

Latent-Shift: Latent Diffusion with Temporal Shift for Efficient Text-to-Video Generation

Jie An^{1,2*} Songyang Zhang^{1,2*} Harry Yang² Sonal Gupta² Jia-Bin Huang^{2,3}
 Jiebo Luo^{1,2} Xi Yin²

¹University of Rochester ²Meta AI ³University of Maryland, College Park

{jan6, jluo}@cs.rochester.edu szhang83@ur.rochester.edu

{harryyang, sonalgupta, yinxixi}@meta.com jbhuang@umd.edu

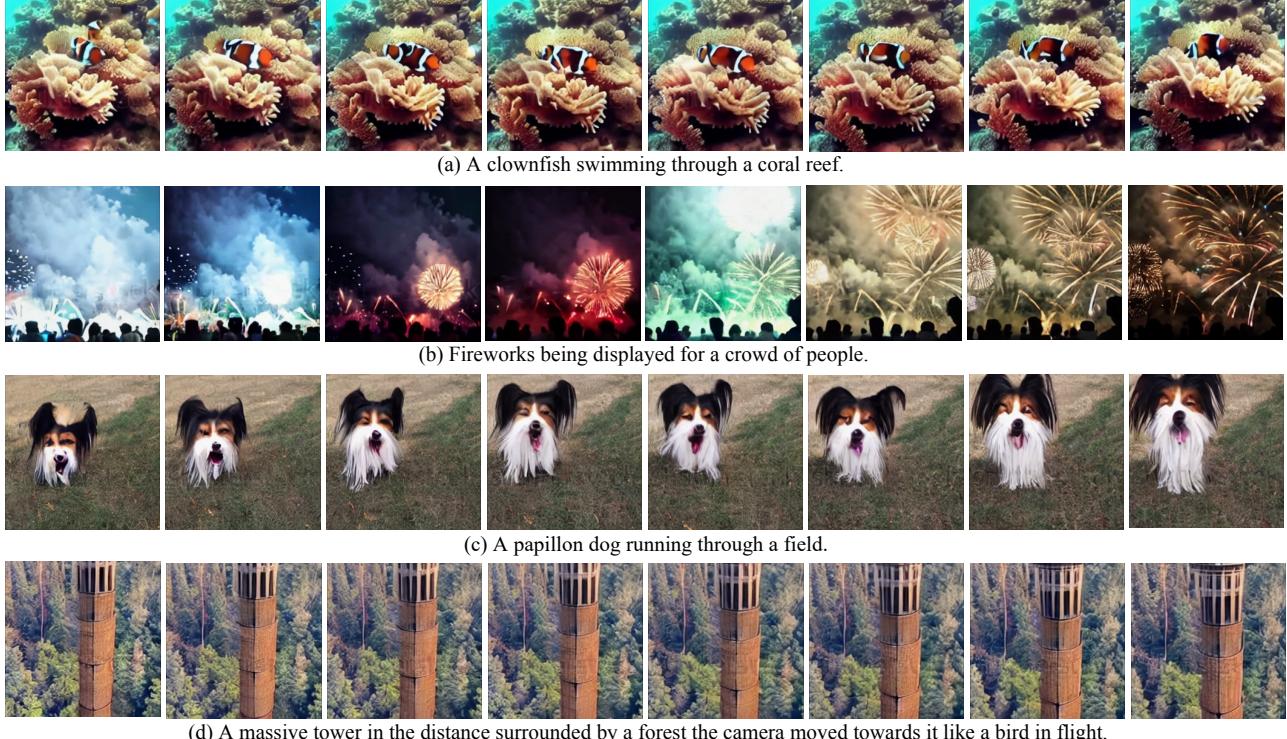


Figure 1. Text-to-Video generation results. Our method can generate rich contents with meaningful motions including both object/scene motions ((a-c)) and camera motion ((d)). please check our project page <https://latent-shift.github.io> for video samples.

Abstract

We propose Latent-Shift — an efficient text-to-video generation method based on a pretrained text-to-image generation model that consists of an autoencoder and a U-Net diffusion model. Learning a video diffusion model in the latent space is much more efficient than in the pixel space. The latter is often limited to first generating a low-resolution video followed by a sequence of frame interpolation and

*Equal contribution, ordered alphabetically. This work was done when Jie and Songyang interned at Meta AI.

super-resolution models, which makes the entire pipeline very complex and computationally expensive. To extend a U-Net from image generation to video generation, prior work proposes to add additional modules like 1D temporal convolution and/or temporal attention layers. In contrast, we propose a parameter-free temporal shift module that can leverage the spatial U-Net as is for video generation. We achieve this by shifting two portions of the feature map channels forward and backward along the temporal dimension. The shifted features of the current frame thus receive the features from the previous and the subse-

quent frames, enabling motion learning without additional parameters. We show that Latent-Shift achieves comparable or better results while being significantly more efficient. Moreover, Latent-Shift can generate images despite being finetuned for T2V generation.

1. Introduction

In the last two years, tremendous progress has been made in generative modeling in AI. Text-to-Image (T2I) generation systems [7, 21, 27, 29, 42] trained on large-scale text-image pairs can now generate high-quality images with novel scene compositions. In particular, latent diffusion models for T2I generation have garnered high interest because of the efficient modeling with fewer modules.

Recent work extends T2I models for text-to-video (T2V) generation. The two main challenges in T2V generation are the lack of high-quality text-video data at scale and the complexity of modeling the temporal dimension. There are two mainstream frameworks: 1) Transformer with Variational Auto Encoders (VAE); 2) diffusion models with U-Net. CogVideo [15] and Phenaki [34] are based on VAE and Transformer to learn T2V generation in the latent space. Make-A-Video [31] and Imagen Video [11] are based on diffusion models to learn video generation in the pixel space and have shown better performance than their Transformer+VAE counterparts. However, due to the complexity of video modeling, pixel-based T2V diffusion models must compromise to generate a low-resolution video first (64×64 in Make-A-Video and 40×24 in Imagen Video), followed by a sequence of super-resolution and frame interpolation models (see Tab. 4 for details). This makes the entire pipeline complicated and computationally expensive.

