Eurographics Symposium on Rendering 2025
B. Wang and A. Wilkie
(Guest Editors)

COMPUTER GRAPHICS *forum*
Volume 44 (2025), Number 4

StructuReiser: A Structure-preserving Video Stylization Method

R. Spetlik[1], D. Futschik[2], D. Sýkora[1]

[1]Czech Technical University in Prague, Faculty of Electrical Engineering, Czech Republic
[2]Google Research, USA

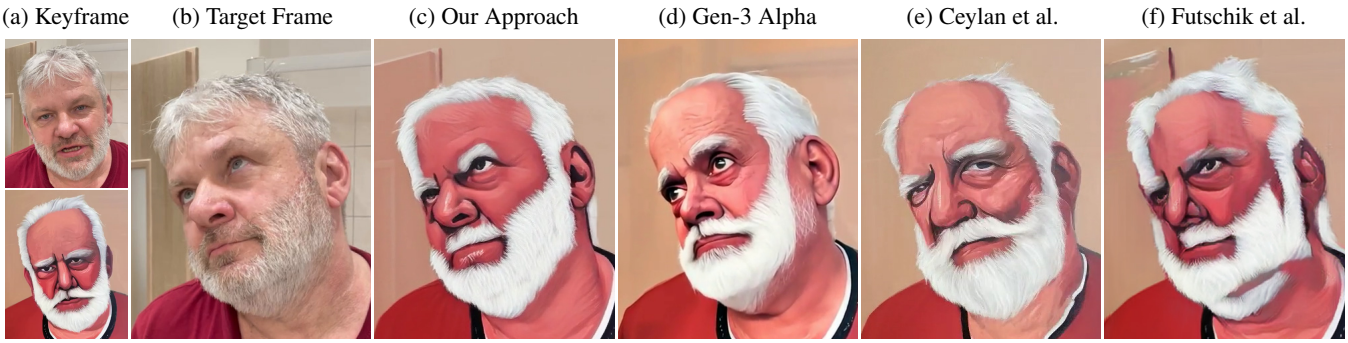


Figure 1: *StructuReiser transfers the style from a single stylized keyframe (a) to the entire video sequence (b) generating stylized frames (c) that are both stylistically consistent and structurally faithful. The keyframe (a) was created using the text-guided video-to-video diffusion model by Ceylan et al. [CHM23]. However, when applied directly to other frames in the sequence, this model often introduces significant structural inconsistencies (e). A large video model Gen-3 Alpha [Run25] introduces both structural and style inconsistencies. The state-of-the-art keyframe-based video stylization method of Futschik et al. [FKL*21] faces similar issues (f). In contrast, our approach (c) maintains the structural integrity of the target video sequence while ensuring coherent stylization throughout.*

Abstract

We introduce StructuReiser, a novel video-to-video translation method that transforms input videos into stylized sequences using a set of user-provided keyframes. Unlike most existing methods, StructuReiser strictly adheres to the structural elements of the target video, preserving the original identity while seamlessly applying the desired stylistic transformations. This provides a level of control and consistency that is challenging to achieve with text-driven or keyframe-based approaches, including large video models. Furthermore, StructuReiser supports real-time inference on standard graphics hardware as well as custom keyframe editing, enabling interactive applications and expanding possibilities for creative expression and video manipulation.

CCS Concepts
• ***Computing methodologies*** → ***Non-photorealistic rendering; Image processing;***

1. Introduction

Guided video stylization aims to modify the visual appearance of an input sequence while preserving its high-level structure and motion. Existing solutions fall largely into two groups: keyframe-based methods [JST*19, TFK*20, FKL*21] that allow users to directly manipulate visual appearance through one or more stylized keyframes, and text-driven approaches [YZLL23, CHM23, GBTBD24] where the appearance is specified using text prompts. Recently, large video models (Sora [Ope25], Veo2 [Goo25] or Gen-3 Alpha [Run25]) have been introduced, which also demonstrate capabilities to perform keyframe-based stylization (c.f. Gen-3 Alpha ReStyle).

Despite their practicality and impressive results, the mentioned keyframe-based and text-driven methods were not originally designed to preserve the content structure of the unstyled video, which can lead to structural elements being lost or degraded in the stylized output. For example, when the input video features a distinct character whose identity is crucial (see Fig. 1a), there is no guarantee that the stylized sequence will retain this identity (see Fig. 1d–e). To address this issue in keyframe-based ap-

proaches, users must provide a set of keyframes that accurately capture existing and newly appearing structural elements in the input video – a process that can be tedious and time-consuming. The challenge is more pronounced with text-driven methods, as it can be difficult to craft a prompt that reliably preserves all structural details.

To overcome this limitation, we propose a novel approach to keyframe-based video stylization. We formulate a guided video stylization problem that, in addition to ensuring fidelity to the transferred style, also focuses on preserving the structural elements of the input video (cf. Fig. 1b–c) – an aspect that has not been discussed in previous approaches. Although a similar objective was previously set in neural style transfer [GEB16] (and its video extension [RDB18]), our work targets a different scenario—semantically meaningful style transfer in which spatial alignment between original and stylized content is essential. In this context, preserving arbitrary structural elements in stylized video remains a challenging problem that we address in this paper.

To validate the effectiveness of our method, we conducted a series of qualitative and quantitative evaluations including an online user study that demonstrate the importance of explicit modeling of the appearance-to-structure relationship and demonstrate our method's ability to preserve both stylistic and structural aspects. Moreover, because our stylization method is based on a feed-forward neural network, it can perform inference in real time on commodity graphics hardware. This makes our approach suitable for interactive scenarios, such as video conferencing, where diffusion-based techniques or large video models are difficult to apply.

To summarize our contributions:

(i) We formulate a task of structurally faithful video stylization, emphasizing both semantically meaningful stylistic fidelity and structural preservation.
(ii) We provide a solution that trains a feed-forward neural network with the assistance of a pre-trained diffusion model to effectively transfer style from a stylized keyframe while preserving essential structural elements in the rest of the video sequence.
(iii) We validate our approach through extensive qualitative and quantitative evaluations, which demonstrate a significant improvement over current state-of-the-art in maintaining structural fidelity while closely adhering to the desired stylization.

Codes & models at https://github.com/radimspetlik/structureiser.

2. Related work

The origin of image and video stylization techniques can be traced back several decades. Early stylization approaches were typically based on hand-crafted algorithmic solutions that were restricted to a certain range of styles and specific target domains. For instance, Curtis et al. [CAS*97] present a physically-based simulation to mimic the appearance of a watercolor media, Salisbury et al. [SWHS97] produce painterly artworks automatically using a set of predefined brush strokes, while Praun et al. [PHWF01] can generate brush strokes procedurally. Despite the impressive results these early stylization techniques produce, their main limitation lies in the fact that slight modification of an existing style or creation of a new one usually requires a significant effort and expertise.

To overcome this limitation, Hertzmann et al. [HJO*01] introduced the idea of image analogies. In their approach, the user can provide an example pair of unstyled and stylized images that specify the intended stylization analogy. The resulting image is then constructed by copying patches from a stylized exemplar so that the corresponding pixels in the unstyled patches match the patches in the target unstyled image. The framework of image analogies later proved to be a viable solution also for example-based video stylization [BCK*13, JST*19] that can deliver temporally consistent sequences that faithfully preserve the user-specified visual style. A key limitation of those techniques is that they treat the target video as a guide for style transfer, and thus larger structural changes that may appear in the target domain are not taken into account. To overcome this limitation, the user needs to provide multiple consistently stylized keyframes, of which manual preparation can be labor intensive. When stylized keyframes are generated synthetically, it is, on the other hand, difficult to ensure their consistency.

Those limitations were addressed by Frigo et al. [FSDH16] and Gatys et al. [GEB16] who perform stylization using only the style image and try to better respect the structural changes in the target domain. Frigo et al. search for the optimal mapping between the adaptively sized patches in the target image and patches in the style-exemplar while Gatys et al. iteratively optimizes the output image so that when fed into the VGG network [SZ14] its responses correspond to VGG responses of the style exemplar and the target image. This approach inspired follow-up works [LFY*17, KSS19] of which aim is to increase the faithfulness of the generated image to the style exemplar by employing more sophisticated loss functions. Chen et al. [CLY*17] and Ruder et al. [RDB18] later demonstrated how to extend the framework of Gatys et al. to example-based video stylization delivering temporally consistent sequences. Although these approaches are fully automatic and do not require the preparation of a larger number of stylized keyframes, their artistic control over the final output is fairly limited. The transfer is usually not semantically meaningful and lacks faithfulness to the original artistic media.

To perform a semantically meaningful transfer while respecting structural changes in the target domain, image-to-image translation networks were proposed [JAFF16,IZZE17]. However, these require a large amount of training data to work reliably. Only in some domain-specific scenarios, such as portrait stylization [FCC*19] training pairs can be generated automatically [FJS*17].

