Extended Abstract

Motivation Video style transfer aims to alter the style of a video while preserving its content. The input to our problem is a style image and the original video. The output is the stylized version of the video. 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.

Method One existing solution is to independently apply image style transfer to each frame of the video, but the stylized video could have inconsistent content or temporal artifacts. 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. We formulate video style transfer as a Markov Decision Process in which a policy network predicts a Gaussian residual adjustment δ_t to the Stable Diffusion latent of each frame, conditioned on the latents of two consecutive frames. The agent is trained via policy gradients (REINFORCE) to maximize a reward that combines three components: Gram-matrix style similarity to the reference image, LPIPS content similarity to the original frame, and bidirectional temporal consistency measured by warping stylized frames with RAFT optical flow and L1 penalities masked for occlusion.

Implementation Building on the DiffuseST [5] codebase, we extract and re-encode per-frame latents using Stable Diffusion's image-to-image pipeline. Our lightweight convolution policy network concatenates previous and current latents, processes them through two conv-norm-ReLU layers plus a residual block, and outputs a mean μ for the adjustment (with learnable σ). We integrate our custom training loop–complete with reward calculation, sampling via the reparametrization trick, and gradient updates–alongside all analysis scripts and visualization code.

Results We evaluate our reinforcement learning-based video style transfer method using both quantitative metrics (style loss, temporal consistency, content preservation, CLIP similarity) and qualitative comparisons. Our agent consistently improves over the baseline diffusion model across all loss components by decreasing 1.7% on average, despite limited training data and the stochasticity introduced by latent perturbations.

Training curves show a steady increase in reward and decrease in total loss, suggesting that the learned policy effectively guides latent adjustments to enhance stylization quality. Qualitative results reveal sharper textures and improved temporal coherence, attributes to the use of The Starry Night that provide expressive styles to learn.

Discussion While absolute numerical improvements are modest, they are notable given the limited scale of training data and compute resources. The results support our hypothesis that RL can optimize stylization quality under multi-objective reward settings.

The policy gradient method used in this work is known for high variance and instability. Despite this, our method shows robust performance, suggesting that even lightweight RL frameworks can be beneficial in visual generation domains. This highlights the promise of reinforcement learning as a general framework for optimizing sequential objectives in generative modeling tasks.

Conclusion Our work provides a proof of concept that reinforcement learning can be effectively applied to enhance video style transfer by learning policies that improve temporal and stylistic coherence. While improvements over the baseline are subtle, they are significant in light of limited data, compute, and the stochastic nature of diffusion models.

These findings highlight RL's potential in generative modeling and motivate future work on scaling, stabilizing training, and exploring generalization across styles and video domains.Ultimately, this work lays the groundwork for applying RL to broader challenges in video generation and multimodal visual learning.

Video Style Transfer with Reinforcement Learning

Amelia Kuang[*]
Department of Computer Science
Stanford University
kuangzy@stanford.edu

Sirui (Ariel) Chen[*]
Department of Computer Science
Stanford University
siruic@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.

Our main contributions are as follows:

- **First work** to integrate Stable Diffusion (DiffuseST) with reinforcement learning for this problem with public codebase and demonstrate a proof of concept that using RL can improve the quality of video style transfer.
- Introduce a latent-space **lightweight RL policy** that adjusts diffusion outputs for temporal consistency.

