MAGVIT: Masked Generative Video Transformer Supplementary Materials

Acknowledgements

The authors would like to thank Tom Duerig, Victor Gomes, Paul Natsev along with the Multipod committee for sponsoring the computing resources. We appreciate valuable feedback and leadership support from David Salesin, Jay Yagnik, Tomas Izo, and Rahul Sukthankar thoughout the project. Special thanks to Wolfgang Macherey for supporting the project. We thank David Alexander Ross and Yu-Chuan Su for many helpful comments for improving the paper. We also give thanks to Sarah Laszlo and Hugh Williams for creating the MAGVIT model card, Bryan Seybold and Albert Shaw for extending the features, Jonathan Ho and Tim Salimans for providing the JAX code pointer for FVD computation, and the Scenic team for the infrastructure support. We are thankful to Wenhe Liu, Xinyu Yao, Mingzhi Cai, Yizhi Zhang, and Zhao Jin for proof reading the paper. This project is funded in part by Carnegie Mellon University's Mobility21 National University Transportation Center, which is sponsored by the US Department of Transportation.

Appendix Overview

This supplementary document provides additional details to support our main manuscript, organized as follows:

- Appendix A presents the 3D-VQ architectures and the transformer models in MAGVIT.
- Appendix B includes additional implementation details in training and evaluation.
- Appendix C provides more quantitative evaluation results, which include:
  - Comparisons to more published results on the three benchmarks in the paper: UCF-101 [31], BAIR [12, 36], and Kinetics-600 [6].
  - Multi-task results on Something-Something-v2 (SSv2) [14].
  - Results on three additional datasets: NuScenes [5], Objectron [3] and Web video datasets.
- Appendix D shows more qualitative examples of the generated videos.

We present a demo video for MAGVIT and show more generated examples on this web page[1] .

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

A. MAGVIT Model Architecture

A.1. 3D-VQ Tokenizer

Fig. 7 shows the architectures of the MAGVIT 3D-VQ module and compares it with the 3D-VQ module in TATS [13] which held the previous state-of-the-art for video generation. Compared with TATS, the major design choices in MAGVIT 3D-VQ are listed below.

- Average pooling, instead of strided convolution, is used for down-sampling.
- Nearest resizing and convolution are used for up-sampling.
- We use spatial down- and up-sampling layers near the latent space and spatial-temporal down- and up-sampling layers near the pixel space, resulting in mirrored encoder-decoder architecture.
- A single deeper 3D discriminator is designed rather than two shallow discriminators for 2D and 3D separately.
- We quantize into a much smaller vocabulary of 1,024 as compared to 16,384.
- We use group normalization [43] instead of batch normalization [20] and Swish [26] activation function instead of SiLU [16].
- We use the LeCAM regularization [34] to improve the training stability and quality.

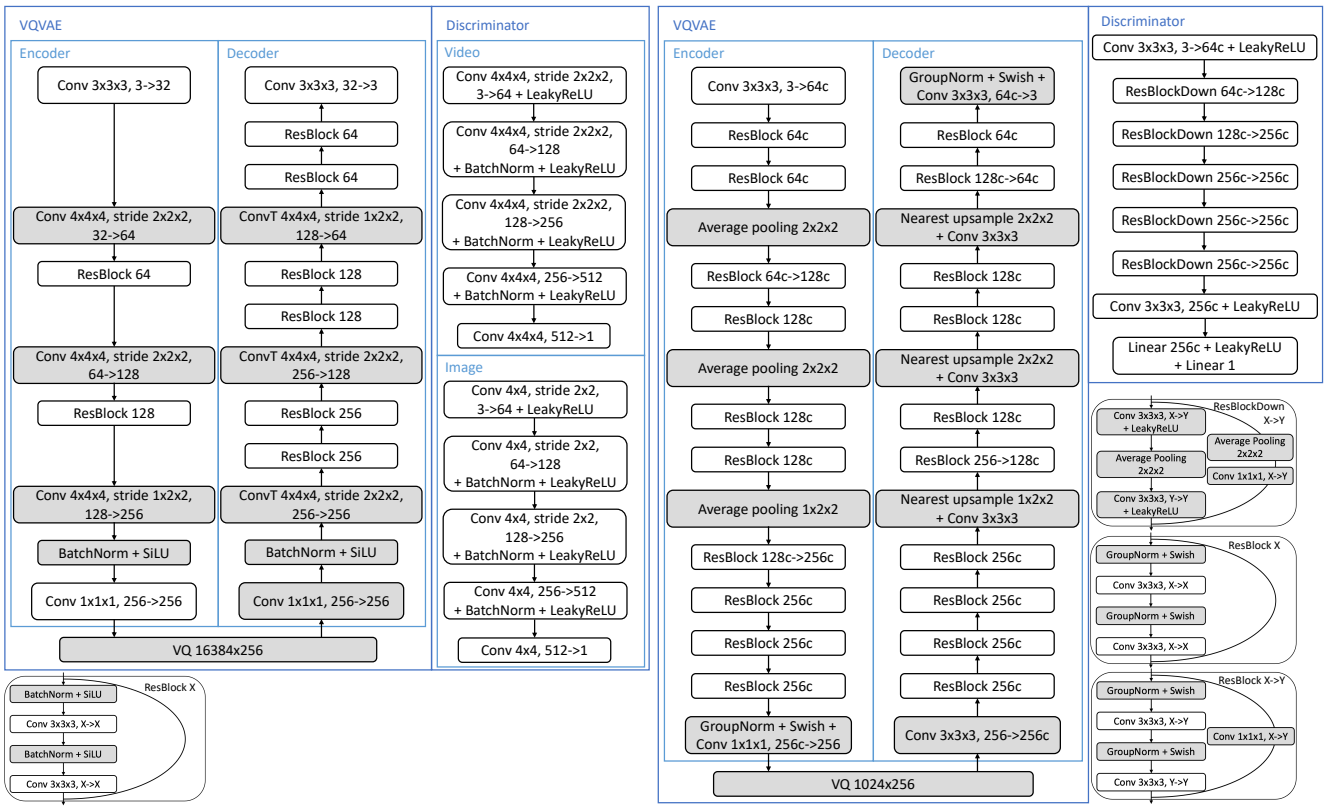
The quantitative comparison of the 3D-VQ from TATS and MAGVIT were presented in Table 6 of the main paper. In addition, Fig. 9 below qualitatively compares their reconstruction quality on UCF-101. Figs. 10 and 11 show MAGVIT's high-quality reconstruction on example YouTube videos.

We design two variants of the MAGVIT 3D-VQ module, *i.e.*, the base (B) with 41M parameters and the large (L) with 158M parameters, excluding the discriminators.

A.2. Transformer

MAGVIT uses the BERT transformer architecture [10] adapted from the Flaxformer implementation[2]. Following the transformer configurations in ViT [11], we use two variants of transformers, *i.e.*, base (B) with 87M parameters and large (L) with 306M in all our experiments. Tab. 8 lists the

[2] https://github.com/google/flaxformer



(a) TATS [13] 3D-VQ. (32M+14M parameters)

(b) *MAGVIT* (ours) 3D-VQ.
(41M+15M parameters at $c = 1$ (B), 158M+61M at $c = 2$ (L))

Figure 7. **Comparison of 3D-VQ model architectures between MAGVIT and the TATS [13].** We highlight the blocks with major differences in gray background and detail their design differences in Appendix A.1. We train the models to quantize 16-frame clips of 128×128 resolution into $4 \times 16 \times 16$ tokens. The number of parameters in parentheses are broken down between VQVAE and discriminators.

detailed configurations for each variant. A huge (H) transformer is only used to train on the large Web video dataset and generate demo videos.

B. Implementation Details

B.1. Task Definitions

We employ a total of ten tasks for multi-task video generation. Each task is characterized by a few adjustable settings such as interior condition shape, padding function, and optionally prefix condition. Fig. 8 illustrates the interior condition regions for each task under the above setup. Given a video of shape $T \times H \times W$, we define the tasks as following:

- Frame Prediction (FP)
  - Interior condition: t frames at the beginning; $t = 1$.
  - Padding: replicate the last given frame.
- Frame Interpolation (FI)
  - Interior condition: t_1 frames at the beginning and t_2 frames at the end; $t_1 = 1, t_2 = 1$.
  - Padding: linear interpolate between the last given frame at the beginning and the first given frame at the end.
- Central Outpainting (OPC)
  - Interior condition: a rectangle at the center with height h and width w; $h = 0.5H$, $w = 0.5W$.
  - Padding: pad the nearest pixel for each location (`edge` padding).
- Vertical Outpainting (OPV)
  - Interior condition: a centered vertical strip with width w; $w = 0.5W$.
  - Padding: `edge` padding.
- Horizontal Outpainting (OPH)
  - Interior condition: a centered horizontal strip with height h; $h = 0.5H$.
  - Padding: `edge` padding.
- Dynamic Outpainting (OPD)
  - Interior condition: a moving vertical strip with width w; $w = 0.5W$.
  - Direction of movement: left to right.
  - Padding: zero padding.

