Video Style Transfer with Reinforcement Learning

Sirui (Ariel) Chen*, Coco Xu*, Amelia Kuang*
Stanford University
450 Jane Stanford Way Stanford, CA 94305
siruic@stanford.edu, cocozxu@stanford.edu, kuangzy@stanford.edu

Abstract

We propose a novel framework for video style transfer that combines image-to-image models with reinforcement learning to address the challenge of temporal inconsistency across stylized frames. Although existing methods for image style transfer yield high-quality results, applying them independently to video frames often results in flickering artifacts and loss of temporal coherence. Our method reformulates video stylization as a sequential decision-making process, where a reinforcement learning agent adapts the latent representation of a Stable Diffusion model to ensure consistent style, content preservation, and smooth transitions. The agent is trained using Policy Gradient methods with a custom reward function that incorporates style similarity, content fidelity, and bidirectional temporal consistency measured via optical flow.

1. Introduction

Video style transfer aims to generate a stylized video that preserves the content of the original video while applying the visual style of a separate reference image. While image style transfer has been well-studied and produces visually appealing results for individual frames, naively applying these methods frame-by-frame to a video often leads to noticeable temporal inconsistencies—stylization varies from frame to frame, causing flickering and other artifacts.

To address this challenge, we propose a novel framework that formulates video style transfer as a sequential decision-making problem, allowing us to enforce temporal coherence across frames. Specifically, we design a reinforcement learning (RL) agent that operates over video frames to guide the stylization process toward consistent results. The agent leverages prior stylized frames as context when generating each new frame, aiming to minimize stylistic variation while maintaining visual fidelity to both the original content and the target style.

The input to our algorithm is a video sequence $\mathcal{V} = I_1, I_2, \ldots, I_T$ consisting of T frames and a single reference style image R. Our algorithm aims to produce a stylized video $\mathcal{S} = S_1, S_2, \ldots, S_T$ where each frame S_t satisfies the following criteria: (1) it reflects the style of R, (2) it preserves the semantic content of the original frame I_t, and (3) it is temporally coherent with neighboring stylized frames. We build on the image-to-image style transfer model DiffuseST [5], which outputs the latent representations for each stylized frame. Our main contribution is to introduce a reinforcement learning (RL) policy that adjusts these latent vectors to promote temporal consistency across frames while maintaining and improving auxiliary loss like style and content. The policy is trained using policy gradients methods and operates on the final latent representations from the encoding stage of previous and current frames. It outputs a residual adjustment term, which is added to the current frame's latent before decoding.

Our motivation for this work stems from the limitations of current video style transfer techniques, which either suffer from temporal artifacts or lack generalization capability. Moreover, to the best of our knowledge, no existing approach integrates Stable Diffusion with reinforcement learning for this task. By casting stylization as a reinforcement learning problem, we open the door to more adaptive and controllable stylization strategies for video generation.

Our method is shown to improve upon a baseline DiffuseST model by reducing temporal loss, while maintaining comparable performance in style and content preservation metrics. Furthermore, we observe a consistent increase in the overall reward signal and a downward trend in the total training loss across epochs. These results indicate that our reinforcement learning agent successfully enhances temporal coherence in stylized videos without compromising visual quality.

2. Related Work

Video style transfer lies at the intersection of image style transfer, temporal consistency in video generation, and reinforcement learning for vision tasks. Prior works can be categorized into three major groups: (1) classical image style transfer, (2) diffusion-based style transfer approaches, and

(3) reinforcement learning (RL) for stylization and diffusion guidance.

Classical Image Style Transfer: The foundational work by Gatys et al.[4] introduced neural style transfer using Gram matrix statistics extracted from VGG features. Follow-up works such as AdaIN[6] and WCT [7] proposed feed-forward architectures for real-time inference. While effective for individual images, these methods often produce artifacts when applied frame-by-frame to video.

Diffusion-Based Style Transfer: Diffusion models have enabled high-quality and semantically aligned generation. DiffuseST [5] introduced zero-shot image style transfer using pre-trained diffusion models with classifier-free guidance. SDEdit [8] allows editing images by partially denoising and re-sampling. ControlNet [15] augments diffusion models with structural guidance for more deterministic generation. However, these methods generally focus on still images, and naive application to video frames yields poor temporal consistency.