A generative AI system’s efficiency is essential because it impacts the user experience when interacting with these tools. Additionally, a simpler model architecture aids further research and development on top of it. In this paper, we propose Latent-Shift, which is an efficient model that can generate a two seconds video clip with 256×256 resolution without additional super-resolution or frame interpolation models.

Our work builds on the T2I latent diffusion model. When expanding the U-Net from T2I to T2V generation, we carefully choose not to increase the model complexity. Unlike prior work that adds additional temporal convolutional layers [31] and/or temporal attention layers [14, 31, 45] to expand the U-Net for temporal modeling, we use a parameter-free temporal shift module as motivated from [19, 22]. During training, we shift a few channels of the spatial U-Net feature maps forward and backward along the temporal dimension. This allows the shifted features of the current frame to observe the features from the previous and the subsequent frames and thus help to learn temporal coherence.

We show in our experiments that Latent-Shift achieves better performance than latent video diffusion models with temporal attention while having fewer parameters and thus being more efficient.

In summary, our main contributions are three-fold:

- We propose a novel temporal shift module to leverage a T2I model as-is for T2V generation without adding any new parameters.
- We show that our Latent-Shift model finetuned for video generation can also be used for T2I generation, which is a unique capability of the parameter-free temporal shift module.
- We demonstrate the effectiveness and efficiency of Latent-Shift through extensive evaluations on MSR-VTT, UCF-101, and a user study.

2. Related Work

Text-to-Image Generation. Early work in T2I generation [40, 44] are focused on GAN-based [25] extensions to generate images in simple domains like flowers [25], birds [35], etc. Recent work leverage better modeling techniques like Transformer with VAE or diffusion models to enable zero-shot T2I generation with compelling results. For example, CogView [3, 9], DALLE [28], and Parti [42] train an auto-regressive Transformer on large-scale text-image pairs for T2I generation. Make-A-Scene [7] additionally adds a scene control to allow more creative expression. On the other hand, GLIDE [23], DALLE2 [27], and Imagen [29] leverage diffusion models and achieve impressive image generation results. These diffusion-based models are trained on the pixel space and require additionally trained super-resolution models to achieve a high resolution. Latent diffusion can generate high-resolution images directly by learning a diffusion model in the latent space to reduce the computational cost. We extend the latent diffusion model for T2V generation.

Text-to-Video Generation. Similar to the evolution in T2I generation, early T2V generation methods [18, 20, 26] are based on GAN and applied to constrained domains like moving digits or simple human actions. Due to the challenges in modeling video data and a need for large-scale, high-quality text-video datasets, the priors of T2I in both modeling and data are leveraged for T2V generation. For example, NÜWA [37] formulates a unified representation space for image and video to conduct multitask learning for T2I and T2V generation. CogVideo [15] adds temporal attention layers to the pretrained and frozen CogView2 [4] to learn the motion. Make-A-Video [31] proposes to finetune from a pretrained DALLE2 [27] to learn the motion from video data alone, enabling T2V generation without training on text-video pairs. Video Diffusion Models [14] and

Imagen Video [11] perform joint text-image and text-video training by considering images as independent frames and disabling the temporal layers in the U-Net. Phenaki [34] also conducts joint T2I and T2V training in the Transformer model by considering an image as a frozen video.

While the advance in video generation is exciting, the entire pipeline for video generation can be very complex. As shown in Tab. 4, Make-A-Video has 6 models to generate a high-resolution video, and Imagen Video has 8 models, as a result of learning video generation in the pixel space.

Latent Diffusion for Video Generation. To reduce the complexity of video generation, latent-based models are explored [5, 10, 15, 34, 37, 38, 45]. Here we focus the discussions on latent diffusion models. Tune-A-Video [38] fine-tunes a pretrained T2I model on a single video to enable one-shot video generation with the same action as the training video. Esser *et al.* [5] leverage monocular depth estimations and content representation to learn the reversion of the diffusion process in the latent space for video editing. The work that is most similar to ours is MagicVideo [45], where the authors use a T2I U-Net with a frame-wise adaptor and a directed temporal attention module for T2V generation. However, similar to [11, 14, 31], these approaches need to use additional parameters to model the temporal dimension in videos. Our work can leverage the T2I U-Net as is for video generation, which is more efficient and enables both T2I and T2V generation in one unified framework.

3. Method

This section introduces our Latent-Shift that extends a latent diffusion model (LDM) from T2I generation to T2V generation through the temporal shift module. In Sec. 3.1, we introduce the background of the LDM for T2I generation. Sec. 3.2 shows the mechanism, rationale, and effects of the temporal shift module. Sec. 3.3 gives an overview of the proposed Latent-Shift for T2V generation.

3.1. Latent Image Diffusion Models

There are two training stages in the latent image diffusion models: 1) an autoencoder is trained to compress images into compact latent representations; 2) a diffusion model based on the U-Net architecture is trained on text-image pairs to learn T2I generation in the latent space.

Latent Representation Learning. The latent space is learned by an autoencoder that consists of an encoder and a decoder. Given an RGB image $\mathbf{x} \in \mathbb{R}^{H \times W \times 3}$, the encoder \mathcal{E} first compresses \mathbf{x} into a latent representation $\mathbf{z} = \mathcal{E}(\mathbf{x}) \in \mathbb{R}^{h \times w \times c}$ and then the decoder \mathcal{D} reconstructs the image $\tilde{\mathbf{x}} = \mathcal{D}(\mathbf{z}) \in \mathbb{R}^{H \times W \times 3}$ from \mathbf{z} , where H and W represent the image height and width in the pixel space, h , w and c represent the height, width, and channel size of the feature maps in the latent space. The encoder downsamples the image by a factor of $f = H/h = W/w = 2^m$ with

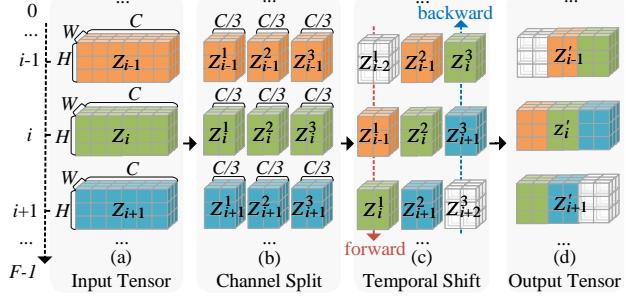


Figure 2. An illustration of temporal shift. C , H , W , F represent channel, height, width, and frame, respectively.