To mitigate the requirement for larger paired datasets, few-shot learning approaches [LHM*19, WLT*19] and deformation-based approaches [SLT*19b, SLT*19a] were proposed. However, these methods require pretraining on large domain-specific datasets and thus are not applicable in the general case. Texler et al. [TFK*20] proposed a few-shot patch-based training strategy for which only a few stylized keyframes are necessary to deliver compelling video stylization results without the requirement of domain-specific pretraining. However, their method has limitations comparable to the image analogies approach of Jamriška et al. [JST*19], i.e., when new structural details appear in the target sequence, it is necessary to provide additional stylized keyframes. Although in the

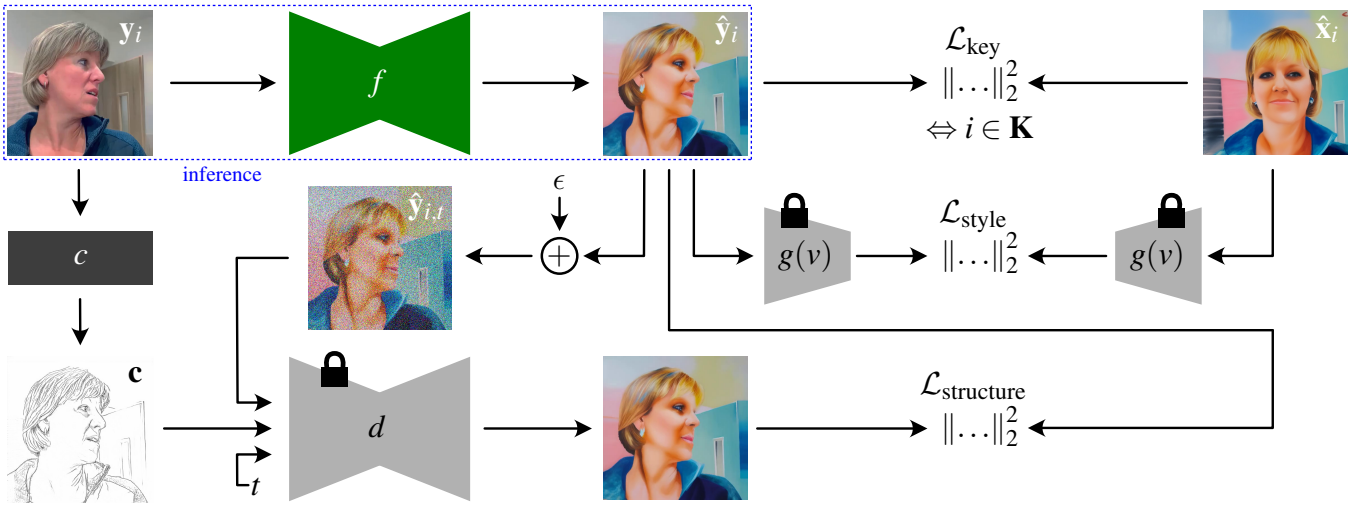


Figure 2: *An overview of our approach's training procedure. Given non-keyframe images* $\mathbf{y}_i \in \mathcal{Y}$ *(which are not stylized), we optimize the operator* f *to produce images* $\hat{\mathbf{y}}_i$ *with a similar appearance as stylized keyframes* $\hat{\mathbf{x}}_i$*. The* key *loss* $\mathcal{L}_{key}$ *(2) encourages reconstruction of keyframes* $\hat{\mathbf{x}}_i$*, the* style *loss* $\mathcal{L}_{style}$ *(3) ensures style consistency between frames and keyframes using Gram correlation matrices* g *of extracted VGG network responses* v*, and finally the* structure *loss* $\mathcal{L}_{structure}$ *(4) enforces fidelity to structural elements present in the input video frames* $\mathbf{y}_i$*. The* structure *loss requires a conditioned diffusion model* d *initialized by adding a random Gaussian noise* $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ *into the synthesized image* $\hat{\mathbf{y}}_i$*, a time step* t*, and a function* c *transforming the input image* $\mathbf{y}_i$ *to a condition* $\mathbf{c}$ *(in this case line art filter).*

follow-up work of Futschik et al. [FKL*21] the amount of required keyframes decreased significantly, the newly appearing structural changes cause difficulties as the underlying approach is still predominantly focused on style preservation.

As an alternative approach to video stylization, unwrapping techniques have been proposed [RAKRF08, KOWD21]. In those approaches, input video frames are first projected onto a static atlas where edits can be performed at one snap and later transferred back to the original video domain. The quality of results is highly dependent on the quality of the generated unwrap. For small local edits, these techniques produce impressive results, but larger changes typically cause difficulties.

Recent approaches to video stylization utilize large pre-trained text-to-image diffusion models [RBL*22] and can edit videos globally using text prompts [Kha23, ZLN*23, YZLL23, YZLL24, CHLC24, GBTBD24, KLC*24, SKL*24]. However, the outputs of these techniques heavily depend on a particular version of the pre-trained text-to-image diffusion model, which may tend to produce unpredictable results that do not consistently reproduce structural changes in the target video sequence. This can sometimes lead to a typical structural flicker that can be disturbing to the observer. Moreover, in addition to the text prompt, a specific control over the stylization process is difficult to achieve, in contrast to keyframe-based methods where the user has full creative freedom [JST*19, TFK*20, FKL*21].

Liu et al. [LXZ*24] introduced a text-to-video diffusion framework that injects pre-trained models with "diffusion adapters" for stylized generation, and further explored depth-based conditioning in their supplementary materials. While their approach can produce compelling results, it relies on training these adapters on a specialized dataset, which limits the variety of possible styles. Moreover, it does not explicitly address the notion of *semantically meaningful* style transfer – aligning the style domain with the underlying content so that key structures are preserved and enhanced. In contrast, our work aims to maintain strong content-style alignment by learning a dedicated style adapter that flexibly operates on diverse data, allowing it to capture intricate details and faithfully reproduce even niche styles. Additionally, our approach supports real-time performance once trained, making it well-suited for interactive stylization scenarios. Consequently, we emphasize both the preservation of essential structural attributes and the faithful rendering of user-specified styles, ensuring that stylized videos remain coherent and semantically aligned from beginning to end.

An online video generation tool based on large video model Gen-3 Alpha [Run25], was recently introduced, allowing users to provide a stylized first frame to guide semantically meaningful style transfer. Because it is a commercial tool with proprietary internals, the technical details remain inaccessible. Nevertheless, we provide extensive comparisons in the supplementary video, allowing readers to observe differences between its results and other published methods discussed in this paper.

3. Our Approach

The input to our method is a set of N video frames $\mathcal{T}$, which we decompose into two complementary subsets:

$$\mathcal{T} = \mathcal{X} \cup \mathcal{Y}.$$

1. *Keyframes* $\mathcal{X} = \{\mathbf{x}_i \in \mathcal{T} \mid i \in \mathbf{K}\}$, where $\mathbf{K} \subset \{1, 2, \ldots, N\}$ is the set of K selected frame indices. For each keyframe $\mathbf{x}_i$, a corresponding stylized ground truth $\hat{\mathbf{x}}_i \in \hat{\mathcal{X}}$ is provided.
2. *Non-keyframes* $\mathcal{Y} = \{\mathbf{y}_i \in \mathcal{T} \mid i \notin \mathbf{K}\}$, i.e. the remaining $N - K$ frames that do *not* have direct stylized references.

We define the set of keyframe pairs by

$$\mathcal{K} = \{(\mathbf{x}_i, \hat{\mathbf{x}}_i) \in \mathcal{X} \times \hat{\mathcal{X}} \mid i \in \mathbf{K}\},$$

which captures the K tuples of the original keyframes and their corresponding stylized counterparts.

Our goal is to learn a stylization operator

$$f \colon \mathcal{T} \to \hat{\mathcal{T}},$$

mapping each frame $\mathbf{y}_i \in \mathcal{T}$ to a stylized version $\hat{\mathbf{y}}_i = f(\mathbf{y}_i) \in \hat{\mathcal{T}}$. In practice, one may view f as an element of a suitable function space, such as a subset of $L^2(\Omega)$ if $\Omega \subset \mathbb{R}^2$ denotes the spatial domain of frames, or a more sophisticated reproducing kernel Hilbert space for higher-level feature embeddings. We train f by minimizing the following objective:

$$\mathcal{L}(\mathcal{K}, \mathcal{Y}) = \lambda_k \mathcal{L}_{\text{key}}(\mathcal{K}) + \lambda_v \mathcal{L}_{\text{style}}(\mathcal{K}, \mathcal{Y}) + \lambda_s \mathcal{L}_{\text{structure}}(\mathcal{Y}), \quad (1)$$

where each term imposes distinct yet complementary constraints.

Keyframe Reconstruction Loss We begin with a *reconstruction loss* $\mathcal{L}_{\text{key}}$, which enforces fidelity on the keyframes:

$$\mathcal{L}_{\text{key}}(\mathcal{K}) = \frac{1}{K} \sum_{i \in \mathbf{K}} \left\| f(\mathbf{x}_i) - \hat{\mathbf{x}}_i \right\|_2^2. \quad (2)$$

Here, $\|\cdot\|_2$ denotes the ℓ^2-norm in the image (or feature) space. By pairing $\mathbf{x}_i$ with $\hat{\mathbf{x}}_i$, this term encourages f to replicate the specific user-defined style on each keyframe.

Style Loss To ensure consistent stylization across the non-keyframes, we incorporate a *style loss* $\mathcal{L}_{\text{style}}$ inspired by Gatys et al. [GEB16]. For each non-keyframe $\mathbf{y}_j$, we compare Gram matrices of VGG [SZ14] features between $f(\mathbf{y}_j)$ and the *ground-truth stylized* references $\{\hat{\mathbf{x}}_i\}$. Specifically, defining

$$g(\mathbf{u}, l) = \text{Gram}\big(\phi_l(\mathbf{u})\big),$$

where ϕ_l is the activation map of the l-th layer in VGG and $\text{Gram}(\cdot)$ computes a normalized correlation matrix, we write:

$$\mathcal{L}_{\text{style}}(\mathcal{K}, \mathcal{Y}) = \frac{1}{|\mathbf{K}|\,|\mathbf{L}|\,|\mathcal{Y}|} \sum_{i \in \mathbf{K}} \sum_{j \notin \mathbf{K}} \sum_{l \in \mathbf{L}} \left\| g(\hat{\mathbf{x}}_i, l) - g(f(\mathbf{y}_j), l) \right\|_2^2, \quad (3)$$

where $\mathbf{L}$ is the set of VGG layers used. This term enforces that textural and color statistics from the keyframe stylization are inherited by all non-keyframe outputs.

Structure Preservation via Diffusion Models A key element of our pipeline is the *structure loss* $\mathcal{L}_{\text{structure}}$, which preserves important geometric and semantic features from the input frames $\mathbf{y}_i$. Specifically, we employ a pre-trained diffusion model d conditioned by a function c designed to extract visually salient details such as edges, outlines, or semantic cues. This choice of conditioning helps maintain the integrity of shapes and objects that are crucial for *semantically meaningful* style transfer, where the target style and content naturally align.