Reinforcement Learning for Stylization and Diffusion: Reinforcement learning offers a compelling framework for sequential adaptation. RL-NST [2] applies RL to image style transfer, tuning parameters for aesthetic outcomes. DDPO [1] shows that RL can be used to control diffusion generation toward high-level objectives by shaping reward functions. While promising, these methods are not focused on the objective of improving temporal loss in terms of video generation.

State-of-the-Art Models: State-of-the-art video style transfer models such as CoDeF [9] and CompoundVST [14] rely on explicit motion modeling and attention for frame alignment. However, most of them are limited to static pipelines with fixed heuristics.

Our proposed method is unique in casting video stylization as an MDP, where an RL agent actively selects latent conditioning strategies for each frame. This allows dynamic adaptation to style, content, and temporal cues. While most current systems are either fully supervised or require handcrafted loss terms, our approach enables learning more flexible, data-driven strategies.

To our knowledge, no prior work has integrated Stable Diffusion with reinforcement learning to address temporal stylization, making our contribution a novel step toward more controllable and temporally-aware video generation.

3. Method

Our approach combines frame-by-frame style transfer via **Stable Diffusion** with **policy gradient reinforcement learning method** to ensure temporal and stylistic consistency, as shown in Figure 1. We build on the existing codebase of DiffuseST [1] to extract latent representation for each frame and perform encoding and decoding stage of the diffusion model. All code for the policy gradient training loop, the policy network architecture, reward calculation, loss functions, and analysis is ours.

[1]https://github.com/I2-Multimedia-Lab/DiffuseST/tree/main

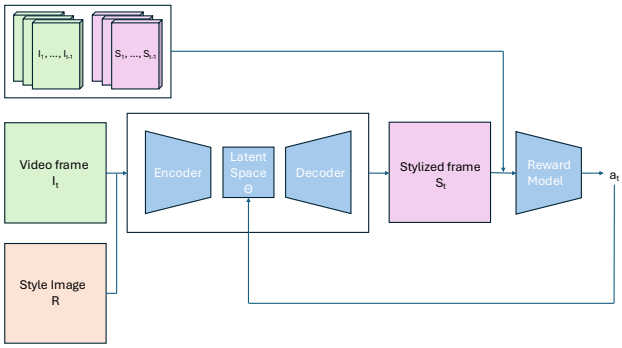

Figure 1: Video style transfer proposed architecture

The pipeline consists of three key components:

3.1. Single Frame Style Transfer

The input video $\mathcal{V}$ is decomposed into individual frames $\{I_1, \ldots, I_T\}$. Each frame I_t is passed through a Stable Diffusion-based image-to-image model $\mathcal{D}_\theta$ to produce a stylized output S_t:

$$S_t = \mathcal{D}_{\theta_t}(I_t, R) \tag{1}$$

where R is the reference style image, and θ_t is the latent-conditioning parameter at timestep t.

3.2. Policy Gradients Formulation

To guide consistent generation across frames, we formulate the problem as a Markov Decision Process (MDP), where at each timestep t:

- **State:**

$$s_t = \{I_{t-1}, I_t, S_{t-1}\} \tag{2}$$

The state includes the previous and current video frames I_{t-1}, I_t and the previous stylized frame S_{t-1}, providing the agent with temporal context through the temporal reward function.

- **Observation:**

$$o_t = \{L_{t-1}, L_t\} \tag{3}$$

The observation consists of the latent representations produced by DiffuseST [5] for the previous and current frame.

- **Action:**

$$a_t = \delta_t \sim \mathcal{N}(\mu, \sigma^2) \tag{4}$$

At each timestep, the policy outputs a residual adjustment term δ_t which is sampled from a learned distribution by the policy network. The distribution is conditioned on the latent representations of the current and previous frames. Specifically, the policy predicts the mean μ and a fixed standard deviation σ of a Gaussian distribution, from which δ_t is sampled. The adjusted latent $L_t + \delta_t$ is decoded to generate the stylized frame S_t.

