Few-Shot Video Generation with Recurrent Motion Prediction using Pre-trained Image Diffusion Models

Ryan Li
Student ID: 06375418
lansong@stanford.edu

Churan He
Student ID: 06684508
churanhe@stanford.edu

Yingying Chen
Student ID: 06287845
ych@stanford.edu

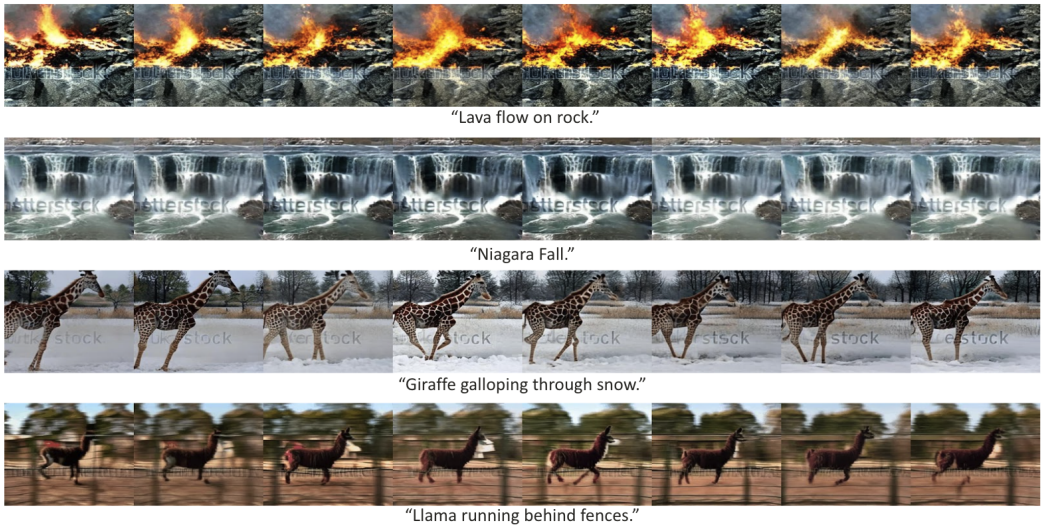

Figure 1: **Samples of our few-shot Text-to-Video model's results.** More output examples are available in Appendix A.

Abstract

This paper addresses the prevailing challenges in Text-to-Video (T2V) generation and introduces a novel approach for producing high-quality videos in computation-constrained environments. While Text-to-Image (T2I) generation has witnessed significant advancements, its extension to T2V generation remains a complex endeavor. Current T2V models often require extensive computational resources and large-scale text-video pair datasets for training. To mitigate these challenges, we explore leveraging pretrained T2I Diffusion models for T2V generation. Existing methods in this space, however, tend to exhibit limitations in generation capabilities and temporal consistency, often relying on external guidance, which restricts their practical applicability. Our study presents a novel solution that facilitates guidance-free video generation through few-shot finetuning, by integrating a recurrent causal-attention layer into Image Diffusion models for efficient motion learning, and employing structured noisy latent sampling for improved inference. The effectiveness of our approach is demonstrated through quantitative and qualitative evaluations, revealing its capability to generate high-quality videos with fluid motion sequences at a reduced computational cost. This advancement not only enriches the field of video generation with potential applications across diverse domains but also contributes to the democratization and efficiency

of T2V models. Our project's code is publicly available at this Github repository - https://github.com/RyanLi0802/CS236-Final-Project.

1 Introduction

Recent advances in generative models, particularly in diffusion models (12; 39; 33), have led to significant strides Text-to-Image (T2I) generation (25; 32; 35). These models have demonstrated remarkable capabilities in synthesizing high-quality, photorealistic images from textual descriptions. Extending these successes to Text-to-Video (T2V) generation, however, presents unique challenges. While diffusion models hold potential for T2V (3; 10), the process is often prohibitively costly in terms of computational resources and data requirements. Training T2V models to achieve comparable fidelity and coherence as their T2I counterparts requires substantial investment, rendering it an impractical endeavor for many research and application contexts.

To overcome such challenges, attempts have been made to leverage pretrained T2I Diffusion models for T2V generation in zero-shot (15), one-shot (43), and few-shot (44) manners. However, these methodologies often fall short in terms of generating videos with consistent quality and temporal coherence. Specifically, they struggle to maintain structural integrity and fluid motion across frames without extensive reliance on external guidance or conditioning on the first frame. This limitation not only hinders the flexibility and autonomy of the generative process but also impacts the practical utility of the generated videos, as they may lack the necessary realism and continuity expected in natural video sequences.