Let $\hat{\mathbf{y}}_{i,t}$ be a noisy version of $f(\mathbf{y}_i)$ by adding Gaussian noise ϵ at a predefined time step t. Formally:

$$\hat{\mathbf{y}}_{i,t} = \sqrt{\bar{\alpha}_t}\, f(\mathbf{y}_i) + \sqrt{1 - \bar{\alpha}_t}\, \epsilon, \quad \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}),$$

where $\bar{\alpha}_t \in (0,1)$ quantifies how much noise is injected at step t. The diffusion model $d\big(\hat{\mathbf{y}}_{i,t}, c(\mathbf{y}_i), t\big)$ predicts the portion of noise that is *incompatible* with the underlying structure in $\mathbf{y}_i$. Consequently, the structure loss is defined as:

$$\mathcal{L}_{\text{structure}}(\mathcal{Y}) = \frac{1}{|\mathcal{Y}|} \sum_{i \notin \mathbf{K}} \left\| d\big(\hat{\mathbf{y}}_{i,t}, c(\mathbf{y}_i), t\big) - \epsilon \right\|_2^2. \quad (4)$$

Minimizing $\|d(\hat{\mathbf{y}}_{i,t}, c(\mathbf{y}_i), t) - \epsilon\|^2$ essentially *projects* $f(\mathbf{y}_i)$ onto a manifold of images consistent with the conditioning $c(\mathbf{y}_i)$. As a result, our stylized frames respect the layout and contours implied by the original video, preserving the semantic integrity needed for high-quality style transfer.

Theoretical Underpinnings of the Structure Term Although the above can be seen as a gradient-based alignment, it is more precisely viewed as a *manifold projection* that keeps $f(\mathbf{y}_i)$ close to images that match the structural content of $\mathbf{y}_i$. The operator d is a learned approximation of the reverse diffusion process, which recovers a clean sample from a noisy version under the constraint of conditioning c. By embedding this denoising operator into our loss, we force the stylized frames $\{f(\mathbf{y}_i)\}$ to remain structurally faithful to $\{\mathbf{y}_i\}$ in ways simpler pixel- or gradient-based losses cannot (see Sec. 5.5 and Fig. 18).

Mathematically, one can regard the denoising step $d(\hat{\mathbf{y}}_{i,t}, c(\mathbf{y}_i), t)$ as seeking a fixed point of an operator

$$\mathcal{D}_{c,t} \colon \mathcal{T} \times \Omega \to \mathcal{T},$$

where Ω encodes external conditioning information (in our case, line art, edges, or semantic cues). The convergence to the correct structural manifold emerges from iterating $\mathcal{D}_{c,t}$ at different noise levels t. By integrating this denoising operator into a loss term, we ensure that the stylized results $\{f(\mathbf{y}_i)\}$ are *pulled* to preserve the geometry of $\{\mathbf{y}_i\}$.

Choice of Conditioning and Flexibility In Fig. 2, we instantiate d as a diffusion model [ZRA23] conditioned on a line art filter c. Other conditioning strategies – such as Canny edges, semantic segmentation, or domain-specific keypoint detectors – can also be used. Changing c redefines which aspects of $\mathbf{y}_i$ qualify as "structure," making the method adaptable to diverse tasks and stylistic preferences (see Sec. 5.4 and Fig. 17).

Why a Diffusion-Based Structural Constraint? By leveraging the learned prior from a generative diffusion model, we gain two important benefits over more direct edge or gradient constraints:

- **High-Level Consistency.** Diffusion models are trained on large datasets of natural images (or sketches), so they capture not just local gradients but also global structures like object boundaries and semantic relationships.
- **Adaptive Gradient Signal.** The denoiser d provides gradients that actively push stylized images toward realistic and coherent shapes, rather than enforcing a single-scale edge or pixel match. This approach better handles strong stylizations while keeping structural details intact.

Definition of "Structure" Throughout this work, we use "structure" to refer to prominent spatial or semantic features such as boundaries, object shapes, or scene layouts. In practice, the definition of structure is governed by the choice of c and the training corpus of d. Adjusting c can emphasize fine contours, coarse silhouettes, or specialized annotations, allowing the preservation of precisely those elements deemed most critical for semantically meaningful stylization.

This formulation highlights that our structure-preservation principle emerges from a rigorous viewpoint: we interpret the stylized image $f(\mathbf{y}_i)$ as a point constrained to lie on a manifold of structurally coherent solutions, with the diffusion model d serving as a learned projection operator guided by c. By positioning the problem in function spaces and ensuring well-defined gradient flows from generative priors, our approach transcends the limitations of simpler, purely local constraints.

Our proposed optimization scheme leverages a pretrained diffusion model for training regularization, drawing on the principle introduced in score distillation sampling (SDS) by Poole et al. [PJBM23]. A key novelty of our approach lies in the way we inject structural details while preserving the unique traits of the given artistic style. Such functionality is difficult to achieve using the original SDS formulation or using the conditioned diffusion model [ZRA23] directly, as these rely solely on the learned prior, and therefore struggle with custom visual styles that are typically out of the domain on which the diffusion model was trained.

3.1. Implementation details

We implemented our approach in PyTorch [PGM*19] using the AdamW [LH19] optimizer with fixed learning rate $3 \cdot 10^{-5}$. To model the stylization operator f, we adopt the network architecture originally proposed by Futschik et al. [FCC*19] that proved to be suitable for style transfer tasks [TFF*20, FKL*21] allowing for reproduction of important high-frequency details, critical for generating complex and believable artistic styles. The batch size was set to 1 and instance normalization [UVL16] used instead of batch normalization. We set $\lambda_k = 1.0$ and $\lambda_v = 100.0$. The parameters for $\mathcal{L}_{\text{structure}}$ (4) were selected experimentally and differ between the presented sequences: $\lambda_s \in \{10^{-5}, 10^{-6}\}$ and $t \in \{20, 28\}$. As d, we adopt the default noise scheduler of ControlNet v1.1 [ZRA23] and the UniPC scheduler [ZBR*24] with the number of steps set to 30. The parameters of the VGG network and diffusion model were fixed and line art conditioning was used for c (if not stated otherwise). The training was performed on a single NVIDIA A100 GPU with 40 GB of RAM for 4 hours and the model with the lowest total loss has been selected to produce the stylized sequences. In practice, however, even notably shorter training times can lead to compelling results (see Sec. 5.1).

4. Results and Comparison

The results are presented in Figures 1, 3, 4, 6, 7, 8, and 9. See also our supplementary material for additional results. We compare them with the output of recent text-driven methods [CHM23, YZLL23, CHLC24, GBTBD24] and keyframe-based approaches [JST*19, TFK*20, FKL*21, Run25].

4.1. Perceptual study

To qualitatively evaluate our approach, we conducted a perceptual study comparing the outputs of our method with those of five state-of-the-art text-driven techniques [CHM23, YZLL23, CHLC24, GBTBD24, YZLL24], three state-of-the-art keyframe-based methods [JST*19, TFK*20, FKL*21], and a large video model [Run25]. The study assessed how well each method reproduced the artistic style, preserved structural content, and maintained temporal consistency of the input video. We collected data from 55 participants through an online survey, where participants were presented with randomized two-alternative forced-choice (2AFC) comparisons. Each participant completed 36 questions, selecting which anonymized stylization better reproduce style (12 questions), preserve content (12 questions), and maintain temporal consistency (12 questions). In each comparison, an output from our method was paired with one from another method using the same input data.

The preference scores for our method versus others are presented in Fig. 10 as a colored heatmap, where dark green denotes 100% of the participants who prefer our method and dark red denotes 0%. Each row corresponds to a different method, and each column corresponds to one of the evaluation criteria: *Structure*, *Style*, and *Temporal Consistency*. Our method offers improved performance over previous works in reproducing input structures, even though it may reproduce the exemplar's style slightly less accurately, which is expected due to our focus on structural preservation. Moreover, our method demonstrates improved temporal consistency in stylized output compared to previous approaches. This is another benefit of our approach: Its ability to preserve structural details helps to ensure temporal consistency. When structures in the target video are consistent, their output stylization will be consistent as well implicitly.

4.2. Quantitative evaluation

To quantitatively evaluate our method's ability to preserve structural elements from input videos, we conducted a comprehensive comparison across published video stylization techniques. We calculated the averages and standard deviations between corresponding input and stylized frames across all available video sequences, using three metrics: SSIM, LPIPS [ZIE*18], and ꟻLIP [ANAM*20]. The style exemplars were sourced from Ceylan et al. [CHM23], Geyer et al. [GBTBD24], Yang et al. [YZLL23, YZLL24], Chu et al. [CHLC24], and a combination including artist-created stylizations.

Since the published text-driven methods cannot perform keyframe-based stylization, we trained our method using a single style exemplar selected from the output of each diffusion method. We then compared the results of our method with those of the text-driven methods, as shown in the first four groups of results in Tab. 1. In the last group, we compare our method with the keyframe-based stylization approaches of Jamriška et al. [JST*19], Futschik et al. [FKL*21], Texler et al. [TFK*20], and Gen-3 Alpha* [Run25] using a set of text-driven and artist-created exemplars. Our method demonstrates superior performance across all sequence groups and metrics, effectively maintaining structural fidelity across diverse style sources.

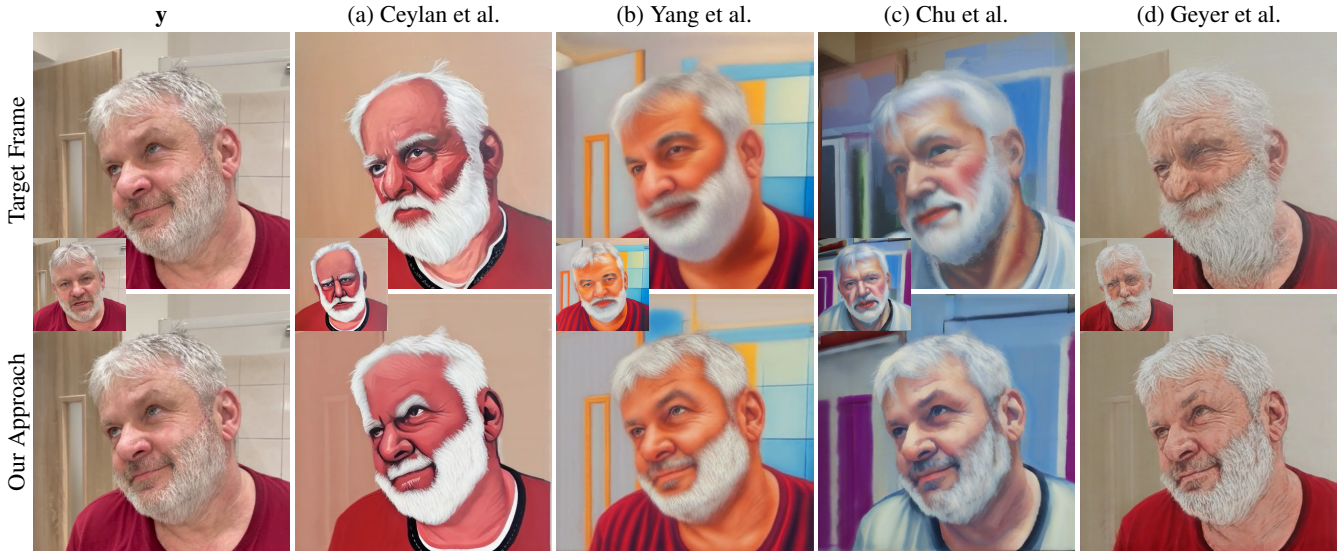


Figure 3: *Comparison with the state-of-the-art in text-driven video stylization: The target video sequence (see a representative target frame* $\mathbf{y}$*) has been stylized using text-driven approaches (top row): (a) Ceylan et al. [CHM23], (b) Yang et al. [YZLL23], (c) Chu et al. [CHLC24], and (d) Geyer et al. [GBTBD24]. One frame from those stylized sequences was used as a keyframe (see small insets). The style of this keyframe has been propagated to the rest of the target sequence* $\mathbf{y} \in \mathcal{Y}$ *using our approach (bottom row). Note how our approach better preserves the structural details seen in the target frame. See also our supplementary video to compare consistency across the entire sequence.*

