TCFLOW: TEMPORALLY CONSISTENT FLOW FOR SINGLE-REFERENCE VIDEO COLOR TRANSFER

Anonymous authors
Paper under double-blind review

ABSTRACT

We study *single-reference video color transfer*: given a content video and one reference image, the goal is to generate a photorealistic and temporally stable video whose color palette matches the reference. This task is fundamentally constrained by temporal transport inconsistency, where minor frame-level distribution shifts caused by motion, occlusion, or lighting induce divergent color mappings and perceptible flickering. Existing approaches expose a stability-expressiveness trade-off: clip-level global transforms are temporally stable but limited in modeling complex reference-specific correspondences, whereas frame-wise methods are more expressive but often temporally incoherent. To address this limitation, we propose **TCFlow**, a unified meta-flow framework for robust and temporally consistent video color transfer. TCFlow predicts a clip-shared flow that transports the aggregated video-level content distribution toward a single reference palette. This clip-shared formulation inherently decouples global color structure from frame-specific variations, effectively suppressing color flickering. To ensure high-fidelity transfer, we introduce a color aggregator that synthesizes global color statistics across the video clip, enabling expressive and accurate color mapping without optical flow, dense motion supervision, or test-time optimization. Experiments show that TCFlow substantially suppresses color flickering and reduces frame-wise mapping variation. Beyond standard benchmarks, we validate TCFlow on a dataset of real-world media art exhibitions, where it consistently outperforms state-of-the-art methods in stability and photorealism. Further source code is provided in the supplementary material. Additional results, including qualitative comparisons and more video results, are available at `https://tc-flow-iclr.vercel.app/`.

1 INTRODUCTION

Video has become a dominant medium for visual communication, spanning film production, advertising, social media, virtual production, and large-scale media art. In these applications, color is not merely a low-level appearance attribute, but a central carrier of mood, style, brand identity, and artistic intent. A practical video color transfer system should therefore allow users to impose the color impression of a desired reference onto an existing video while preserving scene structure, object identity, and photorealism (Bonneel et al., 2013; Yao et al., 2016; Zhang et al., 2019; Akimoto et al., 2020; Faridul et al., 2016). Since creative users typically specify a desired palette with a single reference image, rather than semantic annotations, segmentation masks, paired examples, or textual descriptions, we study *single-reference video color transfer*: given a content video and one reference image, the goal is to generate a video that matches the reference palette while remaining realistic and temporally stable.

This task requires two properties that are difficult to satisfy simultaneously: expressive reference-conditioned color mapping and temporal consistency. Video color grading methods typically favor stability by applying a clip-level global transform, such as a 3D LUT (Li et al., 2025). While efficient and temporally consistent, such global transforms have limited capacity to model complex, reference-specific color correspondences. Image color transfer methods (Gatys et al., 2016; Luan et al., 2017) in contrast, can learn more expressive mappings (Li et al., 2017; 2018; Yoo et al., 2019; He et al., 2019), but applying them independently to each frame often produces flickering (Ke et al., 2023; Larchenko et al., 2025), because each frame induces a slightly different color transformation. As illustrated in Fig. 1, existing approaches expose a fundamental trade-off between reference fidelity and temporal

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

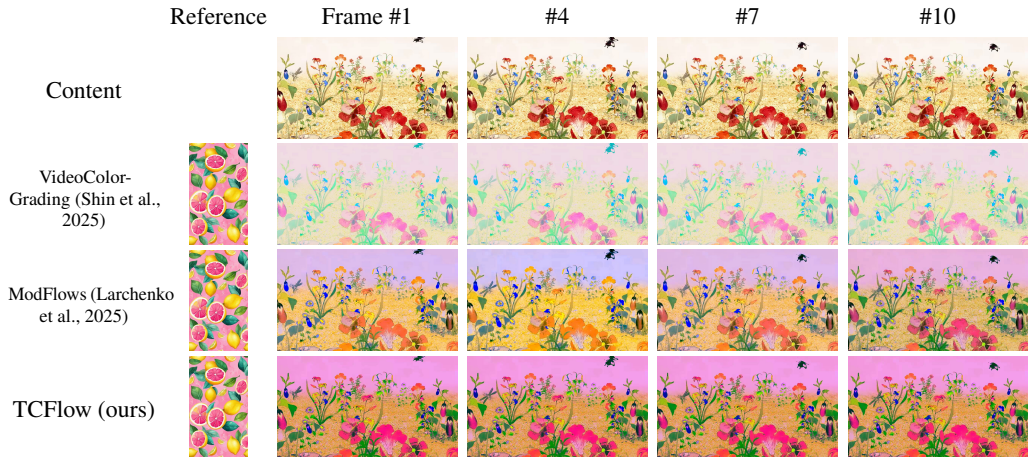

Figure 1: **Color Flickering Problem.** Frame-wise transfer methods like ModFlows suffer from temporal transport inconsistency, where the same semantic regions (e.g., the sky and the flower) exhibit abrupt color shifts over time. TCFlow predicts a clip-shared flow that ensures a stable and consistent color mapping across the entire sequence, effectively reducing flickering while accurately matching the reference color.

consistency. VideoColorGrading (Shin et al., 2025) produces temporally stable outputs, but its global transformation fails to faithfully capture the reference palette. In contrast, ModFlows (Larchenko et al., 2025) provides more expressive reference-conditioned color transfer, but its frame-wise mappings vary across time, resulting in visible color flickering.

Viewing color transfer as a transport between the content and reference color distributions (Villani, 2008; Reinhard et al., 2001; Pitie & Kokaram, 2007; Pitié et al., 2007), expressive transport-based methods can achieve strong reference color matching by learning flexible correspondences between the two distributions. However, this flexibility becomes problematic in video: when a separate transport is estimated for each frame, even small frame-to-frame changes in the empirical color distribution can induce different color correspondences over time. We refer to this failure mode as *temporal transport inconsistency* (Kar & Su, 2025; Liu et al., 2021). Such distributional changes arise from motion, occlusion, lighting variation, and changes in object composition, causing corresponding colors to drift across time even when individual frames match the reference palette. This suggests that temporal consistency should be enforced not merely after color transfer (Bonneel et al., 2013; Yao et al., 2016; Lai et al., 2018), but directly at the level of the color transport itself.

To address this problem, we propose **TCFlow**, a temporally consistent meta-flow framework for single-reference video color transfer. TCFlow is built on a simple principle: the color transformation should be inferred from the video as a whole, rather than independently from each frame. Given a content video and a reference image, TCFlow predicts a *clip-shared* rectified flow that transports the video-level content color distribution toward the reference color distribution (Liu et al., 2022; Liu, 2022). By sharing a single transport map across all frames, TCFlow inherently ensures temporal consistency and eliminates the flickering artifacts typical of frame-wise methods.

To further enhance the quality of the transferred colors, we employ a color aggregator that aggregates global color statistics from the video. This allows the model to infer a high-fidelity shared flow that accurately captures the desired palette and preserves structural integrity, overcoming the limitations of traditional color transfer. Extensive experiments demonstrate that TCFlow significantly outperforms frame-wise flow baselines in terms of temporal stability, achieving a superior trade-off between color transfer fidelity and temporal coherence. We further introduce a media-art exhibition benchmark with abstract structures and extreme lighting shifts, where TCFlow maintains artistic intent and high-fidelity transfer.

2 Related Work

Video Style and Color Transfer Early methods used optical flow or temporal losses to mitigate flickering (Ruder et al., 2016; Huang et al., 2017; Wu et al., 2022; Wen et al., 2023; Chen et al.,

2017; Gao et al., 2020; Liu & Zhang, 2021). While effective for artistic stylization, these often introduce structural distortions and "style leakage." In high-end color transfer, such artifacts undermine photorealism, as subtle shifts in hue and luminance are critical to perceived quality (Zhang et al., 2026).

LUT-based Video Color Grading Professional workflows rely on 3D Look-Up Tables (LUTs) for absolute stability. Recent learning-based approaches automate this by predicting image-adaptive LUTs (Shin et al., 2025; Chen et al., 2023). Although computationally efficient, the global nature of LUTs limits their degrees of freedom, making it difficult to capture complex, spatially-varying color correspondences from sophisticated reference images.

Flow-based Color Mapping Color transfer can be framed as a distribution alignment problem using Normalizing Flows (Liu, 2022). For instance, ModFlows Larchenko et al. (2025) learns a conditioned flow between color palettes. However, frame-wise flows are vulnerable to *temporal transport inconsistency*: slight statistical variations between frames induce divergent transport paths (Liu et al., 2025), causing visible color drift even when individual frames appear plausible.

Unlike methods relying on smoothing or rigid LUTs, TCFlow redefines color transport at the video level. By treating a clip as a unified distribution, we introduce a *clip-shared flow* via meta-learning. This allows for expressive, non-linear mappings that remain strictly consistent over time without the overhead of optical flow or test-time optimization, offering a robust solution that simultaneously enhances color transfer expressivity and stability.

3 METHOD

3.1 PROBLEM FORMULATION

Let $V = \{I_n\}_{n=1}^N$ denote a content video with N frames, where $x_{n,p} = I_n(p) \in [0,1]^3$ is the RGB color at pixel location p of frame n. Given a single reference image R, we denote its reference colors by $r_q = R(q) \in [0,1]^3$. Our goal is to generate a color-transferred video $\widehat{V} = \{\widehat{I}_n\}_{n=1}^N$ whose frames follow the color of R, while preserving the spatial structure of V and avoiding unnatural temporal variations such as flickering.