In this paper, we present a novel solution for guidance-free video generation with consistent movement predictions under the few-shot tuning paradigm. Figure 2 shows a high-level overview of few-shot T2V generation pipelines. During few-shot tuning, the model is provided with text-video pairs for the objective of reconstructing the original video based on the provided text prompt. At inference time, the model is provided with just the text prompt to generate a video as described in the prompt.

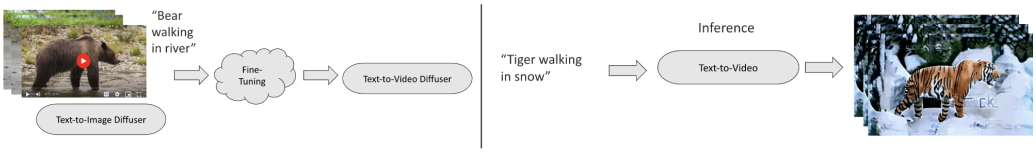

Figure 2: High level overview of few-shot Text-to-Video generation pipelines.

To improve the model's generation capability and outputs' temporal consistency, we propose several modifications to the T2I Diffusion model architecture. First of all, a recurrent causal attention layer is inserted into the attention mechanism of pre-trained T2I diffusion models in order to learn motion patterns. In this layer, future movements are predicted by attending over the noised latents of the first video frame and "history" vector that learns motion information from all previously generated frames, computed recursively in an RNN manner (36). Secondly, inspired by the shared-noise sampling proposed in LAMP (44), we look into noise generation algorithms so that instead of sampling randomly at each frame, the sampled noise would be structured in a way that potentially better represents a continuous motion across the frames. Details of our technical implementation are available in section 3.

The Frozen in Time dataset (2) is chosen for its richness and quality. We explore video datasets including the Frozen in Time dataset (2), the DAVIS dataset (29) and the DeepMind Kinetics (1). Videos from DAVIS are of high quality but the dataset suffers from very limited sample size. While the DeepMind Kinetics is rich in size and category, videos from this dataset requires extensive data pre-processing, which leads to an extra layer of complexities and prelonged training periods. Therefore, Frozen in Time was selected as the most suitable dataset for this project.

To evaluate our method, we select Tune-A-Video (43) as the baseline model, and compare the generated outputs under both quantitative and qualitative evaluation methods. For quantitative evaluation, adopted from EvalCrafter (21), we use 4 metrics: 1. Text-Video Consistency (CLIP-Score) that evaluates the similarity of semantic embeddings on each frame and the text prompt. 2.

Sementic Consistency (CLIP-Temp) (7; 30) that compares the similarity of semantic embeddings on each of the two frames of the generated videos. 3. Warping Error (18; 17) that measures the visual consistency from sequential frame transformations and optical flow. 4. User Study Ratings. For qualitative evaluation, we present the prompt texts and generated video frames from both the baseline and our enhanced model in figures, along with human evaluation and comparison on frame coherence and temporal consistency, especially on key subjects. Further details on model evaluation can be viewed in section 4.

2 Related Work

2.1 Text-to-Image Generation

Advancements in deep visual generation neural networks (e.g., VQ-GAN (8), diffusion (5; 33)), combined with large-scale multi-dimensional datasets (38; 20; 45; 22; 2; 26), have led to remarkable progremss in Text-to-Image (T2I) Generation (25; 32; 35). Diffusion models have shown remarkable effectiveness. DALL-E (32) overcomes the constraints in expressiveness and text-image alignment that Generative Adversarial Networks (GANs) face, by integrating a pre-trained generator, CLIP (31). GLIDE (25) takes this one step further by introducing classifier-free guidance, which not only allows for text-driven image editing but also for inpainting. Additionally, methods such as variational autoencoders (16), VQ-diffusion (9), and Latent Diffusion Models (LDMs) (33) transform the input into a latent space representation, aiming to create a more stable optimization process.

2.2 Text-to-Video Generation

Early endeavors in video generation were mainly limited to basic tasks, such as animating moving digits (27) or capturing specific human actions (4). The pioneering approach in Text-to-Video (T2V) generation, Sync-DRAW (23), is the first that employed a Variational Autoencoder (VAE) (16) coupled with recurrent attention. Furthermore, Vondrick et al. (40) have expanded the application of GANs from generating images to T2V generation.