- **Rewards:**

$$r_t = -(\lambda_{\text{style}} \cdot \mathcal{R}_{\text{style}}(S_t, R) + \lambda_{\text{content}} \cdot \mathcal{R}_{\text{content}}(S_t, I_t) \tag{5}$$

$$+ \lambda_{\text{temp}} \cdot \mathcal{R}_{\text{temp}}(S_t, S_{t-1} | I_t, I_{t-1})) \tag{6}$$

Each term is scaled by a distinct λ value to ensure consistency in magnitude and contribution to the overall reward. We define each reward as the negative of the corresponding loss, since lower loss is better but higher reward is preferred. The total reward encourages the agent to generate frames that are

  - **Stylistically consistent:** similarity of S_t to the reference style R
  - **Content preserving:** similarity of S_t to I_t
  - **Temporally consistent:** consistency between S_t and S_{t-1} using I_{t-1} and I_t as references

The policy is trained via policy gradient methods to optimize this reward, enabling adaptive adjustment of latent representations that maximize the cumulative reward across the video and preserve both visual fidelity and temporal coherence across frames.

3.3. Policy Network Architecture

The policy network is implemented as a lightweight convolutional model that takes the latent representations of two consecutive frames–L_{t-1} and L_t–and predicts a distribution over the residual adjustment δ_t. The network first concatenates L_{t-1} and L_t along the channel dimension and processes the result through two convolutional layers followed by instance normalization and ReLU activation. A residual block further refines the feature representation.

The output is passed through a 1×1 convolution to produce the adjustment mean μ. A learnable gating parameter modulates this mean to stablize early training, and the standard deviation σ is modeled as a learnable scalar. The final action δ_t is sampled from a Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$ via reparameterization, and the log probability of the return is fed back to the optimization loop to compute the policy gradient updates. Figure 2 shows the policy network architecture.

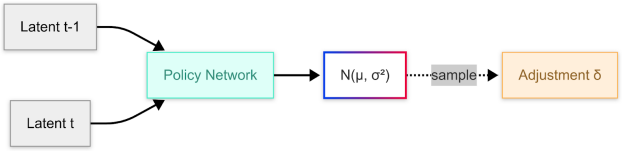

Figure 2: Policy predicts a Gaussian distribution from which the adjustment term δ is sampled and added to the latent.

3.4. Frame-by-Frame Style Transfer Baseline

To establish a strong baseline, we adopt DiffuseST [5] to perform per-frame style transfer. DiffuseST is a training-free diffusion-based framework by disentangling content and style representations, combining spatial and textual embeddings, and using step-wise nature of diffusion model for content and style injection.

In the baseline setting, each video frame is processed independently. Given a content frame and a reference style image, DiffuseST generates the stylized frame. This setup inherits the strengths of DiffuseST, including rich expression of style and good content preservations. However, as it lacks notion of temporal structure, the resulting video may have inconsistency across frames, particularly in regions with fine textures or motion details. To tackle this limitation, we introduce a policy-gradient-based refinement stage that learns to enforce consistency across time while retaining the aesthetic quality of the style transfer. We adopt three quantitative metrics: style loss, content loss and temporal consistency score.

3.5. Reward Functions

Style Loss. To ensure each stylized frame adheres to the desired artistic style, we compute the style loss between each stylized frame and the style image using the Gram matrix [3]. This loss captures the correlations between feature activations in a pretrained network, effectively measuring stylistic similarity. With this value, we guide the RL agent to maintain the desired artistic style across frames.

$$\mathcal{L}_{\text{style}} = \sum_l \|G_l(S_t) - G_l(R)\|_F^2$$

where $G_l(x)$ denotes the Gram matrix of the feature activations at layer l for image x. $G_l(x) = \phi_l(x)\phi_l(x)^\top$ is computed from the vectorized feature map $\phi_l(x)$ from VGG.

Content Loss. To quantify content preservation in our stylized outputs, we adopt the Learned Perceptual Image Patch Similarity (LPIPS) metric [16]. LPIPS measures perceptual similarity based on deep feature activations from pretrained neural networks, correlating well with human visual judgments. A lower LPIPS score indicates that the styl-

(a) Style image.

(b) Input Frame 1.

(c) Input Frame 8.

(d) Input Frame 16.

Figure 3: Example style image and input video frames from the dataset.

ized image remains perceptually closer to the original content. In our experiments, we use the AlexNet-based version (version 0.1) of LPIPS as it provides a balanced trade-off between performance and computational efficiency. This metric enables us to objectively evaluate how well the semantic content of the original video frames is preserved after style transfer.