We formulate the task as learning a reference-conditioned color mapping $\mathcal{G}_{V,R} : \mathcal{C} \to \mathcal{C}$, where $\mathcal{C} = [0,1]^3$ is the normalized RGB color space. For each input color, $\widehat{x}_{n,p} = \mathcal{G}_{V,R}(x_{n,p})$ and $\widehat{I}_n(p) = \widehat{x}_{n,p}$. Let $P_{\widehat{I}_n}$ be the empirical color distribution of the transferred frame $\widehat{I}_n$, and let P_R be the empirical color distribution of the reference image. The desired output should satisfy frame-wise reference alignment,

$$\frac{1}{N}\sum_{n=1}^{N} \mathcal{D}_{\mathrm{col}}(P_{\widehat{I}_n}, P_R) \to 0, \tag{1}$$

and video-level temporal consistency,

$$\mathcal{D}_{\mathrm{temp}}(\widehat{V}, V) \text{ should be small.} \tag{2}$$

Here, $\mathcal{D}_{\mathrm{col}}$ measures color-distribution discrepancy, while $\mathcal{D}_{\mathrm{temp}}$ measures unnatural temporal instability in the transferred video, such as flickering or abrupt frame-to-frame color changes. Detailed 3D visualizations of the color transformation trajectories for TCFlow and ModFlows are available in Fig. 5 of the Appendix A.1.

3.2 OVERVIEW

TCFlow learns a clip-shared color flow for single-reference video color transfer. Instead of estimating an independent image-to-latent flow for each frame, TCFlow constructs one video-specific flow from colors sampled across all frames. Each frame is still transformed at the pixel level, but all frames in the same clip share the same video-to-latent transport.

The method has two steps. First, for each training video, we fit a teacher rectified flow using temporally balanced color samples. Given a sampling budget of K colors and N frames, we sample

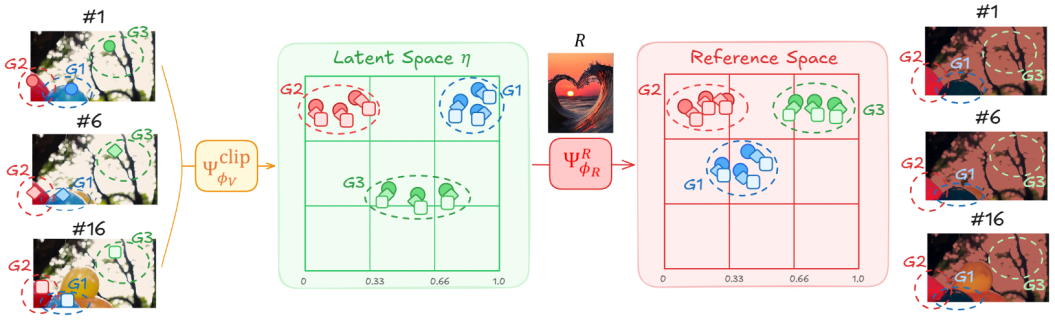

Figure 2: **Overview of the proposed Clip-shared Latent Color Transport.** TCFlow estimates a single video-specific shared content flow $\Psi^{\text{clip}}_{\phi_V}$ by pooling color samples from multiple frames (e.g., Frame $\#1, \#6, \#16$). These colors (grouped as G1–G3) are first mapped to the latent space η via the shared content flow $\Psi^{\text{clip}}_{\phi_V}$. Subsequently, they are decoded into the reference space using an inverse flow $(\Psi^{R}_{\phi_R})^{-1}$ derived from the reference image. This shared transport mechanism ensures temporal consistency across the entire video clip. More analysis is in Appendix A.1.

approximately K/N pixels from each frame and pool them. The resulting teacher flow is therefore optimized to explain the color distributions of all frames in the clip, rather than a single frame. Second, we train a color aggregator Γ_ω to infer this clip-shared flow directly from an input video. The aggregator utilizes a lightweight feature-aggregation block to synthesize global color statistics across the temporal axis.

At inference, the same predictor is used for both the content video and the reference image. The reference image is repeated along the temporal axis to form a pseudo-video, from which the predictor estimates a reference-to-latent flow. Color transfer is then performed by mapping content colors to the latent space with the content flow and decoding them to the reference space through the inverse reference flow.

3.3 Clip-shared Latent Color Transport

Let P_{I_n} denote the empirical color distribution of frame I_n. For a video clip, we define the balanced clip-level sampling distribution as

$$P_V = \frac{1}{N} \sum_{n=1}^{N} P_{I_n}. \tag{3}$$

In practice, P_V is constructed by sampling approximately K/N pixels from each frame and pooling them across the clip.

We introduce a fixed latent color distribution η supported on $[0,1]^3$, e.g., $\eta = \mathcal{U}([0,1]^3)$. TCFlow decomposes color transfer into a content video-to-latent transport and a reference-to-latent transport. For the content video, all frames share the same transport:

$$\Psi^{(1)}_{\phi_V} = \cdots = \Psi^{(N)}_{\phi_V} = \Psi^{\text{clip}}_{\phi_V}, \qquad \Psi^{\text{clip}}_{\phi_V} : P_{I_n} \to \eta \quad \text{for all } n.$$

For the reference image, the transport $\Psi^{R}_{\phi_R} : P_R \to \eta$ is obtained at inference by applying the same predictor to a repeated pseudo-video of R. The final video-to-reference map is

$$\mathcal{G}_{V,R} = \left(\Psi^{R}_{\phi_R}\right)^{-1} \circ \Psi^{\text{clip}}_{\phi_V}. \tag{4}$$

Both transports are parameterized by rectified flows. For a content color $x_{n,p}$, the clip-shared video-to-latent trajectory is

$$\frac{dz^V_{n,p}(t)}{dt} = f_{\phi_V}(z^V_{n,p}(t), t), \qquad z^V_{n,p}(0) = x_{n,p}, \quad t \in [0,1]. \tag{5}$$

The latent color is

$$u_{n,p} = z^V_{n,p}(1) = x_{n,p} + \int_0^1 f_{\phi_V}(z^V_{n,p}(t), t)\, dt. \tag{6}$$

To decode $u_{n,p}$ into the reference space, we integrate the reference flow backward:

$$\widehat{x}_{n,p} = u_{n,p} + \int_1^0 f_{\phi_R}(z^R_{n,p}(t), t)\, dt, \qquad z^R_{n,p}(1) = u_{n,p}. \tag{7}$$

Finally, $\widehat{I}_n(p) = \widehat{x}_{n,p}$.

The overall pipeline of this transport is illustrated in Fig. 2. As depicted, color samples from multiple frames across the clip (e.g., Frame $\#1, \#6, \#16$) are first projected into a unified latent space η via the shared flow $\Psi^{\text{clip}}_{\phi_V}$, and subsequently mapped to the reference space. The transport to a uniform distribution yields a learned latent representation that exhibits robust continuity and low sensitivity to reference selection, as demonstrated by our experiments in Appendix A.2.

3.4 Latent Teacher Flow Construction

For each training video clip $V_m = \{I_{m,n}\}_{n=1}^N$, we construct a balanced clip-level distribution $P_{V_m} = \frac{1}{N}\sum_{n=1}^N P_{I_{m,n}}$ by sampling approximately K/N pixels per frame. This temporal pooling prevents the resulting flow map from being biased toward any isolated frame and provides a target that represents the optimal color transformation for the whole video.

We optimize a teacher flow model f_{ϕ_m} that transports the content distribution P_{V_m} to the canonical latent distribution η. Given a content color $x_0 \sim P_{V_m}$, a target latent sample $x_1 \sim \eta$, and time $t \sim \mathcal{U}(0,1)$, we define the linear interpolation path $z_t = (1-t)x_0 + tx_1$. The teacher flow parameters $\phi^\star_m$ are optimized via the Rectified Flow objective:

$$\phi^\star_m = \arg\min_{\phi} \mathbb{E}_{x_0 \sim P_{V_m}, x_1 \sim \eta, t \sim \mathcal{U}(0,1)} \left[\| f_\phi(z_t, t) - (x_1 - x_0) \|_2^2 \right]. \tag{8}$$

The converged parameters $\phi^\star_m$ serve as high-fidelity ground-truth flow targets for the subsequent learning stage.

3.5 Color Aggregator

While teacher flows are highly accurate, they require per-video optimization, which is computationally expensive during inference. To enable efficient and real-time transfer, we introduce a color aggregator Γ_ω that predicts the shared flow parameters in a single forward pass: $\widehat{\phi}_m = \Gamma_\omega(V_m)$.

Rather than relying on individual frames or simple feature averaging, the color aggregator is designed to synthesize information across the entire temporal axis to infer a flow that best aligns the content video's global palette with the reference. We train the aggregator to match the teacher's vector field in function space:

$$\mathcal{L}_{\text{meta}}(\omega) = \mathbb{E}_{V_m, x_0 \sim P_{V_m}, x_1 \sim \eta, t \sim \mathcal{U}(0,1)} \left[\left\| f_{\widehat{\phi}_m}(z_t, t) - f_{\phi^\star_m}(z_t, t) \right\|_2^2 \right], \tag{9}$$

where $\widehat{\phi}_m = \Gamma_\omega(V_m)$. This objective forces the aggregator to capture the essential color statistics of the clip to produce a high-fidelity transport map without test-time optimization.

