Learning Blind Video Temporal Consistency

Wei-Sheng Lai[1(✉)], Jia-Bin Huang[2], Oliver Wang[3], Eli Shechtman[3], Ersin Yumer[4], and Ming-Hsuan Yang[1,5]

[1] UC Merced, Merced, USA
wlai24@ucmerced.edu
[2] Virginia Tech, Blacksburg, USA
[3] Adobe Research, Seattle, USA
[4] Argo AI, Mountain View, USA
[5] Google Cloud AI, Sunnyvale, USA

Abstract. Applying image processing algorithms independently to each frame of a video often leads to undesired inconsistent results over time. Developing temporally consistent video-based extensions, however, requires domain knowledge for individual tasks and is unable to generalize to other applications. In this paper, we present an efficient approach based on a deep recurrent network for enforcing temporal consistency in a video. Our method takes the original and per-frame processed videos as inputs to produce a temporally consistent video. Consequently, our approach is agnostic to specific image processing algorithms applied to the original video. We train the proposed network by minimizing both short-term and long-term temporal losses as well as a perceptual loss to strike a balance between temporal coherence and perceptual similarity with the processed frames. At test time, our model does not require computing optical flow and thus achieves real-time speed even for high-resolution videos. We show that our single model can handle multiple and unseen tasks, including but not limited to artistic style transfer, enhancement, colorization, image-to-image translation and intrinsic image decomposition. Extensive objective evaluation and subject study demonstrate that the proposed approach performs favorably against the state-of-the-art methods on various types of videos.

1 Introduction

Recent advances of deep convolutional neural networks (CNNs) have led to the development of many powerful image processing techniques including, image filtering [30,37], enhancement [10,24,38], style transfer [17,23,29], colorization [19,41], and general image-to-image translation tasks [21,27,43]. However,

Electronic supplementary material The online version of this chapter (https://doi.org/10.1007/978-3-030-01267-0_11) contains supplementary material, which is available to authorized users.

V. Ferrari et al. (Eds.): ECCV 2018, LNCS 11219, pp. 179–195, 2018.
https://doi.org/10.1007/978-3-030-01267-0_11

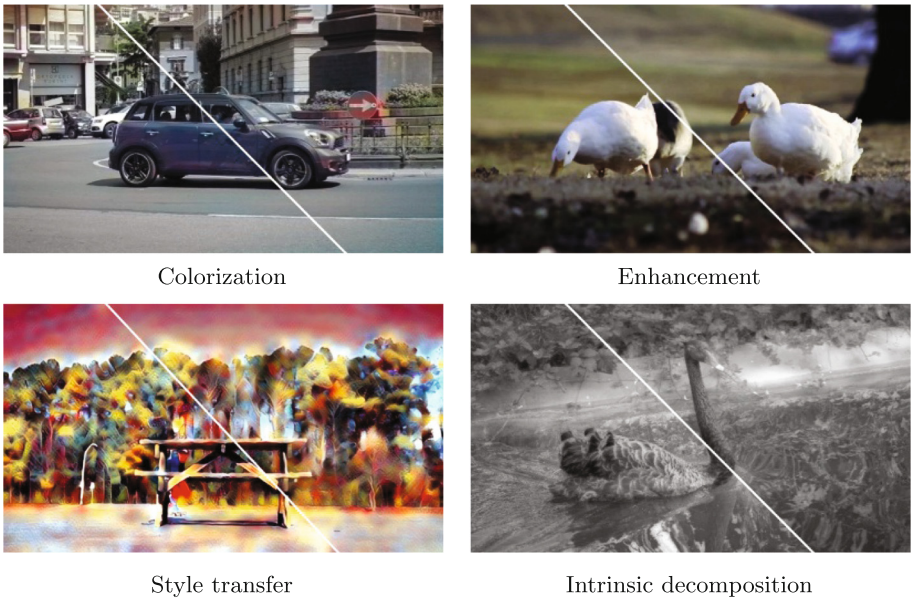


Fig. 1. Applications of the proposed method. Our algorithm takes per-frame processed videos with serious temporal flickering as inputs (lower-left) and generates temporally stable videos (upper-right) while maintaining perceptual similarity to the processed frames. Our method is blind to the specific image processing algorithm applied to input videos and runs at a high frame-rate. This figure contains *animated videos* (see supplementary material).

