# Enhancing Scene Transition Awareness in Video Generation via Post-Training

**Anonymous ACL submission**

## Abstract

Recent advances in AI-generated video have shown strong performance on *text-to-video* tasks, particularly for short clips depicting a single scene. However, current models struggle to generate longer videos with coherent scene transitions, primarily because they cannot infer when a transition is needed from the prompt. Most open-source models are trained on datasets consisting of single-scene video clips, which limits their capacity to learn and respond to prompts requiring multiple scenes. Developing scene transition awareness is essential for multi-scene generation, as it allows models to identify and segment videos into distinct clips by accurately detecting transitions.

To address this, we propose the **Transition-Aware Video** (TAV) dataset, which consists of preprocessed video clips with multiple scene transitions. Our experiment shows that post-training on the **TAV** dataset improves prompt-based scene transition understanding, narrows the gap between required and generated scenes, and maintains image quality.

## 1 Introduction

The ability to generate visual content from natural language has rapidly improved in recent years, driven by the emergence of powerful generative models such as diffusion (e.g., Ho et al. (2020), Song et al. (2021), Rombach et al. (2022), OpenAI (2023)) and visual autoregressive model (e.g., van den Oord et al. (2016), Kalchbrenner et al. (2017), Chen et al. (2020), Chen et al. (2023a)). These methods have become central to modern text-to-image and text-to-video systems, enabling high-quality results from simple prompts and forming the foundation of advanced T2V models such as *Sora* (Brooks et al., 2024) and *Kling* (Kuaishou Technology, 2024).

We observe that existing video generation models perform well on short clips with a single scene, but often struggle to maintain quality and coherence in longer, story-level videos. Open-source models like *EasyAnimate* (Xu et al., 2024) and *CogVideo* (Hong et al., 2022), typically struggle to recognize the needs for scene transitions, i.e, they fail to generate the correct number of scenes as specified in the prompt, even when multiple distinct scenes are explicitly described. We evaluate the open-sourced models using 50 prompts that explicitly require the generation of two distinct scenes. As shown in Table 1, the average number of scenes generated is approximately one, supporting our conclusion about the models' limited ability to handle multi-scene prompts.

One possible reason is that widely used video-text datasets, such as *WebVid-10M* (Bain et al., 2022), *Panda-70M* (Chen et al., 2024b), and *MiraData* (Ju et al., 2024a), are largely composed of single-scene clips (over 90%), typically extracted using simple scene segmentation techniques. Thus, current models are rarely exposed to explicit scene transitions during training, which results in an out-of-distribution issue when a scene change is required at inference time. Given the strong generation capabilities of current models, we explore whether post-training them to recognize scene transitions in prompts can enhance overall performance, improving both coherence and visual quality.

| OpenSora | CogVideo | EasyAnimate |
|---|---|---|
| 1.12 | 1.48 | 1.22 |

Table 1: Average number of scenes generated by models given prompts that explicitly indicate two scenes.

Our contributions in this work includes:

- We design the **TAV** dataset to explicitly teach models how to handle scene transitions from prompt by post-training. The **TAV** dataset consists of pairs of 10-second video clips with scene transitions and their corresponding scene-wise descriptions. The clips are extracted from the *Panda-70M* dataset, and for each clip, a large language model (LLM) is used to generate separate descriptions for each individual scene.

- We conduct an experiment to compare the number of scenes generated by the original OpenSora model and the OpenSora model post-trained on the **TAV** dataset, using the same set of prompts. The results show that post-training with the **TAV** dataset increases the average number of scene, indicating improved understanding of scene transition requirements specified in the prompts. Notably, image quality remains unaffected, as measured by VBench (Huang et al., 2023).

## 2 Related Work

**Long video generation.** Increasing attention has been paid to generating long, story-driven videos in recent research. Early approaches leveraged GANs and VAEs to model video distributions, while models like VideoGPT (Yan et al., 2021) and TATS (Ge et al., 2022) introduced discrete latent spaces and transformer-based architectures for improved temporal coherence. Transformer-based methods such as Phenaki (Villegas et al., 2022) further extended video length by generating token sequences conditioned on textual input. More recently, diffusion models have emerged as a powerful framework. Methods like LEO (Wang et al., 2023b) and LVDM (He et al., 2022) leverage hierarchical or latent motion spaces to synthesize long videos with enhanced continuity. NUWA-XL (Yin et al., 2023) and GAIA-1 (Hu et al., 2023) adopt structured diffusion or world model approaches, while FreeNoise (Qiu et al., 2023) and Gen-L-Video (Wang et al., 2023a) extend generation by aggregating noise-sampled or overlapping segments. StreamingT2V (Henschel et al., 2024) proposes an autoregressive framework with memory mechanisms to maintain appearance consistency over time.

**Transition generation.** Scene transitions are essential for storytelling, enabling smooth shifts in time, space, or perspective. Traditional techniques such as fades, dissolves, wipes, and cuts are often implemented using predefined patterns, while morphing methods (Wolberg (1998), Shechtman et al. (2010)) allow for smoother transitions by estimating pixel-level correspondences. Generative approaches like latent-space interpolation (Van Den Oord et al., 2017) have been used to model semantic transitions, with applications in style transfer (Chen et al., 2018) and object transfiguration (Sauer et al. (2022),Kang et al. (2023)). Recent advances explore data-driven methods for generative scene transitions. Seine (Chen et al., 2023b) introduces a short-to-long video diffusion model focused on transitions and predictions. Loong (Wang et al., 2024) leverages autoregressive language models for minute-level multi-scene video generation, while VideoDirectorGPT (Lin et al., 2023) incorporates LLM-guided planning to ensure consistency across multiple scenes.