Model	Param.	# heads	# layers	Hidden size	MLP dim
MAGVIT-**B**	87 M	12	12	768	3072
MAGVIT-**L**	305 M	16	24	1024	4096
MAGVIT-**H**	634 M	16	32	1280	5120

Table 8. **Transformer architecture configurations used in MAGVIT.**

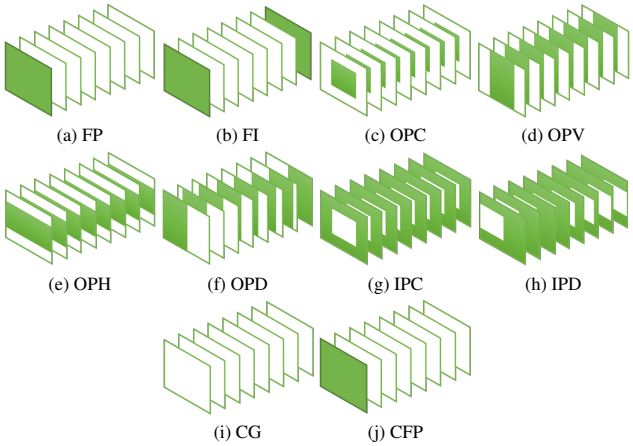


Figure 8. **Interior condition regions for each task**, where green denotes valid pixels and white pixels denote the task-specific paddings discussed in Appendix B.1. The tasks are Frame Prediction (FP), Frame Interpolation (FI), Central Outpainting (OPC), Vertical Outpainting (OPV), Horizontal Outpainting (OPH), Dynamic Outpainting (OPD), Central Inpainting (IPC), Dynamic Inpainting (IPD), Class-conditional Generation (CG), and Class-conditional Frame Prediction (CFP).

- Central Inpainting (IPC)
  - Interior condition: everything but a rectangle at the center with height h and width w; $h = 0.5H$, $w = 0.5W$.
  - Padding: zero padding.
- Dynamic Inpainting (IPD)
  - Interior condition: everything but a vertically centered moving rectangle with height h and width w; $h = 0.5H$, $w = 0.5W$.
  - Direction of movement: left to right.
  - Padding: zero padding.
- Class-conditional Generation (CG)
  - Prefix condition: class label.
- Class-conditional Frame Prediction (CFP)
  - Prefix condition: class label.
  - Interior condition: t frames at the beginning; $t = 1$.
  - Padding: replicate the last given frame.

B.2. Training

MAGVIT is trained in two stages where we first train the 3D-VQ tokenizer and then train the transformer with a frozen tokenizer. We follow the same learning recipe across all datasets, with the only variation in the number of training epochs. Here are the training details for both stages:

- 3D-VQ:
  - Video: 16 frames, frame stride 1, 128×128 resolution. (64×64 resolution for BAIR)
  - Base channels: 64 for B, 128 for L.
  - VQVAE channel multipliers: 1, 2, 2, 4. (1, 2, 4 for 64×64 resolution).
  - Discriminator channel multipliers: 2, 4, 4, 4, 4. (2, 4, 4, 4 for 64×64 resolution)
  - Latent shape: 4×16×16.
  - Vocabulary size: 1,024.
  - Embedding dimension: 256.
  - Initialization: central inflation from a 2D-VQ trained on ImageNet with this setup.
  - Peak learning rate: 10^{-4}.
  - Learning rate schedule: linear warm up and cosine decay.
  - Optimizer: Adam with $\beta_1 = 0$ and $\beta_2 = 0.99$.
  - Generator loss type: Non-saturating.
  - Generator adversarial loss weight: 0.1.
  - Perceptual loss weight: 0.1.
  - Discriminator gradient penalty: r1 with cost 10.
  - EMA model decay rate: 0.999.
  - Batch size: 128 for B, 256 for L.
  - Speed: 0.41 steps/sec on 16 TPU-v2 chips for B, 0.56 steps/sec on 32 TPU-v4 chips for L.
- Transformer:
  - Sequence length: 1026.
  - Hidden dropout rate: 0.1.
  - Attention dropout rate: 0.1.
  - Mask rate schedule: cosine.
  - Peak learning rate: 10^{-4}.
  - Learning rate schedule: linear warm up and cosine decay.
  - Optimizer: Adam with $\beta_1 = 0.9$ and $\beta_2 = 0.96$.
  - Weight decay 0.045.
  - Label smoothing: 10^{-4}.
  - Max gradient norm: 1.
  - Batch size: 256.
  - Speed: 1.24 steps/sec on 16 TPU-v2 chips for B, 2.70 steps/sec on 32 TPU-v4 chips for L.

Using more hardware resources can speed up the training. We train MAGVIT models for each dataset separately.

Dataset	3D-VQ		Transformer	
	B	L	B	L
UCF-101	500	2000	2000	2000
BAIR	400	800	400	800
BAIR-MT	400	800	1200	1600
Kinetics-600	45	180	180	360
SSv2	135	400	720	1440
nuScenes	1280	5120	2560	10240
Objectron	1000	2000	1000	2000
Web	5	20	10	20

Table 9. **Training epochs for each dataset.**

The training epochs for each dataset are listed in Tab. 9.

B.3. Evaluation

Evaluation metrics. The FVD [36] is used as the primary evaluation metric. We follow the official implementation[3] in extracting video features with an I3D model trained on Kinetics-400 [7]. We report Inception Score (IS) [28][4] on the UCF-101 dataset which is calculated with a C3D [33] model trained on UCF-101. We further include image quality metrics: PSNR, SSIM [40] and LPIPS [46] (computed by the VGG features) on the BAIR dataset.

Sampling protocols. We follow the sampling protocols from previous works [9, 13] when eveluating on the standard benchmarks, *i.e*. UCF-101, BAIR, and Kinetics-600. We sample 16-frame clips from each dataset without replacement to form the real distribution in FVD and extract condition inputs from them to feed to the model. We continuously run through all the samples required (*e.g*., 40,000 for UCF-101) with a single data loader and compute the mean and standard deviation for 4 folds. When evaluating on other datasets, due to the lack of prior works, we adapt the above protocol based on the dataset size to ensure sample diversity.

For our MAGVIT model, we use the following COMMIT decoding hyperparameters by default: cosine schedule, 12 steps, temperature 4.5. Below are detailed setups for each dataset:

- UCF-101:
  - Dataset: 9.5K videos for training, 101 classes.
  - Number of samples: 10,000×4.
  - Resolution: 128×128.
  - Real distribution: random clips from the training videos.
- BAIR:
  - Dataset: 43K videos for training and 256 videos for evaluation.
  - Number of samples: 25,600×4.
  - Resolution: 64×64.
  - Real distribution: the first 16-frame clip from each evaluation video.
  - COMMIT decoding: exponential schedule, temperature 400.
- Kinetics-600:
  - Dataset: 384K videos for training and 29K videos for evaluation.
  - Number of samples: 50,000×4.
  - Generation resolution: 128×128.
  - Evaluation resolution: 64×64, via central crop and bilinear resize.
  - Real distribution: 6 sampled clips (2 temporal windows and 3 spatial crops) from each evaluation video.
  - COMMIT decoding: uniform schedule, temperature 7.5.
- SSv2:
  - Dataset: 169K videos for training and 24K videos for evaluation, 174 classes.
  - Number of samples: 50,000×4.
  - Resolution: 128×128.
  - Real distribution for the CG task: random clips from the training videos.
  - Real distribution for the other tasks: 2 sampled clips (2 temporal windows and central crop) from each evaluation video.
- nuScenes:
  - Dataset: 5.4K videos for training and 0.6K videos for evaluation, front camera only, 32 frames per video.
  - Number of samples: 50,000×4.
  - Resolution: 128×128.
  - Real distribution: 48 sampled clips (16 temporal windows and 3 spatial crops) from each evaluation video.
- Objectron:
  - Dataset: 14.4K videos for training and 3.6K videos for evaluation.
  - Number of samples: 50,000×4.
  - Resolution: 128×128.
  - Real distribution: 5 sampled clips (5 temporal windows and central crop) from each evaluation video.
- Web videos:
  - Dataset: ∼12M videos for training and 26K videos for evaluation.
  - Number of samples: 50,000×4.
  - Resolution: 128×128.
  - Real distribution: randomly sampled clips from evaluation videos.