The GODIVA model (41), a more recent innovation, stands out as the first to utilize a 2D Vector Quantized-Variational Autoencoder (VQ-VAE) combined with sparse attention mechanisms for T2V generation, facilitating the creation of more lifelike scenes. Building on this, NUWA (42) introduces a versatile representation for various generative tasks by adopting a multi-task learning approach. Further enhancing T2V generation, CogVideo (14) incorporates temporal attention modules into the existing T2I model, CogView2 (6).

In hope to replicate the recent success of T2I diffusion models, several recent works focused on Text-to-Video (T2V) generation by expanding diffusion-based models into the spatio-temporal domain (37; 11; 47; 10; 3). However, despite showing promising results, these models are computationally expensive and require training on millions of text-video pairs. The massive demand for labeled data, training hours, and computational resources is unaffordable for most researchers.

2.3 Few-Shot Video Editing and Generation

To bridge the aforementioned gap, attempts have been made to leverage pre-trained T2I diffusion models directly for video editing and video generation with few-shot tuning. Works such as Dreamix (24), GEN-1 (7), and LoveCon (19) focus on video editing by utilizing a video template and generating a modified video while keeping the original motion structure.

Another line of research involves generating videos directly from text in few-shot or zero-shot manner. T2V-Zero (15) proposes a modification to T2I diffusion models for zero-shot T2V Generation. However, it is challenging to transfer knowledge from spatial to spatial-temporal domain without any data samples, and the videos generated by T2V-Zero are often limited to repeating images with random motion. At the same time, Tune-A-Video (43) suggests a one-shot tuning method for video generation, however, their model relies on structural guidance from the original video, thus aligning it more with video editing rather than video generation. Besides, Tune-A-Video suffers from inconsistencies across frames in the model generated output videos. Most recently, LAMP (44) presents a few-shot-based tuning framework where pre-trained T2I diffusion models learn the pattern of a specific motion through a few selected examples, and generate new videos in the same motion

category. However, despite showing improvements from the previous methods, LAMP (44) seems to be limited to generating videos that are highly similar to the input training videos albeit small differences in styling or background. Our model exhibits high-level performance across all of our evaluation metrics.

3 Method

3.1 Adapting Image Diffusion Models to Video Generation

Contemporary T2I models (25; 33) primarily operate on 2D images. In these models, a typical architecture involves a UNet (34), which consists of a spatial downsampling block, followed by an upsampling block with residual connections. The UNet is built using a series of 2D convolutional residual blocks and transformer blocks. To adapt these T2I models for video generation, we first need to transform the 2D convolutions into pseudo 3D convolutions by inflating the 3x3 convolution filters to 1x3x3 filters with the first axis indexing video frames (13). Consequently, each 3D convolutional layer processes inputs of dimension $F \times H \times W \times C$, where F represents the number of frames, H and W denote the height and width of each frame, respectively, and C stands for the number of channels.

3.2 Recurrent Causal Attention Mechanism

Furthermore, to extend the spatio-only T2I Diffusion Models into the spatio-temporal domain, we can adapt the spatio self-attention layers into spatio-temporal attention layers. Common options for spatio-temporal attention include full attention and causal attention. However, such mechanisms are computationally burdening and does not scale when the number of frames increases. Tune-A-Video (43) proposes a sparse spatio-temporal attention mechanism, where the current frame attends only to the first and former video frames, instead of attending over the full video or all preceding frames. Formally, under sparse spatio-temporal attention, query features are derived from frame z_{v_i}, key and value features from the first frame z_{v_1} and the former frame $z_{v_{i-1}}$, and $Attention(Q, K, V) = Softmax(\frac{QK^T}{\sqrt{d}}) \cdot V$ is defined with

$$Q = W^Q z_{v_i}, K = W^K [z_{v_1}, z_{v_{i-1}}], V = W^V [z_{v_1}, z_{v_{i-1}}]$$

While such an approach removes the prohibitive computational complexity of full and causal attention, the mechanism also ignores all information between the first and former frames. As the result, the learned model often fails generate a continuous sequence of motion, and the outputs are often limited to random variances of the same image. To better learn fluid motion patterns, we introduce a recurrent causal attention mechanism where the motion pattern of a video is learned and encoded in a RNN fashion. Formally, recurrent causal attention is defined as a cross attention between the noisy latents of the current frame (z_{v_i}) as the Query, and the noisy latents of the first frame (z_{v_1}) along with a hidden memory (h_i) as Key and Value. The hidden memory h_i is updated with the noise residuals generated from $z_{v_1}, z_{v_2}, ..., z_{v_{i-1}}$ autoregressively, as shown in figure 3. Recurrent weights W_{hh}, W_{zh}, W_{ho} are initialized at zero to avoid adding noise to performance of the pretrained model at the beginning of the training.

$$h_i = f(W_{hh} h_{i-1} + W_{zh} z_i + b_h)$$
$$Q = W^Q z_{v_i}, K = W^K [g(W_{ho} h_t + b_a)], V = W^V [g(W_{ho} h_t + b_a)]$$

The first frame encapsulates most of the graphical information of a short video, and h_i is trained to extract motion information from the noise residuals of all the past predict frames. With the recurrent causal attention mechanism, we open up the potential of further improvement on temporal consistency while keeping the size of the attention matrices and computation complexity constant regardless of the videos' length.