Bidirectional Temporal Consistency Score. Temporal consistency is measured using optical flow estimated by a pretrained RAFT model [13]. Given two consecutive input frames I_t and I_{t+1}, we compute forward and backward optical flow fields, $F_{t\to t+1}$ and $F_{t+1\to t}$ respectively, and use them to warp the corresponding stylized frames S_t and S_{t+1}. The warped frame $\mathcal{W}(S_t, F_{t\to t+1})$ is expected to align with S_{t+1}, and vise versa. The loss is computed as the L1 differences between the warped and target frames in both directions. To avoid penalizing regions affected by occlusion or flow uncertainty, we apply an occlusion mask based on flow cycle consistency. This loss encourages smooth and temporally consistent transitions in the output video.

4. Dataset and Features

Our dataset consists of two components: (1) style reference images and (2) short content video clips. For style references, we curate high-resolution artworks from the WikiArt dataset [11]. For content videos, we collect 10 short 2-second clips from Pexels [10], covering diverse scenes such as nature, urban landscapes, people, and pets. To ensure that the selected clips exhibit sufficient temporal dynamics for our model to learn from, we further filter them based on the magnitude of motion present in each clip. Each video contains approximately 20 frames, yielding a total of 200 video frames for stylization. Due to the limitation of our compute, we were only able to train on 5 of those videos.

To prepare the data, all videos are downsampled to a spatial resolution of 256×256 and converted to RGB. We then extract individual frames from each clip and store them as PNG images. We split the dataset into 75% training, and 25% test. Due to compute limitation, we finally choose 5 videos from training set and 2 videos from test set for experiments. Each stylization episode consists of a content video clip and the style image. This results in stylization tasks that vary significantly in texture, color palette, and spatial composition.

Our reinforcement learning agent does not directly operate on raw pixel data. Instead, it conditions the Stable Diffusion pipeline via latent vector perturbations, which indirectly control the output stylization. For each frame, we extract 1000 latent vectors, one for each diffusion timestep, and allow the agent to perturb these latents. These latent vectors are learned over the course of training based on a reward that is computed from the extracted perceptual and temporal features described above. Before feeding images to the diffusion model and calculating reward functions, we normalize and transform our images. No other normalization or whitening is applied to the inputs.

By designing the dataset and the preprocessing pipeline in this way, we enable consistent evaluation of temporal stability and visual quality across varied stylistic contexts and scene dynamics.

5. Experiments

We evaluate our baseline model on five videos featuring diverse content types, including animals, human, flowers, and natural scenery. For all experiments, we use The Starry Night by Vincent van Gogh as the target style image (Figure 3a).

Hyperparameters. We conducted experiments varying the number of training epochs and the learning rate. We used the Adam optimizer. We experimented with running for 2, 5, or 10 epochs and learning rate of 1e-4 and 5e-4. Our results indicate that training for more epochs consistently improves performance, likely because the reinforcement learning agent has more opportunity to refine the residual latents used to condition the diffusion model. Among the learning rates tested, we found that 1e-4 yields the most stable and effective training dynamics, striking a good balance between convergence speed and policy stability.

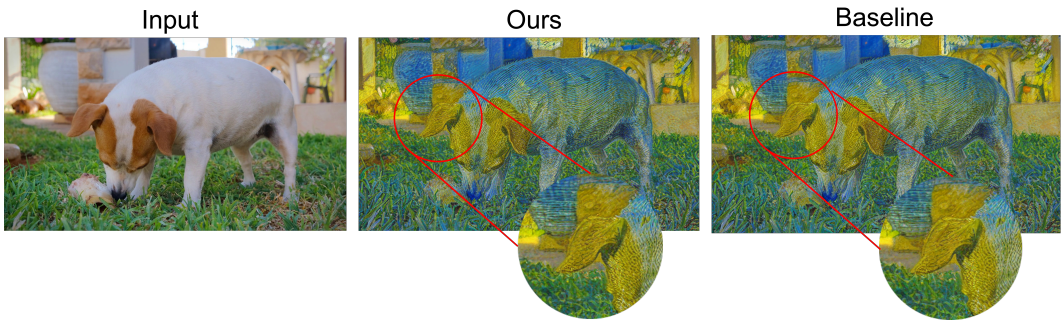

Figure 4: Zoom-in to see detailed texture comparison between our method and the baseline.