Architecture for Color Aggregation. Given an input video tensor $V \in \mathbb{R}^{N\times 3\times H\times W}$, we first extract frame-level descriptors $a_n = \text{GAP}(B_\omega(I_n)) \in \mathbb{R}^D$ using a shared backbone B_ω, resulting in a feature matrix $A = [a_1, \ldots, a_N] \in \mathbb{R}^{D\times N}$.

To synthesize these descriptors into a unified clip-level representation, the feature-aggregation block processes the temporal axis through a lightweight 1D temporal convolution block to capture complex color dependencies and lighting variations:

$$h_V = \text{Pool}_{\text{time}}\left(\text{Conv1D}_{\text{pw}}\left(\text{Conv1D}_{\text{dw}}(A)\right)\right). \tag{10}$$

The final flow parameters are then derived as $\widehat{\phi} = Wh_V + b$. By treating the temporal axis as a source of global color statistics, the color aggregator ensures that the resulting shared flow $\Psi^{\text{clip}}_{\phi_V}$ achieves superior color fidelity while reducing frame-wise temporal flickering.

3.6 INFERENCE

During inference, the content colors $x_{n,p}$ are first projected into the latent distribution η (represented as the green grid in Fig. 2). This intermediate representation allows for a stable decoding into the reference color regardless of frame-specific illumination changes.

Given a test content video V and reference image R, TCFlow predicts the content video-to-latent flow as

$$\widehat{\phi}_V = \Gamma_\omega(V). \tag{11}$$

To obtain the reference-to-latent flow, we repeat the reference image along the temporal axis to form a pseudo-video

$$\widetilde{R} = \{R, R, \ldots, R\}_{n=1}^{N}, \tag{12}$$

and apply the same predictor:

$$\widehat{\phi}_R = \Gamma_\omega(\widetilde{R}). \tag{13}$$

For every frame I_n and pixel p, the content color is transported to the latent space using the shared content flow:

$$u_{n,p} = x_{n,p} + \int_0^1 f_{\widehat{\phi}_V}(z_{n,p}^V(t), t)\, dt. \tag{14}$$

It is then decoded into the reference space by integrating the reference flow backward:

$$\widehat{x}_{n,p} = u_{n,p} + \int_1^0 f_{\widehat{\phi}_R}(z_{n,p}^R(t), t)\, dt, \qquad z_{n,p}^R(1) = u_{n,p}. \tag{15}$$

Finally, $\widehat{I}_n(p) = \widehat{x}_{n,p}$ and $\widehat{V} = \{\widehat{I}_n\}_{n=1}^{N}$. The same $\widehat{\phi}_V$ is applied to all content frames, while $\widehat{\phi}_R$ is inferred from the repeated reference pseudo-video using the same predictor. Thus, TCFlow performs reference-based video color transfer without a separate reference-flow predictor or test-time optimization.

4 EXPERIMENTS

We evaluate the performance of TCFlow and demonstrate its superiority in video color transfer. To ensure a fair and comprehensive comparison, we select several state-of-the-art baselines that support video-based color transfer: CCPL (Wu et al., 2022), CAP-VST (Wen et al., 2023), VideoColorGrading (Shin et al., 2025), and the flow-based method ModFlows (Larchenko et al., 2025).

For consistent evaluation, all baseline models except VideoColorGrading were retrained on the same training dataset as our model. For VideoColorGrading, we used the official well-trained pre-trained checkpoint released by the authors, as the resources required to reproduce its Look-Up Table training were not fully available in the public implementation. All experiments were conducted under identical benchmark settings to validate the relative improvements across all evaluated metrics.

4.1 EXPERIMENTAL SETTINGS

Datasets. For the training of TCFlow, we utilize UltraVideo (Xue et al., 2025), a large-scale dataset comprising approximately 58,000 UHD video clips. This dataset encompasses a broad spectrum of scenes, including natural landscapes, urban environments, human activities, and diverse object trajectories. To balance computational tractability with the representative diversity of the dataset, we randomly sub-sampled 5,000 video clips, all resized to a resolution of 960×540, to serve as our training set.

To rigorously evaluate the generalization performance of TCFlow and facilitate quantitative comparisons with existing generative models, we established a test benchmark using DL3DV-10K (Ling et al., 2024). Our benchmark consists of 500 video pairs, strategically partitioned into two subsets to provide a multi-faceted evaluation. Specifically, 250 pairs were curated using the Hungarian algorithm(Kuhn, 1955; Munkres, 1957) to maximize the color discrepancy between the content and the reference, thereby simulating extreme color transformation scenarios. The remaining 250 pairs were selected via random sampling to represent standard operating conditions. This dual-track design ensures a balanced assessment, verifying the model's robustness across both typical and adversarial color transfer environments.

Table 1: **Quantitative comparison on the DL3DV-10K dataset.** TCFlow achieves superior performance with respect to color flickering and Lipschitz scores, resulting in more stable and consistent results. Performance values are color-coded: red for best, blue for second best.

Methods	Aggregated scores			Color scores		Lipschitz(↓)	$D_{\text{temp}}^{\text{global}}$ (↓)	$D_{\text{temp}}^{\text{patch}}$ (↓)
	Grayscale(↓)	Depth(↓)	Edge(↓)	SWD(↓)	MS-SWD(↓)			
CCPL (Wu et al., 2022)	0.1300 ± 0.0017	0.1320 ± 0.0019	0.1552 ± 0.0019	0.0698 ± 0.0011	1.4064 ± 0.0141	89.62 ± 1.36	0.9275 ± 0.0193	1.3499 ± 0.0228
CAP-VST (Wen et al., 2023)	0.1185 ± 0.0022	0.1290 ± 0.0021	0.1338 ± 0.0021	0.0816 ± 0.0015	1.4273 ± 0.0188	37.87 ± 0.81	0.8503 ± 0.0156	1.1541 ± 0.0225
VideoColorGrading (Shin et al., 2025)	0.1727 ± 0.0037	0.1944 ± 0.0039	0.1970 ± 0.0036	0.1417 ± 0.0035	2.2607 ± 0.0424	17.86 ±0.25	0.7606 ± 0.0167	1.1419 ± 0.0238
ModFlows (Larchenko et al., 2025)	0.0889 ± 0.0030	0.1122 ± 0.0028	0.1263 ± 0.0029	0.0556 ± 0.0021	1.4382 ± 0.0403	52.91 ± 2.32	1.4163 ± 0.0277	1.5005 ± 0.0274
TCFlow (ours)	**0.0788 ± 0.0015**	**0.1052 ± 0.0016**	**0.1207 ± 0.0023**	**0.0462 ± 0.0006**	**1.2355 ± 0.0161**	**14.77 ± 0.36**	**0.7382 ± 0.0173**	**1.0543 ± 0.0231**

Implementation Details. In the teacher flow construction stage, we optimize individual flow parameters to map the clip-level content distribution P_V of each training video to the canonical latent distribution η. The velocity field model f_ϕ is parameterized as a Multi-Layer Perceptron (MLP) consisting of two linear layers with 8,192 hidden units and Tanh activations. Each teacher flow is optimized for 10,000 iterations using the Adam optimizer with a learning rate of 5×10^{-4}, forming a distillation dataset $\mathcal{D}_V = \{(V_m, \phi_m^\star)\}_{m=1}^{5000}$.

To train the color aggregator Γ_ω, which directly infers the clip-shared flow parameters $\widehat{\phi}_V$ from an input clip V, we adopt EfficientNet-B4 (Tan & Le, 2019) as the spatial image backbone B_ω. The temporal convolution block processes frame descriptors a_n to capture global color trends. Model training was conducted on a single GPU server equipped with 96GB VRAM, employing the Adam optimizer with a learning rate of 1×10^{-4} and a batch size of 8. Training for 1,200 epochs required approximately seven days. Additional details regarding computational complexity, latency, and FLOP comparisons are provided in Appendix A.3.

Evaluation Metrics. We use an aggregated score as our primary metric, defined as the Euclidean distance from an ideal reference point p (Larchenko et al., 2025), which integrates content preservation and color fidelity. The content score evaluates structural integrity using normalized grayscale, DepthFM (Gui et al., 2025), and HED edge maps (Xie & Tu, 2015) via the DISTS metric (Ding et al., 2020). Color alignment is assessed using Sliced Wasserstein Distance (SWD) (Larchenko et al., 2025) and its multi-scale variant, MS-SWD (He et al., 2024). To evaluate temporal coherence, we introduce a global color flickering metric $D_{\text{temp}}^{\text{global}}$ based on the Bhattacharyya distance (Bhattacharyya, 1943) and a patch-based metric $D_{\text{temp}}^{\text{patch}}$ for local consistency. Additionally, we estimate the average Lipschitz constant of f_ϕ to assess mapping regularity and robustness (Larchenko et al., 2025). All metrics are frame-averaged with variability estimated via the bootstrap method (Efron & Tibshirani, 1994); detailed formulations are provided in Appendix A.4.