To leverage the most out of the pretrained weights and parameters, we fix most of the model's parameters and only update some of the projection matrices in attention layers and weight in recurrent neural network. Similar to methodology proposed by Wu et al (43), we freeze all weights and parameters except for $W^Q, W_{hh}, W_{zh}, and b_h$ in recurrent causal attention layer, W^Q in cross-attention layers and all projection matrices in temporal self-attention layers.

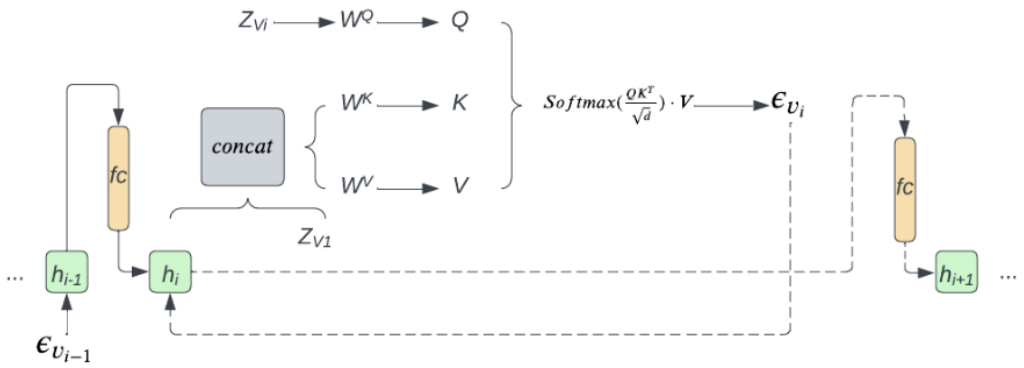

Figure 3: Illustration of Recurrent Causal Attention Mechanism

3.3 Structured Sampling from Noise

In Tune-A-Video (43), each frame is generated in the following way

$$\mathcal{V} = \mathcal{D}(\text{DDIM-samp}(\text{DDIM-inv}(\varepsilon(\mathcal{V})), \mathcal{T}^*)$$

With $\mathcal{D}$ being the decoder, ε being the encoder, and $\mathcal{T}^*$ being the text prompt for the output. Our aim is to generate output videos in a guidance-free manner, without the need for DDIM inverted latents. However, since the output frames are highly correlated with each other, having good structural heuristics for sampling noised latents could benefit the output quality. LAMP (44) introduced the concept of shared-noise sampling, where all noised latents share the same base noise. Formally, the noise latents for the ith frame is defined as

$$\epsilon^i = \alpha\epsilon^s + (1-\alpha)\epsilon^i$$

In this paper we propose another noise sampling method, autoregressive noise sampling, where the noise latents of each frame is dependent on the noise latents of the previous frame. Since the recurrent causal attention layer predicts the score of each output frame autoregressively, we would like to explore whether sampling the noise autoregressively would help generate better quality results. Formally, we define the autoregressive noise sampling procedure as:

$$\epsilon_0 \sim \mathcal{N}(0, \mathrm{I}), \epsilon_i' \sim \mathcal{N}(0, \mathrm{I})$$

$$\epsilon_i = \alpha * \epsilon_{i-1} + (1-\alpha) * \epsilon_i'$$

In this way, the noise used to generate the following frame will be based upon the noise used to generate the previous frame. This constraint can help to reduce the significant alternation from sampling purely randomized noise.

3.4 Few-shot Adaptation

The baseline model Tune-A-Video (43) is a one-shot video generation model. The input video will become the structural guidance of the generated video through DDIM inversion. Therefore, the generated video will have significant similarity compared to the original regarding the movement or the action of the subject. In this work, we eliminate the requirement for structural guidance by switching from one-shot to few-shot settings, where we feed our model multiple text-video pairs describing the same motion during finetuning. Our video generation model will extract and learn motion features from input videos, and apply such motion patterns to other objects and settings to generate custom output videos, instead of strictly following the inverse latents of a specific input video.