Naturally, incorporating specialized modules designed to improve consistency and capture scene transitions is essential for enhancing long video generation. From our perspective, evaluating the quality of the training dataset and identifying the most suitable and efficient data for this task should also be a top priority. To the best of our knowledge, this aspect has received limited attention, and our proposed **TAV** dataset aims to highlight its importance.

**Datasets.** Public video–text datasets can be roughly grouped by scale and focus. *Web-scale corpora*—*MiraData* (330 k long clips) (Ju et al., 2024b), HD-VILA 100M (Xue et al., 2022), and auto-captioned sets like *Panda-70M* (Chen et al., 2024a) and *InternVid* (Wang et al., 2023c)—supply hundreds of millions of paired frames that power minute-level diffusion/Transformer training. *General short-video caption sets*, led by *WebVid-10M* (Bain et al., 2022) and *HowTo100M* (Miech et al., 2019), dominate text-to-video pre-training, while action-label datasets such as *Kinetics-700* (Carreira et al., 2019) and *Moments-in-Time* (Monfort et al., 2019) emphasize clip-level semantics. Finally, a spectrum of *domain-specific benchmarks* remains essential for evaluation and niche tasks: classical recognition/caption corpora (*UCF101* (Soomro et al., 2012), *MSR-VTT* (Xu et al., 2016), *ActivityNet-Captions* (Krishna et al., 2017), *YouCook2* (Zhou et al., 2018)); egocentric *Ego4D* (Grauman et al., 2022); face-centric *CelebV-Text* (Gu et al., 2023); robotic *BAIR* (Finn et al., 2017); synthetic *Moving MNIST* (Srivastava et al., 2015); and interpolation-oriented *Vimeo-90K* (Xue et al., 2019). Together, these resources span from thousands to hundreds of millions of videos, underpinning contemporary generative models across training, fine-tuning, and evaluation.

## 3 Method

In this section, we present the pipeline on preparing the **TAV** dataset.

**Data source.** We first draw a sample of 500 videos from the validation set of *Panda-70M* dataset, which contains a total of $2,000$ videos. The sample was carefully constructed to ensure that its category distribution closely approximates that of the full dataset, effectively serving as a representative subset of the population. For the post-training stage, the sample is split into 480 videos for training, 50 for validation, and 50 for testing.

**Scene transition detection.** We modified the method in *PySceneDetect*. Let $L(i, j)$ and $R(i, j)$ represent pixel values at position $(i, j)$ in two image frames for the same channel, and $N$ be the total number of pixels. We define the average pixel difference as:

$$D(L, R) = \frac{1}{N} \sum_{i,j} |L(i, j) - R(i, j)|.$$

Then we compute the average pixel difference in each HSV channel between consecutive frames, and define the overall frame change value as

$$\begin{aligned} V_t =& w_H \cdot D(H_t, H_{t-1}) \\ &+ w_S \cdot D(S_t, S_{t-1}) + w_V \cdot D(V_t, V_{t-1}). \end{aligned}$$

Here $w_H, w_S, w_V$ are the weights assigned to each channel by the user. A scene cut is detected if

$$V_t > \text{threshold}.$$

**Scene transition extraction.** We apply the aforementioned scene transition detection method to the previously selected 500 video samples. For each video, we retain only the first detected scene cut and extract a 10-second clip centered around it (combining 5 seconds before and 5 seconds after the transition point) to obtain a segment that contains a clear scene transition. If either side of the transition point does not contain a full 5 seconds of footage, we include as much as is available.

**Video Data Caption.** After obtaining the 10-second clip containing two distinct scenes, we use *BLIP* to generate separate textual descriptions for each scene. These descriptions are then combined into a single prompt that explicitly indicates a scene transition. For example: *{Previous scene: Superman is flying across the city; Next scene: He sees Batman fighting the Joker on a rooftop}*. The **TAV** dataset consists of 500 video–prompt pairs constructed in this manner.

## 4 Experiments

For consistency and brevity, we refer readers to Appendix A-C for implementation details, including code, and an example frame strip.

**Implementation.** We fine-tune the *OpenSora-Plan v1.3.1* (Lin et al., 2024) model using the **TAV** dataset under a video-to-text generation setting. The training is performed using a single process with DeepSpeed Zero Stage 2 optimization. We utilize the *google/mt5-xxl* text encoder and adopt a WFVAEModel (*D8_4x8x8*) pretrained from *OpenSora-Plan v1.3.0* as the video autoencoder. The model processes 33-frame video clips at a resolution of 256×256, with a sampling rate of 1 and frame rate of 8 FPS. Using a single H200 GPU, each training epoch completes in about 2 hours.

Key hyperparameters include a batch size of 1, 100 total training steps, a learning rate of $1 \times 10^{-5}$ with a constant scheduler, and *bf16* mixed precision training. We use Exponential Moving Average (EMA) with a decay rate of 0.9999 starting from step 0. Gradient checkingpoint is enabled, and training is resumed from the latest checkpoint. Additional strategies include sparse 1D attention (with *sparse_n = 4*), temporal and spatial interpolation scales set to 1.0, and a guidance scale of 0.1. The model uses SNR-weighted loss (*snr_gamma = 5.0*) and adopts a *v_prediction* type for diffusion.

**Experiment design.** To assess the model's performance, we construct three evaluation groups, each using a different version of the prompt to test the effectiveness of post-training.