$m \in \mathbb{N}$. VQGAN [6] and VAE [17] are two widely used architectures of the autoencoder. We use a pretrained VAE model in our work.

Conditional Latent Diffusion Models. Diffusion models [12, 24] are generative models that are learned to recursively denoise from a normal distribution to a data distribution. There are different ways to parameterize the model. It can be trained by adding noise to the data and estimating the noise at different time steps. Specifically, given an image \mathbf{x} that is encoded to the latent space \mathbf{z} , we add Gaussian noise into \mathbf{z} defined as:

$$\mathbf{z}_t = \alpha_t \mathbf{z}_0 + \sigma_t \epsilon, \epsilon \sim \mathcal{N}(0, 1), \quad (1)$$

where α_t and σ_t are functions of t following the definition in [14] that control the noise schedule, t is the diffusion step that is uniformly sampled from $\{1, \dots, T\}$ during training where T is the total number of time steps. $\mathbf{z}_0 = \mathbf{z}$ is the original latent space before adding noise.

The T2I LDM is trained on text-image pairs (\mathbf{x}, \mathbf{y}) . The text \mathbf{y} is encoded through a text encoder \mathcal{C} to a representation $\mathcal{C}(\mathbf{y})$, which is mapped to the U-Net’s spatial attention layers through the cross attention scheme. The conditional latent diffusion model is trained to estimate the noise ϵ given a noisy input and conditioned on the text representation. A mean squared error loss is used:

$$\mathcal{L}_{img} = \mathbb{E}_{\mathcal{E}(\mathbf{x}), \mathbf{y}, \epsilon, t} \left[\|\epsilon - \epsilon_\theta(\mathbf{z}_t, t, \mathcal{C}(\mathbf{y}))\|_2^2 \right], \quad (2)$$

3.2. Temporal Shift

Motivated by [19, 22], the temporal shift operation can leverage a 2D spatial network to handle both spatial and temporal information by mingling the information from neighboring frames with the current frame. Figure 2 illustrates the temporal shift operation with an example of three frames. Let $Z \in \mathbb{R}^{C \times F \times H \times W}$ be the input to a temporal shift module, where $Z_i \in \mathbb{R}^{C \times H \times W}$ is the feature map for the i^{th} frame. We first split each Z_i into Z_i^1 , Z_i^2 , and Z_i^3 along the channel dimension C , where $Z_i^j \in \mathbb{R}^{\frac{C}{3} \times H \times W}$.

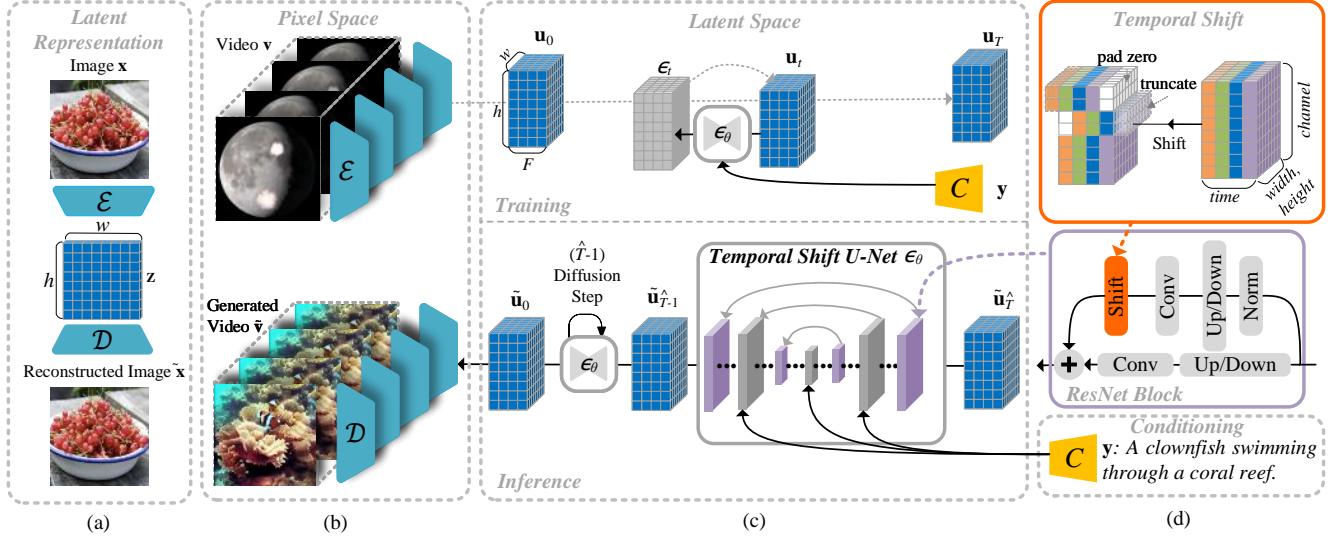


Figure 3. An illustration of our framework. From left to right: (a) An autoencoder is trained on images to learn latent representation. (b) The pretrained autoencoder is adapted to encode and decode video frames independently. (c) During training, a temporal shift U-Net ϵ_θ learns to denoise latent video representation at a uniformly sampled diffusion step $t \in [1, T]$. During inference, the U-Net gradually denoises from a normal distribution from step $\hat{T} - 1$ to 0 where \hat{T} is the number of resampled diffusion steps in inference. (d) The U-Net ϵ_θ is composed of two key building blocks: the 2D ResNet blocks with convolutional layers, highlighted in **violet**, and the transformer blocks with spatial attention layers, colored in **gray**. The temporal shift module, highlighted in **red**, shifts the feature maps along the temporal dimension. It is inserted into the residual branch of each 2D ResNet block. The text condition is applied to the transformer blocks via cross-attention. The channel dimension c in the latent space representation of \mathbf{z} and \mathbf{u} are omitted for clarity.