4.2 Results

Qualitative Results. As shown in Fig. 3, TCFlow achieves a qualitative comparison with state-of-the-art methods on the DL3DV-10K dataset. As indicated by the red boxes, CCPL suffers from severe visual artifacts and color distortion, particularly around object boundaries, which degrades both local appearance and structural details. CAP-VST and ModFlows produce more expressive color transformations, but their frame-wise or locally varying mappings lead to inconsistent sky tones and noticeable color fluctuations across frames. VideoColorGrading provides more temporally stable results than CAP-VST and ModFlows, but its lower color expressiveness limits its ability to accurately reproduce the reference color, particularly in the tree regions. In contrast, TCFlow consistently transfers the reference colors to diverse scene regions, including the sky, trees, and foreground, while preserving sharp structural boundaries. Moreover, its clip-shared transport maintains consistent color mappings across frames, resulting in fewer temporal color fluctuations and more faithful reference matching.

Quantitative Results. As shown in Table 1, TCFlow outperforms state-of-the-art methods across nearly all metrics on the DL3DV-10K dataset. Specifically, it achieves the best aggregated scores (grayscale, depth, and edge), demonstrating superior preservation of structural integrity and perceptual fidelity. It also yields the lowest SWD and MS-SWD, confirming the most accurate alignment of color distributions with the reference. Notably, TCFlow exhibits exceptional temporal stability and mapping



Figure 3: **Qualitative comparison on the DL3DV-10K dataset.** TCFlow achieves the best balance between accurate color transformation and structural preservation.

regularity, evidenced by the lowest Lipschitz constant and color flickering score; the former suppresses local artifacts, while the latter mitigates the perceptible flickering common in existing methods. While VideoColorGrading shows competitive stability, it fails to match TCFlow's aggregated and color scores. Thus, TCFlow achieves a favorable balance between color fidelity and temporal coherence. Similar trends are observed in our real-world media art benchmark in Appendix A.5, additional evaluations on dynamic scenes provided in Appendix A.6.

4.3 Ablation Studies

Clip-shared latent color transport analysis We analyze the effect of our clip-shared transport map $\Psi_{\phi_V}^{\text{clip}}$, which constrains the entire video to a unified mapping to prevent temporal discontinuities. In Fig. 4, the frame-wise flow exhibits color inconsistency on the ground between Frames #8 and #9 due to over-optimization of individual frames. In contrast, our clip-shared flow ensures temporal color consistency by regularizing transitions across the entire temporal sequence. Quantitatively, the color flicker instability graph shows that clip-shared flow (green) maintains significantly lower and more stable scores compared to the erratic fluctuations of the frame-wise flow (gray). These results confirm that clip-shared transport inherently enforces temporal coherence by preventing stochastic drift, effectively reducing flickering artifacts.

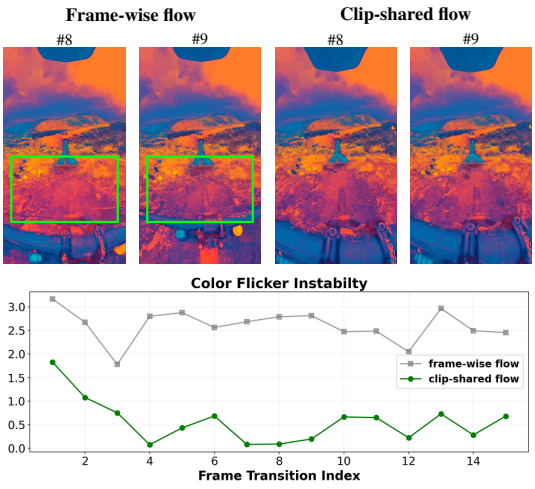

Figure 4: **Analysis of color flicker instability.**

Table 2: **Impact of the color aggregator on the DL3DV-10K dataset.** TCFlow with the proposed color aggregator achieves substantially better color fidelity, while exhibiting a trade-off in temporal and structural consistency metrics. Performance values are color-coded: red for the best.

Methods	Aggregated scores			Color scores		Lipschitz(↓)	$D_{\text{temp}}^{\text{global}}$ (↓)	$D_{\text{temp}}^{\text{patch}}$ (↓)
	Grayscale(↓)	Depth(↓)	Edge(↓)	SWD(↓)	MS-SWD(↓)			
TCFlow w/o color agg	0.0802 ± 0.0017	0.1006 ± 0.0017	0.1165 ± 0.0018	0.0599 ± 0.0008	1.6606 ± 0.0218	10.68 ± 0.22	0.5822 ± 0.0124	0.8104 ± 0.0188
TCFlow	**0.0788 ± 0.0015**	**0.1052 ± 0.0016**	**0.1207 ± 0.0023**	**0.0462 ± 0.0006**	**1.2355 ± 0.0161**	**14.77 ± 0.36**	**0.7382 ± 0.0173**	**1.0543 ± 0.0231**

Impact of Color Aggregator We evaluate the contribution of the color aggregator by comparing the full TCFlow with a version employing simple mean aggregation for temporal features (TCFlow w/o color agg), as shown in Table 2.

The color aggregator substantially enhances color fidelity, as evidenced by a significant reduction in both SWD and MS-SWD. While it slightly improves the grayscale score, we observe a moderate increase in depth and edge scores. These results indicate that the aggregator is particularly effective at capturing and aligning global color statistics across the temporal axis to ensure precise palette matching.

However, the inclusion of the color aggregator leads to a slight increase in the Lipschitz constant and temporal consistency scores, suggesting a trade-off between color fidelity and structural/temporal stability. Nevertheless, the dramatic gains in color-related metrics justify its use, demonstrating that the proposed aggregation mechanism is essential for achieving high-fidelity, accurate color transfer.

Analysis of Color Transformation Trajectories We analyze the color transfer behavior of TCFlow in comparison with ModFlows. As illustrated in Appendix A.1 (Fig. 5), TCFlow projects pixels with similar color characteristics into a more compact region of the uniform space before mapping them to the reference space. We quantitatively measure this cluster dispersion using standard deviation, as summarized in Table3.

TCFlow consistently achieves lower cluster variance than ModFlows in the uniform space, indicating more cohesive representations of similar colors. This compactness is also reflected in the final transferred space, where TCFlow maintains lower color variance than ModFlows and the original content. These results suggest that the proposed transformation produces more consistent color mappings across frames.

Table 3: Comparison of color cluster Std. Dev.

	Content	ModFlows[Larchenko et al., 2025]		TCFlow	
Cluster	(RGB)	Uniform	Reference	Uniform	Reference
G1	0.0435	0.1249	0.0906	**0.0772**	**0.0084**
G2	0.0449	0.0896	0.0762	**0.0360**	**0.0201**
G3	0.0172	0.0678	0.0257	**0.0236**	**0.0171**

5 Conclusion

In this work, we presented **TCFlow** as a robust solution to the problem of temporal instability in video color transfer. By shifting the perspective from frame-wise mapping to video-level distribution transport, we have formalized and addressed the temporal transport inconsistency that fundamentally limits existing methods. Our experiments confirm that the proposed clip-shared flow successfully captures global color correspondences while remaining resilient to local variations, all without the computational burden of motion estimation. We have demonstrated that TCFlow not only suppresses flickering but also preserves high fidelity. We hope our framework serves as a versatile foundation for consistent video-to-video translation and inspires further exploration of meta-learning for distribution-based tasks.

Limitation TCFlow focuses on efficient single-reference video color transfer without requiring semantic maps, masks, motion supervision, or test-time optimization. Since it performs clip-level distribution-to-distribution transport, the same mapping is applied regardless of pixel position or semantic region. Extending TCFlow to support fine-grained region-level control with semantic maps or additional conditions is an important future direction.

AI USE STATEMENT

In this work, we leveraged generative AI tools primarily for enhancing the manuscript's linguistic quality. Specifically, AI was used for proofreading, refining the academic tone of the prose, and rephrasing complex sentences to ensure maximum clarity and readability for the reader. We also utilized AI to double-check the consistency of mathematical notations and formulations to prevent clerical errors.

We explicitly state that generative AI tools were not used for the core conceptualization of the research, including the derivation of the clip-shared flow framework, the design of the experimental pipeline, or the interpretation of the results. All theoretical contributions and research hypotheses were developed independently by the authors. Additionally, no synthetic data was generated using AI for this study.

Furthermore, generative AI assisted in accelerating the implementation of non-core software components, such as boilerplate code for result visualization and script automation for ablation studies. Every piece of AI-generated code was scrutinized and rigorously tested by two authors to ensure correctness and reproducibility. We take full responsibility for the final content of this paper, including all claims, mathematical derivations, and technical artifacts.

REPRODUCIBILITY STATEMENT

To ensure the reproducibility of our experimental results, we provide the complete implementation code, training scripts, and ablation scripts in the supplementary material. DL3DV-10K, REDS and MOSE-v2 datasets used in this work are publicly available. Detailed descriptions of the data preprocessing pipelines, model architectures, training procedures, and hardware specifications are provided in Section 4.

REFERENCES

Naofumi Akimoto, Akio Hayakawa, Andrew Shin, and Takuya Narihira. Reference-based video colorization with spatiotemporal correspondence, 2020. URL https://arxiv.org/abs/2011.12528.

Anil Bhattacharyya. On a measure of divergence between two statistical populations defined by their probability distribution. *Bulletin of the Calcutta Mathematical Society*, 35:99–110, 1943.

Nicolas Bonneel, Kalyan Sunkavalli, Sylvain Paris, and Hanspeter Pfister. Example-based video color grading. *ACM Trans. Graph.*, 32(4):39–1, 2013.