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

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

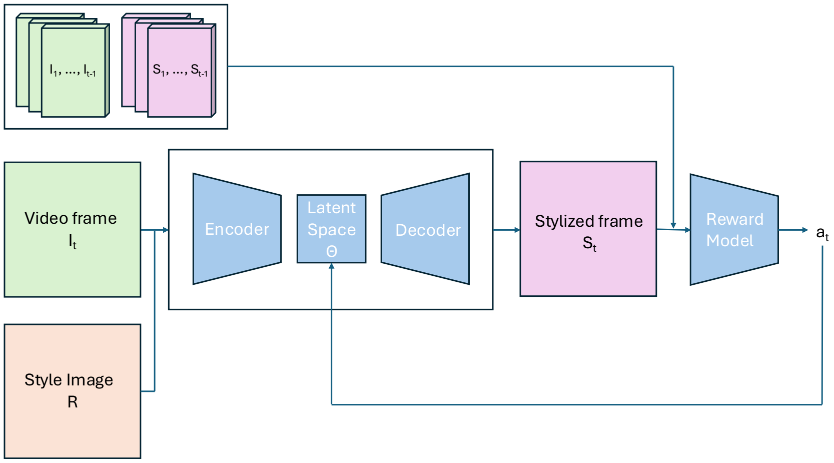


Figure 1: Video style transfer proposed architecture

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.

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:**
  $$r_t = -(\lambda_{\text{style}} \cdot \mathcal{R}_{\text{style}}(S_t, R) + \lambda_{\text{content}} \cdot \mathcal{R}_{\text{content}}(S_t, I_t) + \lambda_{\text{temp}} \cdot \mathcal{R}_{\text{temp}}(S_t, S_{t-1}|I_t, I_{t-1})) \tag{5}$$

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 the difference between the reward generated by its outputs and the reward generated by the baseline diffusion model, enabling it improvement beyond the baseline in 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.

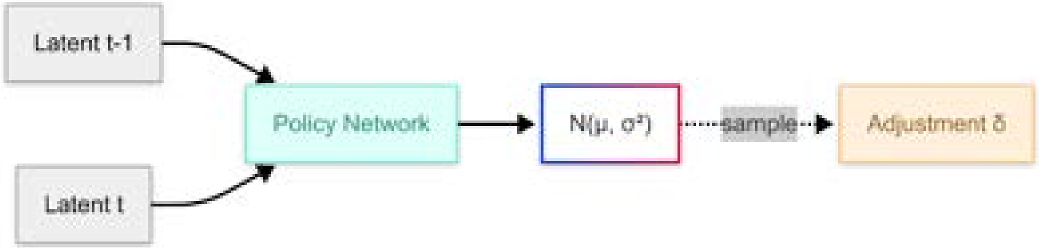


Figure 2: Policy predicts a Gaussian distribution from which the adjustment term δ is sampled and added to the latent.

3.4 Reward Functions

Style Loss. We guide the RL agent to maintain the target art style by penalizing the squared Frobenius norm between the Gram matrices [3] of the agent's stylized frame S_t and the reference style image R:

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

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 reward the RL agent for preserving semantic content, we use the Learned Perceptual Image Patch Similarity (LPIPS) metric [16]. LPIPS is correlated with human perceptual judgments. A lower LPIPS score between a stylized frame and its original content frameindicates better frame preservation. We employ the AlexNet-based LPIPS (v0.1) for its balance of accuracy and efficiency.

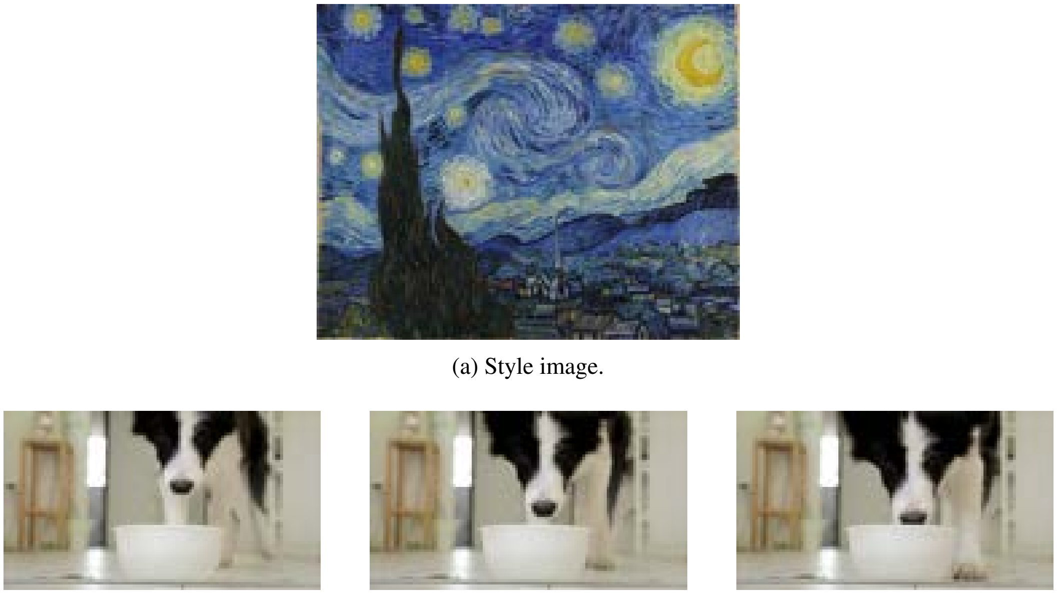
Bidirectional Temporal Consistency Score. We encourage smooth frame-to-frame transitions by warping consecutive stylized frames S_t and S_{t-1} using forward/backward RAFT flows [13] $F_{t-1\to t}$ and $F_{t-1\to t}$ estimated from I_t and I_{t-1}, then penalizing the L1 difference between each warped frame and its target. A occlusion mask derived from the flow cycle consistency prevents spurious penalties in unreliable regions.