For the "random clips" above, we refer to the combination of a random temporal window and a random spatial crop on a random video. For the fixed number of "tempo-

[3] https://github.com/google-research/google-research/tree/master/frechet_video_distance

[4] https://github.com/pfnet-research/tgan2

Method	Extra Video	Class	FVD↓	IS↑
VGAN [39]		✓	-	8.31±0.09
TGAN [27]			-	11.85±0.07
MoCoGAN* [35]		✓	-	12.42±0.07
ProgressiveVGAN [2]		✓	-	14.56±0.05
TGAN [27]		✓	-	15.83±0.18
RaMViD [19]			-	21.71±0.21
LDVD-GAN [21]			-	22.91±0.19
StyleGAN-V*# [30]			-	23.94±0.73
VideoGPT [44]			-	24.69±0.30
TGANv2 [28]		✓	1209±28	28.87±0.67
MoCoGAN-HD# [32]			838	32.36
DIGAN [45]			655±22	29.71±0.53
DIGAN# [45]			577±21	32.70±0.35
DVD-GAN# [9]		✓	-	32.97±1.70
Video Diffusion*# [17]			-	57.00±0.62
TATS [13]			420±18	57.63±0.24
CCVS+StyleGAN# [22]			386±15	24.47±0.13
Make-A-Video* [29]		✓	367	33.00
TATS [13]		✓	332±18	79.28±0.38
CogVideo* [18]	✓	✓	626	50.46
Make-A-Video* [29]	✓	✓	81	82.55
MAGVIT-B-CG (ours)		✓	159±2	83.55±0.14
MAGVIT-L-CG (ours)		✓	**76**±2	**89.27**±0.15

Table 10. **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. Methods marked with # additionally used the test set in training.

Method	K600 FVD↓	BAIR FVD↓
LVT [25]	224.7	126±3
Video Transformer [41]	170.0±5.0	94±2
CogVideo* [18]	109.2	-
DVD-GAN-FP [9]	69.1±1.2	110
CCVS [22]	55.0±1.0	99±2
Phenaki [37]	36.4±0.2	97
VideoGPT [44]	-	103
TrIVD-GAN-FP [23]	25.7±0.7	103
Transframer [24]	25.4	100
MaskViT [15]	-	94
FitVid [4]	-	94
MCVD [38]	-	90
NÜWA [42]	-	87
RaMViD [19]	16.5	84
Video Diffusion [17]	16.2±0.3	-
MAGVIT-B-FP (ours)	24.5±0.9	76±0.1 (47±0.1)
MAGVIT-L-FP (ours)	**9.9**±0.3	**62**±0.1 (31±0.2)

Table 11. **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. Methods marked with * is pretrained on additional large video data.

ral windows" or "spatial crops", deterministic uniform sampling is used.

For the image quality metrics on BAIR in Table 3 of the main paper, CCVS [22] generates at 256×256 while the others are at 64×64. When calculating PSNR and SSIM, we follow [38] in using the best value from 100 trials for each evaluation video.

Debiased FVD on BAIR Computing FVD is difficult on the BAIR dataset due to its small evaluation target of only 256 16-frame clips. Following the standard evaluation protocol, we generate 100 predictions for each clip to create 256,00 samples [4].

The real distribution to compute FVD in this way is highly biased with the insufficient evaluation videos [36]. We can see this by a simple experiment where we compute the training FVD with only 256 training videos. We observe that this 256-sample training FVD (64) is far worse than the regular training FVD with all 43K videos (13), showing the biased FVD computation.

To bridge the gap, we use uniformly sampled 16-frame clips from the 256 30-frame evaluation videos, which results in $256 \times 15 = 3840$ clips. The uniform sampling yields a better representation of the evaluation set. Under this new protocol, MAGVIT-L-FP achieves FVD 31 instead of 62, which is more aligned with its training set performance (FVD=8).

We report this "debiased FVD" in addition to the standard FVD computation on the BAIR dataset, with the default COMMIT decoding hyperparameters. We also use it for BAIR multi-task evaluation and ablation studies on BAIR .

C. Additional Quantitative Evaluation

Class-conditional generation. Tab. 10 shows a detailed comparison with the previously published results on the UCF-101 [31] class-conditional video generation benchmark, where the numbers are quoted from the cited papers. Note that CogVideo [18] and Make-A-Video [29] are pretrained on additional 5-10M videos before finetuning on UCF-101, where Make-A-Video further uses a text-image prior trained on a billion text-image pairs. The remaining models, including MAGVIT, are only trained on 9.5K train-

Method	Task	**Avg↓**	FP	FI	OPC	OPV	OPH	OPD	IPC	IPD	CG	CFP
MAGVIT-B-UNC	Single	258.8	278.8	91.0	67.5	27.3	36.2	711.5	319.3	669.8	107.7	279.0
MAGVIT-B-FP	Single	402.9	59.3	76.2	213.2	81.2	86.3	632.7	343.1	697.9	1780.0	59.3
MAGVIT-B-*MT*	Multi	43.4	71.5	38.0	38.8	23.3	26.1	33.4	23.3	25.3	94.7	59.3
MAGVIT-L-*MT*	Multi	**27.3**	33.8	25.0	21.1	16.8	17.0	23.5	13.5	15.0	79.1	28.5
Masked pixel	-	-	94%	87%	75%	50%	50%	50%	25%	25%	100%	94%
Masked token	-	-	75%	50%	75%	50%	50%	50%	25%	25%	100%	75%

Table 12. **Multi-task generation performance on Something-Something-V2 evaluated by FVD.** Gray values denote unseen tasks during training. The bottom two rows list the proportions of masked pixels and tokens for each task.

Method	nuScenes-FP	Objectron-FI	Web-**MT8**	FP	FI	OPC	OPV	OPH	OPD	IPC	IPD
MAGVIT-B	29.3	-	33.0	84.9	33.9	34.4	21.5	22.1	26.0	20.7	20.4
MAGVIT-L	20.6	26.7	**21.6**	45.5	30.9	19.9	15.3	14.5	20.2	12.0	14.7

Table 13. **Generation performance on NuScenes, Objectron, and Web videos evaluated by FVD.**

ing videos of UCF-101, or 13.3K training and testing videos of UCF-101 for those marked with #. Fig. 12 provides a visual comparison to the baseline methods.

As shown, even the smaller MAGVIT-B performs favorably against previous state-of-the-art model TATS [13] by a large margin. MAGVIT-L pushes both the FVD (332 $\rightarrow$ 76, $\downarrow$ 77%) and IS (79.28 $\rightarrow$ 89.27, $\uparrow$ 13%) to a new level, while outperforming the contemporary work Make-A-Video [29] which is pretrained on significantly large extra training data.

Frame prediction. For the frame prediction task on BAIR Robot Pushing [12, 36] (1-frame condition) and Kinetics-600 [6] (5-frame condition), Tab. 11 provides a detailed comparison with previously published results. We use "-" to mark the FVDs that either is unavailable in their paper or incomparable to others. For example, Video Diffusion [17]'s FVD reported in their paper was on a different camera angle (top-down view `image_main`[5]) and is hence incomparable to others.

MAGVIT achieves state-of-the-art quality in terms of FVD on both datasets, with a 39% relative improvement on the large-scale Kinetics benchmark than the highly-competitive Video Diffusion baseline [17]. Fig. 13 and Fig. 14 below provide visual comparisons to the baseline methods on BAIR and Kinetics-600, respectively.

Multi-task video generation. Having verified single-task video generation, Tab. 12 shows per-task performance of the ten tasks on the large-scale Something-Something-v2 (SSv2) [14] dataset, with the proportions of masks in both pixel and token spaces. SSv2 is a challenging dataset commonly used for action recognition, whereas this work benchmarks video generation on it for the first time. On this dataset, a model needs to synthesize 174 basic actions with everyday objects. Fig. 15 shows examples of generated videos for each task on this dataset.

We compare the multi-task models (MT) with two single-task baselines trained on unconditional generation (UNC) and frame prediction (FP). The multi-task models show consistently better average FVD across all tasks compared with the single-task baselines.

Results on nuScenes, Objectron, and 12M Web Videos. Tab. 13 shows the generation performance on three additional datasets, *i.e.*, nuScenes [5], Objectron [3], and 12M Web videos which contains 12 million videos we collected from the web. We evaluate our model on the frame prediction task on nuScenes, the frame interpolation task on Objectron, and the 8-task suite on the Web videos. Fig. 16 shows examples of generated videos for each task. The results substantiate the generalization performance of MAGVIT on videos from distinct visual domains and the multi-task learning recipe on large-scale data.

Tokenizer reconstruction. We report the image quality metrics (PSNR, SSIM, LPIPS) for the VQGAN reconstruction in Tab. 14. We compare MAGVIT 3D against the baseline MaskGIT 2D to highlight our 3D design while keeping the remaining components the same. As shown, the results in Tab. 14 are consistent with the findings by FVD in Tab. 6.