- **Group A.** This group uses prompts consisting of a single sentence without indicating any scene transition (e.g., *{Superman flying across the building}*). It serves to demonstrate that the model is also capable of handling single-scene generation, highlighting its versatility beyond multi-scene transitions.

- **Group B.** This group uses prompts containing two sentences that imply, but do not explicitly

Table 2: Evaluation metrics for baseline and post-trained models across different training epochs.

| | group | epoch | average segments | aesthetic quality | overall consistency | dynamic degrees | imaging quality |
|---|---|---|---|---|---|---|---|
| Baseline | A | - | 1.180 | 0.510 | 0.045 | 0.203 | 0.652 |
| Baseline | B | - | 1.060 | 0.551 | 0.042 | 0.038 | 0.648 |
| Baseline | C | - | 1.120 | 0.517 | 0.049 | 0.089 | 0.643 |
| Post-trained | A | 16 | 1.840 | 0.401 | 0.060 | 0.783 | 0.575 |
| Post-trained | B | 16 | 1.800 | 0.405 | 0.062 | 0.789 | 0.592 |
| Post-trained | C | 16 | 1.740 | 0.395 | 0.062 | 0.816 | 0.584 |
| Post-trained | A | 24 | **2.380** | 0.436 | 0.052 | 0.538 | 0.647 |
| Post-trained | B | 24 | **2.700** | 0.419 | 0.054 | 0.526 | 0.630 |
| Post-trained | C | 24 | **2.900** | 0.429 | 0.060 | 0.517 | 0.599 |
| Post-trained | A | 36 | 2.300 | 0.430 | 0.053 | 0.643 | 0.608 |
| Post-trained | B | 36 | 2.520 | 0.425 | 0.054 | 0.643 | 0.622 |
| Post-trained | C | 36 | 2.400 | 0.443 | 0.057 | 0.515 | 0.616 |

indicate, a scene transition. For example: *{Superman is flying across the building, and then sees Batman fighting the Joker on a rooftop}*.

- **Group C.** This group uses prompts that explicitly instruct a scene transition. For example: *{Previous scene: Superman is flying across the building; Next scene: Superman sees Batman fighting the Joker on a rooftop}*.

We revise the prompts (originally generated from text descriptions of the 50 test videos in the **TAV** dataset) into the three groups described above. These prompts are then applied to both the baseline and post-trained models to evaluate the average number of scene transitions and overall image quality. We show the results in Table 2.

## 5 Result and Analysis

As shown in Table 2, the average number of scenes increases significantly after post-training. In the baseline model, particularly for Groups B and C, the average number of segments remains around 1, indicating a limited ability to recognize the need for multi-scene generation. In contrast, the post-trained model shows a substantial improvement, with the average number of segments even exceeding 2. These results demonstrate that post-training with the **TAV** dataset effectively enhances the model's capability for multi-scene generation.

Furthermore, post-training does not noticeably degrade video quality. On the contrary, it improves both *dynamic consistency* and *temporal smoothness*, enabling the model to generate more coherent motion and fluid scene transitions.

As training progresses, we also observe gradual improvements in *aesthetic quality* and *imaging quality*, with metrics approaching or matching those of the baseline. These results suggest that post-training with the **TAV** dataset enhances multi-scene generation without compromising visual fidelity.

Moreover, even though post-training is conducted on a multi-scene dataset, the model still performs well on prompts requiring only a single scene (Group A). As shown in Table 2, the post-trained model demonstrates strong performance not only when prompts explicitly indicate a two-scene structure (Group C), but also when the transition is only implicitly suggested (Group B).

## 6 Conclusion

In conclusion, our experiments demonstrate that prompt design plays a crucial role in controlling the number of scenes generated by T2V models. Furthermore, post-training on the **TAV** dataset significantly enhances the model's ability to recognize and fulfill multi-scene generation requirements, especially when such intent is expressed explicitly in the prompt. Notably, we also observe that, despite being trained on prompts with explicit scene transition instructions, the post-trained model shows improved understanding and response to prompts that imply two scenes without clearly stating the transition.

## Limitation

First, this study is a preliminary experiment. We only evaluate open-sourced state-of-the-art T2V models and use only a subset of videos from the Panda70M dataset due to limited computational resources. Second, our experiments currently focus solely on multi-scene clips. A more comprehensive evaluation should include a mixture of both single-scene and multi-scene clips. Third, it remains unclear which scene detection algorithm performs best in the T2V setting. The threshold configuration in our experiments is heuristically determined based on our prior experience.

## Ethical Considerations

**Usage Rights.** Our Data is collected from Panda-70M, which is an open sourced dataset. The Panda-70M dataset is provided by Snap Inc. under a license for *non-commercial, research purposes only*. Redistribution is permitted provided the original copyright notice, license terms, and disclaimers are retained.(Chen et al., 2024a) Content are safe, covering diverse video domains, including animals, scenery, food, sports, activities, vehicles, tutorials, news and TV, gaming. Prompts are generated in English.

Because our main result is about the quantity of scene generated by T2V models, not about the content of scene, the risk of ethical concerns is minimal. All prompts are generated by open-sourced models.

## Acknowledgment

We gratefully acknowledge the use of AI-assisted tools solely for grammatical corrections during manuscript preparation. No other aspects of the research— including conceptualization, experimental design, data analysis, or interpretation of results—were generated or modified by AI. All substantive content and conclusions were developed independently by the authors.

## Implementation Details

Code will come soon at anonymous space.