extending these CNN-based methods to video is non-trivial due to memory and computational constraints, and the availability of training datasets. Applying image-based algorithms independently to each video frame typically leads to temporal flickering due to the instability of global optimization algorithms or highly non-linear deep networks. One approach for achieving temporally coherent results is to explicitly embed flow-based temporal consistency loss in the design and training of the networks. However, such an approach suffers from two drawbacks. First, it requires domain knowledge to re-design the algorithm [1,16], re-train a deep model [12,15], and video datasets for training. Second, due to the dependency of flow computation at test time, these approaches tend to be slow.

Bonneel et al. [6] propose a general approach to achieve temporal coherent results that is *blind* to specific image processing algorithms. The method takes the original video and the per-frame processed video as inputs and solves a gradient-domain optimization problem to minimize the temporal warping error between consecutive frames. Although the results of Bonneel et al. [6] are temporally stable, their algorithm highly depends on the quality of dense correspondence (e.g., optical flow or PatchMatch [2]) and may fail when a severe occlusion

occurs. Yao et al. [39] further extend the method of Bonneel et al. [6] to account for occlusion by selecting a set of key-frames. However, the computational cost increases linearly with the number of key-frames, and thus their approach cannot be efficiently applied to long video sequences. Furthermore, both approaches assume that the gradients of the original video are similar to the gradients of the processed video, which restricts them from handling tasks that may generate new contents (e.g., stylization).

In this work, we formulate the problem of video temporal consistency as a learning task. We propose to learn a deep recurrent network that takes the input and processed videos and generates temporally stable output videos. We minimize the short-term and long-term temporal losses between output frames and impose a perceptual loss from the pre-trained VGG network [34] to maintain the perceptual similarity between the output and processed frames. In addition, we embed a convolutional LSTM (ConvLSTM) [36] layer to capture the spatial-temporal correlation of natural videos. Our network processes video frames sequentially and can be applied to videos with arbitrary lengths. Furthermore, our model does not require computing optical flow at *test* time and thus can process videos at real-time rates (400+ FPS on 1280×720 videos).

As existing video datasets typically contain low-quality frames, we collect a high-quality video dataset with 80 videos for training and 20 videos for evaluation. We train our model on a wide range of applications, including colorization, image enhancement, and artistic style transfer, and demonstrate that a *single* trained model generalizes well to *unseen* applications (e.g., intrinsic image decomposition, image-to-image translation, see Fig. 1). We evaluate the quality of the output videos using temporal warping error and a learned perceptual metric [42]. We show that the proposed method strikes a good balance between maintaining the temporal stability and perceptual similarity. Furthermore, we conduct a user study to evaluate the subjective preference between the proposed method and state-of-the-art approaches.

We make the following contributions in this work:

1. We present an efficient solution to remove temporal flickering in videos via learning a deep network with a ConvLSTM module. Our method does not require pre-computed optical flow or frame correspondences at *test* time and thus can process videos in real-time.
2. We propose to minimize the short-term and long-term temporal loss for improving the temporal stability and adopt a perceptual loss to maintain the perceptual similarity.
3. We provide a *single* model for handling multiple applications, including but not limited to colorization, enhancement, artistic style transfer, image-to-image translation and intrinsic image decomposition. Extensive subject and objective evaluations demonstrate that the proposed algorithm performs favorably against existing approaches on various types of videos.

Table 1. Comparison of blind temporal consistency methods. Both the methods of Bonneel et al. [6] and Yao et al. [39] require dense correspondences from optical flow or PatchMatch [2], while the proposed method does not explicitly rely on these correspondences at test time. The algorithm of Yao et al. [39] involves a key-frame selection from the entire video and thus cannot generate output in an online manner.