[5] https://www.tensorflow.org/datasets/catalog/bair_robot_pushing_small

VQ Tokenizer	From Scratch			ImageNet Initialization					
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
MaskGIT 2D	21.4	0.667	0.139	21.5	0.685	0.114		-	
					Average			Central	
MAGVIT 3D-L	21.8	0.690	0.113	21.9	0.697	0.103	**22.0**	**0.701**	**0.099**

Table 14. **Image quality metrics of different tokenizers** on UCF-101 training set reconstruction.

D. Qualitative Examples

D.1. High-Fidelity Tokenization

Comparison of tokenizers. Fig. 9 compares the reconstruction quality of the three VQ tokenizers on the UCF-101, including the 2D-VQ from MaskGIT [8], the 3D-VQ from TATS [13], and MAGVIT 3D-VQ, where the videos are taken from the UCF-101 training set. We obtain the TATS model from their official release [6]. We train the MaskGIT 2D-VQ and MAGVIT 3D-VQ using the same protocol on the UCF-101 dataset.

We can see that the MaskGIT 2D-VQ produces a reasonable image quality, but falls short of frame consistency which causes significant flickering when played as a video (*e.g.*, the curtain color in the first row and the wall color in the third row). TATS 3D-VQ has a better temporal consistency but loses details for moving objects (*e.g.*, the woman's belly in the second row). In contrast, our 3D VQ produces consistent frames with greater details reconstructed for both static and moving pixels.

Scalable tokenization. Since the tokenizers are trained in an unsupervised manner, they exhibit remarkable generalization performances and can be scaled to big data as no labels are required. To demonstrate this, we train a large MAGVIT 3D-VQ on the large YouTube-8M [1] dataset while ignoring the labels, and use the model to quantize randomly sampled videos on YouTube.

Figs. 10 and 11 show the original and reconstructed videos from YouTube at 240p (240 $\times$ 432) resolution with arbitrary lengths (*e.g.* 4,096 frames). Although the tokenizer is only trained with 16-frame 128$\times$128 videos, it produces high reconstruction fidelity for high spatial-temporal resolutions that are unseen in training. Our 3D-VQ model compresses the video by a factor of 4 temporally, by 8$\times$8 spatially, and by 2.4 (24 bits $\rightarrow$ 10 bits) per element, yielding a 614.4$\times$ compression rate. Despite such high compression, the reconstructed results show stunning details and are almost indistinguishable from the real videos.

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

D.2. Single-Task Generation Examples

Fig. 12 compares the generated samples from CCVS+StyleGAN [22], the prior state-of-the-art TATS [13], and MAGVIT on the UCF-101 class-conditional generation benchmark. As shown in Fig. 12, CCVS+StyleGAN [22] gets a decent single-frame quality attributing to the pretrained StyleGAN, but yields little or no motion. TATS [13] generates some motion but with clear artifacts. In contrast, our model produces higher-quality frames with substantial motion.

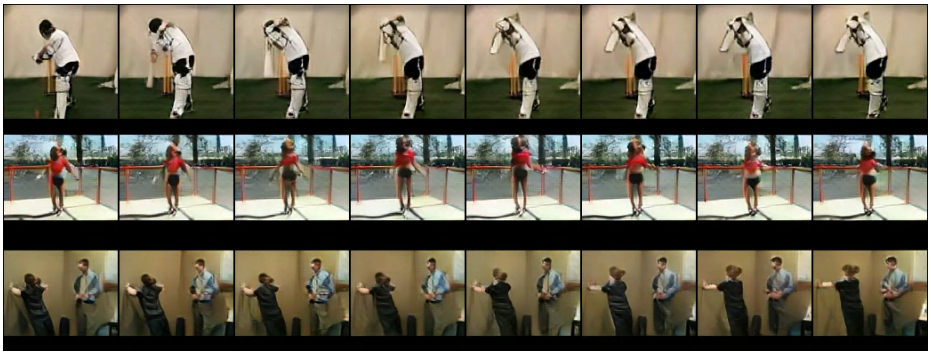
Fig. 13 compares the generated samples between the state-of-the-art RaMViD [19] and MAGVIT on the BAIR frame prediction benchmark given 1-frame condition. As shown, the clips produced by MAGVIT maintaining a better visual consistency and spatial-temporal dynamics.

Fig. 14 compares the generated samples from RaMViD [19] and MAGVIT on the Kinetics-600 frame prediction benchmark given 5-frame condition. Note that RaMViD generates video in 64$\times$64 and MAGVIT in 128$\times$128 where the standard evaluation is carried out on 64$\times$64. As shown, given the conditioned frames, MAGVIT generates plausible actions with greater details.

D.3. Multi-Task Generation Examples

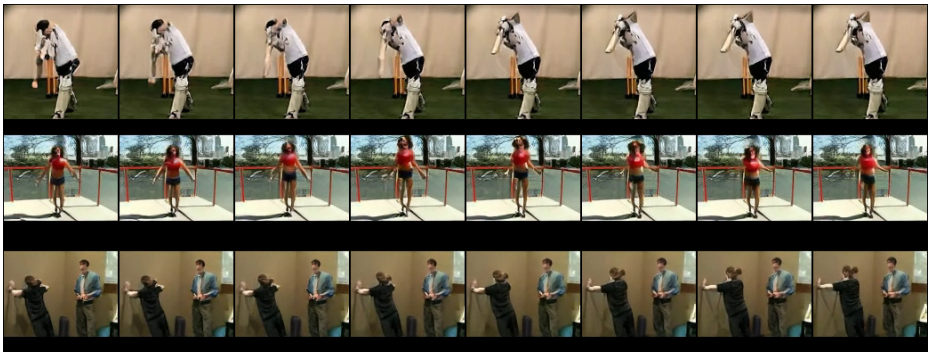
Fig. 15 shows multi-task generation results on 10 different tasks from a single model trained on SSv2. Fig. 16 shows multi-task samples for three other models trained on nuScenes, Objectron, and Web videos. These results substantiate the multi-task flexibility of MAGVIT.

The diverse video generation tasks that MAGVIT is capable of can enable many useful applications. For example, Figs. 17 and 18 show a few untrawide outpainting samples by repeatedly performing the vertical outpainting task. MAGVIT can easily generate nice large panorama videos given a small condition.

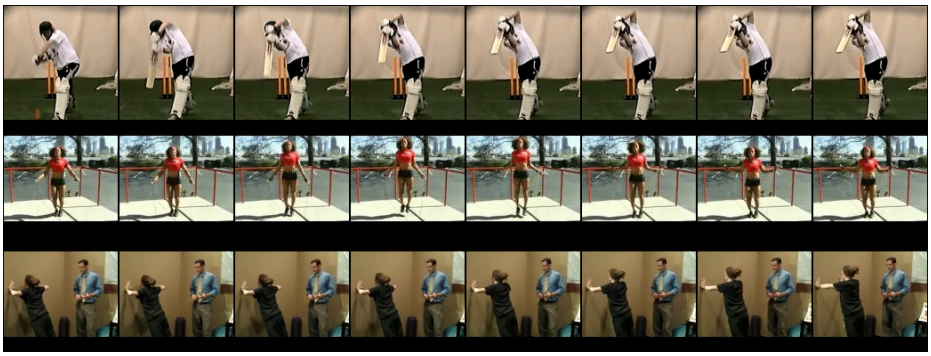


(a) MaskGIT [8] 2D-VQ

(b) TATS [13] 3D-VQ



(c) MAGVIT 3D-VQ-L (ours)



(d) Real

Figure 9. **Comparison of tokenizers on UCF-101 training set reconstruction.** Videos are reconstructed at 16 frames 64×64 resolution 25 fps and shown at 12.5 fps, with the ground truth in (d). MaskGIT 2D-VQ produces a reasonable image quality, but falls short of frame consistency which causes significant flickering when played as a video (*e.g.*, the curtain color in the first row and the wall color in the third row). TATS 3D-VQ has a better temporal consistency but loses details for moving objects (*e.g.*, the woman's belly in the second row). In contrast, our 3D VQ produces consistent frames with greater details reconstructed for both static and moving pixels.