Dongdong Chen, Jing Liao, Lu Yuan, Nenghai Yu, and Gang Hua. Coherent online video style transfer, 2017. URL https://arxiv.org/abs/1703.09211.

Yaosen Chen, Han Yang, Yuexin Yang, Yuegen Liu, Wei Wang, Xuming Wen, and Chaoping Xie. Nlut: Neural-based 3d lookup tables for video photorealistic style transfer, 2023. URL https://arxiv.org/abs/2303.09170.

Henghui Ding, Kaining Ying, Chang Liu, Shuting He, Xudong Jiang, Yu-Gang Jiang, Philip HS Torr, and Song Bai. Mosev2: A more challenging dataset for video object segmentation in complex scenes. *arXiv preprint arXiv:2508.05630*, 2025.

Keyan Ding, Kede Ma, Shiqi Wang, and Eero P Simoncelli. Image quality assessment: Unifying structure and texture similarity. *IEEE transactions on pattern analysis and machine intelligence*, 44(5):2567–2581, 2020.

Bradley Efron and Robert J Tibshirani. *An introduction to the bootstrap*. Chapman and Hall/CRC, 1994.

H. Sheikh Faridul, T. Pouli, C. Chamaret, J. Stauder, E. Reinhard, D. Kuzovkin, and A. Tremeau. Colour mapping: A review of recent methods, extensions and applications. *Computer Graphics Forum*, 35(1):59–88, 2016. doi: 10.1111/cgf.12671.

Wei Gao, Yijun Li, Yihang Yin, and Ming-Hsuan Yang. Fast video multi-style transfer. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 3211–3219, 2020. doi: 10.1109/WACV45572.2020.9093420.

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)*, June 2016.

Ming Gui, Johannes Schusterbauer, Ulrich Prestel, Pingchuan Ma, Dmytro Kotovenko, Olga Grebenkova, Stefan Andreas Baumann, Vincent Tao Hu, and Björn Ommer. Depthfm: Fast generative monocular depth estimation with flow matching. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 3203–3211, 2025.

Jiaqi He, Zhihua Wang, Leon Wang, Tsein-I Liu, Yuming Fang, Qilin Sun, and Kede Ma. Multiscale sliced wasserstein distances as perceptual color difference measures. In *European Conference on Computer Vision*, pp. 425–442. Springer, 2024.

Mingming He, Jing Liao, Dongdong Chen, Lu Yuan, and Pedro V. Sander. Progressive color transfer with dense semantic correspondences. *ACM Trans. Graph.*, 38(2), April 2019. doi: 10.1145/3292482.

Haozhi Huang, Hao Wang, Wenhan Luo, Lin Ma, Wenhao Jiang, Xiaolong Zhu, Zhifeng Li, and Wei Liu. Real-time neural style transfer for videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Aupendu Kar and Guan-Ming Su. Temporal consistent semantic video color transfer from multiple references. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 6253–6261, June 2025.

Zhanghan Ke, Yuhao Liu, Lei Zhu, Nanxuan Zhao, and Rynson W. H. Lau. Neural preset for color style transfer, 2023. URL https://arxiv.org/abs/2303.13511.

Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, and Ming-Hsuan Yang. Learning blind video temporal consistency. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

Maria Larchenko, Alexander Lobashev, Dmitry Guskov, and Vladimir Vladimirovich Palyulin. Color transfer with modulated flows. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 4464–4472, 2025.

Mujing Li, Guanjie Wang, Xingguang Zhang, Qifeng Liao, and Chenxi Xiao. D-lut: Photorealistic style transfer via diffusion process. In *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, pp. 9188–9196, February 2025.

Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Universal style transfer via feature transforms, 2017. URL https://arxiv.org/abs/1705.08086.

Yijun Li, Ming-Yu Liu, Xueting Li, Ming-Hsuan Yang, and Jan Kautz. A closed-form solution to photorealistic image stylization, 2018. URL https://arxiv.org/abs/1802.06474.

Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22160–22169, 2024.

Hanyuan Liu, Chengze Li, Tien-Tsin Wong, and Xueting Liu. Chromaflow: End-to-end flow matching for efficient video colorization. In *Proceedings of the SIGGRAPH Asia 2025 Technical Communications*, SA Technical Communications '25, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400721366. doi: 10.1145/3757376.3771391. URL https://doi.org/10.1145/3757376.3771391.

Qiang Liu. Rectified flow: A marginal preserving approach to optimal transport, 2022. URL https://arxiv.org/abs/2209.14577.

Shiguang Liu and Yu Zhang. Temporal-consistency-aware video color transfer. pp. 464–476, Berlin, Heidelberg, 2021. Springer-Verlag. ISBN 978-3-030-89028-5. doi: 10.1007/978-3-030-89029-2_36.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow, 2022. URL https://arxiv.org/abs/2209.03003.

Yihao Liu, Hengyuan Zhao, Kelvin C. K. Chan, Xintao Wang, Chen Change Loy, Yu Qiao, and Chao Dong. Temporally consistent video colorization with deep feature propagation and self-regularization learning, 2021. URL https://arxiv.org/abs/2110.04562.

Fujun Luan, Sylvain Paris, Eli Shechtman, and Kavita Bala. Deep photo style transfer, 2017. URL https://arxiv.org/abs/1703.07511.

James Munkres. Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics*, 5(1):32–38, 1957.

Seungjun Nah, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte, and Kyoung Mu Lee. Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1996–2005. IEEE, 2019.

F. Pitie and A. Kokaram. The linear monge-kantorovitch linear colour mapping for example-based colour transfer. In *4th European Conference on Visual Media Production*, pp. 1–9, 2007. doi: 10.1049/cp:20070055.

François Pitié, Anil C. Kokaram, and Rozenn Dahyot. Automated colour grading using colour distribution transfer. *Computer Vision and Image Understanding*, 107(1):123–137, 2007. doi: https://doi.org/10.1016/j.cviu.2006.11.011.

E. Reinhard, M. Adhikhmin, B. Gooch, and P. Shirley. Color transfer between images. *IEEE Computer Graphics and Applications*, 21(5):34–41, 2001. doi: 10.1109/38.946629.

Manuel Ruder, Alexey Dosovitskiy, and Thomas Brox. Artistic style transfer for videos. In *Pattern Recognition*, pp. 26–36. Springer, 2016.

Seunghyun Shin, Dongmin Shin, Jisu Shin, Hae-Gon Jeon, and Joon-Young Lee. Video color grading via look-up table generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19141–19152, October 2025.

Mingxing Tan and Quoc Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 6105–6114. PMLR, 09–15 Jun 2019.

Cédric Villani. *Optimal transport – Old and new*, volume 338, pp. xxii+973. 01 2008. doi: 10.1007/978-3-540-71050-9.

Linfeng Wen, Chengying Gao, and Changqing Zou. Cap-vstnet: Content affinity preserved versatile style transfer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18300–18309, 2023.

Zijie Wu, Zhen Zhu, Junping Du, and Xiang Bai. Ccpl: Contrastive coherence preserving loss for versatile style transfer. In *European conference on computer vision*, pp. 189–206. Springer, 2022.

Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *Proceedings of the IEEE international conference on computer vision*, pp. 1395–1403, 2015.

Zhucun Xue, Jiangning Zhang, Teng Hu, Haoyang He, Yinan Chen, Yuxuan Cai, Yabiao Wang, Chengjie Wang, Yong Liu, Xiangtai Li, et al. Ultravideo: High-quality uhd video dataset with comprehensive captions. *arXiv preprint arXiv:2506.13691*, 2025.

Chun-Han Yao, Chia-Yang Chang, and Shao-Yi Chien. Example-based video color transfer. In *2016 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, 2016. doi: 10.1109/ICME.2016.7552926.

Jaejun Yoo, Youngjung Uh, Sanghyuk Chun, Byeongkyu Kang, and Jung-Woo Ha. Photorealistic style transfer via wavelet transforms, 2019. URL https://arxiv.org/abs/1903.09760.

Bo Zhang, Mingming He, Jing Liao, Pedro V. Sander, Lu Yuan, Amine Bermak, and Dong Chen. Deep exemplar-based video colorization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Shiwen Zhang, Xiaoyan Yang, Bojia Zi, Haibin Huang, Chi Zhang, and Xuelong Li. Telestyle: Content-preserving style transfer in images and videos, 2026. URL https://arxiv.org/abs/2601.20175.

A Appendix

A.1 Visualization of Color Transformation Trajectories

We study 3D visualizations of color cluster trajectories in Fig. 5 to further validate the color transfer performance of TCFlow. This visualization illustrates the distribution of three representative color clusters G1, G2, and G3 within their respective latent and reference spaces.

In the projection to the uniform space, ModFlows exhibits broad and sparse cluster distributions, whereas TCFlow aggregates the same pixel groups into highly compact and consistent locations. This visual density aligns with the lower standard deviation values reported in Table 3 of the main manuscript, demonstrating that TCFlow effectively preserves semantic information even under complex color transformations.

Similarly, in the reference space, TCFlow maintains the structural cohesion of each cluster while accurately converging toward the color coordinates of the reference space. The high cohesion observed in these latent spaces suppresses abrupt color fluctuations between frames. Consequently, the stable cluster formation identified in this visualization confirms that TCFlow ensures superior spatio-temporal continuity in the final video color transfer results.