## References

Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. 2022. Frozen in time: A joint video and image encoder for end-to-end retrieval. *Preprint*, arXiv:2104.00650.

Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, and 1 others. 2024. Video generation models as world simulators. Technical report, OpenAI. Introduces the Sora text-to-video diffusion-transformer model.

João Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. 2019. A short note on the kinetics-700 human action dataset. In *arXiv:1907.06987*.

Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. 2020. Image gpt. *arXiv preprint arXiv:2006.03622*.

Ting Chen, Saurabh Saxena, Geoffrey Hinton, and Ishan Misra. 2023a. Magvit: Masked generative video transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6710–6719.

Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang-Wei Chao, Byung Eun Jeon, Yuwei Fang, and 1 others. 2024a. Panda-70m: Captioning 70m videos with multiple cross-modality teachers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang wei Chao, Byung Eun Jeon, Yuwei Fang, Hsin-Ying Lee, Jian Ren, Ming-Hsuan Yang, and Sergey Tulyakov. 2024b. Panda-70m: Captioning 70m videos with multiple cross-modality teachers. *Preprint*, arXiv:2402.19479.

Xinyuan Chen, Yaohui Wang, Lingjun Zhang, Shaobin Zhuang, Xin Ma, Jiashuo Yu, Yali Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. 2023b. Seine: Short-to-long video diffusion model for generative transition and prediction. ArXiv preprint, 2023.

Xinyuan Chen, Chang Xu, Xiaokang Yang, Li Song, and Dacheng Tao. 2018. Gated-gan: Adversarial gated networks for multi-collection style transfer. *IEEE Transactions on Image Processing*, 28(2):546–560.

Chelsea Finn, Ian Goodfellow, and Sergey Levine. 2017. Deep visual foresight for planning robot motion. In *Proceedings of the IEEE International Conference on Robotics and Automation*. BAIR Robot Pushing dataset.

Songwei Ge, Thomas Hayes, Harry Yang, Xi Yin, Guan Pang, David Jacobs, Jia-Bin Huang, and Devi Parikh. 2022. Long video generation with time-agnostic vqgan and time-sensitive transformer. In *European Conference on Computer Vision (ECCV)*.

Kristen Grauman, Andrew Westbury, Rohit Girdhar, and 1 others. 2022. Ego4d: Around the world in 3,000 hours of egocentric video. *arXiv preprint arXiv:2110.07058*.

Yufei Gu, Wenhao Huang, Xinyang Zhang, and 1 others. 2023. Celebv-text: A large-scale video-text dataset for realistic human generation. *arXiv preprint arXiv:2312.00734*.

Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. 2022. Latent video diffusion models for high-fidelity long video generation. *arXiv preprint arXiv:2211.13221*. Updated version v2 on 20 Mar 2023.

Roberto Henschel, Levon Khachatryan, Daniil Hayrapetyan, Hayk Poghosyan, Vahram Tadevosyan, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. 2024. Streamingt2v: Consistent, dynamic, and extendable long video generation from text. *arXiv preprint arXiv:2403.14773*.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 6840–6851.

Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. 2022. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*.

Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. 2023. Gaia-1: A generative world model for autonomous driving. arXiv preprint arXiv:2309.17080.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. 2023. Vbench: Comprehensive benchmark suite for video generative models. *Preprint*, arXiv:2311.17982.

Xuan Ju, Yiming Gao, Zhaoyang Zhang, Ziyang Yuan, Xintao Wang, Ailing Zeng, Yu Xiong, Qiang Xu, and Ying Shan. 2024a. Miradata: A large-scale video dataset with long durations and structured captions. *Preprint*, arXiv:2407.06358.

Xuan Ju, Yiming Gao, Zhaoyang Zhang, Ziyang Yuan, Xintao Wang, Ailing Zeng, Yu Xiong, Qiang Xu, and Ying Shan. 2024b. Miradata: A large-scale video dataset with long durations and structured captions. *arXiv preprint arXiv:2407.06358*.

Nal Kalchbrenner, Aaron van den Oord, Karen Simonyan, Ivo Danihelka, Oriol Vinyals, Alex Graves, and Koray Kavukcuoglu. 2017. Video pixel networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 1771–1779.

MinGuk Kang, Joonghyuk Shin, and Jaesik Park. 2023. Studiogan: A taxonomy and benchmark of gans for image synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. 2017. Dense-captioning events in videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Kuaishou Technology. 2024. Kuaishou unveils proprietary video generation model 'kling'; testing now available. Press release via PR Newswire. Introduces Kling with DiT backbone, 3D VAE and spatiotemporal attention.

Bin Lin, Yunyang Ge, Xinhua Cheng, Zongjian Li, Bin Zhu, Shaodong Wang, Xianyi He, Yang Ye, Shenghai Yuan, Liuhan Chen, Tanghui Jia, Junwu Zhang, Zhenyu Tang, Yatian Pang, Bin She, Cen Yan, Zhiheng Hu, Xiaoyi Dong, Lin Chen, and 5 others. 2024. Open-sora plan: Open-source large video generation model. *Preprint*, arXiv:2412.00131.

Han Lin, Abhay Zala, Jaemin Cho, and Mohit Bansal. 2023. Videodirectorgpt: Consistent multi-scene video generation via llm-guided planning. *arXiv preprint arXiv:2309.15091*. Updated July 2024.

Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. 2019. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Mathew Monfort, Alex Andonian, Bolei Zhou, and 1 others. 2019. Moments in time dataset: One million videos for event understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

OpenAI. 2023. GPT-4 technical report. Technical Report arXiv:2303.08774, OpenAI. Technical report.