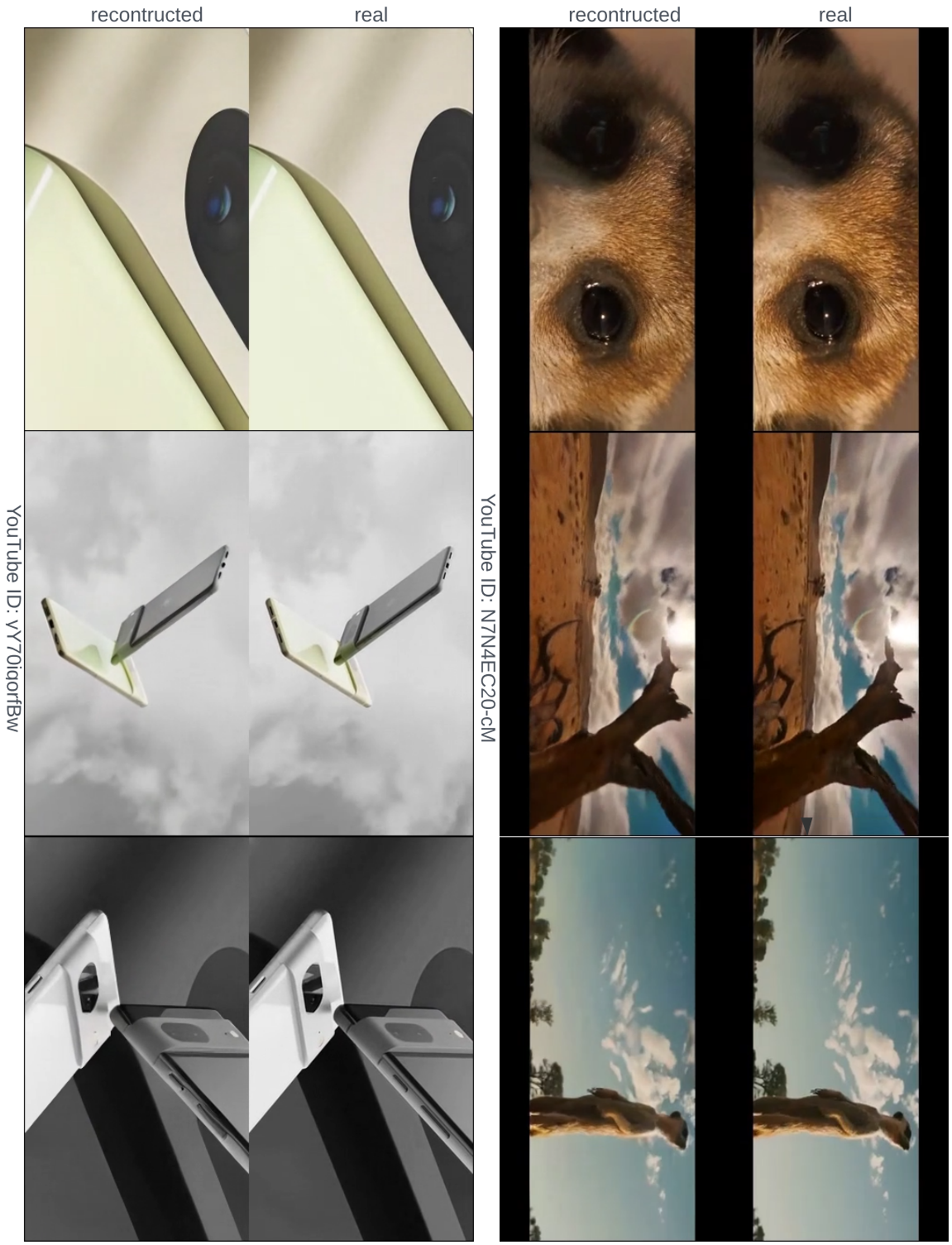


Figure 10. **Our 3D-VQ model produces high reconstruction fidelity with scalable spatial-temporal resolution.** For each group, the top row contains real YouTube videos and the bottom row shows the reconstructed videos from the discrete tokens. The original videos are in 240p (240×432) resolution with N frames. Our 3D-VQ model represents the video as $\frac{N}{4} \times 30 \times 54$ discrete tokens with a codebook of size 1024, representing a total compression rate of 614.4. Despite such high compression, the reconstructed results show stunning details and are almost indistinguishable from the real videos.

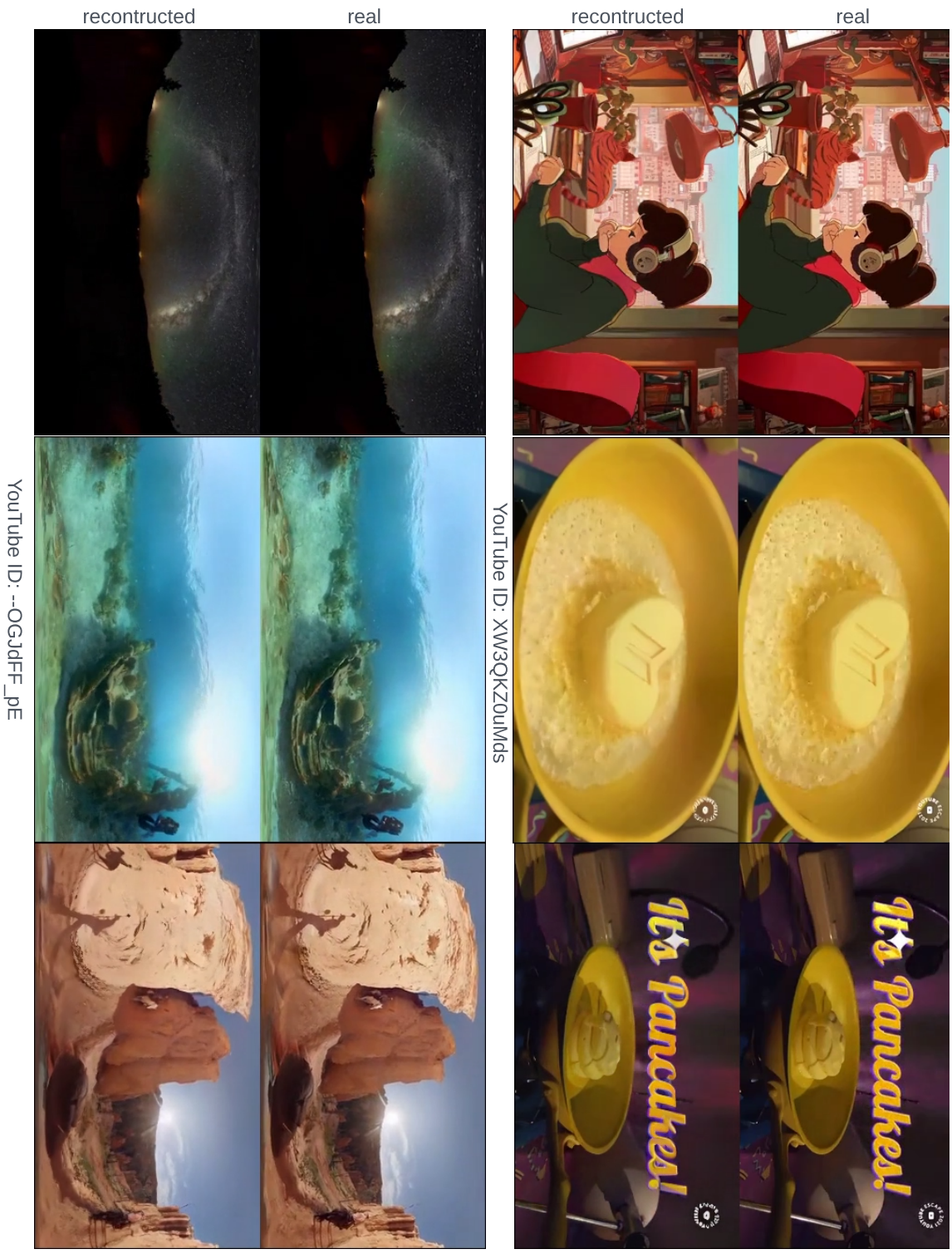
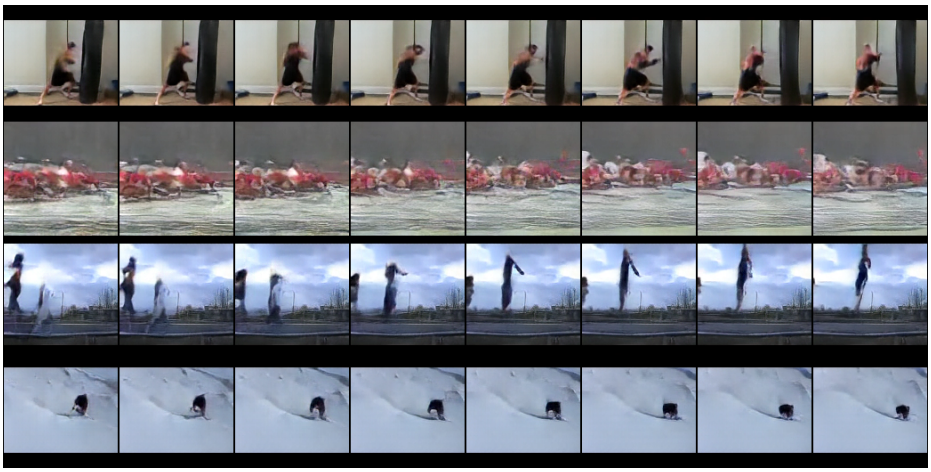


Figure 11. **Our 3D-VQ model produces high reconstruction fidelity with scalable spatial-temporal resolution.** For each group, the top row contains real YouTube videos and the bottom row shows the reconstructed videos from the discrete tokens. The original videos are in 240p (240×432) resolution with N frames. Our 3D-VQ model represents the video as $\frac{N}{4} \times 30 \times 54$ discrete tokens with a codebook of size 1024, representing a total compression rate of 614.4. Despite such high compression, the reconstructed results show stunning details and are almost indistinguishable from the real videos.