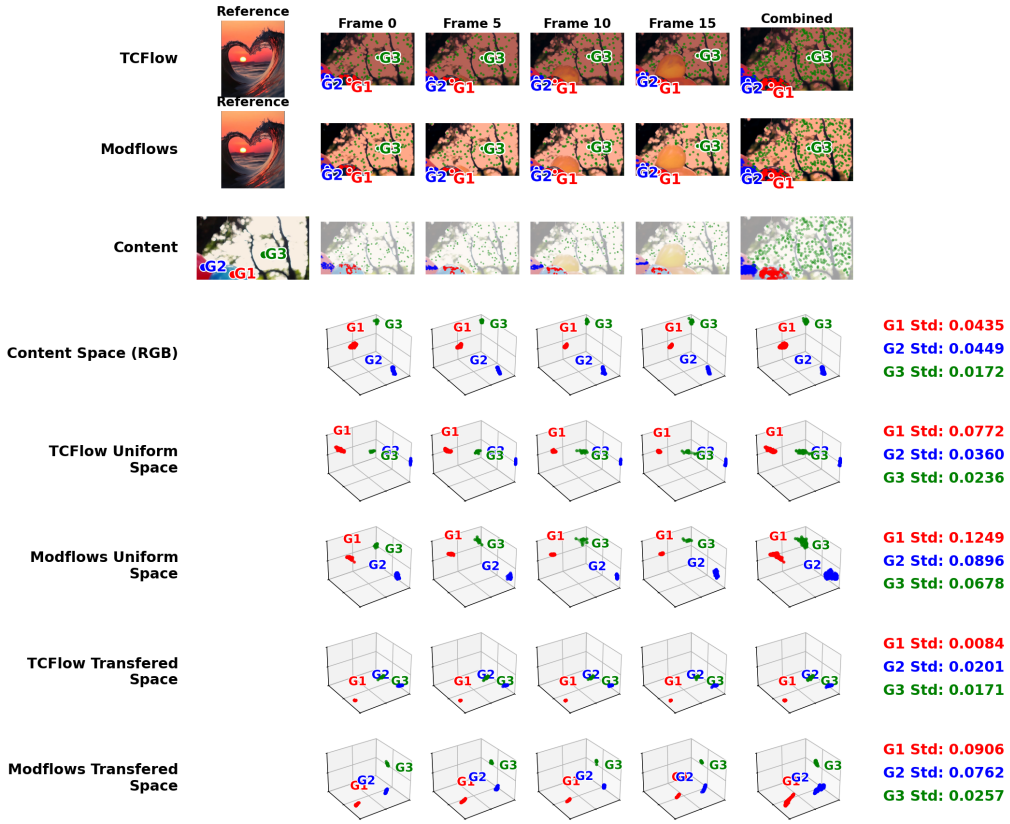

Figure 5: 3D visualization of color cluster trajectories

A.2 Canonical Color Space Analysis and Reference Sensitivity

A.2.1 Canonical Latent Space Validation.

To evaluate whether the uniform latent distribution behaves as a canonical color space, we measured the consistency of the corresponding latent representations using the same scene rendered with six different color styles. As summarized in Table 4, the transported latent space reduces the average pairwise L_2 distance by 38.3% and the style-induced variance by $5.81\times$ compared to the original

Table 4: **Latent consistency across different color styles.** We evaluate pairwise L_2 distance and style-induced variance between original RGB space and transported Latent space.

Metric	RGB	Latent	Improvement
Pairwise L_2 Distance ↓	0.3710	0.2289	$38.3\% \downarrow$
Style-induced Variance ↓	$1.00\times$	$0.1721\times$	$5.81\times$ reduction

Table 5: **Reference sensitivity evaluation using adjacent reference frames.**

Metric	Value
Pearson's $r \uparrow$	0.779
Regression slope ↓	0.393
Sensitivity ratio ↓	0.267 ± 0.084

RGB space. These results demonstrate that corresponding pixels across diverse color styles are mapped to substantially more consistent latent coordinates, supporting the effectiveness of the uniform latent space as a canonicalized color representation.

A.2.2 Reference Sensitivity and Continuity.

We also evaluated the model's sensitivity to reference frame selection using adjacent frames from the same reference video. For consecutive reference frames $\{R_i, R_{i+1}\}_{i=1}^{M-1}$ and their corresponding transferred output videos $\{\widehat{V}_i, \widehat{V}_{i+1}\}$, we define the input perturbation $\Delta_{\text{in}}^{(i)} = \mathcal{D}(R_i, R_{i+1})$ and the output video discrepancy $\Delta_{\text{out}}^{(i)} = \frac{1}{N}\sum_{n=1}^{N} \mathcal{D}(\widehat{V}_{i,n}, \widehat{V}_{i+1,n})$, where $\mathcal{D}(\cdot,\cdot)$ denotes the normalized RGB L_2 distance. To quantify linear dependency, we compute Pearson's correlation coefficient r:

$$r = \frac{\sum_{i=1}^{M-1}\left(\Delta_{\text{in}}^{(i)} - \bar{\Delta}_{\text{in}}\right)\left(\Delta_{\text{out}}^{(i)} - \bar{\Delta}_{\text{out}}\right)}{\sqrt{\sum_{i=1}^{M-1}\left(\Delta_{\text{in}}^{(i)} - \bar{\Delta}_{\text{in}}\right)^2}\sqrt{\sum_{i=1}^{M-1}\left(\Delta_{\text{out}}^{(i)} - \bar{\Delta}_{\text{out}}\right)^2}}, \tag{16}$$

where $\bar{\Delta}_{\text{in}}$ and $\bar{\Delta}_{\text{out}}$ denote the sample means over all $M-1$ adjacent pairs. Additionally, the sensitivity ratio for pair i and its mean $\bar{S}$ are evaluated as:

$$S_i = \frac{\Delta_{\text{out}}^{(i)}}{\Delta_{\text{in}}^{(i)}}, \quad \bar{S} = \frac{1}{M-1}\sum_{i=1}^{M-1} S_i. \tag{17}$$

As detailed in Table 5, changes in the transferred outputs exhibit a strong linear correlation with changes in the reference frames ($r = 0.779, p < 0.001$). Furthermore, the regression slope ($0.393 < 1$) and low sensitivity ratio (0.267 ± 0.084) confirm that perturbations in the reference frame are propagated smoothly rather than amplified. These results indicate that TCFlow responds smoothly to minor variations in the selected reference frame without amplifying the induced changes.

A.3 Computational Efficiency Analysis

As summarized in Table 6, we report a detailed efficiency comparison against existing baselines in terms of parameters, FLOPs, latency, and throughput. TCFlow exhibits comparable computational complexity to ModFlows (230.89 vs. 230.69 GFLOPs) while substantially reducing inference latency by 45.35% (224.60 vs. 410.94 ms/frame) and achieving an 83.13% higher throughput (4.45 vs. 2.43 FPS). This efficiency gain stems from our clip-shared inference design: while ModFlows estimates a separate reference-conditioned flow per frame (requiring repeated flow inference across time), TCFlow predicts a single video-level flow from the entire clip and reuses the inferred mapping. Consequently, the flow inference cost is amortized over the clip, significantly accelerating processing while maintaining temporal consistency.

Compared to large-scale models like VideoColorGrading, TCFlow requires approximately 98.1% fewer parameters (35.49 M vs. $1,887.70$ M) and 79.4% fewer GFLOPs (230.89 vs. $1,122.09$), achiev-

Table 6: **Quantitative comparison of computational efficiency.** We report parameter counts, GFLOPs, latency (ms per frame), and throughput (FPS) across baseline methods measured on 1280×720 resolution with 16 frames.

Method	Params (M)	GFLOPs	Latency (ms)	FPS
CCPL (Wu et al., 2022)	2.05	728.34	42.57	23.49
CAP-VST (Wen et al., 2023)	4.09	1695.48	251.04	3.98
VideoColorGrading (Shin et al., 2025)	1887.70	1122.09	128.17	7.80
ModFlows (Larchenko et al., 2025)	32.25	230.69	410.94	2.43
TCFlow (Ours)	**35.49**	**230.89**	**224.60**	**4.45**

ing significantly lower computational overhead despite VideoColorGrading's lightweight architectural design tailored for fast frame-by-frame processing. Overall, TCFlow achieves expressive and temporally consistent color transfer with a favorable balance between output quality and computational complexity.

A.4 Details on Evaluation Metrics

Following Larchenko et al. (2025), the aggregated score evaluates the balance between content and color metrics by measuring the distance from an ideal reference point p in the joint score space:

$$\text{Aggregated Score} = \sqrt{(p - \text{Content Score})^2 + (p - \text{Color Score})^2}. \tag{18}$$

The content score captures structural fidelity through a combination of normalized grayscale images, DepthFM (Gui et al., 2025) depth maps, and HED (Xie & Tu, 2015) edge maps, which are robust to global color shifts.

To quantify temporal artifacts without requiring pixel-wise correspondence, we define the global color flickering metric $D_{\text{temp}}^{\text{global}}$ as the average discrepancy in temporal histogram shifts between the content V and the transferred output $\widehat{V}$. Specifically, to enhance sensitivity to subtle fluctuations, we apply a scaled log-transformation to the absolute distance discrepancy:

$$D_{\text{temp}}^{\text{global}} = \frac{1}{N-1} \sum_{n=1}^{N-1} \ln\left(1 + 100 \cdot \left| D_B(H_n^{\widehat{V}}, H_{n+1}^{\widehat{V}}) - D_B(H_n^{V}, H_{n+1}^{V}) \right|\right), \tag{19}$$

where $D_B(H_n, H_{n+1}) = \sqrt{1 - \sum_i \sqrt{H_n(i) \cdot H_{n+1}(i)}}$ represents the Bhattacharyya distance between the normalized RGB histograms H of adjacent frames. The factor 100 is used to amplify small discrepancies before the logarithmic transformation.