4 Results and Analysis

4.1 Experiment Setups

We evaluate our model against Tune-A-Video (43) as our baseline on videos retrieved from the Frozen in Time dataset (2). The models are benchmarked on 8 groups of videos, with each group consisting of 8-16 videos sharing the same theme and motion sequence. Specifically, our model is evaluated against the baseline on the following 8 themes: "animal walking", "human running", "birds flying", "fireworks", "helicopter flying", "horse running", "raining", and "waterfalls". We manually annotate the text prompt for each video.

The baseline model and its hyperparameters are set up exactly according to Wu et al's original paper and their github implementation, with 300 training steps, 3e-5 constant learning rate, 50 inference steps, and a guidance scale of 12.5. Since the baseline uses one-shot finetuning and requires structural guidance from DDIM Inversion, we select the first video of each theme to be the input video for training and for structural guidance.

We finetune our model on all 8-16 videos with the same theme for 16,000 steps. Since our pipeline requires substantially more training steps than the baseline, we schedule our learning using linear decay with an initial learning rate of 3e-5, and 1600 warm-up steps. We fix most of the pre-trained layers and only tune parameters in the added recurrent causal attention layer and the linear query layer of the cross attention blocks. Lastly, we initialize all parameters of the added recurrent causal attention layer to zero to avoid polluting the generation capability of the pretrained T2I model at the beginning of training, as done in LAMP (44) and ControlNet (46). A more in-depth analysis of the model's hyperparameters can be found in section 4.3.

All experiments are conducted on a single NVIDIA V100 GPU with 16GB vRAM.

4.2 Baseline Comparisons

4.2.1 Quantitative Results

Automatic Metrics We evaluate our model with the baseline quantitatively for text-video alignment and temporal consistency. The quantitative results contain CLIP-Score, CLIP-Temp(7; 30) , Warping Error(18; 17), and user studies.

CLIP-Score calculates the correlation between each frame and the input prompt by their embeddings (normalized to scale from 0 - 100);

$$\text{CLIP-Score} = \frac{1}{M}\sum_{i=1}^{M}(\frac{1}{N}\sum_{t=1}^{N}(\text{C}(emb(x^i{}_t), emb(p^i))))$$

CLIP-Temp calculates the consistency across frames (normalized to scale from 0 - 100);

$$\text{CLIP-Temp} = \frac{1}{M}\sum_{i=1}^{M}(\frac{1}{N-1}\sum_{t=1}^{N-1}(\text{C}(emb(x^i{}_t), emb(x^{i+1}{}_t))))$$

Finally, warping Error evaluates the graphical consistency (lower is better). As shown in table 1, across all three evaluation criteria, our model out-performs the baseline model by a non-trivial margin.

User Study To provide a comprehensive analysis of our model's output quality against the baseline's, we conduct a user study with 12 participants. Each participant is provided with 6 pairs of videos, one from baseline and one from our model, along with the text prompt used to generate the videos. Then, each participant is asked to rank each video's ***temporal consistency***, ***faithfulness to input text***, ***video fidelity***, and ***overall video quality*** on a scale of 5. As shown in table 1, we observe a significant improvement in temporal consistency from the baseline model. Our model performs on par with the baseline in faithfulness and fidelity, with a slightly lower rating on faithfulness due to overfitting issues in our training pipeline. Overall, our model receives a nontrivial higher preference score over the baseline model.

Evaluation Metrics	Tune-A-Video (Raw)	Tune-A-Video (Guided)	Our Model
CLIP-Score ↑	25.44	29.11	**31.29**
CLIP-Temp ↑	95.51	96.29	**98.00**
Warping Error ↓	33325.53	9956.31	**5756.69**
Fidelity ↑	–	3.2	**3.6**
Temp. Consistency ↑	–	3.6	**4.3**
Text Faithfulness ↓	–	**4.1**	4.0
Overall Preference ↑	–	3.6	**3.9**

Table 1: Quantitative evaluation of our model against the baselines. The first three rows represent automatic metrics, and the last four rows are obtained from user studies

4.2.2 Qualitative Results

A qualitative comparison between our model and the baseline can be shown in figure 4. The baseline model fails to generate any meaningful frames without the structural guidance from DDIM inverted latents. With structural guidance, the baseline model generates fairly realistic images, despite not following the prompt "in tall grass". However, the more important issue with the baseline output is that the generated frames do not form a continuous motion. Due to the limitations in its finetuning methodology, the generated outputs are more akin to random variances of the same images instead of a meaningful sequence of movements.