Then we shift Z_i^1 forward and Z_i^3 backward along the temporal (frame) dimension F . Finally, we merge the temporally-shifted features together. The output of the temporal shift module for the i^{th} frame is:

$$Z'_i = \begin{cases} [\mathbf{0}, Z_0^2, Z_1^3] & i = 0, \\ [Z_{i-1}^1, Z_i^2, Z_{i+1}^3] & 0 < i < F - 1, \\ [Z_{F-2}^1, Z_{F-1}^2, \mathbf{0}] & i = F - 1. \end{cases} \quad (3)$$

Here $\mathbf{0}$ denotes zero-padded feature maps.

The temporal shift module enables each frame's feature Z_i to contain the channels of the adjacent frames Z_{i-1} and Z_{i+1} and thus enlarge the temporal receptive field by 2. The 2D convolutions after the temporal shift, which operate independently on each frame, can capture and model both the spatial and temporal information as if running an additional 1D convolution with a kernel size of 3 along the temporal dimension [19].

3.3. Latent-Shift for T2V Generation

We adopt a pretrained autoencoder and a U-Net latent diffusion model. The autoencoder is fixed to encode and decode videos independently for each frame. We finetune the U-Net with the added temporal shift modules to enable video modeling for T2V generation.

The pretrained U-Net comprises two key building blocks: 1) 2D ResNet blocks that consist of mainly convolutional layers, and 2) spatial transformer blocks that mainly

include attention layers; both are designed only to model the spatial relationships. It is essential to enable the U-Net to model temporal information between video frames to learn meaningful motion. One straightforward direction is to add additional layers, as widely used in prior work. For example, VDM [14] and Magic Video [45] add a temporal attention layer after each spatial attention layer. Make-A-Video [31] adds 1D convolutional layers in the ResNet blocks and temporal attention layers in the transformer blocks. While it is intuitive to add new layers to extend the U-Net from modeling images to videos, we explore ways to use the U-Net as is for video generation.

To this end, we propose to incorporate the aforementioned temporal shift modules into the U-Net for T2V generation. Our framework is illustrated in Fig. 3. Specifically, we insert a temporal shift module inside the residual branch, which shifts the feature maps along the temporal dimension with zero padding and truncation, as shown in Fig. 3 (d).

Given a video $\mathbf{v} \in \mathbb{R}^{F \times H \times W \times 3}$, where F represents the number of frames, we use the pretrained encoder \mathcal{E} to encode each frame to get the latent video representation as $\mathbf{u} \in \mathbb{R}^{F \times h \times w \times c}$. The diffusion model is learned on this encoded latent space. We use a pretrained text encoder and add the text representation the same way in the attention layers as a condition. Similar to prior work, we also use classifier-free guidance [13, 23] to improve sample fidelity

Table 1. Zero-Shot T2V generation comparison on MSR-VTT. The results of CogVideo are cited from [31]. The best and second-best results are marked in bold and underlined, respectively.

Method	Zero-Shot	FID \downarrow	CLIPSIM \uparrow
GODIVA [36]	No	-	0.2402
NÜWA [37]	No	47.68	0.2439
CogVideo [15]	Yes	23.59	0.2631
Make-A-Video [31]	Yes	13.17	0.3049
Latent-VDM	Yes	<u>14.25</u>	0.2756
Latent-Shift (ours)	Yes	15.23	<u>0.2773</u>

to the input text. This is enabled by randomly dropping out the text input with a certain probability to learn unconditional video denoising during training. Similar to the diffusion process for image generation, we add noise ϵ into \mathbf{u} at each training time step as:

$$\mathbf{u}_t = \alpha_t \mathbf{u}_0 + \sigma_t \epsilon, \epsilon \sim \mathcal{N}(0, 1), \quad (4)$$

where α_t , σ_t , t are defined the same way as in Eqn. 1. $\mathbf{u}_0 = \mathbf{u}$ is the initial latent video representation before adding noise. The training objective is to estimate the added noise from the noisy input, which is defined as:

$$\mathcal{L}_{vid} = \mathbb{E}_{\mathcal{E}(\mathbf{x}), \mathbf{y}, \epsilon, t} \left[\|\epsilon - \epsilon_\theta(\mathbf{u}_t, t, \mathcal{C}(\mathbf{y}))\|_2^2 \right], \quad (5)$$

where θ denotes the learnable parameters from the pre-trained T2I U-Net model and \mathcal{C} is the pretrained text encoder which is fixed during training.

Diffusion models are typically trained on a large number of discrete time steps (*e.g.*, 1000) but can be used to sample data with fewer time steps to improve efficiency during inference. We use the DDPM sampler [12] with classifier-free guidance [13] and conduct sampling with $\hat{T} = 100$ steps.

4. Experiments

4.1. Implementation Details

Training is conducted on the WebVid [1] dataset with 10M text-video pairs. Following prior work, we report results on UCF-101 [32], MSR-VTT [39] with commonly used metrics including Inception Score (IS), Fréchet Image Distance (FID), Fréchet Video Distance (FVD), and CLIP similarity (CLIPSIM) between the generated video frames and the text. In addition, we conduct a user study comparing to CogVideo [15] via video quality and text-video faithfulness metrics. For all evaluations, we generate a random sample for each text without any automatic ranking. More details on hyperparameter settings are available in the supplementary materials.

Table 2. T2V generation comparison on UCF-101.

Method	IS \uparrow	FVD \downarrow
VideoGPT [41]	24.69 ± 0.3	-
TGANv2 [30]	26.60 ± 0.47	-
DIGAN [43]	32.70 ± 0.35	577 ± 22
DVD-GAN [2]	32.97 ± 1.7	-
MoCoGAN-HD [33]	33.95 ± 0.25	700 ± 24
CogVideo [15]	50.46	626
VDM [14]	57.80 ± 1.3	-
TATS-base [8]	79.28 ± 0.38	278 ± 11
Make-A-Video [31]	82.55	81.25
Latent-VDM	<u>90.74</u>	358.34
Latent-Shift (ours)	92.72	360.04

Table 3. User study results. The numbers show the percentages of raters who prefer our Latent-Shift or Latent-VDM over CogVideo [15].