To complement this global measure, we additionally evaluate a patch-based color flickering metric $D_{\text{temp}}^{\text{patch}}$ that captures local temporal color consistency. Each frame is partitioned into K_p non-overlapping spatial patches $\{p^{(k)}\}_{k=1}^{K_p}$. We first compute the spatially averaged patch distance for each temporal transition as

$$\bar{D}_B^{V}(n, n+1) = \frac{1}{K_p} \sum_{k=1}^{K_p} D_B\left(H_n^{V,k}, H_{n+1}^{V,k}\right), \tag{20}$$

and similarly define

$$\bar{D}_B^{\widehat{V}}(n, n+1) = \frac{1}{K_p} \sum_{k=1}^{K_p} D_B\left(H_n^{\widehat{V},k}, n+1^{\widehat{V},k}\right). \tag{21}$$

The patch-based temporal discrepancy is then measured as

$$D_{\text{temp}}^{\text{patch}} = \frac{1}{N-1} \sum_{n=1}^{N-1} \ln \left(1 + 100 \cdot \left| \bar{D}_B^{\widehat{V}}(n, n+1) - \bar{D}_B^{V}(n, n+1) \right| \right). \tag{22}$$

Here, $H_n^{V,k}$ and $H_n^{\widehat{V},k}$ denote the RGB histograms of the k-th patch at frame n for the content and output videos, respectively.

Furthermore, we estimate the average Lipschitz constant L of the learned color mapping f_ϕ, defined by the regularity condition $\|f_\phi(x_1) - f_\phi(x_2)\| \leq L\|x_1 - x_2\|$. A lower Lipschitz constant L indicates a smoother mapping with reduced sensitivity to input perturbations.

All metrics are calculated independently per frame and subsequently averaged. To ensure the reliability of the reported results, we compute standard deviations using the bootstrap method, which allows for a robust estimation of variability across diverse video sequences.

A.5 Real-World Media Art Exhibition Benchmark

We curated a novel Real-world Media art Benchmark to evaluate the generalization capabilities and robustness of TCFlow in demanding creative scenarios. This dataset comprises 135 high-fidelity video-reference pairs sourced from professional large-scale projection mappings, immersive installations, and 3D digital motion graphics. Unlike standard datasets, these sequences are characterized by extreme luminance fluctuations, high-contrast abstract structures, and non-linear color transitions, which are typical in high-stakes visual exhibitions. By leveraging actual exhibition data, we rigorously verify TCFlow's ability to maintain temporal constancy and structural integrity under adversarial lighting conditions.

A.5.1 Qualitative Results.

Figure 6 illustrates a visual comparison on a representative sequence from the Media Art benchmark. As highlighted in the red inset boxes, existing baselines exhibit clear visual failure modes: CCPL (Wu et al., 2022) applies visual patterns rigidly across localized regions, causing unnatural structural distortions that undermine 3D geometric realism. CAP-VST (Wen et al., 2023) suffers from severe color bleeding and detail blurring, where dynamic objects (highlighted in the zoomed-in boxes of Frames #6 and #7) lose sharp structural boundaries during frame transitions. VideoColorGrading (Shin et al., 2025) fails to reflect the expressive, vibrant hues of the reference image, yielding a muted and desaturated palette that lacks artistic impact. Furthermore, ModFlows (Larchenko et al., 2025) induce noticeable temporal instability; as indicated by the red box on the right vertical pillar, the illumination and background color tones shift abruptly between Frame #6 and Frame #10, causing visible color flickering. In sharp contrast, TCFlow accurately maps the rich blue and green reference palette across all scene elements while ensuring flawless temporal coherence and boundary preservation across consecutive video frames.

A.5.2 Quantitative Results.

These visual advantages are corroborated by the quantitative evaluation in Table 7. TCFlow achieves the best overall performance across all aggregated structural preservation metrics (Grayscale, Depth, and Edge scores). While CCPL and ModFlows achieve competitive color scores (such as MS-SWD and SWD, respectively), they suffer from significant transport instability. Specifically, CCPL exhibits an exceptionally high Lipschitz constant (138.31), whereas ModFlows yields high temporal flickering (1.1542), reflecting frame-wise color drift. Conversely, while VideoColorGrading exhibits low Lipschitz and flickering metrics due to its rigid LUT-based design, it yields poor color transfer metrics (SWD 0.3118, MS-SWD 4.99), failing to express the target palette. TCFlow successfully bridges this gap, suppressing temporal flickering to levels comparable with clip-level LUT grading while preserving high reference color fidelity.

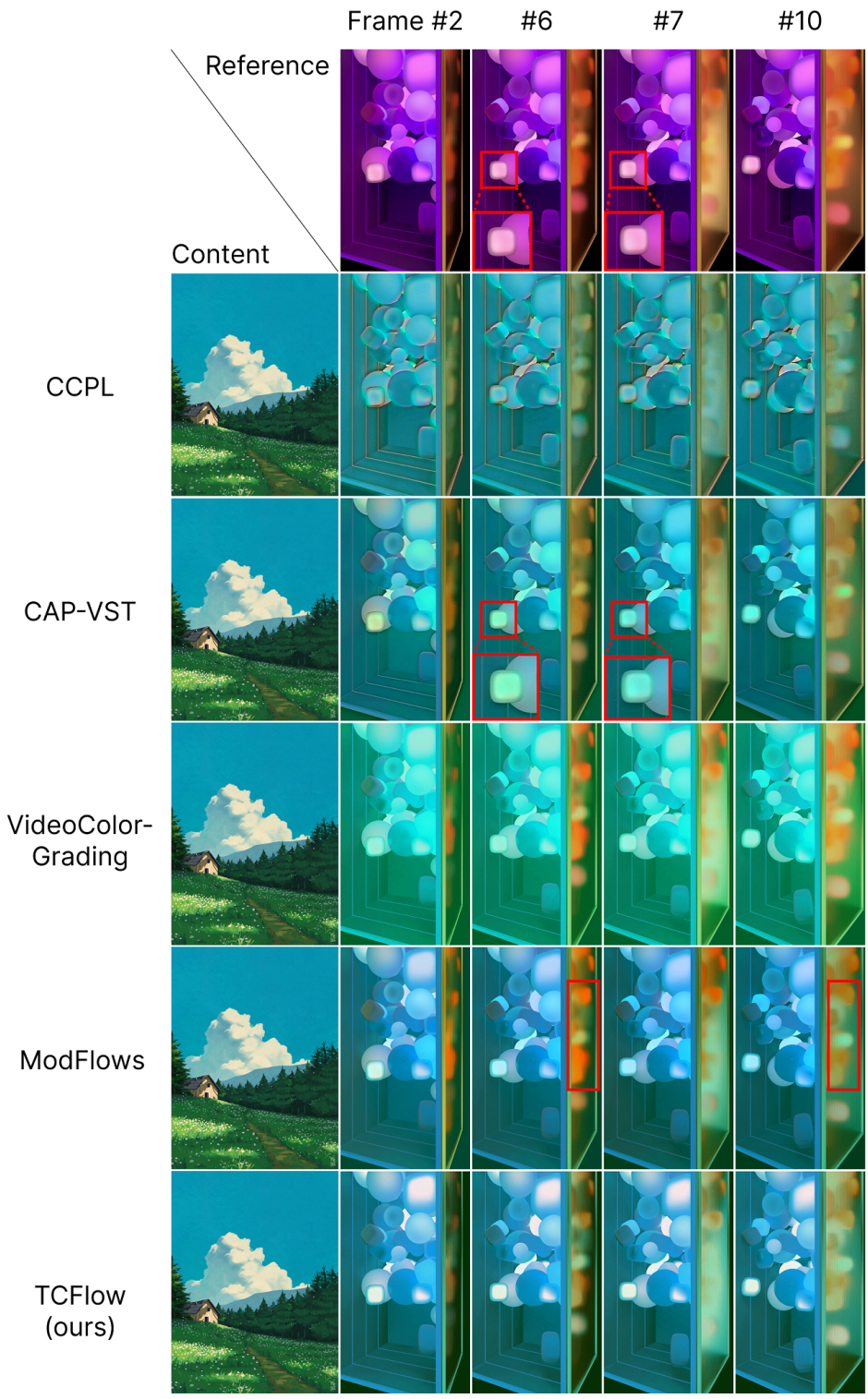

Figure 6: **Qualitative comparison on Media art Benckmark.** TCFlow achieves the best balance between accurate color transformation and structural preservation.

Table 7: **Quantitative comparison on the Media art Benchmark.** TCFlow achieves superior performance in aggregated scores. Performance is color coded: red for best, blue for second best.

