burnaev. Latent video transformer. In *VISIGRAPP (5: VISAPP)*, 2021. 1, 8

[46] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021. 1, 2, 8

[47] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 1

[48] Masaki Saito, Eiichi Matsumoto, and Shunta Saito. Temporal generative adversarial nets with singular value clipping. In *ICCV*, 2017. 1, 8

[49] Masaki Saito, Shunta Saito, Masanori Koyama, and Sosuke Kobayashi. Train sparsely, generate densely: Memory-efficient unsupervised training of high-resolution temporal gan. *IJCV*, 128(10):2586–2606, 2020. 1, 5

[50] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv:2209.14792*, 2022. 5, 8

[51] 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. 5, 8

[52] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015. 8

[53] Kihyuk Sohn, Yuan Hao, José Lezama, Luisa Polania, Huiwen Chang, Han Zhang, Irfan Essa, and Lu Jiang. Visual prompt tuning for generative transfer learning. *arXiv preprint arXiv:2210.00990*, 2022. 8

[54] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *NeurIPS*, 2019. 8

[55] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv:1212.0402*, 2012. 2, 3, 5

[56] 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. 8

[57] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3D convolutional networks. In *ICCV*, 2015. 5

[58] Hung-Yu Tseng, Lu Jiang, Ce Liu, Ming-Hsuan Yang, and Weilong Yang. Regularizing generative adversarial networks under limited data. In *CVPR*, 2021. 3

[59] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. MoCoGAN: Decomposing motion and content for video generation. In *CVPR*, 2018. 1, 8

[60] Belinda Tzen and Maxim Raginsky. Neural stochastic differential equations: Deep latent gaussian models in the diffusion limit. *arXiv:1905.09883*, 2019. 8

[61] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv:1812.01717*, 2018. 1, 2, 5

[62] Aäron van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *ICML*, 2016. 8

[63] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. In *NeurIPS*, 2017. 2

[64] Ruben Villegas, Mohammad Babaeizadeh, Pieter-Jan Kindermans, Hernan Moraldo, Han Zhang, Mohammad Taghi Saffar, Santiago Castro, Julius Kunze, and Dumitru Erhan. Phenaki: Variable length video generation from open domain textual description. *arXiv:2210.02399*, 2022. 5, 8

[65] Vikram Voleti, Alexia Jolicoeur-Martineau, and Christopher Pal. Masked conditional video diffusion for prediction, generation, and interpolation. In *NeurIPS*, 2022. 1, 5

[66] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *NeurIPS*, 2016. 1, 8

[67] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, et al. Image as a foreign language: BEiT pretraining for all vision and vision-language tasks. *arXiv:2208.10442*, 2022. 2

[68] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, 13(4):600–612, 2004. 5

[69] Dirk Weissenborn, Oscar Täckström, and Jakob Uszkoreit. Scaling autoregressive video models. In *ICLR*, 2019. 1, 8

[70] Chenfei Wu, Jian Liang, Lei Ji, Fan Yang, Yuejian Fang, Daxin Jiang, and Nan Duan. NÜWA: Visual synthesis pre-training for neural visual world creation. In *ECCV*, 2022. 1, 5, 8

[71] Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. VideoGPT: Video generation using vq-vae and transformers. *arXiv:2104.10157*, 2021. 2, 3, 8

[72] Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, and Yonghui Wu. Vector-quantized image modeling with improved VQGAN. In *ICLR*, 2022. 2

[73] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv:2206.10789*, 2022. 1, 8

[74] 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. 5, 8

[75] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 5

[76] Zhu Zhang, Jianxin Ma, Chang Zhou, Rui Men, Zhikang Li, Ming Ding, Jie Tang, Jingren Zhou, and Hongxia Yang. M6-UFC: Unifying multi-modal controls for conditional image synthesis. *arXiv:2105.14211*, 2021. 2, 8