Our model, on the other hand, displays a smooth walking movement, where the tiger starts with its limb reaching forward, and then gradually straightens it back under its body as it propels forward, mimicking the natural and fluid motion of a walking tiger. Moreover, the baseline only works well with 8 output frames, and the video quality degrades when the number of output frames increases. However, with our model, we observe smooth motion sequences and solid temporal consistency even when the number of output frames scale up to 16 and 32.

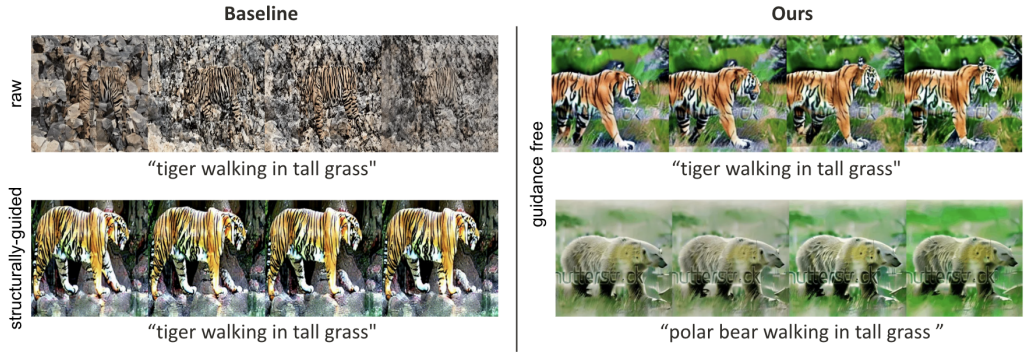

Figure 4: Qualitative comparison between the baseline output (left) and our model (right).

4.3 Model Performance Analysis

In this section, we conduct a in-depth analysis on a number of factors that impact our model's performance. First, we examine how the number of training steps impact the model's output. As shown in figure 5, with 600 training steps, the model fails to learn meaningful connections between the frames, and the outputs remain the same as the original image diffusion model. With 4800 training steps, the model can already output a sequence of related frames. However, the background is unstable and inconsistencies and still observed in the tiger's movement. Under 16000 training steps, the model is capable of constructing a fluid motion among the frames with a consistent background. We do not observe, on the other hand, any significant improvements from 16000 steps to 30000 steps. Instead, we sometimes even observe degradation in video quality under 30000 training steps, potentially due to model overfitting.

Next, we compare the effects of different noise sampling strategies. The hypothesis is that since the frames in the actual video are connected to each other, and DDIM provides a one-to-one mapping

Figure 5: Comparison among model's outputs under different training steps.

between the noise and data distributions, the noise latents used to generate the videos should also be structurally related. To explore this domain, we apply three different sampling strategies and compare each of their outputs: random noise sampling, shared noise sampling, and autoregressive noise sampling. Results are shown in figure 6. While all of these sampling strategies yield reasonable results, we do observe subtle differences among these strategies. Random sampling produces the most diverse though sometimes inconsistent samples. With shared noise sampling, the output frames tend to be highly identical unless the α value is small (e.g., 0.2). Autoregressive sampling produces a smooth transition in output frames, though the background and the overall image tend to be more blurry than the results generated from random sampling. We are not able to derive a definitive conclusion on the optimal noise sampling strategy based on our limited experiment samples. Further investigation in this space is left for future work.

Figure 6: Comparison between different noise sampling strategies.

Finally, we examine the generalizability of our finetuned model. Since the goal of our model is to learn a meaning motion sequence that can be applied to various settings, we probe the model's generalizability by modifying both the object (e.g., tiger) and the background (e.g., in tall grass), while keeping the same motion (e.g., walking). The results (figure 7) show that the learned model generalizes fairly well in terms of object, but fails to generalize the background. One possible explanation could be that the model has encountered much more training samples featuring objects than samples featuring background during its pretraining. Even though our finetuning dataset only contains a very limited set of samples, the pretrained layers are still able to generate a variety of objects that do not exist in the finetuning dataset (e.g., pandas, polar bear) as the model has encountered these objects many times during pretraining. However, since there aren't as many samples featuring backgrounds in the pretraining dataset, the pretrained layers' capability to generates various backgrounds is weaker, and our recurrent motion learning layer, which is heavily overfitted to our finetuning dataset, will take more effect. While the model can generate backgrounds it has seen during finetuning with decent quality, it fails to generalize to backgrounds outside the finetuning dataset. As one can see in the third example ("tiger walking on Mars"), while the first frame still

somewhat resembles the environment on Mars, the recurrent motion learning layer would tilt the score towards backgrounds it has seen during finetuning. As the error accumulates over the frames, the final frame is changed to a green-ish background, a common background in most of the finetuning data samples.