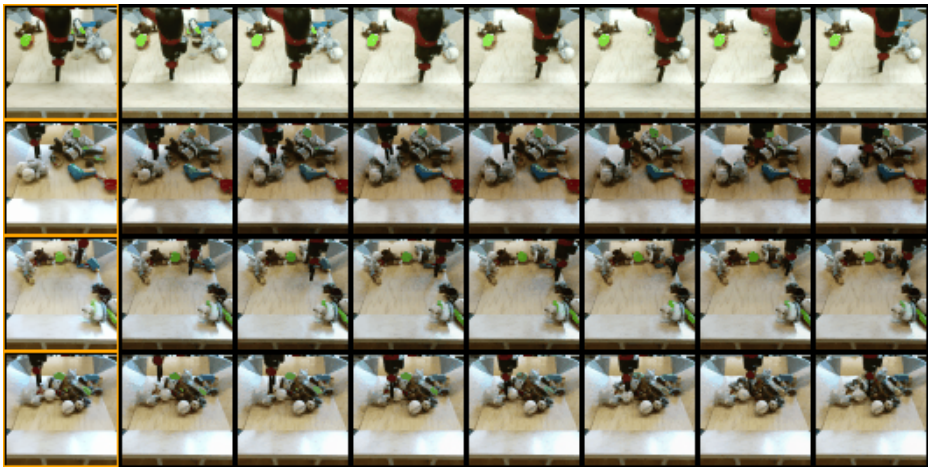
(a) CCVS+StyleGAN [22]



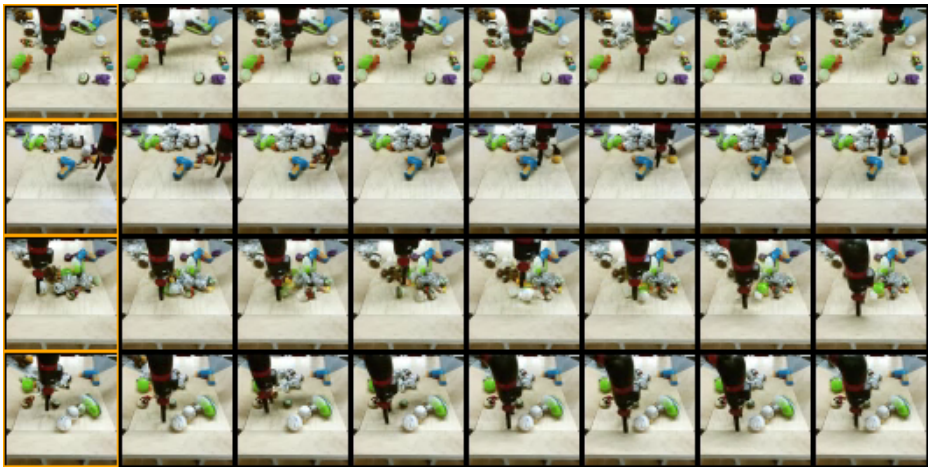
(b) TATS [13]

(c) MAGVIT-L-CG (ours)

Figure 12. **Comparison of class-conditional generation samples on UCF-101.** 16-frame videos are generated at 128×128 resolution 25 fps and shown at 12.5 fps. Samples for [13, 22] are obtained from their official release (https://songweige.github.io/projects/tats/). CCVS+StyleGAN gets a decent single-frame quality attributing to the pretrained StyleGAN, but yields little or no motion. TATS generates some motion but with clear artifacts. In contrast, our model produces higher-quality frames with substantial motion.

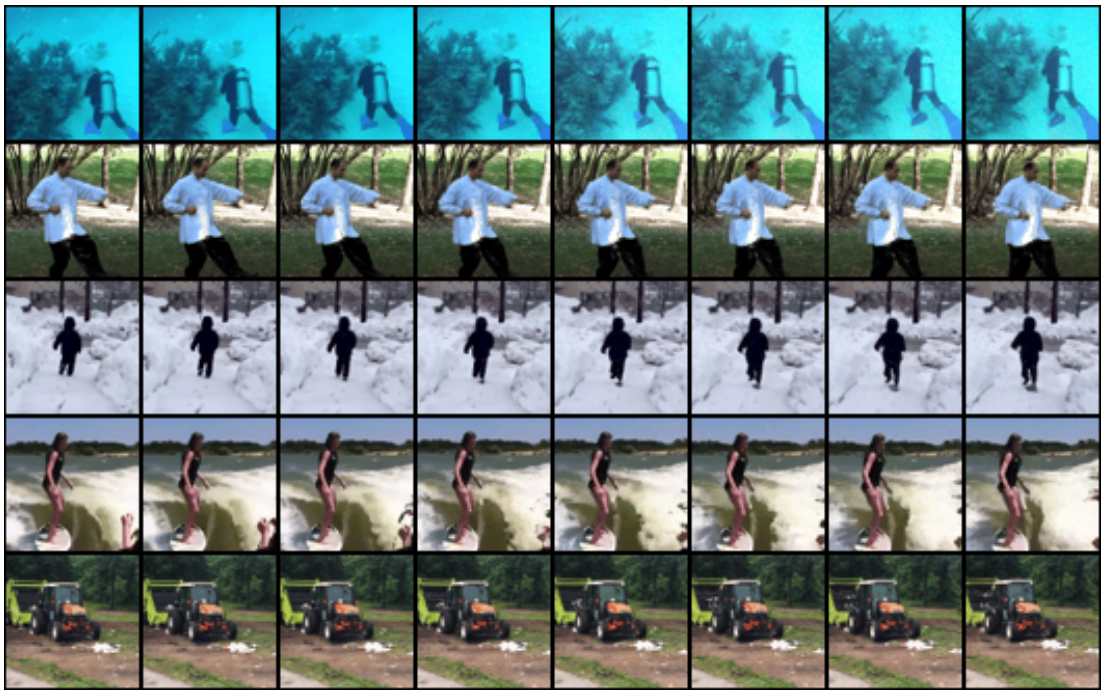


(a) RaMViD [19]

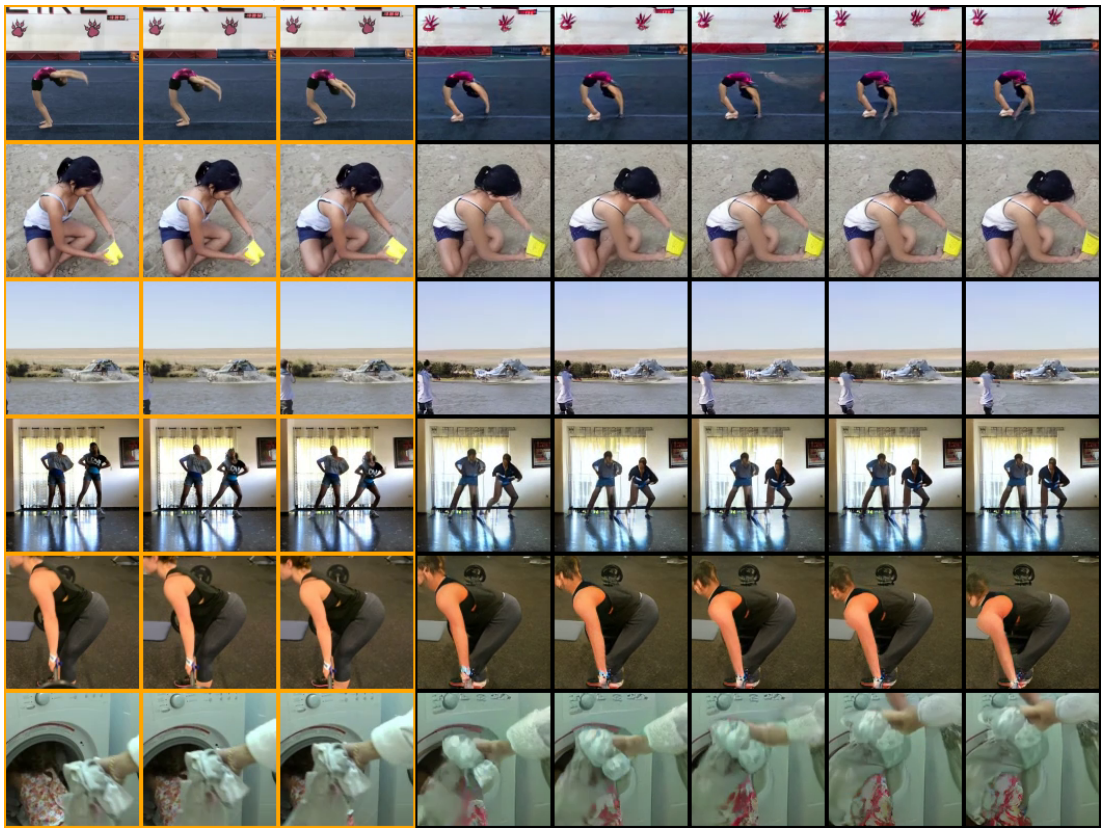


(b) MAGVIT-L-FP (ours)

Figure 13. **Comparison of frame prediction samples on BAIR unseen evaluation set.** 16-frame videos are generated at 64×64 resolution 10 fps given the first frame as condition and shown at 5 fps where condition frames are marked in orange. Samples for [19] are obtained from their official release (https://sites.google.com/view/video-diffusion-prediction). As shown, the clips produced by MAGVIT maintaining a better visual consistency and spatial-temporal dynamics.