Haonan Qiu, Menghan Xia, Yong Zhang, Yingqing He, Xintao Wang, Ying Shan, and Ziwei Liu. 2023. Freenoise: Tuning-free longer video diffusion via noise rescheduling. arXiv preprint arXiv:2310.15169.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695.

Axel Sauer, Katja Schwarz, and Andreas Geiger. 2022. Stylegan-xl: Scaling stylegan to large diverse datasets. In *Proceedings of SIGGRAPH '22: Special Interest Group on Computer Graphics and Interactive Techniques Conference*, pages 49:1–49:10, Vancouver, BC, Canada.

Eli Shechtman, Alex Rav-Acha, Michal Irani, and Steven M. Seitz. 2010. Regenerative morphing. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 615–622, San Francisco, CA, USA.

Jiaming Song, Chenlin Meng, and Stefano Ermon. 2021. Denoising diffusion implicit models. In *International Conference on Learning Representations (ICLR)*. ArXiv preprint arXiv:2010.02502.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. Ucf101: A dataset of 101 human action classes from videos in the wild. In *arXiv:1212.0402*.

Nitish Srivastava, Elman Mansimov, and Ruslan Salakhutdinov. 2015. Unsupervised learning of video representations using lstms. In *Proceedings of the International Conference on Machine Learning*. Moving MNIST.

Aaron van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. 2016. Conditional image generation with pixelcnn decoders. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 29, pages 4790–4798.

Aaron Van Den Oord, Oriol Vinyals, and 1 others. 2017. Neural discrete representation learning. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Ruben Villegas, Mohammad Babaeizadeh, Pieter-Jan Kindermans, Hernan Moraldo, Han Zhang, Mohammad Taghi Saffar, Santiago Castro, Julius Kunze, and Dumitru Erhan. 2022. Phenaki: Variable length video generation from open domain textual descriptions. In *International Conference on Learning Representations (ICLR)*.

Fu-Yun Wang, Wenshuo Chen, Guanglu Song, Han-Jia Ye, Yu Liu, and Hongsheng Li. 2023a. Gen-l-video: Multi-text to long video generation via temporal co-denoising. arXiv preprint arXiv:2305.18264.

Yaohui Wang, Xin Ma, Xinyuan Chen, Antitza Dantcheva, Bo Dai, and Yu Qiao. 2023b. Leo: Generative latent image animator for human video synthesis. arXiv preprint arXiv:2305.03989.

Yi Wang, Yinan He, Yizhuo Li, and 1 others. 2023c. Internvid: A large-scale video-text dataset for multimodal understanding and generation. *arXiv preprint arXiv:2307.06942*.

Yuqing Wang, Tianwei Xiong, Daquan Zhou, Zhijie Lin, Yang Zhao, Bingyi Kang, Jiashi Feng, and Xihui Liu. 2024. Loong: Generating minute-level long videos with autoregressive language models. *arXiv preprint arXiv:2410.02757*.

George Wolberg. 1998. Image morphing: A survey. *The Visual Computer*, 14(8-9):360–372.

Jiaqi Xu, Xinyi Zou, Kunzhe Huang, Yunkuo Chen, Bo Liu, MengLi Cheng, Xing Shi, and Jun Huang. 2024. Easyanimate: A high-performance long video generation method based on transformer architecture. *arXiv preprint arXiv:2405.18991*.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Hongwei Xue, Tiankai Hang, Yanhong Zeng, Yuchong Sun, Bei Liu, Huan Yang, Jianlong Fu, and Baining Guo. 2022. Advancing high-resolution video-language representation with large-scale video transcriptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T. Freeman. 2019. Video enhancement with task-oriented flow. In *International Journal of Computer Vision*. Vimeo-90K dataset.

Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. 2021. Videogpt: Video generation using vq-vae and transformers. arXiv preprint arXiv:2104.10157.

Shengming Yin, Chenfei Wu, Huan Yang, Jianfeng Wang, Xiaodong Wang, Minheng Ni, Zhengyuan Yang, Linjie Li, Shuguang Liu, Fan Yang, Jianlong Fu, Ming Gong, Lijuan Wang, Zicheng Liu, Houqiang Li, and Nan Duan. 2023. Nuwa-xl: Diffusion over diffusion for extremely long video generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*.

Luowei Zhou, Chenliang Xu, Jason J. Corso, Richard Socher, and Caiming Xiong. 2018. Towards automatic learning of procedures from web instructional videos. In *Proceedings of AAAI*.