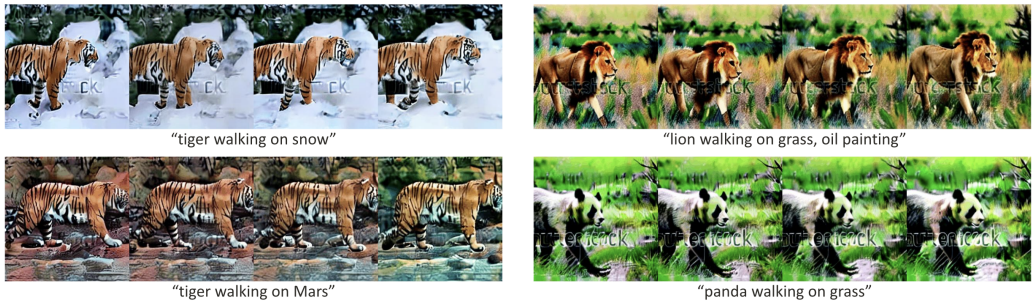

Figure 7: Our model's output under different text prompts finetuned on the same dataset

5 Conclusion

This work proposes a novel approach to few-shot video generation. By leveraging recurrent causal attention for motion learning, we are able to finetune pretrained T2I Diffusion Models on a small set of training data and generate photorealistic videos with fluid motion sequence without the reliance of structural guidance or first frame conditioning. The paper showcases the success of learning a sequence of movements from a small number of videos and applying the learned motion to a variety of objects. With all training implemented on a single V100 GPU in Google Colab, this work presents a significant improvement in the trade-off between training cost and generation freedom.

However, the current approach also has its own limitations. The most significant limitation is that the recurrent motion learning layer affects the model's generalizability and causes instabilities in the output's background. One way to address this problem is by disentangling image generation from motion prediction. Instead of training the recurrent motion learning layer on entire frames, we may instead choose to train the layer on residual images or optical flows. Moreover, due to the cumulative effects of recurrent networks, generation errors accumulates across the frames, leading to degradation in video quality as the number of frames increases. To generate longer videos, one might resort to splitting a long video into consecutive windows (19) or employ other conditioning or normalization techniques to improve output stability.

In this paper, the impact of different noise sampling strategies was explored, but failed to obtain conclusive results. Further investigation of structured noise sampling and more complex sampling techniques such as noised latent learning are left for future work.

Finally, the autoregressive nature of the recurrent causal attention mechanism makes both training and inference times longer. While its effect is not observable when the number of frames is small, the inefficiencies of such mechanisms could prevent the model from scaling up. Recent innovation to the RNN architecture (28) allows efficient parallelizable training while preserving constant computational and memory complexity at inference. Further optimizations to the network can be done to improve the model's training efficiency while maintaining good motion-learning capabilities.

In conclusion, our work marks a significant step toward guidance-free video generation that balances between high-quality output and computational efficiency. It showcases how video generation quality can be improved without demanding computational power by small innovations to the network architecture. We hope this research paves the way for future advancements in T2V generation, contributing to the democratization and efficiency of T2V models.

References

[1] Deepmind kinetics video dataset. `https://www.deepmind.com/open-source/kinetics`. Accessed: 2023-10-24.

[2] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval, 2022.

[3] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models, 2023.

[4] Haoye Cai, Chunyan Bai, Yu-Wing Tai, and Chi-Keung Tang. Deep video generation, prediction and completion of human action sequences. In *Computer Vision – ECCV 2018*, pages 374–390. Springer International Publishing, 2018.

[5] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021.

[6] Ming Ding, Wendi Zheng, Wenyi Hong, and Jie Tang. Cogview2: Faster and better text-to-image generation via hierarchical transformers, 2022.

[7] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models, 2023.

[8] Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis, 2021.

[9] 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, 2022.

[10] Yuwei Guo, Ceyuan Yang, Anyi Rao, Yaohui Wang, Yu Qiao, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning, 2023.

[11] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P. Kingma, Ben Poole, Mohammad Norouzi, David J. Fleet, and Tim Salimans. Imagen video: High definition video generation with diffusion models, 2022.

[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020.

[13] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J. Fleet. Video diffusion models, 2022.

[14] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers, 2022.

[15] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators, 2023.

[16] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022.

[17] Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, and Ming-Hsuan Yang. Learning blind video temporal consistency, 2018.