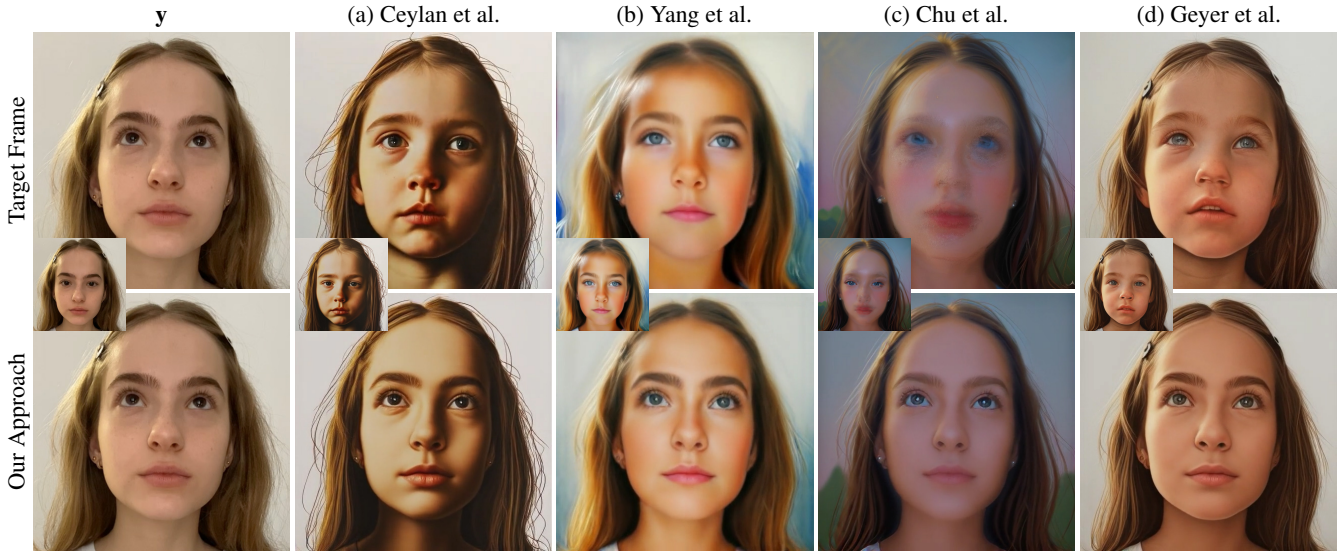


Figure 4: *Comparison with the state-of-the-art in text-driven video stylization (cont.): See Fig. 3 for a detailed explanation.*

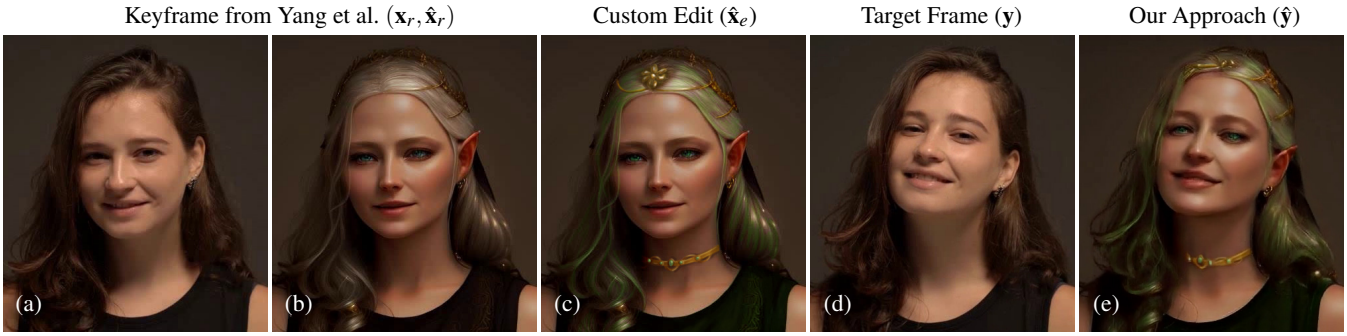


Figure 5: *Custom edit of results generated by the text-driven method of Yang et al. [YZLL23]. Left to right: (a, b) keyframe $(\mathbf{x}_r, \hat{\mathbf{x}}_r)$ is the result of Yang et al. [YZLL23] conditioned with the text prompt "Galadriel, the royal Elf, silver-golden hair," (c) custom edit $\hat{\mathbf{x}}_e$ of the stylized keyframe $\hat{\mathbf{x}}_r$, (d) target frame $\mathbf{y}$, (e) stylization $\hat{\mathbf{y}}$ produced by our method with a single keyframe $(\mathbf{x}_r, \hat{\mathbf{x}}_e)$. A key advantage of our method is that it allows custom edits of videos stylized by text-driven methods, which typically offer only limited control over the generated results through textual prompts.*

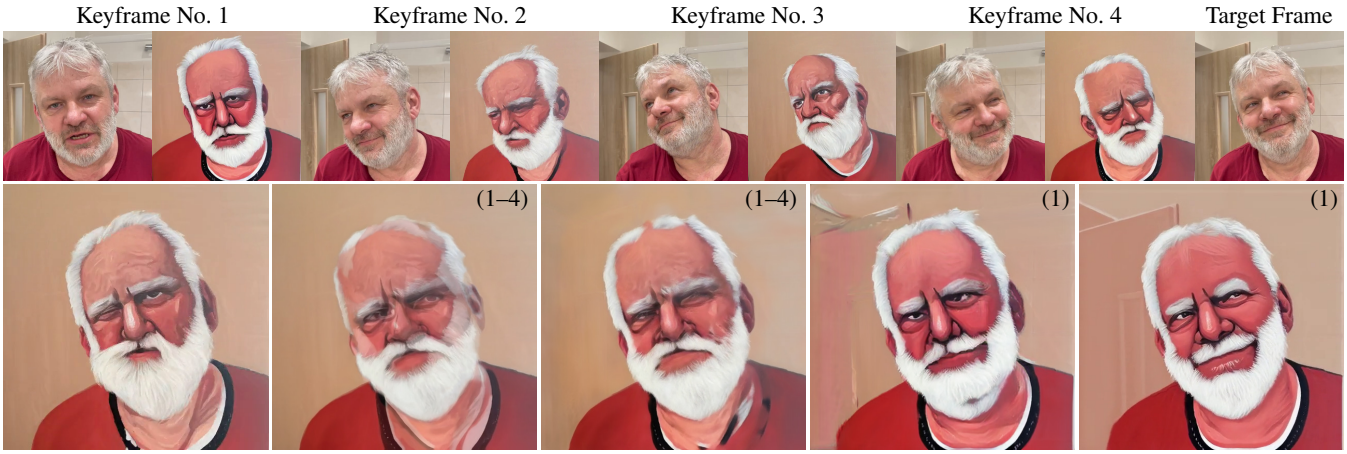


Figure 6: *Comparison with the state-of-the-art in keyframe-based video stylization: The text-driven method of Ceylan et al. [CHM23] has been used to generate a stylized sequence (a) from which four keyframes (No. 1–4) were selected to perform video stylization using methods of Jamriška et al. [JST*19] (b) and Texler et al. [TFK*20] (c), and one keyframe (No. 1) was selected for the method of Futschik et al. [FKL*21] (d) and for our approach (e). Note how our approach better preserves the structural details seen in the target frame. In our supplementary video, it is also visible that our approach keeps the structure consistent.*

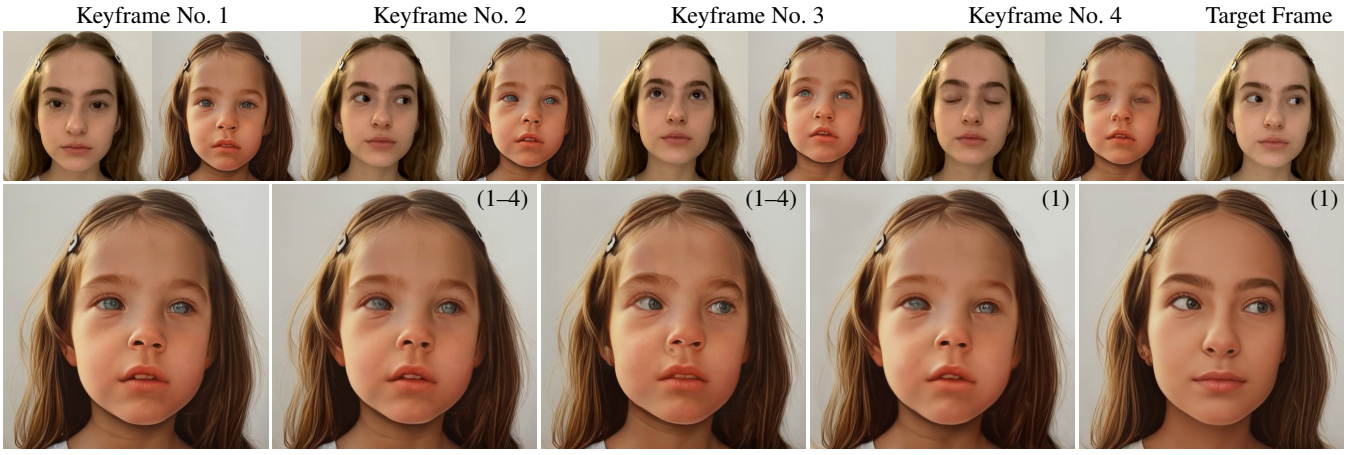
4.3. Comparison with text-driven methods

Since text-driven methods only support textual guidance and cannot perform keyframe-based stylization directly, we adopt a two-step approach for each video sequence. First, we generate a stylized version using a combined text prompt $\mathbf{p} = \mathbf{p}_{\text{edit}}\mathbf{p}_{\text{desc}}$, where $\mathbf{p}_{\text{edit}}$ (e.g., "hyperrealistic detailed oil painting of") defines the desired style, and $\mathbf{p}_{\text{desc}}$ (e.g., "an old man with a white beard") describes the content. For instance, $\mathbf{p}$ = "hyperrealistic detailed oil painting of an old man with a white beard". From the resulting stylized sequence, we then select one frame as a keyframe and propagate this keyframe's style throughout the entire target sequence $\mathbf{y} \in \mathcal{Y}$ using our proposed keyframe-based stylization method.