	Bonneel et al. [6]	Yao et al. [39]	Ours
Content constraint	Gradient	Local affine	Perceptual loss
Short-term temporal constraint	✓	–	✓
Long-term temporal constraint	–	✓	✓
Require dense correspondences (at test time)	✓	✓	–
Online processing	✓	–	✓

2 Related Work

We address the temporal consistency problem on a wide range of applications, including automatic white balancing [14], harmonization [4], dehazing [13], image enhancement [10], style transfer [17,23,29], colorization [19,41], image-to-image translation [21,43], and intrinsic image decomposition [3]. A complete review of these applications is beyond the scope of this paper. In the following, we discuss task-specific and task-independent approaches that enforce temporal consistency on videos.

Task-Specific Approaches. A common solution to embed the temporal consistency constraint is to use optical flow to propagate information between frames, e.g., colorization [28] and intrinsic decomposition [40]. However, estimating optical flow is computationally expensive and thus is impractical to apply on high-resolution and long sequences. Temporal filtering is an efficient approach to extend image-based algorithms to videos, e.g., tone-mapping [1], color transfer [5], and visual saliency [25] to generate temporally consistent results. Nevertheless, these approaches assume a specific filter formulation and cannot be generalized to other applications.

Recently, several approaches have been proposed to improve the temporal stability of CNN-based image style transfer. Huang et al. [15] and Gupta et al. [12] train feed-forward networks by jointly minimizing content, style and temporal warping losses. These methods, however, are limited to the specific styles used during training. Chen et al. [7] learn flow and mask networks to adaptively blend the intermediate features of the pre-trained style network. While the architecture design is independent of the style network, it requires the access to intermediate features and cannot be applied to non-differentiable tasks. In contrast, the proposed model is entirely blind to specific algorithms applied to the input frames and thus is applicable to optimization-based techniques, CNN-based algorithms, and combinations of Photoshop filters.

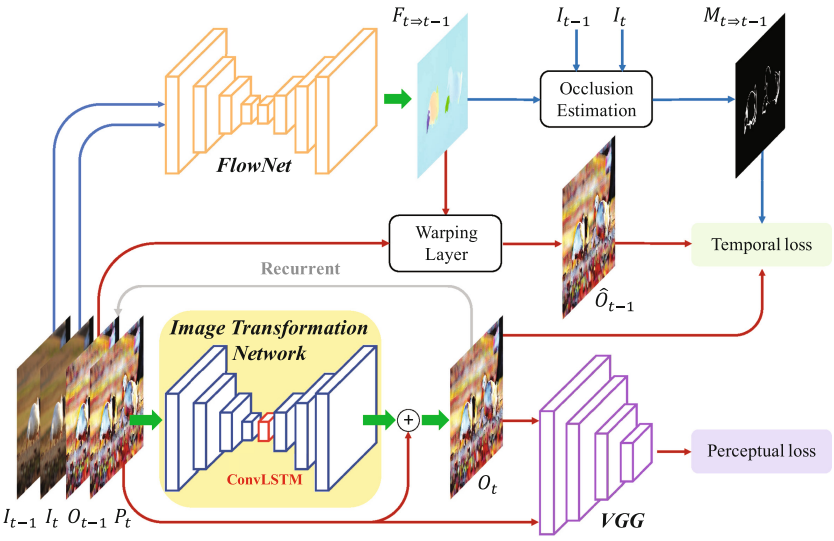


Fig. 2. Overview of the proposed method. We train an image transformation network that takes I_{t-1}, I_t, O_{t-1} and processed frame P_t as inputs and generates the output frame O_t which is temporally consistent with the output frame at the previous time step O_{t-1}. The output O_t at the current time step then becomes the input at the next time step. We train the image transformation network with the VGG perceptual loss and the short-term and long-term temporal losses.

Task-Independent Approaches. Several methods have been proposed to improve temporal consistency for multiple tasks. Lang et al. [25] approximate global optimization of a class of energy formulation (e.g., colorization, optical flow estimation) via temporal edge-aware filtering. In [9], Dong et al. propose a segmentation-based algorithm and assume that the image transformation is spatially and temporally consistent. More general approaches assume gradient similarity [6] or local affine transformation [39] between the input and the processed frames. These methods, however, cannot handle more complicated tasks (e.g., artistic style transfer). In contrast, we use the VGG perceptual loss [23] to impose high-level perceptual similarity between the output and processed frames. We list the feature-by-feature comparisons between Bonneel et al. [6], Yao et al. [39] and the proposed method in Table 1.