(a) RaMViD [19] at 64×64 resolution, condition information is unavailable.



(b) MAGVIT-L-FP (ours) at 128×128 resolution, condition frames are marked in orange.

Figure 14. **Comparison of frame prediction samples on Kinetics-600 unseen evaluation set.** 16-frame videos are generated at 25 fps given 5-frame condition. Samples for [19] are obtained from their official release (https://sites.google.com/view/video-diffusion-prediction). As shown, given the conditioned frames, MAGVIT generates plausible actions with greater details.

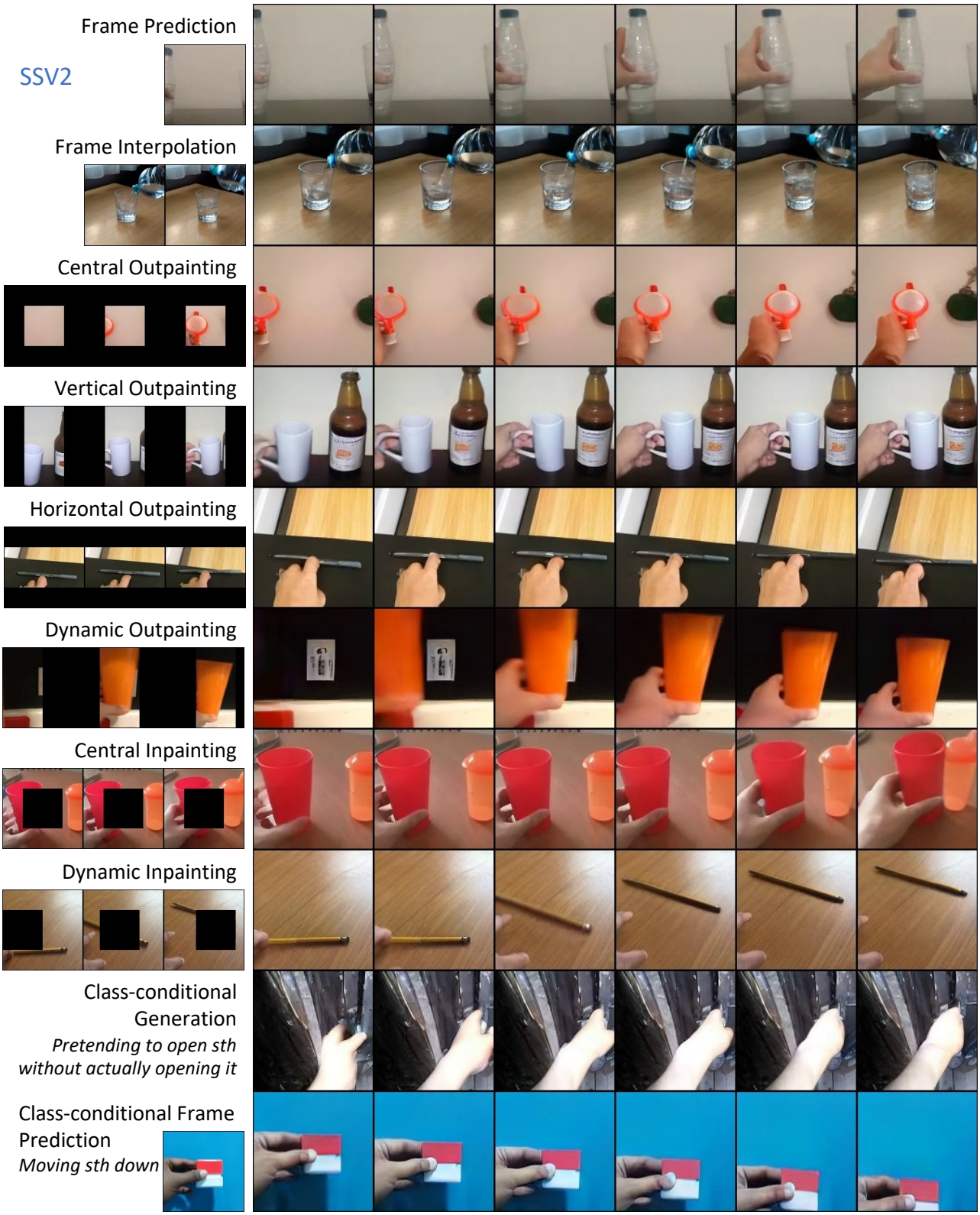


Figure 15. **Multi-task generation results** for the model only trained on the Something-Something-V2 dataset [14]. The condition used to generate the shown videos are taken from the Something-Something-V2 evaluation videos.

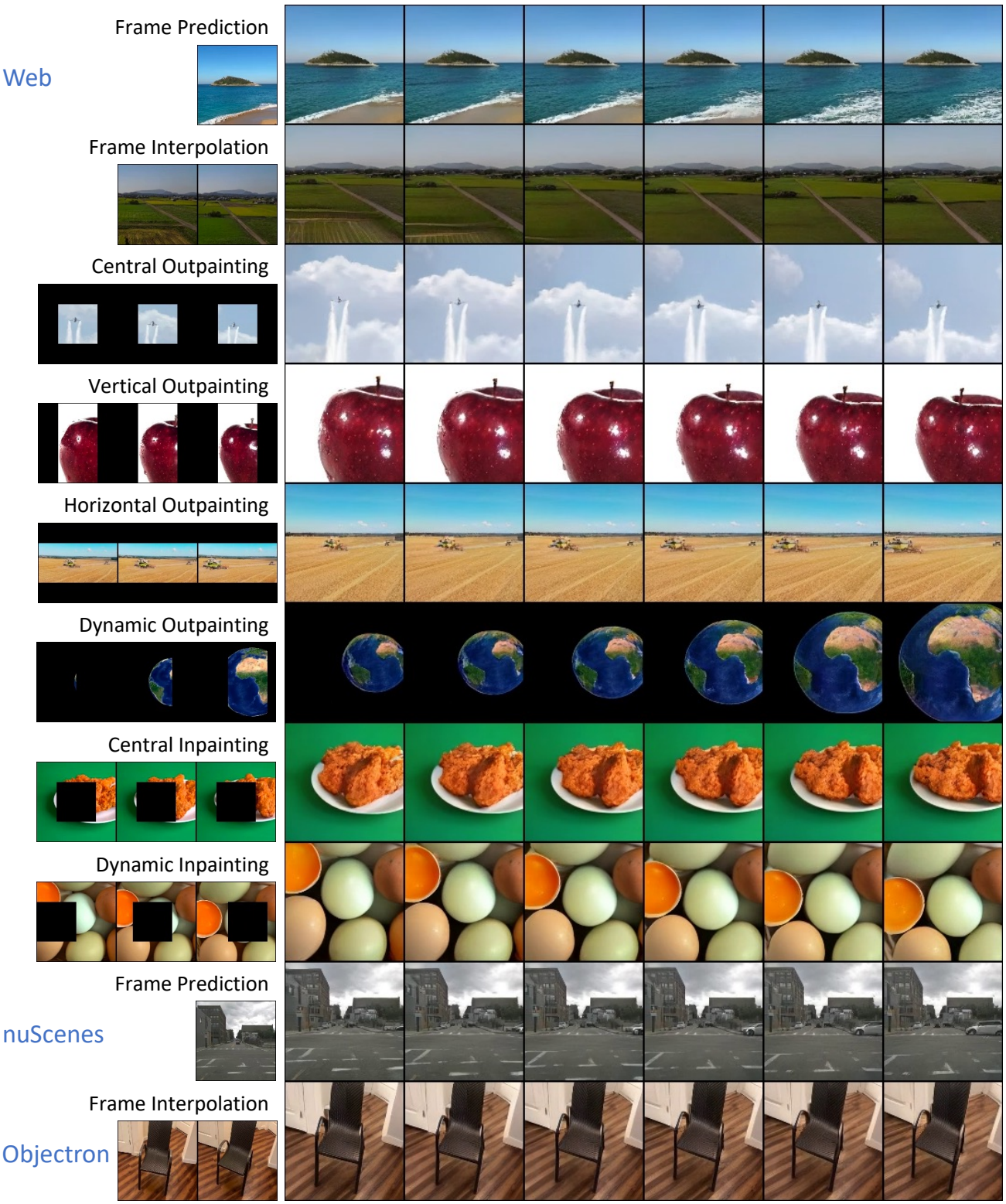


Figure 16. **Multi-task generation results** for three models trained on nuScenes [5], Objectron [3], and 12M Web videos, respectively. The condition used to generate the shown videos are taken from the evaluation set.