As a result, each text-driven method is presented with a unique stylization. Note that in the methods proposed by Ceylan et al. [CHM23] and Geyer et al. [GBTBD24], the context description $\mathbf{p}_{\text{desc}}$ serves as an inversion prompt.

From the results presented in Figures 3, 4, 8 and in our supplementary material it is apparent that our approach maintains structural details better than text-driven approaches. See also our supplementary video that demonstrates structural consistency across the entire sequence contrasting the flicker common in text-driven methods.

A key advantage of our approach in contrast to text-driven techniques is that it enables the user to incorporate custom edits by manually modifying the output of the text-driven method and use it as a newly stylized keyframe for training. This option is beneficial especially in cases when it is difficult to find an accurate text prompt that precisely expresses the desired artistic vision. In



(a) Geyer et al. (b) Jamriška et al. (c) Texler et al. (d) Futschik et al. (e) Our Approach

Figure 7: *Comparison with the state-of-the-art in keyframe-based video stylization (cont.): Text-driven method of Geyer et al. [GBTBD24] has been used to generate the initial stylized sequence (a). See Fig. 6 for a detailed explanation.*

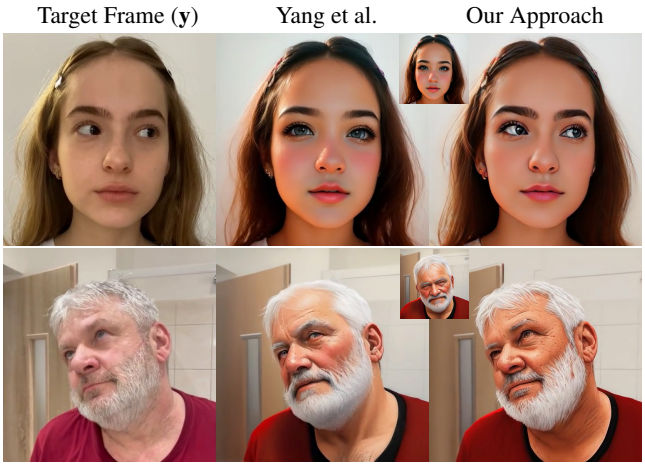


Figure 8: *Comparison with the state-of-the-art in text-driven video stylization (cont.): The target video sequence (see a representative target frame* $\mathbf{y}$*) has been stylized using text-driven approach of Yang et al. [YZLL24] (middle). One frame from those stylized sequences was used as a keyframe (see small insets). The style of this keyframe has been propagated to the rest of the target sequence* $\mathbf{y} \in \mathcal{Y}$ *using our approach (right). Note how our approach better preserves the structural details seen in the target frame. Also, see our supplementary video to compare consistency across the entire sequence.*

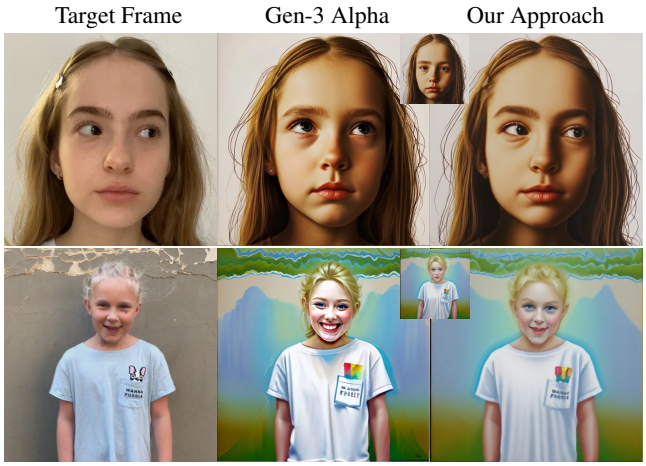


Figure 9: *Comparison with the state-of-the-art in keyframe-based video stylization (cont.): The text-driven method of Ceylan et al. [CHM23] (top row) and Chu et al. [CHLC24] (bottom row) and have been used to generate a stylized sequences from which one keyframe was selected (see small insets) to perform video stylization using large video model Gen-3 Alpha [Run25] (middle). Note how our approach (right) better preserves the structural details seen in the target frame (left) as well as the style in the given keyframe.*

Fig. 5, we show the results of a custom edit of an image $\hat{\mathbf{x}}_r$ stylized by the method of Yang et al. [YZLL23] with the text prompt "Galadriel, the royal Elf, silver-golden hair." This image was edited by an artist producing the image $\hat{\mathbf{x}}_e$. Our method was then trained with the keyframe $(\mathbf{x}_r, \hat{\mathbf{x}}_e)$, rendering the stylization $\hat{\mathbf{y}}$.

4.4. Comparison with keyframe-based methods

To compare our method with other keyframe-based approaches (see Figures 6, 7, 9, and our supplementary material), we used sequences generated by text-driven methods. From each stylized sequence, we selected a keyframe to train different methods: four keyframes were chosen for the methods of Jamriška et al. [JST*19] and Texler et al. [TFK*20], while one keyframe was used for the method of Futschik et al. [FKL*21], the large video model Gen-3

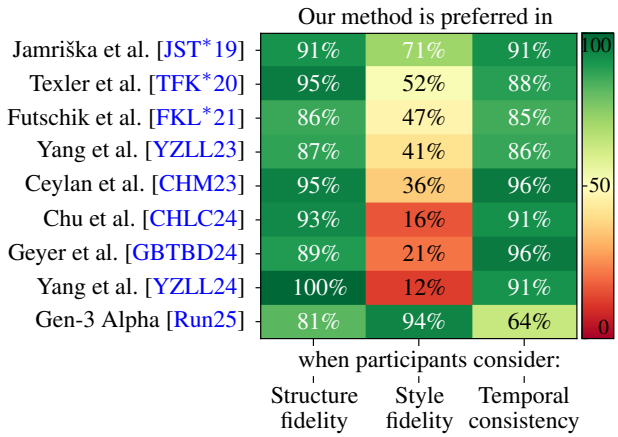


Figure 10: *Perceptual study. Each cell represents the percentage of votes preferring the results of our method over those of other methods, based on responses from a total of* 55 *participants. Comparisons were made against three keyframe-based methods – Jamriška et al. [JST*19], Texler et al. [TFK*20], and Futschik et al. [FKL*21], five text-driven methods – Yang et al. [YZLL23, YZLL24], Ceylan et al. [CHM23], Chu et al. [CHLC24], Geyer et al. [GBTBD24], and a large video model Gen-3 Alpha [Run25]. The heatmap illustrates that our approach offers improved performance over previous methods in reproducing input structures and maintaining temporal consistency. It is noteworthy that our approach fell below the 25% preference mark for style preservation in only three of the nine comparisons, which appears somewhat counterintuitive given our primary emphasis on structural fidelity.*

Alpha [Run25], and our approach. In all the results presented, our approach demonstrates stronger preservation of structural details of the target frame $\mathbf{y}$ while faithfully replicating important style features of the stylized keyframe. Please refer to our supplementary video to compare the structural consistency across the entire sequence.

5. Experiments

In this section, we present a set of experiments that provide better insight into how our approach performs in various settings. We first examine its convergence rate (Sec. 5.1), then we present an ablation study on our loss components (Sec. 5.2). We investigate the influence of parameters λ_s and t (Sec. 5.3) and conditioning prior $\mathbf{c}$ (Sec. 5.4). Finally, we compare the results of training with $\mathcal{L}_{\text{structure}}$ and without it using only the line art guidance function $c(\mathbf{y})$ (Sec. 5.5).

5.1. Convergence rate

In the first experiment, we explore the optimization convergence speed of our method. The results presented in Fig. 11 indicate that a reasonably stylized output could be obtained after 6 minutes of training, and training for more than 90 minutes does not bring a significant improvement in stylization quality. Once the network has been trained, it is capable of performing a real-time stylization of a live video stream (see Fig. 12 and our supplementary videos).