Method	Quality (%)	Faithfulness (%)
Latent-VDM	56.4	53.8
Latent-Shift (ours)	58.0	62.9

4.2. Main Results

Evaluation on MSR-VTT. We conduct a zero-shot evaluation on the MSR-VTT test set. Following prior works [37], we use all the captions in the test set and calculate frame-level metrics. We compare Latent-Shift with prior works that are evaluated on MSR-VTT. In addition, we also implement a baseline of latent video diffusion model with the widely used temporal attention, termed as Latent-VDM. Both Latent-Shift and Latent-VDM are trained with the same setting. The results are shown in Tab. 1. The performance of Latent-Shift is competitive with prior works. In most cases, it already outperforms several methods with noticeable margins. Even though Latent-Shift does not outperform Make-A-Video due to our limited model size (see Tab. 4), the performance is much closer than other models.

Evaluation on UCF-101. We evaluate the performance on UCF-101 by finetuning on the dataset. The UCF-101 dataset consists of 13,320 videos* from 101 human action labels. We construct templated sentences for each class to form a text prompt. Then we finetune our pretrained T2V model to fit the UCF-101 data distribution. During inference, we perform class-conditional sampling to generate videos with the same class distribution as the training set for evaluation, following [31]. As shown in Tab. 2, our approach achieves state-of-the-art results on IS and a competitive score on FVD.

*Following prior work, we train on all the samples from both train and test splits, and evaluate on the train split.

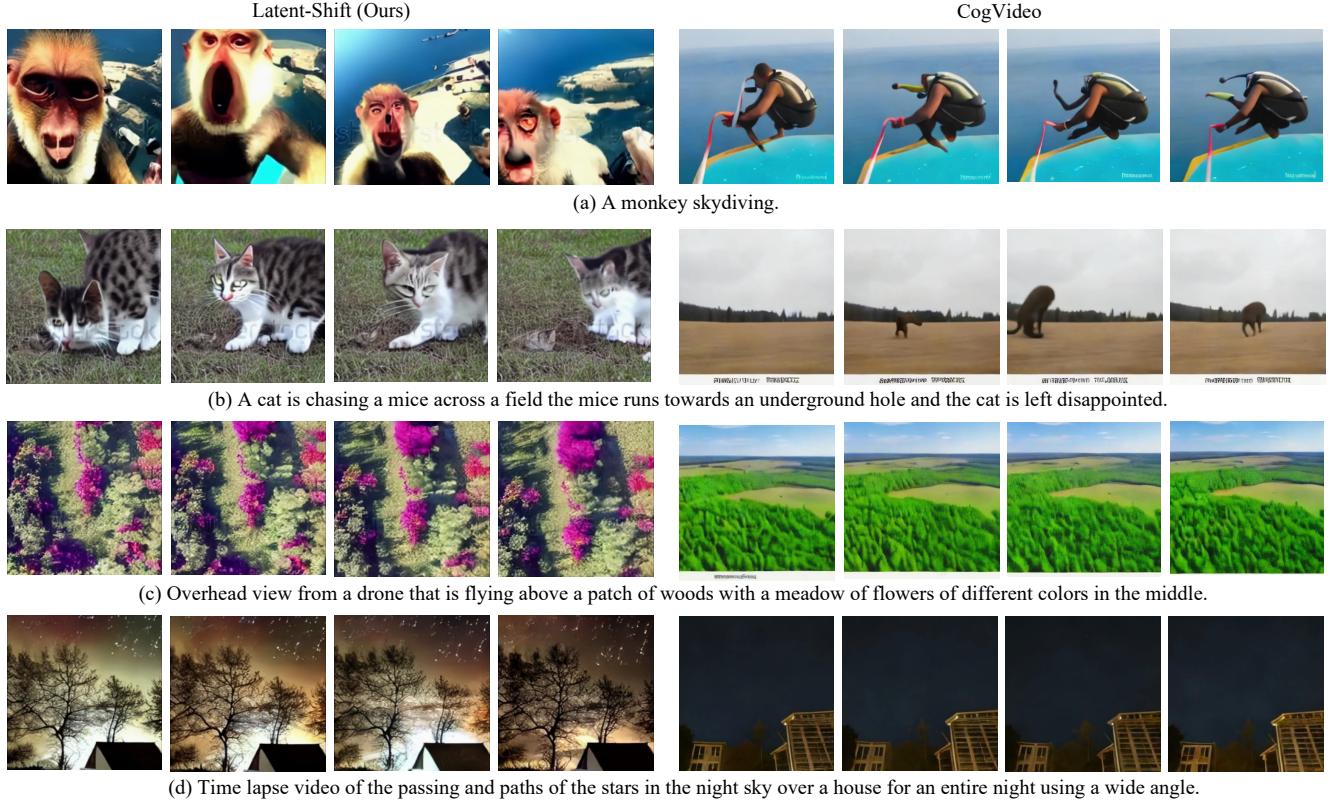


Figure 4. Text-to-Video generation comparison with CogVideo [15] on the user study evaluation set. Our model can generate more semantically correct content with meaningful motions.

Table 4. Model size and inference speed comparisons. The speed is measured in seconds on one A100 (80GB) GPU. (For CogVideo, there are 6B parameters that are shared among the T2V model and the frame interpolation model.)