5.1. Batched Gradient Accumulation Experiment

To improve training stability, we conducted an experiment introducing batched gradient accumulation during policy optimization. In our initial setup, the policy was updated after computing rewards from each pair of consecutive frames. However, this approach resulted in unstable training and poor stylization quality. We attribute this to the high variance in reward signals between adjacent frames, caused by both the stochastic nature of the diffusion model and the variability in scene dynamics.

To address this, we introduced gradient accumulation across multiple steps—specifically, accumulating gradients over 4 consecutive frame pairs before applying an update. This technique effectively reduces the variance of policy gradient estimates and improves training stability. Moreover, for longer video sequences, gradient accumulation over local temporal windows enables the agent to better capture short-term dependencies without processing the entire sequence at once.

This strategy serves as a lightweight alternative to Truncated Backpropagation Through Time (TBPTT), enabling localized temporal credit assignment while maintaining computational efficiency. By updating the policy based on multi-step observations rather than single-frame transitions, the agent learns smoother, more coherent stylization policies that generalize better to longer and more complex videos without suffering from temporal fragmentation.

5.2. Reward Weights Design Experiment

Our reward function is composed of three key components: style fidelity, content preservation, and temporal consistency. To ensure that each component contributes meaningfully during training, we scaled the individual reward terms to be of comparable magnitude, as referenced in Table 1.

We experimented with multiple weighting schemes and ultimately found that assigning a weight of 10 to both the style and temporal rewards, and a weight of 1 to the content reward, consistently produced the most balanced and visually coherent results. This configuration reflects our goal of improving temporal coherence without sacrificing visual quality or semantic content.

We found that this balanced weighting encourages the agent to generate stylized videos that are not only temporally stable, but also visually coherent and faithful to the original content.

6. Results

6.1. Evaluation Metrics

We evaluate the final outputs of our method and the baseline using a combination of quantitative and qualitative metrics. For quantitative evaluation, we rely on optimization-based losses such as temporal consistency, content preservation, and style similarity. Additionally, we use CLIP-based semantic similarity, all of which are commonly adopted in prior work on video style transfer and related domains. For qualitative comparison, we provide visualizations of representative stylized frames and conduct detailed inspections across the full set of output frames for each video, enabling a comprehensive assessment of visual coherence and stylization fidelity.

6.2. Qualitative Results

We compare stylized output frames produced by our method against those from the baseline. Notably, although we introduce additional noise into the latent space before the diffusion decoding process—a step that typically disrupts generation quality—the final outputs from our method remain visually comparable to the baseline. This suggests that our reinforcement learning agent effectively learns a meaningful adjustment to the latent representations generated by the pretrained diffusion encoder.

When examining the fine-grained textures in the stylized frames (see Figure 4), our method produces noticeably stronger line strokes and richer texture details compared to the baseline. We attribute this improvement to the use of the Starry Night style image, which contains distinct and

	Data Set	Content Loss	Style Loss	Temporal Loss
baseline	train	0.3421	8.8020	0.1724
	test	0.3681	5.8211	0.1535
Diffusion w/ rl (Ours)	train	0.3288	8.7992	0.1627
	test	0.3693	5.8312	0.1508

Table 1: Comparison of style, content, and temporal losses between the baseline and our proposed method. We then apply scaling before calculating the total rewards.

	data set	Content preservation	Style Similarity	Temporal Consistency
Baseline	train	0.7249	0.6318	0.9767
	test	0.6723	0.6316	0.9839
Diffusion w/ RL (Ours)	train	0.7166	0.6322	0.9755
	test	0.6690	0.6314	0.9833

Table 2: Comparison of CLIP-based style, content, and temporal evaluation between the baseline and our proposed method.

expressive artistic textures and that our model is better able to capture and preserve.

Additional qualitative comparisons are shown in Figure 5, further demonstrating the coherence and quality achieved by our approach.

6.3. Quantitative Results

6.3.1 Loss Evaluation

We first evaluate the individual loss components defined in our reward function on the final stylized outputs produced by our method and the baseline.

As shown in Table 1, our method outperforms the baseline across all three metrics, demonstrating that our reinforcement learning framework provides consistent improvements over the base DiffuseST model in terms of temporal stability, stylistic fidelity, and content preservation. Furthermore, we observe that two of the three losses are lower on the test set compared to the training set. This could be due to the small size and the lack of diversity within the test set. While this might not reflect the generalization of the results, the consistent improvement across all losses still shows that the model is able to learn meaningful policies.