	SSIM ↑	LPIPS ↓	FLIP ↓
text-driven methods			
Geyer et al. [GBTBD24]	0.72 ± 0.10	0.30 ± 0.08	0.31 ± 0.05
ours	**0.75** ± 0.12	**0.28** ± 0.09	**0.30** ± 0.06
Yang et al. [YZLL23]	0.65 ± 0.09	0.42 ± 0.09	0.47 ± 0.08
ours	**0.71** ± 0.10	**0.37** ± 0.09	**0.43** ± 0.08
Yang et al. [YZLL24]	0.64 ± 0.09	0.34 ± 0.07	0.37 ± 0.05
ours	**0.67** ± 0.12	**0.32** ± 0.08	**0.34** ± 0.07
Ceylan et al. [CHM23]	0.67 ± 0.08	0.41 ± 0.10	0.42 ± 0.10
ours	**0.71** ± 0.09	**0.38** ± 0.10	**0.40** ± 0.10
Chu et al. [CHLC24]	0.56 ± 0.07	0.55 ± 0.07	0.61 ± 0.10
ours	**0.62** ± 0.07	**0.48** ± 0.10	**0.59** ± 0.11
keyframe-based methods			
Jamriška et al. [JST*19]	0.67 ± 0.08	0.42 ± 0.11	**0.48** ± 0.08
Texler et al. [TFK*20]	0.68 ± 0.08	0.42 ± 0.11	**0.48** ± 0.08
Futschik et al. [FKL*21]	0.66 ± 0.10	0.43 ± 0.11	0.49 ± 0.08
Gen-3 Alpha [Run25]	0.54 ± 0.06	0.50 ± 0.09	0.56 ± 0.09
ours	**0.69** ± 0.10	**0.40** ± 0.11	**0.48** ± 0.08

Table 1: *Quantitative comparison of structural fidelity. Averages and standard deviations between input and stylized frames shown in two groups. Since text-driven methods cannot perform keyframe-based stylization, we trained our method using style exemplars generated by these methods for the top group (see Sec. 4.2). The bottom group includes keyframe-based approaches trained on a union of the diffusion-generated and artist-created exemplars. Our method achieves consistently strong performance across all evaluated metrics.*

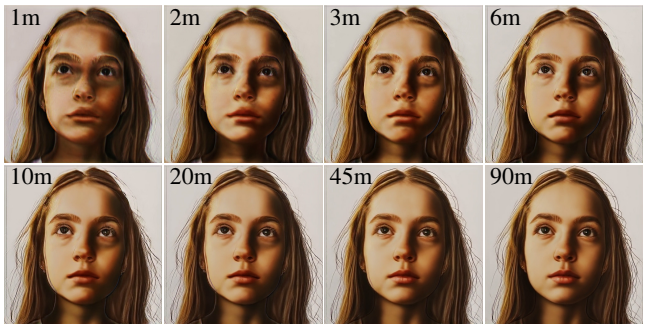


Figure 11: *Convergence speed of our method. Results captured at* 1, 2, 3, 6, 10, 20, 45, *and* 90 *minutes of training reveal the progressive improvement of the stylization. Reasonable results are retrieved after* 6 *minutes of training, with no notable improvement beyond* 90 *minutes on NVIDIA A100 GPU.*

This gives our method a key advantage over the text-driven methods [CHM23, YZLL23, CHLC24, GBTBD24, YZLL24], which require seconds of computation per frame on a single NVIDIA A100 GPU (see Tab. 2).

In Fig. 13, we compare the stylization results of our method (Figures 13a–c, blue color plot) with Futschik et al. [FKL*21] (Figures 13d–f, orange color plot) after 30, 60, and 90 minutes of

	Method	Avg. Time (s/frame)	Frames Per Second
text-driven	Ceylan et al. [CHM23]	6.439	0.16
	Yang et al. [YZLL23]	4.933	0.20
	Yang et al. [YZLL24]	2.956	0.34
	Geyer et al. [GBTBD24]	7.301	0.14
	Chu et al. [CHLC24]	1.402	0.71
keyframe-based	Gen-3 Alpha* [Run25]	0.238	4.20
	Jamriška et al. [JST*19]	0.062	16.12
	Texler et al. [TFK*20]	0.031	32.26
	Futschik et al. [FKL*21]	0.030	33.33
	ours	0.032	31.25

Table 2: *Average generation times (in seconds per frame) and corresponding frames per second for stylized frame rendering on an NVIDIA A100 GPU. All methods were executed in their native environment. The commercial tool Gen-3 Alpha* [Run25] in Turbo mode was running on the server side (unknown conditions). Our results were obtained using a simple non-optimized Python script.*

Figure 12: *Our approach enables real-time identity-preserving stylization of live video streams (see our supplementary video).*

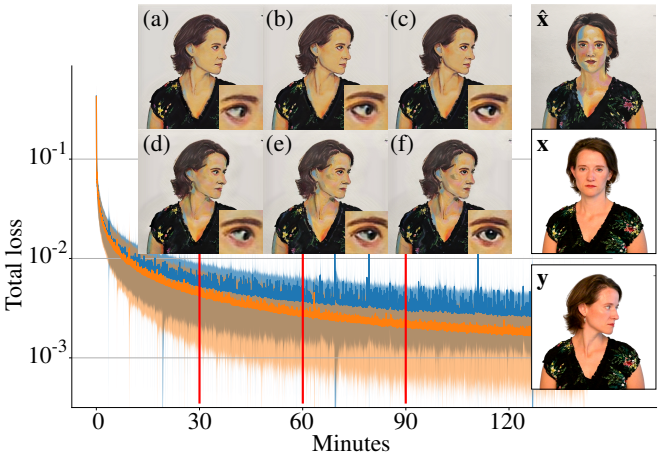


Figure 13: *Comparative analysis of stylization results over training duration. In each of the first three columns, a stylized target frame* **y** *is shown after* 30, 60, *and* 90 *minutes of training. Top row (a), (b), (c) and blue color: our method, bottom row (d), (e), (f) and orange color: Futschik et al. [FKL*21]. The method of Futschik et al. prioritizes the preservation of the original style. However, over time, this leads to a gradual loss of fine details in the structure of the target sequence (cf. zoom-in insets). A key advantage of our method is that we explicitly strive to retain these details, thereby maintaining their fidelity. The plots of the total loss are averaged over* 15 *different sequences, the border of the opaque area depicts the measured standard deviation.*

training (plots are averages over 15 different sequences; the border of the opaque area is the standard deviation). The approach of Futschik et al. [FKL*21] emphasizes the preservation of the original style. However, this emphasis results in a gradual loss of fine details within the structure of the target sequence, as illustrated by the zoom-in insets in Fig. 13. Note how the eye in Fig. 13f is directly copied from the stylized keyframe $\mathbf{y}_k$. A key advantage of our approach is that we explicitly retain these fine structural details, thus maintaining fidelity to the target frames **y**.

5.2. Ablation study on our loss components

In this experiment, we performed an ablation study to analyze the impact of individual loss terms within the optimization of our model. We systematically set the multiplication constant of each of the three loss terms λ_k, λ_v, and λ_s in Eq. (1) to zero. The results are presented in Fig. 14 for the model with the lowest total loss after training on a GPU for 4 hours. We specifically selected an input frame **x** with a significant appearance change with respect to the style exemplar **y** to highlight the ability of our method to produce results in the style of the exemplar even when new, previously unseen structures appear in the input frame ($t = 28$, $\lambda_{\text{structure}} = 5 \times 10^{-6}$).

Excluding $\mathcal{L}_{\text{key}}$ from the loss in Fig. 14a results in an output that preserves the structure of the target frame **x**; however, the transfer is not semantically meaningful – colors from the stylized keyframe **y** are placed in improper locations (cf. white spots on the nose and forehead).

When the term $\mathcal{L}_{\text{style}}$ is omitted (Fig. 14b), we obtain a result with the overall structure of **x** but with corrupted style details because $\mathcal{L}_{\text{structure}}$ enforces structure without maintaining the fine details that are otherwise preserved by $\mathcal{L}_{\text{style}}$.

Conversely, omitting $\mathcal{L}_{\text{structure}}$ (Fig. 14c) retains the style details of **y** through $\mathcal{L}_{\text{style}}$ and the semantic consistency enforced by $\mathcal{L}_{\text{key}}$, but newly appearing structures (e.g., novel eye shapes) are poorly reconstructed.

Only by combining the three loss terms (Fig. 14d) do we obtain results that simultaneously preserve structural integrity and faithfully transfer style features.

Quantitative comparison. Table 3 presents the quantitative assessment of structural fidelity across 18 test sequences. For all metrics considered, adding the structural term $\mathcal{L}_{\text{structure}}$ consistently improves similarity between input and stylized frames, confirming the visual trends observed above.

Figure 14: *An ablation study demonstrating the importance of individual terms in our objective function (1). A neural network is trained to transfer the style from the stylized keyframe* $\mathbf{y}$ *to the target frame* $\mathbf{y}$*. Each of the three terms* $\mathcal{L}_{key}$ *(2),* $\mathcal{L}_{style}$ *(3), and* $\mathcal{L}_{structure}$ *(4) is left out in the training. Leaving out* $\mathcal{L}_{key}$ *(a) causes the style transfer to be less semantically meaningful (see, e.g., white nose), excluding* $\mathcal{L}_{style}$ *(b) leads to complete style washout, and leaving out* $\mathcal{L}_{structure}$ *(c) results in poor replication of target frame structures. Only the combination of all three terms yield satisfactory results (d).*

	SSIM ↑	LPIPS ↓	ƎLIP ↓
$\mathcal{L}_{\text{style}} + \mathcal{L}_{\text{structure}}$	0.72 ± 0.01	0.37 ± 0.01	0.44 ± 0.01
$\mathcal{L}_{\text{key}} + \mathcal{L}_{\text{structure}}$	0.71 ± 0.02	0.42 ± 0.01	0.43 ± 0.01
$\mathcal{L}_{\text{key}} + \mathcal{L}_{\text{style}} + \mathcal{L}_{\text{structure}}$	0.72 ± 0.02	0.36 ± 0.01	0.45 ± 0.01
average with $\mathcal{L}_{\text{structure}}$	$\mathbf{0.71} \pm 0.01$	$\mathbf{0.38} \pm 0.01$	$\mathbf{0.44} \pm 0.01$
$\mathcal{L}_{\text{key}} + \mathcal{L}_{\text{style}}$	0.64 ± 0.01	0.43 ± 0.01	0.48 ± 0.01

Table 3: *Loss ablation – quantitative comparison of structural fidelity. We report the mean ± standard deviation of structural similarity between input and stylized frames across 18 sequences for each ablated loss variant. All evaluated metrics consistently show that including the term* $\mathcal{L}_{structure}$ *enhances preservation of the original video structures.*

Perceptual user study. To assess perceived quality, we conducted a two-alternative forced-choice (2AFC) user study with 28 participants (see Fig. 15). In each trial, subjects were shown a pair of stylized frames – one produced with the full loss $\mathcal{L}_{\text{structure}} + \mathcal{L}_{\text{style}} + \mathcal{L}_{\text{key}}$ and the other with an ablated variant – and asked which they preferred. The full loss was favored in 84 % of comparisons against $\mathcal{L}_{\text{structure}} + \mathcal{L}_{\text{style}}$, 88 % against $\mathcal{L}_{\text{key}} + \mathcal{L}_{\text{style}}$, and 100 % against $\mathcal{L}_{\text{structure}} + \mathcal{L}_{\text{key}}$. These results, averaged over all participants and scenes, corroborate the quantitative findings: users prefer outputs generated with the complete loss formulation.