Method	Parameters (Billion)							Speed (s)
	T2V Core	Auto Encoder	Text Encoder	Prior Model	Super Resolution	Frame Interpolation	Overall	
CogVideo [15]	7.7	0.10	—	—	—	7.7	15.5	434.53
Make-A-Video [31]	3.1	—	0.12	1.3	1.4 + 0.7	3.1	9.72	—
Imagen Video [11]	5.6	—	4.6	—	1.2 + 1.4 + 0.34	1.7 + 0.78 + 0.63	16.25	—
Latent-VDM	0.92	0.08	0.58	—	—	—	<u>1.58</u>	<u>28.62</u>
Latent-Shift (ours)	0.87	0.08	0.58	—	—	—	1.53	23.40

User Study. It is well known that automatic evaluation metrics are far from perfect. Therefore, it is more desirable to conduct user studies. To this end, we use the evaluation set from [31] that consists of 300 text prompt collected from Amazon Mechanical Turk (AMT). We compare to CogVideo [15] and evaluate both video quality and text-video faithfulness. The user study is conducted on AMT where 5 different raters evaluate each comparison and the majority vote is taken.

The results are shown in Tab. 3. Our approach achieves better results in both video quality and text-video faithfulness compared to CogVideo. This is consistent with the au-

Table 5. Evaluation on MSR-VTT in the zero-shot setting. We use LDM and our model to generate images and treated them as frozen videos for this comparison.

Method	FID \downarrow	CLIPSIM \uparrow
LDM	15.36	0.2910
Latent-Shift (T2I)	15.64	0.2737
Latent-Shift	15.23	0.2773

tomatic evaluations. As shown in Tab. 4, our model is also much more efficient than CogVideo.

Model Size and Inference Speed. We compare the model



Figure 5. Text-to-video generation comparison with CogVideo [15] on the UCF101 [32] dataset.

size and inference speed in Tab. 4. Only Cogvideo is chosen for speed comparison since it is the only open-sourced zero-shot T2V model. Latent-Shift is much smaller than prior works and much faster than CogVideo. Without a large number of parameters, Latent-Shift achieves better results than CogVideo in various benchmarks. This validates the effectiveness of Latent-Shift.

4.3. Ablation Study

Temporal Shift v.s. Temporal Attention. We compare our temporal shift module (Latent-Shift) with the widely used temporal attention layers (Latent-VDM) in the U-Net extension from image to video modeling. As already shown in Tabs. 1, 2, 3, Latent-Shift performs better than Latent-VDM in most cases, especially with a large margin in the user study. Furthermore, Latent-Shift requires fewer model parameters and thus enables relatively faster inference than Latent-VDM, as shown in Tab. 4.

Image Generation as a Frozen Video. Our finetuned Latent-Shift can be used for both image and video generation where an image can be considered as a frozen video with a single frame. In Tab. 5, we compare Latent-Shift with LDM on MSR-VTT. We observe that after training Latent-Shift for T2V generation, it can still perform reasonable T2I

generation. However, this also suggests that the metrics on MSR-VTT evaluation is not ideal as they do not account for the motion information in the videos. A better metric for the automatic evaluation of zero-shot T2V generation is needed.

4.4. Qualitative Results

T2V Generation. We show visual comparisons with CogVideo in Figs. 4 and 5 for different evaluation sets.

In both cases, Latent-Shift can generate semantically richer content with a meaningful motion that is faithful to the input text. This validates the effectiveness of our approach.

T2I and T2V Generation without Temporal Shift. The temporal shift module is parameter-free, *i.e.*, the U-Net for T2V generation is with the same parameters as the T2I model that it is initialized from. In more detail, for Latent-VDM, all context information from all frames is always available for each individual frame during training. Therefore, the model collapses during T2I inference with a single frame due to the lack of context information from the missing frames and the relative position inputs (column 4). We also tried to remove the temporal attention layer during inference, but it does not help to enable T2I generation. In

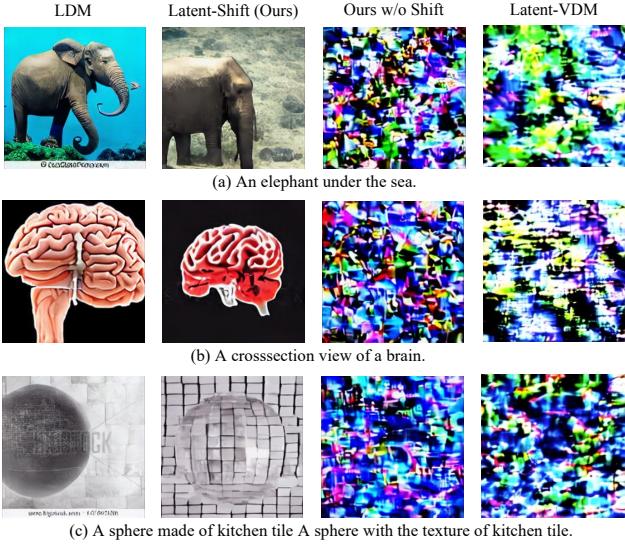


Figure 6. T2I generation comparison. Latent-Shift can generate meaningful images although it is finetuned for video generation but the latent-VDM with temporal attention cannot. Our method fails to generate images if the temporal shift module is removed. This demonstrates the shift module’s importance even though adding temporal shift on a single image means dropping a part of the feature maps.



Figure 7. T2V generation with and without temporal shift during inference. Our model would not generate meaningful videos if the temporal shift module is disabled.

contrast, in training Latent-Shift, the succeeding convolutional layers of the temporal shift module learn local context only from the previous and the next frames (all convolutional kernels are set to 3). Meanwhile, the padded zeros in the first frame and the last frame enable the kernels to learn generation with missing context during training. Therefore, when we use Latent-Shift for T2I generation, it can still generate reasonable images (column 2). Noted that the temporal shift module is necessary for T2I generation and cannot be removed, since the padded zeros indicate whether the context information is missing or not (Column 2 vs Column 3).

Similarly, we perform T2V generation with and without the temporal shift module. As shown in Fig. 7, the video



Figure 8. Failure cases of our approach. (a) Content distortion when generating videos with complex objects and action compositions. (b) Partially aligned text where not everything mentioned in the text can be generated. (c) Limited motion when the action is very subtle in the input text.

generation fails if not adding the temporal shift module.