6.3.2 CLIP Similarity

We also evaluate the stylized outputs from our method and the baseline in a learned representation space. Specifically, we extract CLIP [12] features for all video frames, leveraging the strong semantic encoding capabilities of CLIP due to its large-scale pretraining. This allows us to assess how well the semantic content and style characteristics are preserved in an embedding space aligned with human perception.

For each frame in the video, we extract the CLIP image embedding and compare it to the embedding of the corresponding input frame (content reference) or style reference. We compute cosine similarity between these embeddings to quantify how closely the stylized output matches the target in CLIP space:

$$\text{CLIPSim}(x, y) = \frac{\langle f(x), f(y) \rangle}{\|f(x)\| \cdot \|f(y)\|} \tag{7}$$

where $f(x)$ and $f(y)$ denote the CLIP embeddings of frames x and y, respectively, and $\langle \cdot, \cdot \rangle$ represents the dot product.

As shown in Table 2, our method achieves performance comparable to the baseline, despite the introduction of additional noise in the latent space. This indicates that our approach maintains semantic integrity in the stylized outputs. However, we believe there is room for improvement, and a more detailed discussion of potential limitations and influencing factors is provided in Section 6.4.1.

6.4. Discussion

Our experiments demonstrate consistent improvement across temporal, style, and content loss metrics while optimizing the overall reward as could be seen in Figure 6 in the Appendix. Although the absolute improvements in numerical values are small, this outcome is expected due to the scale of the loss components and the limited dataset used for training.

Notably, we observe a promising upward trend in the reward signal and a steady decline in total loss over training epochs. As shown in Table 1, while the final loss values do not differ drastically from the baseline, their consistent reduction provides evidence that the reinforcement learning agent is effectively learning and improving its policy.

These trends suggest that our RL framework is capable of gradually optimizing stylization quality across multiple objectives, even under resource-constrained settings. This proves our hypothesis that RL is beneficial for video style transfer tasks in terms of improving loss.

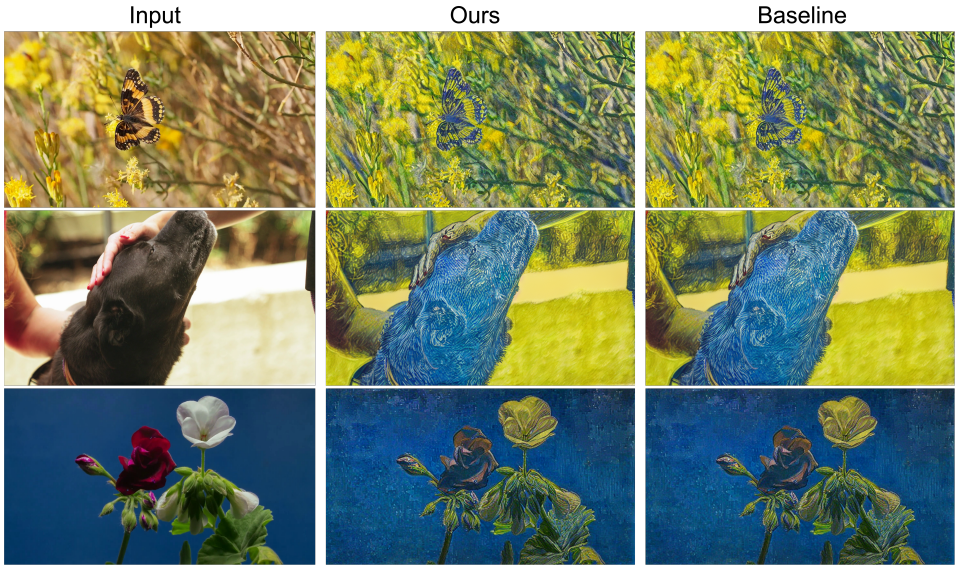

Figure 5: More examples of output from our methods and baseline

6.4.1 Limitations

Through the experiments and analysis to our model, we also identify several limitations that hinder our methods performance. Understanding these limitations are crucial for future improvements and optimizations.

Instability of Policy Gradient Training. We chose to train with policy gradient method as it provides a lightweight and straightforward testbed for our ideas. Despite its simplicity, policy gradient is known to exhibit high variance and less stable convergence compared to methods that use critic networks and value estimates. To address this in future work, one could introduce a critic network to evaluate the latent action to reduce variance in the optimization.