5.3. The influence of parameters λ_s and t

In this experiment, we manipulate the parameter t in $\mathcal{L}_{\text{structure}}$ (4) and the loss weight λ_s (1). We trained the network for 4 hours on a GPU and selected the network with the lowest total loss.

The results in Fig. 16 show that the stronger λ_s, the more pronounced the structure of $\mathbf{x}$ is in the result. Since the value of λ_s modifies the strength of $\mathcal{L}_{\text{structure}}$ in the total loss term $\mathcal{L}$ (1), setting it to a higher value results in a stronger pronunciation of the input video structures in the output of our method. Another interesting dimension of structure reinforcement control strength is the parameter t. The diffusion model we utilize for the structure enforcement uses a 30 step denoising schedule. Every time the loss term $\mathcal{L}_{\text{structure}}$ is evaluated, a convex combination $\hat{\mathbf{y}}_{i,t}$ (4) of the output of our method and a Gaussian noise $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is performed. Then, the pre-trained diffusion model estimates the original noise ϵ, and the residual between the estimated noise and ϵ is propagated through our method. This means that when $t = 0$, the diffusion model gets a pure Gaussian noise and when $t = 29$, it gets the output of our network with only a little Gaussian noise present. According to this, we see that the lower the value of t, the stronger the focus is on the high-level structures compared to the higher values of t.

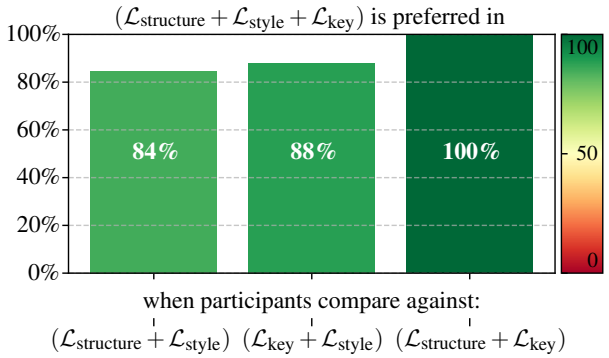


Figure 15: *Loss ablation – perceptual study. Each column shows the average preference of 28 participants for the full loss combination* $\mathcal{L}_{structure} + \mathcal{L}_{style} + \mathcal{L}_{key}$ *(1) when compared in a 2AFC against ablated variants:* $\mathcal{L}_{structure} + \mathcal{L}_{style}$ *(84 %),* $\mathcal{L}_{key} + \mathcal{L}_{style}$ *(88 %), and* $\mathcal{L}_{structure} + \mathcal{L}_{key}$ *(100 %).*

5.4. The influence of image conditioning prior c

In our method, we use a pre-trained diffusion model [ZRA23] as regularizer to ensure that the structural elements of the target frame $\mathbf{y}$ are accurately reflected in the stylized output. We hypothesize that by sampling from the diffusion model, we leverage both the structural conditioning provided by the guidance image $\mathbf{c}$ and

Figure 16: *Reinforcement of input video structure as function of parameters λ_s (1) and t (4). The stronger λ_s, the more pronounced the structure of* $\mathbf{x}$ *is in the result. Note that the structural loss $\mathcal{L}_{structure}$ (4) uses a 30 step denoising schedule. At $t = 0$, the guidance input is pure Gaussian noise $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and at $t = 29$, it contains minimal noise.*

the model's inherent ability to guide denoised samples toward the learned manifold of real images.

In this experiment, we evaluate the effectiveness of four different diffusion models. Each model uses a specific conditioning function c: (i) "line art detection" model, (ii) "depth estimation" model, (iii) Canny edge detector [Can86], and (iv) "soft edge estimation" model.

The results for $t = 16$ and $t = 28$ are presented in Fig. 17. Both depth and soft edge conditioning result in stylization that lacks the finer details of the target frame $\mathbf{y}$, likely due to the coarse structure of their respective guidance images $\mathbf{c}$. The Canny edge detector provides a more detailed signal, while line art guidance offers the most precise structural information. For applications requiring rapid training, the quality of line art can be sacrificed for the approximately 10-fold faster Canny edge guidance. In contrast, depth guidance is less practical, taking about 10 times longer than line art, with soft edge guidance between the two.

In this experiment, we kept $\lambda_s = 5 \cdot 10^{-6}$ and trained on a single NVIDIA A100 GPU for 4 hours. We selected the model with the lowest total loss. One-time optimization for a given style keyframe often converges well within minutes (see Fig. 11). Once converged, each new frame is then stylized in real time at 30+ fps (see Table 2).

Figure 17: *Structure reinforcement as function of image guidance* $\mathbf{c}$ *and diffusion time steps t. The best results are achieved with the line art-conditioned diffusion model $\mathcal{L}_{structure}$ (4) at $t = 28$.*

5.5. Direct optimization with the image conditioning prior

In our experiments, we show that our proposed $\mathcal{L}_{\text{structure}}$ (4) loss enforces structures from the target frame $\mathbf{y}$ to the stylized output $\hat{\mathbf{y}}$ of our method. The purpose of this experiment is two fold: first, to demonstrate that the $\mathcal{L}_{\text{structure}}$ loss cannot be replaced by a simpler loss that uses the line art image conditioning directly; and second, to present this simpler loss as an alternative structure-preserving mechanism, serving as a baseline for comparison. In Fig. 18, we show that replacing $\mathcal{L}_{\text{structure}}$ with the line art conditioning term

$$\mathcal{L}_{\text{lineart}}(\mathcal{Y}) = \frac{1}{|\mathcal{Y}|} \sum_{i \notin \mathbf{K}} \|c(\hat{\mathbf{y}}_i) - c(\mathbf{y}_i)\|_2^2 \tag{5}$$

fails to enforce the transfer of structural details from the target frames $\mathbf{y}_i$ to the stylized outputs $f(\mathbf{y}_i) = \hat{\mathbf{y}}_i$. In this experiment, we trained on a single NVIDIA A100 GPU for 4 hours and selected the model with the lowest total loss.

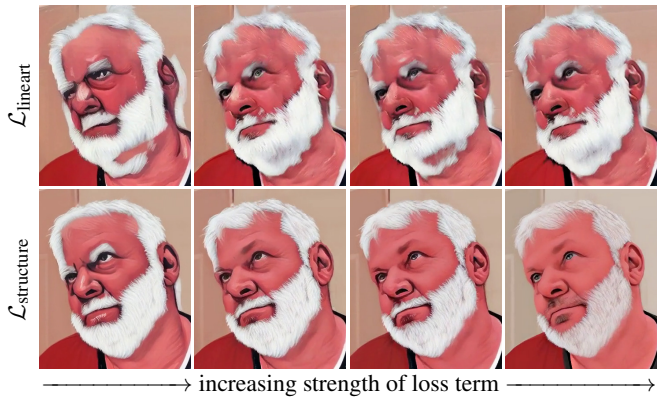


Figure 18: *Comparison of stylization results using the line art loss term $\mathcal{L}_{lineart}$ (5) versus our line art-guided structure loss term $\mathcal{L}_{structure}$ (4). Increasing the strength of $\mathcal{L}_{lineart}$ fails to effectively transfer structural elements from the target frame to the stylized output, whereas $\mathcal{L}_{structure}$ successfully preserves the target's structural details.*

6. Limitations

Although our approach represents an improvement over the current state-of-the-art in both text-driven and keyframe-based video stylization methods, we have identified the following limitations in its application.

In style transfer methods, there exists a delicate balance between maintaining fidelity to the features of the style exemplar images and preserving the structural characteristics present in the content that is being stylized. The current state-of-the-art in keyframe-based video stylization mainly emphasizes the fidelity to the original features in the style image. Our approach aims to produce stylized content that aligns both with the style exemplar and the structural characteristics of the target video sequence. Although we enable users to find the right balance between style and structure using parameters λ_s and t, we acknowledge that in some situations, decreasing the fidelity to the style features may be perceived as a potential limitation.

Our method achieves training times comparable to those of Futshik et al. [FKL*21], which can be relatively long and can restrict interactive updates, a feature offered by Texler et al. [TFK*20]. To address this drawback, in future work we plan to combine Texler et al.'s rapid patch-based training strategy with the calculation of the *style* loss in a full-frame setting.

In our proposed workflow, the user is expected to select a keyframe that will be used for training. Although certain guidelines can be followed, a mechanism that enables the automatic selection of suitable keyframes could simplify the preparation phase.

7. Conclusion

In this work, we introduced a novel keyframe-based video stylization method that balances the preservation of essential structural elements with adherence to a prescribed visual style. By integrating the line art-guided structure loss term $\mathcal{L}_{\text{structure}}$, our approach overcomes limitations of existing text-driven and keyframe-based stylization techniques by preserving structural detail from the input video, enhancing the quality and coherence of stylized sequences, and reducing the need for additional correction keyframes. Real-time inference of our method enables interactive applications such as consistently stylized video calls, which are challenging with existing approaches. By blending style fidelity with structural preservation, our method aims to improve video stylization in both quality and usability.

Acknowledgments This research was supported by the Research Center for Informatics, grant No. CZ.02.1.01/0.0/0.0/16_019/0000765, by the Grant Agency of the Czech Technical University in Prague, grants No. SGS23/173/OHK3/3T/13 and No. SGS25/150/OHK3/3T/13, and by the Ministry of Education, Youth and Sports of the Czech Republic through the e-INFRA CZ (ID:90254).