4 Experimental Setup

4.1 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. Each video presents unique dynamics and appearance variations, providing a challenging testbed for stylization policy learning. Across all tasks, the agent aims to produce a sequence of stylized frames conditioned on a fixed target style image—The Starry Night by Vincent van Gogh (Figure 3a).

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.



(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.

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.

4.2 Frame-by-Frame Style Transfer Baseline

We use DiffuseST [5] — a training-free, diffusion-based framework that disentangles content and style via spatial and textual embeddings — to stylize each frame independently, yielding rich style expression and strong content preservation but suffering from temporal inconsistency across frames. To address this, we introduce a policy-gradient refinement stage that enforces frame-to-frame consistency while maintaining aesthetic quality, evaluated using style loss, content loss, and temporal consistency metrics.

4.3 Hyperparameters

We investigate the effect of training time and learning rate on the performance of our RL agent. Using the Adam optimizer, we conduct experiments with varying numbers of training epochs (2, 5, and 10) and learning rates (1e-4 and 5e-4). Our findings suggest that longer training horizons consistently yield better policy performance, as the agent has more information about the environment to refine the residual latent inputs that modulate the diffusion-based decoder. Among the tested learning rates, 1e-4 provides the most stable policy updates, striking a favorable balance between learning progress and training stability.

4.4 Experiments

4.4.1 Batched Gradient Accumulation for Policy Optimization

To address high variance in policy gradient estimates caused by frame-to-frame reward fluctuations, we introduce batched gradient accumulation during policy optimization. Initially, our agent updated its policy after receiving feedback from each environment step (i.e., stylizing one frame transition). However, this yielded unstable learning and suboptimal stylization policies due to the stochasticity of the diffusion model and rapid scene changes between consecutive frames.

We instead accumulate gradients across four successive transitions before performing a policy update. This reduces variance in gradient estimation and stabilizes learning. Moreover, for longer video trajectories, this method allows the agent to reason over local temporal windows, learning to assign credit across short temporal contexts without incurring the memory and complexity costs of full-sequence backpropagation.

This approach can be viewed as a lightweight alternative to Truncated Backpropagation Through Time (TBPTT), providing localized credit assignment while maintaining computational efficiency. By optimizing the policy over multi-step returns rather than single-step transitions, the agent learns to produce smoother and more temporally coherent stylizations across diverse video trajectories.

4.4.2 Reward Weighting Strategy

The agent's reward signal comprises three components: style fidelity, content preservation, and temporal consistency. To ensure balanced learning, we normalize these reward terms to comparable magnitudes, as shown in Table 1.

We experiment with multiple weighting schemes to assess how different weightings influence policy behavior. Empirically, we find that setting the weights to 10 for style and temporal rewards, and 1 for content reward, leads to the most visually and temporally consistent policies. This configuration reflects our goal of improving temporal coherence without sacrificing visual quality or semantic content.