Methods	Aggregated scores			Color scores		Lipschitz(↓)	$D_{\text{temp}}^{\text{global}}$ (↓)	$D_{\text{temp}}^{\text{patch}}$ (↓)
	Grayscale(↓)	Depth(↓)	Edge(↓)	SWD(↓)	MS-SWD(↓)			
CCPL (Wu et al., 2022)	0.2904 ± 0.0080	0.3156 ± 0.0093	0.2620 ± 0.0063	0.1260 ± 0.0046	2.60 ± 0.093	138.31 ± 4.46	0.9552 ± 0.0449	1.7070 ± 0.0535
CAP-VST (Wen et al., 2023)	0.2826 ± 0.0090	0.3516 ± 0.0093	0.2658 ± 0.0073	0.1771 ± 0.0073	3.09 ± 0.115	67.23 ± 3.13	0.8137 ± 0.0421	1.4046 ± 0.0530
VideoColorGrading (Shin et al., 2025)	0.3725 ± 0.0146	0.3983 ± 0.0137	0.3620 ± 0.0135	0.3118 ± 0.0145	4.99 ± 0.183	15.80 ± 1.18	0.4731 ± 0.0316	1.0024 ± 0.0513
ModFlows (Larchenko et al., 2025)	0.2416 ± 0.0104	0.2630 ± 0.0097	0.2163 ± 0.0076	0.1150 ± 0.0061	2.91 ± 0.133	47.90 ± 5.39	1.1542 ± 0.0747	1.6422 ± 0.0609
TCFlow (ours)	**0.2350 ± 0.0097**	**0.2565 ± 0.0092**	**0.2148 ± 0.0062**	**0.1164 ± 0.0056**	**2.82 ± 0.120**	**26.68 ± 1.60**	**0.4951 ± 0.0383**	**1.2170 ± 0.0533**

Table 8: **REDS benchmark**: Evaluation under strong camera motion. Performance is color coded: red for best, blue for second best.

Methods	Aggregated scores			Color scores		Lipschitz(↓)	$D_{\text{temp}}^{\text{global}}$ (↓)	$D_{\text{temp}}^{\text{patch}}$ (↓)
	Grayscale(↓)	Depth(↓)	Edge(↓)	SWD(↓)	MS-SWD(↓)			
CCPL (Wu et al., 2022)	0.1346	0.1452	0.1543	0.0702	1.2647	83.29	1.3560	1.5954
CAP-VST (Wen et al., 2023)	0.1192	0.1311	0.1300	0.0775	1.3310	41.31	1.0956	1.3036
VideoColorGrading (Shin et al., 2025)	0.1533	0.1738	0.1729	0.1327	2.2476	13.35	0.7140	1.1953
ModFlows (Larchenko et al., 2025)	0.0794	0.1127	0.1140	0.0518	1.3026	43.06	1.2221	1.4301
TCFlow (ours)	**0.0794**	**0.1127**	**0.1153**	**0.0503**	**1.2349**	**10.45**	**0.7209**	**1.1564**

A.6 Robustness to Dynamic Scenes

To evaluate the generalization capability and mapping robustness of the proposed clip-shared flow under complex dynamic real-world scenes, we evaluate on two challenging video benchmarks: REDS (Nah et al., 2019) (featuring rapid camera motion and severe motion blur) and MOSE-v2 (Ding et al., 2025) (characterized by frequent object entrances/exits, dynamic occlusions, and complex movements).

As summarized in Table 8 and Table 9, TCFlow demonstrates robust spatial structure preservation and temporal consistency under severe motion and composition shifts.

On the REDS benchmark (Table 8), TCFlow achieves state-of-the-art color alignment (SWD of 0.0503 and MS-SWD of 1.2349) while exhibiting high transport stability, reducing the Lipschitz constant from 43.06 (ModFlows) to 10.45. Furthermore, TCFlow outperforms existing frame-wise and global baselines in localized temporal coherence, achieving the lowest patch-based flickering score ($D_{\text{temp}}^{\text{patch}} = 1.1564$).

Similarly, on the MOSE-v2 benchmark (Table 9), which features complex object movements and dynamic scene transitions, TCFlow maintains its superior performance. It achieves the best color distribution alignment (SWD of 0.0668) along with top-tier temporal stability ($D_{\text{temp}}^{\text{global}} = 0.1718$ and $D_{\text{temp}}^{\text{patch}} = 0.3594$).

A.7 More results

Due to space and file size constraints, we present only a representative result in the main text. We provide additional results demonstrating TCFlow transferring a single source content video with three distinct reference colors, along with their corresponding temporally consistent transferred results. Additional results are available at `https://tc-flow-iclr.vercel.app/`.

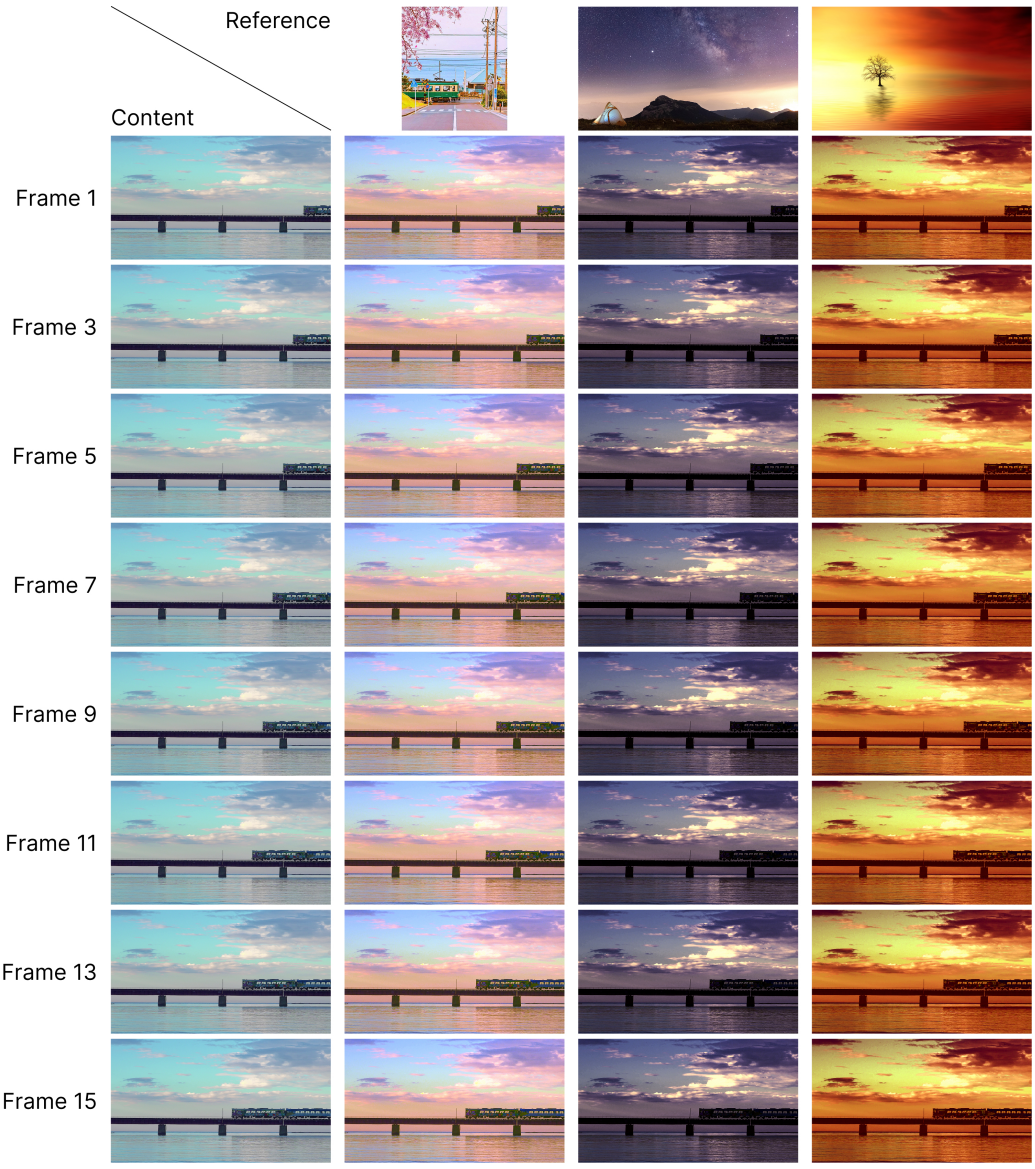

Figure 7: **Visual results for three different reference colors.**

Table 9: **MOSE-v2 benchmark**: Evaluation under object entrance/exit and dynamic camera motion. Performance is color coded: red for best, blue for second best.

Methods	Aggregated scores			Color scores		Lipschitz(↓)	$D_{\text{temp}}^{\text{global}}$ (↓)	$D_{\text{temp}}^{\text{patch}}$ (↓)
	Grayscale(↓)	Depth(↓)	Edge(↓)	SWD(↓)	MS-SWD(↓)			
CCPL (Wu et al., 2022)	0.2049	0.1868	0.2348	0.0772	1.5442	89.12	0.4899	1.0371
CAP-VST (Wen et al., 2023)	0.1786	0.1978	0.1962	0.0953	1.7238	41.90	0.3925	0.7914
VideoColorGrading (Shin et al., 2025)	0.2171	0.2432	0.2393	0.2010	3.2481	12.18	0.2107	0.4253
ModFlows (Larchenko et al., 2025)	0.1424	0.1459	0.1715	0.0691	1.8968	26.97	0.3112	0.4675
TCFlow (ours)	**0.1419**	**0.1459**	**0.1747**	**0.0668**	**1.7341**	**16.03**	**0.1718**	**0.3594**