## Appendix A: Frame and Timeline Comparison

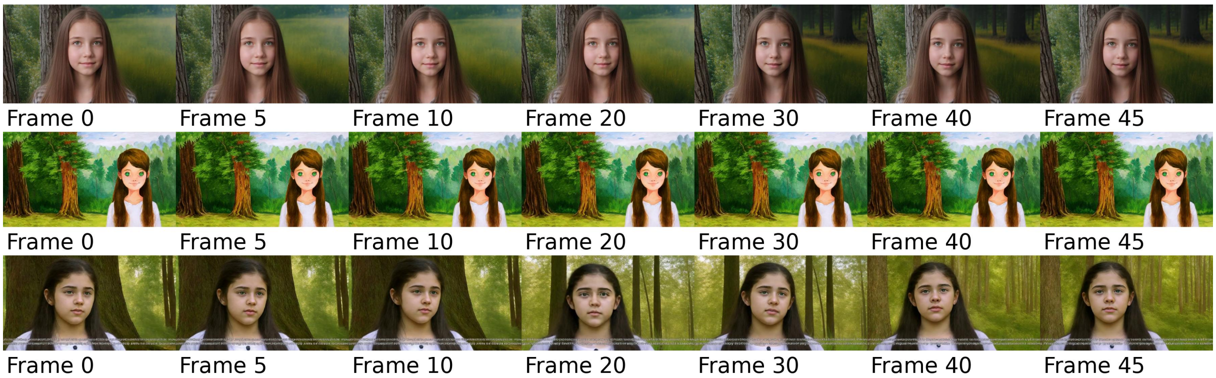


Figure 1: Frame–timeline comparison of three video generations. From **top** to **bottom**: (i) output generated by *EasyAnimate*; (ii) output generated by *OpenSora-Plan prior* to post-training; and (iii) output generated by *OpenSora-Plan after* 24 epochs of post-training. The prompt used to generate is: *"Previous scene: a girl with long hair and green eyes stands in front of a tree. Next scene: a painting of a forest with trees and grass"*

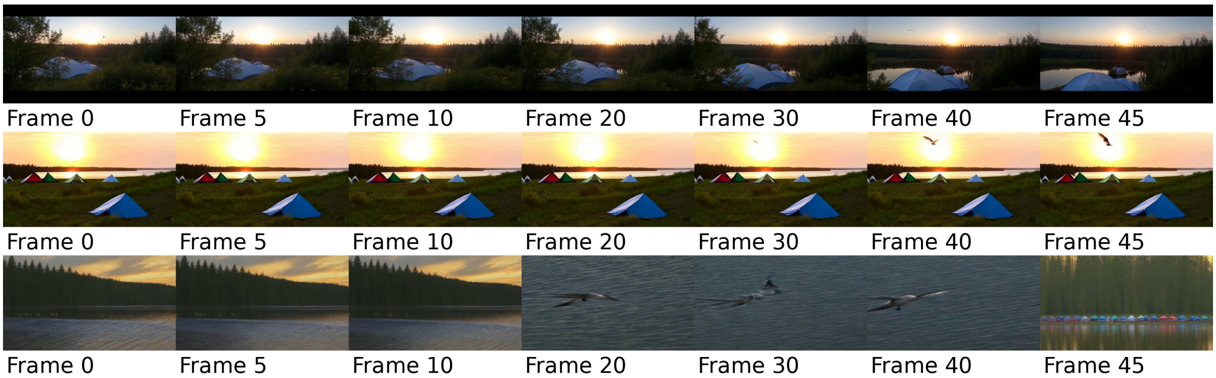


Figure 2: Frame–timeline comparison of three video generations. From **top** to **bottom**: (i) output generated by *EasyAnimate*; (ii) output generated by *OpenSora-Plan prior* to post-training; and (iii) output generated by *OpenSora-Plan after* 24 epochs of post-training. The prompt used to generate is: *"previous scene: a group of tents are set up in the woods; then next scene: a bird flying over the water at sunset "*

## Appendix B: Prompt Examples

This appendix provides a detailed listing of the prompt examples utilized in our experiments. Prompts are grouped into three categories: *Single Scene Prompts*, *Multi-Scene Prompts with Format*, and *Multi-Scene Prompts without Format*. Out of a total of 50 prompts, 42 can be successfully rendered within the paper, whereas 8 fail to display correctly due to formatting or compatibility issues.

### B.1 Single Scene Prompts

1. A man and woman sitting at a table on the beach.
2. A group of tents are set up in the woods.
3. A man and woman sitting at a table with drinks.
4. A girl with long hair and green eyes stands in front of a tree.
5. A boat is in the water near a rocky mountain.
6. A yellow and black bird flying through a blue sky.
7. A little girl in a wheelchair with a toy.
8. A group of women holding signs in front of a crowd.
9. A tall tower with a clock on top.
10. A man in a suit and tie is talking to a woman.
11. Get that superheroie by the - girl.
12. A woman in a black dress and glasses is on the news.
13. A woman in a bikini is talking to a man.
14. A bunch of bottles of liquor on a shelf.
15. A close up of a camera with a pen on it.
16. A person holding a white card with a black and white pattern.
17. A doll is standing on a bed.
18. Blur of a person walking.
19. A group of people are gathered around a tree.
20. A woman is sitting down on the news.
21. A group of people walking around a street.
22. A man in a blue shirt is standing next to a motorcycle.
23. A person is putting a bag of food into a box.
24. A person walking in the snow near a fence.
25. A white plate with the words news brief on it.
26. A man in a hat and a baseball cap.
27. A white microwave oven.

28. A white pot and a silver spoon on a table.
29. A bunch of books on a table.
30. The adobe file in adobe.
31. A table with bowls of food and a bowl of food.
32. A bowl filled with food sitting on top of a table.
33. A bunch of plastic bags sitting on top of a table.
34. Two dolls are sitting in a hospital bed.
35. A flooded street in the suburbs of detroit, michigan.
36. A small white mouse is sitting on the floor.
37. A cat is sitting on the floor next to a bottle of liquid.
38. A baseball player is being hit by a umpire.
39. A cartoon character holding a white cat.
40. A cat is sitting on the floor next to a bottle of liquid.
41. A snow covered parking lot with a sign.
42. A flooded street in phoenix, arizona.