Failure Cases. Latent-Shift works well for most text inputs but can struggle with some. We have observed three main types of failure cases (which are common), as shown in Fig. 8. First, there might be artifacts of object distortion and frame flickering. It happens when the text contains mixed concepts that are not commonly seen in the real world ((a)). It could be limited by the scale of the text-video training data and the fact that the VAE model is trained on images only. To learn the latent space from both image and video patches as [34] has the potential to alleviate this issue. Second, Latent-Shift may not always generate videos that match the text exactly. There may be missing contents ((b)). Third, some generated videos will have limited motion. This is a common issue of T2V generation methods [14, 31, 45], which often happens when the action is subtle in the text ((c)).

5. Conclusion

In this paper, we present Latent-Shift, a simple and efficient framework for T2V generation. We finetune a pre-trained T2I model with temporal shift on video-text pairs. The temporal shift module can model temporal information without adding any new parameters. Our model also preserves the T2I generation capability even though it is finetuned on videos, which is a unique property compared to many existing methods. The experimental results on MSRVTT and UCF101, along with user studies, demonstrate the effectiveness and efficiency of our approach.

References

- [1] Max Bain, Arsha Nagrani, GÜl Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021. [5](#)
- [2] Aidan Clark, Jeff Donahue, and Karen Simonyan. Adversarial video generation on complex datasets. *arXiv preprint arXiv:1907.06571*, 2019. [5](#)
- [3] Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, and Jie Tang. Cogview: Mastering text-to-image generation via transformers. In *NeurIPS*, 2021. [2](#)
- [4] Ming Ding, Wendi Zheng, Wenyi Hong, and Jie Tang. Cogview2: Faster and better text-to-image generation via hierarchical transformers. In *NeurIPS*, 2022. [2](#)
- [5] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. *arXiv preprint arXiv:2302.03011*, 2023. [3](#)
- [6] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *CVPR*, 2021. [3](#)
- [7] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. In *ECCV*, 2022. [2](#)
- [8] 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. [5](#)
- [9] Deeptha Girish, Vineeta Singh, and Anca Ralescu. Understanding action recognition in still images. In *CVPRW*, 2020. [2](#)
- [10] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity video generation with arbitrary lengths. *arXiv preprint arXiv:2211.13221*, 2022. [3](#)
- [11] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey A. Gritsenko, Diederik P. Kingma, Ben Poole, Mohammad Norouzi, David J. Fleet, and Tim Salimans. Imagen video: High definition video generation with diffusion models. *ArXiv*, abs/2210.02303, 2022. [2, 3, 6](#)
- [12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. [3, 5](#)
- [13] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS Workshop on Deep Generative Models and Downstream Applications*, 2021. [4, 5](#)
- [14] Jonathan Ho, Tim Salimans, Alexey A Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In *NeurIPS*, 2022. [2, 3, 4, 5, 8, 11](#)
- [15] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *ICLR*, 2023. [2, 3, 5, 6, 7, 11](#)
- [16] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. [11](#)
- [17] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014. [3](#)
- [18] Yitong Li, Martin Renqiang Min, Dinghan Shen, David E. Carlson, and Lawrence Carin. Video generation from text. In *AAAI*, 2018. [2](#)
- [19] Ji Lin, Chuang Gan, and Song Han. TSM: temporal shift module for efficient video understanding. In *ICCV*, 2019. [2, 3, 4](#)
- [20] Gaurav Mittal, Tanya Marwah, and Vineeth N. Balasubramanian. Sync-draw: Automatic video generation using deep recurrent attentive architectures. In *ACMMM*, 2017. [2](#)
- [21] Chong Mou, Xintao Wang, Liangbin Xie, Jian Zhang, Zhonggang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *arXiv preprint arXiv:2302.08453*, 2023. [2](#)
- [22] Andres Munoz, Mohammadreza Zolfaghari, Max Argus, and Thomas Brox. Temporal shift gan for large scale video generation. In *WACV*, 2021. [2, 3](#)
- [23] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *ICML*, 2022. [2, 4](#)
- [24] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *ICML*, 2021. [3](#)
- [25] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *ICVGIP*, 2008. [2](#)
- [26] Yingwei Pan, Zhaofan Qiu, Ting Yao, Houqiang Li, and Tao Mei. To create what you tell: Generating videos from captions. In *ACMMM*, 2017. [2](#)
- [27] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. [2](#)
- [28] 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. [2](#)
- [29] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *NeurIPS*, 2022. [2](#)
- [30] Masaki Saito, Shunta Saito, Masanori Koyama, and Sosuke Kobayashi. Train sparsely, generate densely: Memory-efficient unsupervised training of high-resolution temporal gan. *IJCV*, 2020. [5](#)
- [31] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. Make-a-video: Text-to-video generation without text-video data. In *ICLR*, 2023. [2, 3, 4, 5, 6, 8](#)
- [32] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. [5, 7](#)
- [33] 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. [5](#)

- [34] 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 descriptions. In *ICLR*, 2023. [2](#), [3](#), [8](#)
- [35] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. Caltech-ucsd birds 200. 2010. [2](#)
- [36] Chenfei Wu, Lun Huang, Qianxi Zhang, Binyang Li, Lei Ji, Fan Yang, Guillermo Sapiro, and Nan Duan. Godiva: Generating open-domain videos from natural descriptions. *arXiv preprint arXiv:2104.14806*, 2021. [5](#)
- [37] Chenfei Wu, Jian Liang, Lei Ji, Fan Yang, Yuejian Fang, Dixin Jiang, and Nan Duan. Nüwa: Visual synthesis pre-training for neural visual world creation. In *ECCV*, 2022. [2](#), [3](#), [5](#)
- [38] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. *arXiv preprint arXiv:2212.11565*, 2022. [3](#)
- [39] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. MSR-VTT: A large video description dataset for bridging video and language. In *CVPR*, 2016. [5](#)
- [40] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *CVPR*, 2018. [2](#)
- [41] Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*, 2021. [5](#)
- [42] 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 preprint arXiv:2206.10789*, 2022. [2](#)
- [43] 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](#)
- [44] Han Zhang, Tao Xu, and Hongsheng Li. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*, 2017. [2](#)
- [45] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models. *arXiv preprint arXiv:2211.11018*, 2022. [2](#), [3](#), [4](#), [8](#)

Latent-Shift: Latent Diffusion with Temporal Shift for Efficient Text-to-Video Generation

– Supplementary Material –

A. Hyperparameter Settings

For video data, we evenly sample 16 frames from a two seconds clip. We perform image resizing and center cropping to 256×256 . The latent space is $32 \times 32 \times 4$. To apply temporal shift on the feature maps of each frame, we keep $1/3$ channels from the previous frame, $1/3$ from the current frame, and $1/3$ from the subsequent frame. Adam [16] is used for optimization, the learning rate is set to 1×10^{-5} , the batch size is set to 256, the number of diffusion steps T is set to 1000, and bounds β_1 and β_T are set to 8.5×10^{-4} and 1.2×10^{-2} . During inference, the number of sampling steps \tilde{T} is set to 100, and the guidance scale s is set to 7.5. Table 6 shows the hyper-parameter settings of our models.