3 Learning Temporal Consistency

In this section, we describe the proposed recurrent network and the design of the loss functions for enforcing temporal consistency on videos.

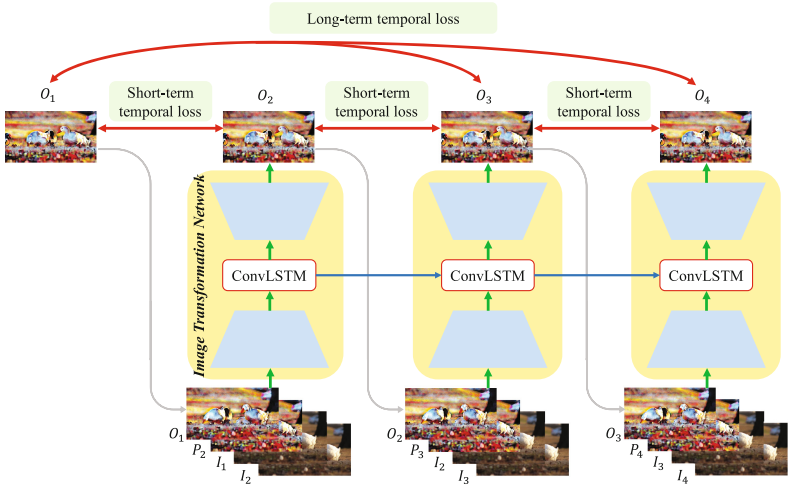


Fig. 3. Temporal losses. We adopt the short-term temporal loss on neighbor frames and long-term temporal loss between the first and all the output frames.

3.1 Recurrent Network

Figure 2 shows an overview of the proposed recurrent network. Our model takes as input the original (unprocessed) video $\{I_t|t = 1 \cdots T\}$ and per-frame processed videos $\{P_t|t = 1 \cdots T\}$, and produces temporally consistent output videos $\{O_t|t = 1 \cdots T\}$. In order to efficiently process videos with arbitrary length, we develop an image transformation network as a *recurrent convolutional network* to generate output frames in an online manner (i.e., sequentially from $t = 1$ to T). Specifically, we set the first output frame $O_1 = P_1$. In each time step, the network learns to generate an output frame O_t that is temporally consistent with respect to O_{t-1}. The current output frame is then fed as the input at the next time step. To capture the spatial-temporal correlation of videos, we integrate a ConvLSTM layer [36] into our image transformation network. We discuss the detailed design of our image transformation network in Sect. 3.3.

3.2 Loss Functions

Our goal is to reduce the temporal inconsistency in the output video while maintaining the perceptual similarity with the processed frames. Therefore, we propose to train our model with (1) a perceptual content loss between the output frame and the processed frame and (2) short-term and long-term temporal losses between output frames.

Content Perceptual Loss. We compute the similarity between O_t and P_t using the perceptual loss from a pre-trained VGG classification network [34], which

is commonly adopted in several applications (e.g., style transfer [23], super-resolution [26], and image inpainting [31]) and has been shown to correspond well to human perception [42]. The perceptual loss is defined as:

$$\mathcal{L}_p = \sum_{t=2}^{T} \sum_{i=1}^{N} \sum_{l} \left\| \phi_l(O_t^{(i)}) - \phi_l(P_t^{(i)}) \right\|_1, \tag{1}$$

where $O_t^{(i)}$ represents a vector $\in R^3$ with RGB pixel values of the output O at time t, N is the total number of pixels in a frame, and $\phi_l(\cdot)$ denotes the feature activation at the l-th layer of the VGG-19 network ϕ. We choose the 4-th layer (i.e., `relu4-3`) to compute the perceptual loss.