Limited Compute and Data Due to constrained computational resources, our experiments were conducted on a relatively small dataset to ensure manageable comopute. While the results demonstrate the learning capability of our proposed method, we believe its generalization performance could be significantly improved with access to larger-scale datasets and greater compute. Scaling up training would likely allow the reinforcement learning agent to better capture diverse motion patterns and style-content variations across videos.

7. Conclusion and Future Work

Our RL-based method demonstrates a proof of concept for using RL to improve video style transfer. While the improvements over the baseline diffusion model are subtle, they are notable given the limited compute and data available during training. These results suggest that even under constrained conditions and in the presence of modeling noise, RL can still yield measurable benefits in terms of video coherence and consistency.

This outcome aligns with expectations, as RL frameworks typically require significantly more data and training steps to converge effectively, a trend also observed in prior work.

Future directions could focus on scaling the RL training with larger datasets and more computational resources, enabling more robust policy learning and more pronounced improvements in temporal consistency and stylization quality. Additionally, it would be valuable to investigate whether the policy trained on Starry Night generalizes to other distinct style transfer tasks, providing insight into the adaptability and transferability of the learned stylization policy.

8. Contributions

Sirui Chen: Implement content loss, policy gradient rewards and update, run experiments and evaluations.

Coco Xu: Data Processing, Implement Style Loss, CLIP evaluation, Ran experiments.

Zhiyi Kuang: Implement temporal loss, policy network architecture, policy gradient training loop, run experiments.

Sirui and Zhiyi are sharing this project with 224R. Specifically, the parts done specifically for 231N include

1. Extraction of latent representations using the baseline diffusion model DiffuseST

2. Formulation of the style loss function (Gram matrix), content loss (LPIPS), temporal loss (RAFT and optical flow)

3. Experiments for better performance and hyperparameters

4. Formulation of evaluation metric using CLIP embeddings

5. Formulation and preprocessing of the Dataset

Appendix

(a) Reward (b) Content Loss

(c) Style Loss (d) Temporal Loss

Figure 6: Loss and reward trends across training epochs. Each subfigure reports a key metric used to monitor learning progress of the RL agent.

References

[1] K. Black, M. Janner, Y. Du, I. Kostrikov, and S. Levine. Training diffusion models with reinforcement learning, 2024. 2

[2] C. Feng, J. Hu, X. Wang, S. Hu, B. Zhu, X. Wu, H. Zhu, and S. Lyu. Controlling neural style transfer with deep reinforcement learning, 2023. 2

[3] L. A. Gatys, A. S. Ecker, and M. Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015. 3

[4] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2414–2423. IEEE, 2016. 2

[5] Y. Hu, C. Zhuang, and P. Gao. Diffusest: Unleashing the capability of the diffusion model for style transfer. In *Proceedings of the 6th ACM International Conference on Multimedia in Asia (MMAsia '24)*, 2024. 1, 2, 3

[6] X. Huang and S. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1501–1510. IEEE, 2017. 2

[7] Y. Li, C. Fang, J. Yang, Z. Wang, X. Lu, and M.-H. Yang. Universal style transfer via feature transforms. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 386–396, 2017. 2

[8] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J.-Y. Zhu, and S. Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021. 2

[9] H. Ouyang, Q. Wang, Y. Xiao, Q. Bai, J. Zhang, K. Zheng, X. Zhou, Q. Chen, and Y. Shen. Codef: Content deformation fields for temporally consistent video processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2

[10] Pexels. Pexels: Free stock photos and videos, 2025. Accessed April 23, 2025. 4

[11] F. Phillips and B. Mackintosh. Wiki art gallery, inc.: A case for critical thinking. *Issues in Accounting Education*, 26(3):593–608, 08 2011. 4

[12] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 6

[13] Z. Teed and J. Deng. RAFT: recurrent all-pairs field transforms for optical flow. *CoRR*, abs/2003.12039, 2020. 4

[14] W. Wang, J. Xu, L. Zhang, Y. Wang, and J. Liu. Consistent video style transfer via compound regularization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12114–12121, 2020. 2

[15] L. Zhang and M. Agrawala. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*, 2023. 2

[16] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 3