The weighted reward encourages the agent to optimize a policy that produces stylized trajectories with high temporal consistency and visual fidelity—key indicators of successful sequential decision-making in this task setting.

4.5 Evaluation Metrics

To assess the effectiveness of the learned policy, we employ a combination of quantitative reward-based metrics and qualitative visual inspections. For quantitative evaluation, we compute the following metrics over full video trajectories generated by the agent's policy: temporal consistency, content preservation, style similarity. We also use CLIP-based semantic similarity, which captures high-level perceptual alignment between the stylized and original sequences, providing a task-agnostic proxy reward grounded in human-perceived semantics. These metrics correspond to the components of our reward function and serve as both training objectives and post-hoc evaluation tools, consistent with prior work in vision-based RL and video stylization.

For qualitative evaluation, we analyze representative stylized trajectories by visualizing selected keyframes and performing detailed walkthroughs of complete video rollouts. This enables a comprehensive evaluation of how well the learned policy generalizes across scenes, maintains visual coherence, and produces consistent stylization across temporally extended sequences.

5 Results

5.1 Quantitative Evaluation

Method	Dataset	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.

Method	Dataset	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 metrics between the baseline and our RL-based method.

5.1.1 Reward Component Evaluation

We begin by analyzing the individual components of the reward function on the final trajectories generated by our agent and by the baseline policy (DiffuseST). This corresponds to a post-training evaluation of the learned policy's performance with respect to the objectives it was optimized for.

As summarized in Table 1, our reinforcement learning approach outperforms the baseline across all reward components: temporal consistency, style fidelity, and content preservation. These improvements demonstrate the effectiveness of training a policy that explicitly optimizes for multi-objective rewards in a sequential decision-making setting.

Furthermore, we observe that two of the three losses reward metrics are lower on the test set than on the training set. We attribute this to the limited diversity and small size of the test videos. While this might not reflect the generalization of the results, the consistent improvement across all metrics still shows that the agent is able to learn meaningful policies that generalize across environments within the given distribution..

5.1.2 CLIP-Based Semantic Evaluation

To evaluate the perceptual and semantic quality of the learned policy's output in a task-agnostic manner, we assess the stylized sequences using CLIP similarity. This allows us to move beyond handcrafted reward terms and evaluate policy outputs in a learned, semantically grounded embedding space.

For each video frame, we extract CLIP [12] image embeddings and compute cosine similarity between the stylized frame and its corresponding content or style reference. This provides a proxy for semantic and perceptual alignment:

$$\text{CLIPSim}(x, y) = \frac{\langle f(x), f(y) \rangle}{\|f(x)\| \cdot \|f(y)\|} \tag{6}$$

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 policy achieves comparable performance to the baseline, despite operating under the added complexity of latent residual conditioning (noises) and stochastic dynamics. This suggests that the learned policy retains semantic fidelity, aligning well with human perception in the embedding space. However, we believe there is room for improvement, and a more detailed discussion of potential limitations and influencing factors is provided in Section 6.2.

5.2 Qualitative Analysis

We qualitatively compare the output trajectories generated by our learned policy to those from the baseline (DiffuseST). Despite introducing stochastic perturbations in the latent space—a step that typically challenge the agent's ability to guide the diffusion decoding process—our method achieves stylizations that are visually comparable to the baseline. This suggests that our reinforcement learning agent effectively learns meaningful policy adjustments that steer the pretrained encoder's latent representations toward desirable outputs.

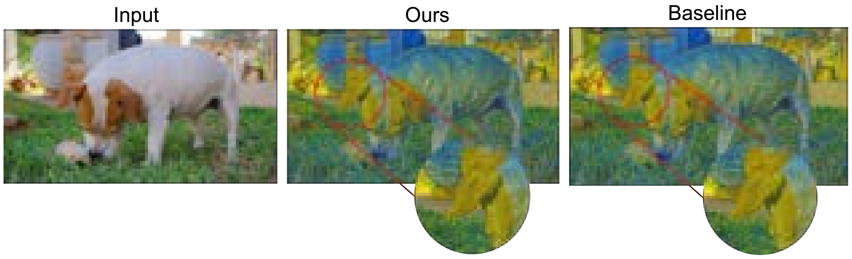


Figure 4: Zoom-in to see detailed texture comparison between our method and the baseline.