B. Text-to-Video Generation

In this section, we compare our proposed Latent-Shift with CogVideo [15] and VDM [14] qualitatively, as shown in Figure 9. We use the text prompts collected from VDM’s website [†]. Comparing all three methods, our generated videos contain richer content and thus with higher visual quality.

[†]<https://video-diffusion.github.io/>

Table 6. The hyper-parameter setting of our models. AE denotes the auto-encoder to encode and decode videos. Common, Latent-Shift and Latent-VDM indicate whether the hyper-parameter belongs to both models, Latent-Shift or Latent-VDM.

Hyper-parameter (common)	Value	Hyper-parameter (common)	Value
Image Size	256	Num Frame	16
Guidance Scale	7.5	Text Seq Length	77
Text Encoder	BERTEmbedder	First Stage Model	AutoencoderKL
AE Double z	True	AE z channel	4
AE Resolution	256	AE In Channel	3
AE Out Channel	3	AE Channel	128
AE Channel Multiplier	[1, 2, 4, 4]	AE Num ResBlock	2
AE Atten Resolution	[]	AE Dropout	0.0
Store EMA	True	EMA FP32	True
EMA Decay	0.9999	Diffusion In Channel	4
Diffusion Out Channel	4	Diffusion Channel	320
Conditioning Key	crossattn	Noise Schedule	quad
Encoder Channel	1280	Atten Resolution	[4, 2, 1]
Num ResBlock	2	Channel Multiplier	[1, 2, 4, 4]
Transformer Depth	1	Batch Size	4
Learn Sigma	False	Diffusion Step	1000
Timestep Respacing	100	Sampling FP16	False
Learning Rate	$1e^{-5}$	Sample Scheduler	DDPM
Num Head	8		
Hyper-parameter (Latent-Shift)	Value	Hyper-parameter (Latent-VDM)	Value
Attention Block Type	SpatialTransformer	Attention Block Type	SpatialTemporalTransformer
Shift Fold	3		

[Prompt:] *Grand Canyon Landscape North Rim.*



(a) Latent-Shift (Ours)



(b) CogVideo



(c) VDM

[Prompt:] *Berlin - Brandenburg Gate at night.*



(a) Latent-Shift (Ours)



(b) CogVideo



(c) VDM

Figure 9. Text-to-Video generation comparison with CogVideo and Video Diffusion Models (VDM).

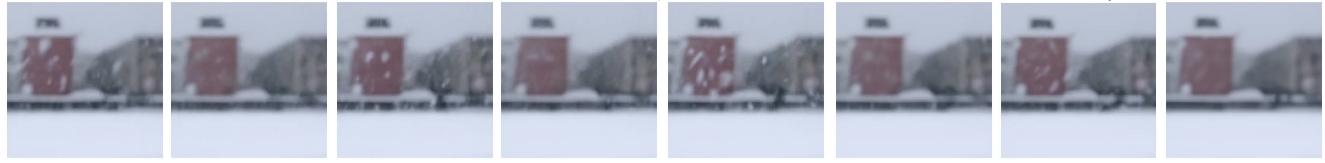
[Prompt:] snowfall in city.



(a) Latent-Shift (Ours)



(b) CogVideo



(c) VDM

[Prompt:] Traffic jam on 23 de Maio avenue both directions south of Sao Paulo.



(a) Latent-Shift (Ours)



(b) CogVideo



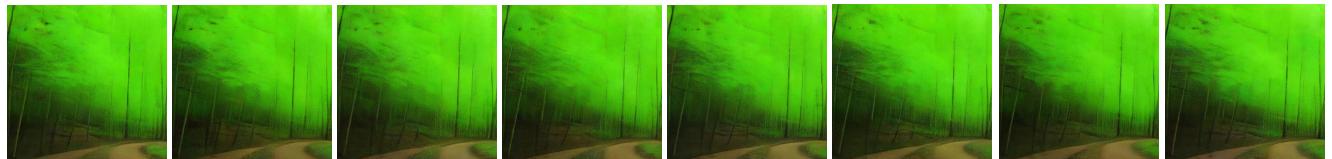
(c) VDM

Figure 9. Text-to-Video generation comparison with CogVideo and Video Diffusion Models (VDM) - continued.

[Prompt:] *path in a tropical forest.*



(a) Latent-Shift (Ours)



(b) CogVideo



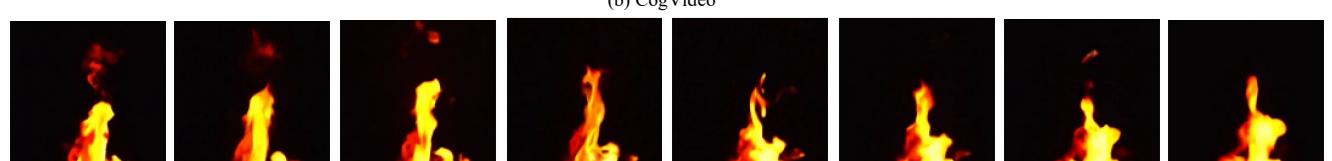
(c) VDM



(a) Latent-Shift (Ours)



(b) CogVideo



(c) VDM

Figure 9. Text-to-Video generation comparison with CogVideo and Video Diffusion Models (VDM) - continued.