### A.2 Multi Scene Prompts with format

**Example format:** *previous scene: ...; then next scene: ...*

1. previous scene: a man and woman sitting at a table on the beach; then next scene: a woman sitting at a table with a drink
2. previous scene: a group of tents are set up in the woods; then next scene: a bird flying over the water at sunset
3. previous scene: a man and woman sitting at a table with drinks; then next scene: a woman in a bikini is standing on the beach
4. previous scene: a girl with long hair and green eyes stands in front of a tree; then next scene: a painting of a forest with trees and grass
5. previous scene: a boat is in the water near a rocky mountain; then next scene: a woman sitting at a table with a drink
6. previous scene: a yellow and black bird flying through a blue sky; then next scene: the girls of the twilight
7. previous scene: a little girl in a wheelchair with a toy; then next scene: a doll sitting in a chair next to a box
8. previous scene: a group of women holding signs in front of a crowd; then next scene: a man and woman are standing in front of a microphone
9. previous scene: a tall tower with a clock on top; then next scene: a man is putting his ballot in the ballot box

10. previous scene: a man in a suit and tie is talking to a woman; then next scene: a man in a suit and tie is talking to another man in a suit
11. previous scene: get that superheroie by the - girl; then next scene: file file for you png file for you my little pony
12. previous scene: a woman in a black dress and glasses is on the news; then next scene: a woman sitting on a couch in front of a tv screen
13. previous scene: a woman in a bikini is talking to a man; then next scene: a man and woman sitting at a table with drinks
14. previous scene: a bunch of bottles of liquor on a shelf; then next scene: a man is standing at the bar
15. previous scene: a close up of a camera with a pen on it; then next scene: a man standing in front of a motorcycle
16. previous scene: a person holding a white card with a black and white pattern; then next scene: a man is holding a cell phone
17. previous scene: a doll is standing on a bed; then next scene: a little girl is putting a gift box
18. previous scene: blur of a person walking; then next scene: a purple vase with a white flower on it
19. previous scene: a group of people are gathered around a tree; then next scene: a cat is standing in the dark
20. previous scene: a woman is sitting down on the news; then next scene: two women sitting on a couch talking to each other women
21. previous scene: a group of people walking around a street; then next scene: a woman walking down a street with a blue jacket
22. previous scene: a man in a blue shirt is standing next to a motorcycle; then next scene: a close up of a cell phone
23. previous scene: a person is putting a bag of food into a box; then next scene: a person is putting food into a container
24. previous scene: a person walking in the snow near a fence; then next scene: a black background with a white and red flower
25. previous scene: a white plate with the words news brief on it; then next scene: a woman standing in front of a brick wall
26. previous scene: a man in a hat and a baseball cap; then next scene: police investigates a man who was shot in the back of a car in the river
27. previous scene: a white microwave oven; then next scene: a white bowl with a spoon and a cup
28. previous scene: a white pot and a silver spoon on a table; then next scene: a white crocked pot
29. previous scene: a bunch of books on a table; then next scene: a table with a bunch of boxes of food
30. previous scene: the adobe file in adobe; then next scene: a computer screen with a green background
31. previous scene: a table with bowls of food and a bowl of food; then next scene: ingredients for making a cake

32. previous scene: a bowl filled with food sitting on top of a table; then next scene: a white cup with a spoon in it
33. previous scene: a bunch of plastic bags sitting on top of a table; then next scene: a pile of plastic bags
34. previous scene: two dolls are sitting in a hospital bed; then next scene: two dolls sitting on a chair
35. previous scene: a flooded street in the suburbs of detroit, michigan; then next scene: a dog is standing in the middle of a flooded street
36. previous scene: a small white mouse is sitting on the floor; then next scene: a small dog is sitting on the floor
37. previous scene: a cat is sitting on the floor next to a bottle of liquid; then next scene: a small white mouse
38. previous scene: a baseball player is being hit by a umpire; then next scene: a baseball player is about to catch the ball
39. previous scene: a cartoon character holding a white cat; then next scene: a cartoon character with a blue background
40. previous scene: a cat is sitting on the floor next to a bottle of liquid; then next scene: a cat is sitting on the floor next to a bottle of sauce
41. previous scene: a snow covered parking lot with a sign; then next scene: a black background with a white and red flower
42. previous scene: a flooded street in phoenix, arizona; then next scene: a police tape is taped around a wall that was covered with graffiti

### B.3 Multi Scene Prompts without format

1. A man and woman sitting at a table on the beach; a woman sitting at a table with a drink.
2. A group of tents are set up in the woods; a bird flying over the water at sunset.
3. A man and woman sitting at a table with drinks; a woman in a bikini is standing on the beach.
4. A girl with long hair and green eyes stands in front of a tree; a painting of a forest with trees and grass.
5. A boat is in the water near a rocky mountain; a woman sitting at a table with a drink.
6. A yellow and black bird flying through a blue sky; the girls of the twilight.
7. A little girl in a wheelchair with a toy; a doll sitting in a chair next to a box.
8. A group of women holding signs in front of a crowd; a man and woman are standing in front of a microphone.
9. A tall tower with a clock on top; a man is putting his ballot in the ballot box.
10. A man in a suit and tie is talking to a woman; a man in a suit and tie is talking to another man in a suit.
11. Get that superheroie by the - girl; file file for you png file for you my little pony.
12. A woman in a black dress and glasses is on the news; a woman sitting on a couch in front of a tv screen.