Figure 17. **Ultrawide outpainting results**. Given a vertical slice of 64×128, MAGVIT expands it into a panorama video of 384×128 by doing vertical outpainting for 5 times on each side.

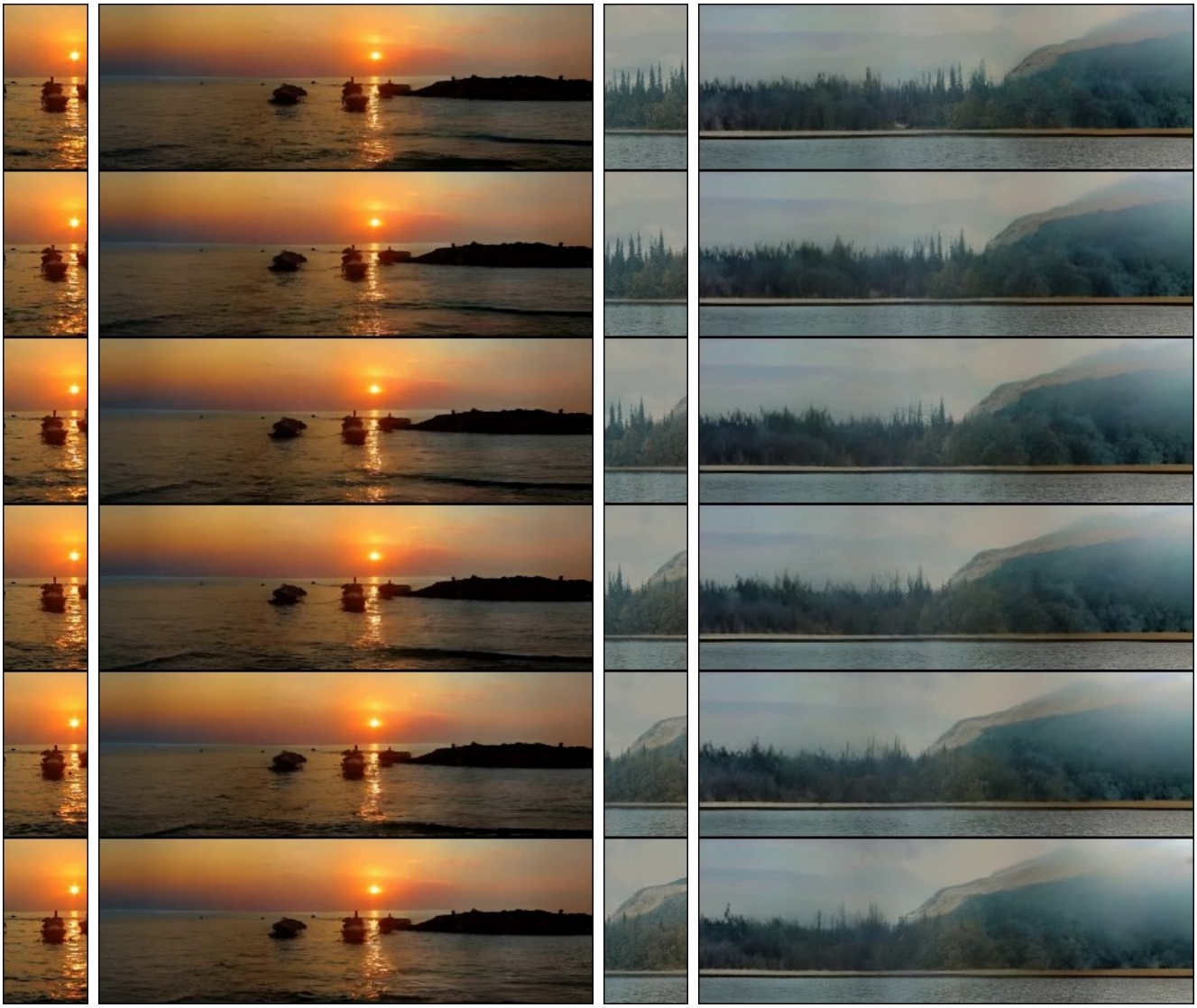


Figure 18. **Ultrawide outpainting results**. Given a vertical slice of 64×128, MAGVIT expands it into a panorama video of 384×128 by doing vertical outpainting for 5 times on each side.

References

[1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*, 2016. 18

[2] 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. 16

[3] 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. 12, 17, 26

[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. 16

[5] 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. 12, 17, 26

[6] Joao Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman. A short note about Kinetics-600. *arXiv:1808.01340*, 2018. 12, 17

[7] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the Kinetics dataset. In *CVPR*, 2017. 15

[8] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. MaskGIT: Masked generative image transformer. In *CVPR*, 2022. 18, 19

[9] Aidan Clark, Jeff Donahue, and Karen Simonyan. Adversarial video generation on complex datasets. *arXiv:1907.06571*, 2019. 15, 16

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019. 12

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2020. 12

[12] Frederik Ebert, Chelsea Finn, Alex X Lee, and Sergey Levine. Self-supervised visual planning with temporal skip connections. In *CoRL*, 2017. 12, 17

[13] 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. 12, 13, 15, 16, 17, 18, 19, 22

[14] 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. 12, 17, 25

[15] 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. 16

[16] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv:1606.08415*, 2016. 12

[17] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In *ICLR Workshops*, 2022. 16, 17

[18] 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. 16

[19] Tobias Höppe, Arash Mehrjou, Stefan Bauer, Didrik Nielsen, and Andrea Dittadi. Diffusion models for video prediction and infilling. *arXiv:2206.07696*, 2022. 16, 18, 23, 24

[20] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015. 12

[21] Emmanuel Kahembwe and Subramanian Ramamoorthy. Lower dimensional kernels for video discriminators. *Neural Networks*, 132:506–520, 2020. 16

[22] Guillaume Le Moing, Jean Ponce, and Cordelia Schmid. CCVS: Context-aware controllable video synthesis. In *NeurIPS*, 2021. 16, 18, 22

[23] 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. 16

[24] 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. 16

[25] Ruslan Rakhimov, Denis Volkhonskiy, Alexey Artemov, Denis Zorin, and Evgeny Burnaev. Latent video transformer. In *VISIGRAPP (5: VISAPP)*, 2021. 16

[26] Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. In *ICLR Workshops*, 2018. 12

[27] Masaki Saito, Eiichi Matsumoto, and Shunta Saito. Temporal generative adversarial nets with singular value clipping. In *ICCV*, 2017. 16

[28] 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. 15, 16

[29] 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. 16, 17

[30] 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. 16

[31] 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. 12, 16

[32] 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. 16

[33] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3D convolutional networks. In *ICCV*, 2015. 15

[34] Hung-Yu Tseng, Lu Jiang, Ce Liu, Ming-Hsuan Yang, and Weilong Yang. Regularizing generative adversarial networks under limited data. In *CVPR*, 2021. 12

[35] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. MoCoGAN: Decomposing motion and content for video generation. In *CVPR*, 2018. 16

[36] 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. 12, 15, 16, 17

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

[38] Vikram Voleti, Alexia Jolicoeur-Martineau, and Christopher Pal. Masked conditional video diffusion for prediction, generation, and interpolation. In *NeurIPS*, 2022. 16

[39] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *NeurIPS*, 2016. 16

[40] 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. 15

[41] Dirk Weissenborn, Oscar Täckström, and Jakob Uszkoreit. Scaling autoregressive video models. In *ICLR*, 2019. 16

[42] 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. 16

[43] Yuxin Wu and Kaiming He. Group normalization. In *ECCV*, 2018. 12

[44] Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. VideoGPT: Video generation using vq-vae and transformers. *arXiv:2104.10157*, 2021. 16

[45] 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. 16

[46] 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. 15