[18] Chenyang Lei, Yazhou Xing, and Qifeng Chen. Blind video temporal consistency via deep video prior, 2020.

[19] Zhenyi Liao and Zhijie Deng. Lovecon: Text-driven training-free long video editing with controlnet, 2023.

[20] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015.

[21] Yaofang Liu, Xiaodong Cun, Xuebo Liu, Xintao Wang, Yong Zhang, Haoxin Chen, Yang Liu, Tieyong Zeng, Raymond Chan, and Ying Shan. Evalcrafter: Benchmarking and evaluating large video generation models, 2023.

[22] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. HowTo100M: Learning a Text-Video Embedding by Watching Hundred Million Narrated Video Clips. In *ICCV*, 2019.

[23] Gaurav Mittal, Tanya Marwah, and Vineeth N. Balasubramanian. Sync-DRAW. In *Proceedings of the 25th ACM international conference on Multimedia*. ACM, oct 2017.

[24] Eyal Molad, Eliahu Horwitz, Dani Valevski, Alex Rav Acha, Yossi Matias, Yael Pritch, Yaniv Leviathan, and Yedid Hoshen. Dreamix: Video diffusion models are general video editors, 2023.

[25] 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, 2022.

[26] Andreea-Maria Oncescu, João F. Henriques, Yang Liu, Andrew Zisserman, and Samuel Albanie. Queryd: A video dataset with high-quality text and audio narrations, 2021.

[27] Yingwei Pan, Zhaofan Qiu, Ting Yao, Houqiang Li, and Tao Mei. To create what you tell: Generating videos from captions, 2018.

[28] Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Huanqi Cao, Xin Cheng, Michael Chung, Matteo Grella, Kranthi Kiran GV, Xuzheng He, Haowen Hou, Przemyslaw Kazienko, Jan Kocon, Jiaming Kong, Bartlomiej Koptyra, Hayden Lau, Krishna Sri Ipsit Mantri, Ferdinand Mom, Atsushi Saito, Xiangru Tang, Bolun Wang, Johan S. Wind, Stansilaw Wozniak, Ruichong Zhang, Zhenyuan Zhang, Qihang Zhao, Peng Zhou, Jian Zhu, and Rui-Jie Zhu. Rwkv: Reinventing rnns for the transformer era, 2023.

[29] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[30] Chenyang Qi, Xiaodong Cun, Yong Zhang, Chenyang Lei, Xintao Wang, Ying Shan, and Qifeng Chen. Fatezero: Fusing attentions for zero-shot text-based video editing, 2023.

[31] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.
[32] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022.
[33] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022.
[34] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.
[35] 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, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding, 2022.
[36] Alex Sherstinsky. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Physica D: Nonlinear Phenomena*, 404:132306, mar 2020.
[37] 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, 2022.
[38] Lucas Smaira, João Carreira, Eric Noland, Ellen Clancy, Amy Wu, and Andrew Zisserman. A short note on the kinetics-700-2020 human action dataset, 2020.
[39] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022.
[40] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics, 2016.
[41] 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, 2021.
[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, 2021.
[43] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Weixian Lei, Yuchao Gu, Yufei Shi, 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, 2023.
[44] Ruiqi Wu, Liangyu Chen, Tong Yang, Chunle Guo, Chongyi Li, and Xiangyu Zhang. Lamp: Learn a motion pattern for few-shot-based video generation, 2023.
[45] Jianhui Yu, Hao Zhu, Liming Jiang, Chen Change Loy, Weidong Cai, and Wayne Wu. Celebv-text: A large-scale facial text-video dataset, 2023.
[46] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023.
[47] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models, 2023.

A Additional Model Outputs

Figure 8: "Waterfall on Mars"

Figure 9: “Hippo running in prairie.”

Figure 10: “Waterfall in mountains.”

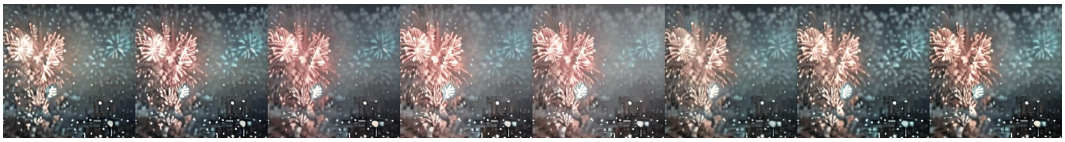

Figure 11: “Fireworks.”

Figure 12: “Bear running in the forest.”

Figure 13: “Birds flying over volcano.”

Figure 14: “Polar bear walking in snow.”

Figure 15: “Fire in forest.”