13. A woman in a bikini is talking to a man; a man and woman sitting at a table with drinks.
14. A bunch of bottles of liquor on a shelf; a man is standing at the bar.
15. A close up of a camera with a pen on it; a man standing in front of a motorcycle.
16. A person holding a white card with a black and white pattern; a man is holding a cell phone.
17. A doll is standing on a bed; a little girl is putting a gift box.
18. Blur of a person walking; a purple vase with a white flower on it.
19. A group of people are gathered around a tree; a cat is standing in the dark.
20. A woman is sitting down on the news; two women sitting on a couch talking to each other women.
21. A group of people walking around a street; a woman walking down a street with a blue jacket.
22. A man in a blue shirt is standing next to a motorcycle; a close up of a cell phone.
23. A person is putting a bag of food into a box; a person is putting food into a container.
24. A person walking in the snow near a fence; a black background with a white and red flower.
25. A white plate with the words news brief on it; a woman standing in front of a brick wall.
26. A man in a hat and a baseball cap; police investigates a man who was shot in the back of a car in the river.
27. A white microwave oven; a white bowl with a spoon and a cup.
28. A white pot and a silver spoon on a table; a white crocked pot.
29. A bunch of books on a table; a table with a bunch of boxes of food.
30. The adobe file in adobe; a computer screen with a green background.
31. A table with bowls of food and a bowl of food; ingredients for making a cake.
32. A bowl filled with food sitting on top of a table; a white cup with a spoon in it.
33. A bunch of plastic bags sitting on top of a table; a pile of plastic bags.
34. Two dolls are sitting in a hospital bed; two dolls sitting on a chair.
35. A flooded street in the suburbs of detroit, michigan; a dog is standing in the middle of a flooded street.
36. A small white mouse is sitting on the floor; a small dog is sitting on the floor.
37. A cat is sitting on the floor next to a bottle of liquid; a small white mouse.
38. A baseball player is being hit by a umpire; a baseball player is about to catch the ball.
39. A cartoon character holding a white cat; a cartoon character with a blue background.
40. A cat is sitting on the floor next to a bottle of liquid; a cat is sitting on the floor next to a bottle of sauce.
41. A snow covered parking lot with a sign; a black background with a white and red flower.
42. A flooded street in phoenix, arizona; a police tape is taped around a wall that was covered with graffiti.

## Appendix C: Video Transition Clip Extraction Code

**Python Code for Scene Transition Detection and Clip Extraction:**

```
import json
import cv2
import numpy as np
import pandas as pd
import ffmpeg
from scenedetect import detect, ContentDetector
from tqdm import tqdm
import os

# Configuration parameters
CLIP_LENGTH = 10 # target duration in seconds
PADDING = 5 # padding before and after transition point
MIN_SCENE_LENGTH = 3
MAX_SCENE_LENGTH = 10

def detect_scenes(video_path):
  ”Detect scene transitions using PySceneDetect”
  scene_list = detect(video_path, ContentDetector())
  return [scene[1].get_seconds() for scene in scene_list]

def extract_transitional_clips(video_path, scene_timestamps):
  video_name = os.path.basename(video_path).split(’.’)[0]
  output_clips = []
  cap = cv2.VideoCapture(video_path)
  fps = cap.get(cv2.CAP_PROP_FPS)
  total_frames = int(cap.get(cv2.CAP_PROP_FRAME_COUNT))
  video_duration = total_frames / fps
  for timestamp in scene_timestamps:
    start_time = max(0, timestamp - PADDING)
    end_time = min(video_duration, timestamp + PADDING)
    if end_time - start_time > MAX_SCENE_LENGTH:
      end_time = start_time + MAX_SCENE_LENGTH
    output_filename = f”{video_name}_{int(start_time)}-{int(end_time)}.mp4”
    output_path = os.path.join(OUTPUT_VIDEO_DIR, output_filename)
    ffmpeg.input(video_path, ss=start_time, to=end_time)
      .output(output_path, vcodec=”libx264”, acodec=”aac”)
      .run(overwrite_output=True, quiet=True)
    output_clips.append({
      ”file_path”: output_path, ”video_name”: video_name,
      ”start_time”: start_time, ”end_time”: end_time,
      ”duration”: end_time - start_time, ”transition_frame”: timestamp
    })
  cap.release()
  return output_clips
```

Figure 3: Python code for detecting scene transitions and extracting fixed-length video clips centered on transitions.

```
def validate_clips(clips):
 filtered_clips = []
 for clip in tqdm(clips, desc="Validating Clips"):
  cap = cv2.VideoCapture(clip["file_path"])
  prev_frame = None; transition_detected = False
  while cap.isOpened():
    ret, frame = cap.read(); if not ret: break
    if prev_frame is not None:
     diff = np.mean(cv2.absdiff(prev_frame, frame))
     if diff > 50: transition_detected = True; break
    prev_frame = frame
  cap.release()
  if transition_detected: filtered_clips.append(clip)
 return filtered_clips

def save_metadata_to_json(filtered_clips):
 output_data = [{"file_path": c["file_path"], "text": ""} for c in filtered_clips]
 with open(OUTPUT_JSON_FILE, "w", encoding="utf-8") as f:
  json.dump(output_data, f, ensure_ascii=False, indent=4)
 print(f"Metadata saved to {OUTPUT_JSON_FILE}")

def main():
 video_path = ".../input_videos/example.mp4"
 scene_timestamps = detect_scenes(video_path)
 video_clips = extract_transitional_clips(video_path, scene_timestamps)
 validated_clips = validate_clips(video_clips)
 save_metadata_to_json(validated_clips)

if __name__ == "__main__": main()
```

Figure 4: Python code for detecting scene transitions and extracting fixed-length video clips centered on transitions.(continued)