Upon closer inspection of fine-grained textures in the outputs (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 expressive artistic textures and that our agent is better able to capture and preserve.

Additional qualitative rollouts in Figure 5 highlight the coherence and fidelity achieved by our policy. These results underscore the agent's ability to generalize the learned behavior across different video sequences, capturing characteristics necessary for high-quality video stylization.

6 Discussion

6.1 Broader Impact

Our experiments demonstrate that reinforcement learning can serve as an effective framework for optimizing stylization quality in video-based tasks.As shown in Figure 6 (Appendix), the overall reward signal steadily increases over reward queries during training, while the total loss—aggregating temporal, style, and content objectives—consistently decreases. Although the absolute improvements in numerical values are small, this outcome is expected due to the scale of the individual reward terms and the limited dataset used for training.

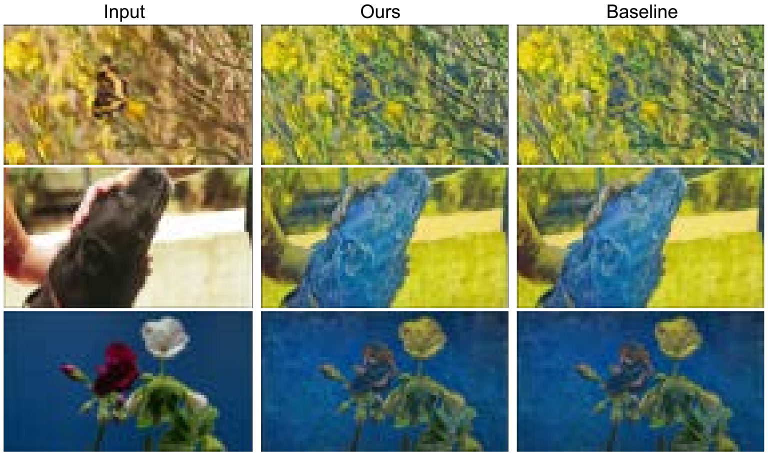


Figure 5: More examples of output from our methods and baseline

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.

Ultimately, this work lays foundational evidence for the broader application of reinforcement learning in video generation and multimodal visual learning, opening up future avenues for controllable and adaptive visual synthesis.

6.2 Limitations and Challenges

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.

Policy Gradient Instability 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 Dataset Size A key limitation of reinforcement learning algorithms–particularly those operating in high-dimensional space like video generation–is their inherent **sample inefficiency**. Learning an effective policy often requires a large number of diverse trajectories, which translates to substantial compute demands, especially when the environment involves computationally expensive components such as diffusion models. For reference, DDPO [1] performs image style transfer using reinforcement learning by collecting 256 samples per training iteration. Each training step in our framework requires forward diffusion inference, reward calculation, and policy updates, making the cost per episode relatively high.

Due to constrained computational resources, our experiments were conducted on a relatively small dataset to ensure manageable comopute. While our method successfully demonstrates qualitative improvements and reduction in temporal loss, it is likely under-optimized compared to what could be achieved in a higher-resource setting.

7 Conclusion

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 Team Contributions

- **Group Member 1:** Zhiyi Kuang: Run baseline DiffuseST, implement temporal loss, policy network architecture, policy gradient training loop, run experiments.
- **Group Member 2:** Sirui Chen: Implement content loss, policy gradient rewards and update, evaluation metrics; Integrate the whole pipeline; Run baseline and RL-base DiffuseST (ours) experiments and evaluations.

Changes from Proposal Our original hypothesis was to have a policy network that updates the parameters of the diffusion model. Our revised hypothesis is to have a policy network that outputs a distribution over the adjustment term. We realize that it is straightforward to perform modification in the latent space directly after examining our baseline diffusion model.