References

[ANAM*20] Andersson P., Nilsson J., Akenine-Möller T., Oskarsson M., Åström K., Fairchild M. D.: FLIP: A Difference Evaluator for Alternating Images. *Proceedings of the ACM on Computer Graphics and Interactive Techniques 3*, 2 (2020), 15:1–15:23. 5

[BCK*13] Bénard P., Cole F., Kass M., Mordatch I., Hegarty J., Senn M. S., Fleischer K. W., Pesare D., Breeden K.: Stylizing animation by example. *ACM Transactions on Graphics 32*, 4 (2013), 119. 2

[Can86] Canny J.: A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence 8*, 6 (1986), 679–698. 12

[CAS*97] Curtis C. J., Anderson S. E., Seims J. E., Fleischer K. W., Salesin D. H.: Computer-generated watercolor. In *SIGGRAPH Conference Proceedings* (1997), pp. 421–430. 2

[CHLC24] Chu E., Huang T., Lin S.-Y., Chen J.-C.: MeDM: Mediating image diffusion models for video-to-video translation with temporal correspondence guidance. In *Proceedings of the AAAI Conference on Artificial Intelligence* (2024). 3, 5, 6, 8, 9, 10

[CHM23] Ceylan D., Huang C.-H. P., Mitra N. J.: Pix2video: Video editing using image diffusion. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* (2023), pp. 23206–23217. 1, 5, 6, 7, 8, 9, 10

[CLY*17] Chen D., Liao J., Yuan L., Yu N., Hua G.: Coherent online video style transfer. In *Proceedings of IEEE International Conference on Computer Vision* (2017), pp. 1105–1114. 2

[FCC*19] Futschik D., Chai M., Cao C., Ma C., Stoliar A., Korolev S., Tulyakov S., Kučera M., Sýkora D.: Real-time patch-based stylization of portraits using generative adversarial network. In *Proceedings of the ACM/EG Expressive Symposium* (2019), pp. 33–42. 2, 5

[FJS*17] Fišer J., Jamriška O., Simons D., Shechtman E., Lu J., Asente P., Lukáč M., Sýkora D.: Example-Based Synthesis of Stylized Facial Animations. *ACM Transactions on Graphics 36*, 4 (2017), 155. 2

[FKL*21] Futschik D., Kučera M., Lukáč M., Wang Z., Shechtman E., Sýkora D.: STALP: Style transfer with auxiliary limited pairing. *Computer Graphics Forum 40*, 2 (2021), 563–573. 1, 3, 5, 7, 8, 9, 10, 13

[FSDH16] Frigo O., Sabater N., Delon J., Hellier P.: Split and Match: Example-based adaptive patch sampling for unsupervised style transfer. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 553–561. 2

[GBTBD24] Geyer M., Bar-Tal O., Bagon S., Dekel T.: TokenFlow: Consistent diffusion features for consistent video editing. In *Proceedings of International Conference on Learning Representations* (2024). 1, 3, 5, 6, 7, 8, 9, 10

[GEB16] GATYS L. A., ECKER A. S., BETHGE M.: Image style transfer using convolutional neural networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 2414–2423. 2, 4

[Goo25] GOOGLE: Veo2, 2025. URL: https://veo2.ai/. 1

[HJO*01] HERTZMANN A., JACOBS C. E., OLIVER N., CURLESS B., SALESIN D. H.: Image analogies. In *SIGGRAPH Conference Proceedings* (2001), pp. 327–340. 2

[IZZE17] ISOLA P., ZHU J.-Y., ZHOU T., EFROS A. A.: Image-to-image translation with conditional adversarial networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 1125–1134. 2

[JAFF16] JOHNSON J., ALAHI A., FEI-FEI L.: Perceptual losses for real-time style transfer and super-resolution. In *Proceedings of European Conference on Computer Vision* (2016), pp. 694–711. 2

[JST*19] JAMRIŠKA O., SOCHOROVÁ Š., TEXLER O., LUKÁČ M., FIŠER J., LU J., SHECHTMAN E., SÝKORA D.: Stylizing video by example. *ACM Transactions on Graphics 38*, 4 (2019), 107. 1, 2, 3, 5, 7, 8, 9, 10

[Kha23] KHANDELWAL A.: InFusion: Inject and attention fusion for multi concept zero shot text based video editing. In *Proceedings of IEEE International Conference on Computer Vision Workshops* (2023). 3

[KLC*24] KIM S., LEE K., CHOI J. S., JEONG J., SOHN K., SHIN J.: Collaborative score distillation for consistent visual editing. In *Advances in Neural Information Processing Systems* (2024). 3

[KOWD21] KASTEN Y., OFRI D., WANG O., DEKEL T.: Layered neural atlases for consistent video editing. *ACM Transactions on Graphics 40*, 6 (2021), 210. 3

[KSS19] KOLKIN N., SALAVON J., SHAKHNAROVICH G.: Style transfer by relaxed optimal transport and self-similarity. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* (2019), pp. 10051–10060. 2

[LFY*17] LI Y., FANG C., YANG J., WANG Z., LU X., YANG M.-H.: Universal style transfer via feature transforms. In *Advances in Neural Information Processing Systems* (2017), pp. 385–395. 2

[LH19] LOSHCHILOV I., HUTTER F.: Decoupled weight decay regularization. In *Proceedings of International Conference on Learning Representations* (2019). 5

[LHM*19] LIU M.-Y., HUANG X., MALLYA A., KARRAS T., AILA T., LEHTINEN J., KAUTZ J.: Few-shot unsupervised image-to-image translation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* (2019), pp. 10551–10560. 2

[LXZ*24] LIU G., XIA M., ZHANG Y., CHEN H., XING J., WANG Y., WANG X., SHAN Y., YANG Y.: StyleCrafter: Taming Artistic Video Diffusion with Reference-Augmented Adapter Learning. *ACM Transactions on Graphics 43*, 6 (2024), 251:1–251:10. 3

[Ope25] OPENAI: SORA, 2025. URL: https://sora.com. 1

[PGM*19] PASZKE A., GROSS S., MASSA F., LERER A., BRADBURY J., CHANAN G., KILLEEN T., LIN Z., GIMELSHEIN N., ANTIGA L., DESMAISON A., KOPF A., YANG E., DEVITO Z., RAISON M., TEJANI A., CHILAMKURTHY S., STEINER B., FANG L., BAI J., CHINTALA S.: PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems* (2019), vol. 32. 5

[PHWF01] PRAUN E., HOPPE H., WEBB M., FINKELSTEIN A.: Real-time hatching. In *SIGGRAPH Conference Proceedings* (2001), pp. 581–586. 2

[PJBM23] POOLE B., JAIN A., BARRON J. T., MILDENHALL B.: DreamFusion: Text-to-3D using 2D diffusion. In *Proceedings of International Conference on Learning Representations* (2023). 5

[RAKRF08] RAV-ACHA A., KOHLI P., ROTHER C., FITZGIBBON A.: Unwrap Mosaics: A new representation for video editing. *ACM Transactions on Graphics 27*, 3 (2008), 17. 3

[RBL*22] ROMBACH R., BLATTMANN A., LORENZ D., ESSER P., OMMER B.: High-resolution image synthesis with latent diffusion models. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* (2022), pp. 10684–10695. 3

[RDB18] RUDER M., DOSOVITSKIY A., BROX T.: Artistic style transfer for videos and spherical images. *International Journal of Computer Vision 126*, 11 (2018), 1199–1219. 2

[Run25] RUNWAY: Gen-3 Alpha, 2025. URL: https://runwayml.com/research/introducing-gen-3-alpha. 1, 3, 5, 8, 9, 10

[SKL*24] SHIN C., KIM H., LEE C. H., LEE S.-G., YOON S.: Edit-A-Video: Single video editing with object-aware consistency. In *Proceedings of Asian Conference on Machine Learning* (2024), pp. 1215–1230. 3

[SLT*19a] SIAROHIN A., LATHUILIÈRE S., TULYAKOV S., RICCI E., SEBE N.: Animating arbitrary objects via deep motion transfer. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* (2019), pp. 2377–2386. 2

[SLT*19b] SIAROHIN A., LATHUILIÈRE S., TULYAKOV S., RICCI E., SEBE N.: First order motion model for image animation. In *Advances in Neural Information Processing Systems* (2019). 2

[SWHS97] SALISBURY M. P., WONG M. T., HUGHES J. F., SALESIN D. H.: Orientable textures for image-based pen-and-ink illustration. In *SIGGRAPH Conference Proceedings* (1997), pp. 401–406. 2

[SZ14] SIMONYAN K., ZISSERMAN A.: Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556* (2014). 2, 4

[TFF*20] TEXLER O., FUTSCHIK D., FIŠER J., LUKÁČ M., LU J., SHECHTMAN E., SÝKORA D.: Arbitrary style transfer using neurally-guided patch-based synthesis. *Computers & Graphics 87* (2020), 62–71. 5

[TFK*20] TEXLER O., FUTSCHIK D., KUČERA M., JAMRIŠKA O., SOCHOROVÁ Š., CHAI M., TULYAKOV S., SÝKORA D.: Interactive video stylization using few-shot patch-based training. *ACM Transactions on Graphics 39*, 4 (2020), 73. 1, 2, 3, 5, 7, 8, 9, 10, 13

[UVL16] ULYANOV D., VEDALDI A., LEMPITSKY V.: Instance normalization: The missing ingredient for fast stylization. *arXiv:1607.08022* (2016). 5

[WLT*19] WANG T.-C., LIU M.-Y., TAO A., LIU G., CATANZARO B., KAUTZ J.: Few-shot video-to-video synthesis. In *Advances in Neural Information Processing Systems* (2019), pp. 5014–5025. 2

[YZLL23] YANG S., ZHOU Y., LIU Z., LOY C. C.: Rerender A Video: Zero-shot text-guided video-to-video translation. In *SIGGRAPH Asia Conference Papers* (2023), p. 95. 1, 3, 5, 6, 7, 8, 9, 10

[YZLL24] YANG S., ZHOU Y., LIU Z., LOY C. C.: FRESCO: Spatial-Temporal Correspondence for Zero-Shot Video Translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2024), pp. 8703–8712. 3, 5, 8, 9, 10

[ZBR*24] ZHAO W., BAI L., RAO Y., ZHOU J., LU J.: Unipc: A unified predictor-corrector framework for fast sampling of diffusion models. *Advances in Neural Information Processing Systems 36* (2024). 5

[ZIE*18] ZHANG R., ISOLA P., EFROS A. A., SHECHTMAN E., WANG O.: The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 586–595. 5

[ZLN*23] ZHANG Z., LI B., NIE X., HAN C., GUO T., LIU L.: Towards consistent video editing with text-to-image diffusion models. In *Advances in Neural Information Processing Systems* (2023), pp. 58508–58519. 3

[ZRA23] ZHANG L., RAO A., AGRAWALA M.: Adding Conditional Control to Text-to-Image Diffusion Models. In *Proceedings of IEEE International Conference on Computer Vision* (2023), pp. 3836–3847. 4, 5, 11