References

[1] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning, 2024.

[2] Chengming Feng, Jing Hu, Xin Wang, Shu Hu, Bin Zhu, Xi Wu, Hongtu Zhu, and Siwei Lyu. Controlling neural style transfer with deep reinforcement learning, 2023.

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

[4] Leon A Gatys, Alexander S Ecker, and Matthias 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.

[5] Ying Hu, Chenyi Zhuang, and Pan 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.

[6] Xun Huang and Serge 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.

[7] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Universal style transfer via feature transforms. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 386–396, 2017.

[8] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021.

[9] Hao Ouyang, Qiuyu Wang, Yuxi Xiao, Qingyan Bai, Juntao Zhang, Kecheng Zheng, Xiaowei Zhou, Qifeng Chen, and Yujun 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.

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

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

[12] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

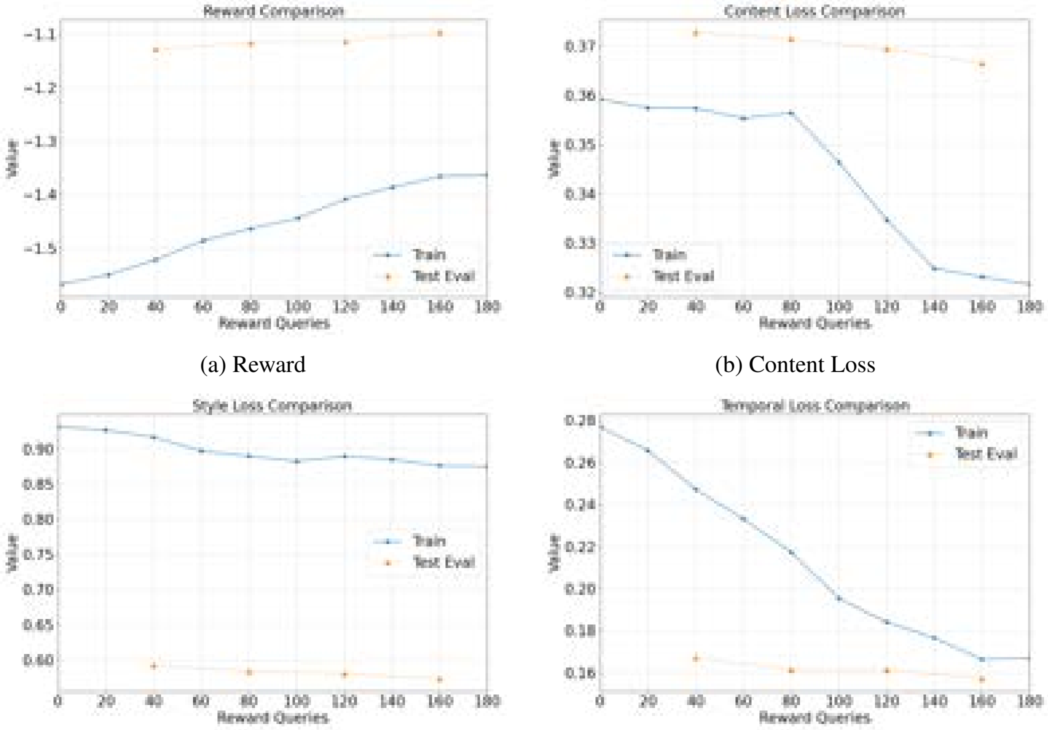
[13] Zachary Teed and Jia Deng. RAFT: recurrent all-pairs field transforms for optical flow. *CoRR*, abs/2003.12039, 2020.

[14] Wenjing Wang, Jizheng Xu, Li Zhang, Yue Wang, and Jiaying Liu. Consistent video style transfer via compound regularization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12114–12121, 2020.

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

[16] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

A Training Loss and Reward Curves.



(a) Reward

(b) Content Loss

(c) Style Loss

(d) Temporal Loss

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