Short-Term Temporal Loss. We formulate the temporal loss as the warping error between the output frames:

$$\mathcal{L}_{st} = \sum_{t=2}^{T} \sum_{i=1}^{N} M_{t\Rightarrow t-1}^{(i)} \left\| O_t^{(i)} - \hat{O}_{t-1}^{(i)} \right\|_1, \tag{2}$$

where $\hat{O}_{t-1}$ is the frame O_{t-1} warped by the optical flow $F_{t\Rightarrow t-1}$, and $M_{t\Rightarrow t-1} = \exp(-\alpha \|I_t - \hat{I}_{t-1}\|_2^2)$ is the visibility mask calculated from the warping error between input frames I_t and warped input frame $\hat{I}_{t-1}$. The optical flow $F_{t\Rightarrow t-1}$ is the backward flow between I_{t-1} and I_t. We use the FlowNet2 [20] to efficiently compute flow on-the-fly during training. We use the bilinear sampling layer [22] to warp frames and empirically set $\alpha = 50$ (with pixel range between $[0, 1]$).

Long-Term Temporal Loss. While the short-term temporal loss Eq. 2 enforces the temporal consistency between consecutive frames, there is no guarantee for long-term (e.g., more than 5 frames) coherence. A straightforward method to enforce long-term temporal consistency is to apply the temporal loss on *all* pairs of output frames. However, such a strategy requires significant computational costs (e.g., optical flow estimation) during training. Furthermore, computing temporal loss between two intermediate outputs is not meaningful before the network converges.

Instead, we propose to impose long-term temporal losses between the *first* output frame and all of the output frames:

$$\mathcal{L}_{lt} = \sum_{t=2}^{T} \sum_{i=1}^{N} M_{t\Rightarrow 1}^{(i)} \left\| O_t^{(i)} - \hat{O}_1^{(i)} \right\|_1. \tag{3}$$

We illustrate an unrolled version of our recurrent network as well as the short-term and long-term losses in Fig. 3. During the training, we enforce the long-term temporal coherence over a maximum of 10 frames ($T = 10$).

Overall Loss. The overall loss function for training our image transformation network is defined as:

$$\mathcal{L} = \lambda_p \mathcal{L}_p + \lambda_{st} \mathcal{L}_{st} + \lambda_{lt} \mathcal{L}_{lt}, \tag{4}$$

where λ_p, λ_{st} and λ_{lt} are the weights for the content perceptual loss, short-term and long-term losses, respectively.

3.3 Image Transformation Network

The input of our image transformation network is the concatenation of the currently processed frame P_t, previous output frame O_{t-1} as well as the current and previous unprocessed frames I_t, I_{t-1}. As the output frame typically looks similar to the currently processed frame, we train the network to predict the *residuals* instead of actual pixel values, i.e., $O_t = P_t + \mathcal{F}(P_t)$, where $\mathcal{F}$ denotes the image transformation network. Our image transformation network consists of two strided convolutional layers, B residual blocks, one ConvLSTM layer, and two transposed convolutional layers.

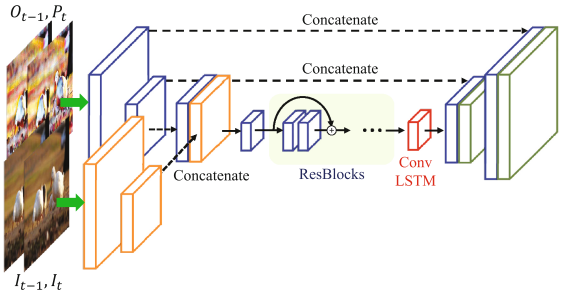


Fig. 4. Architecture of our image transformation network. We split the input into two streams to avoid transferring low-level information from the input frames to output.

We add skip connections from the encoder to the decoder to improve the reconstruction quality. However, for some applications, the processed frames may have a dramatically different appearance than the input frames (e.g., style transfer or intrinsic image decomposition). We observe that the skip connections may transfer low-level information (e.g., color) to the output frames and produce visual artifacts. Therefore, we divide the input into two streams: one for the processed frames P_t and O_{t-1}, and the other stream for input frames I_t and I_{t-1}. As illustrated in Fig. 4, the skip connections only add skip connections from the processed frames to avoid transferring the low-level information from the input frames. We provide all the implementation details in the supplementary material.

4 Experimental Results

In this section, we first describe the employed datasets for training and testing, followed by the applications of the proposed method and the metrics for evaluating the temporal stability and perceptual similarity. We then analyze the effect of each loss term in balancing the temporal coherence and perceptual similarity, conduct quantitative and subjective comparisons with existing approaches, and finally discuss the limitations of our method. The source code and datasets are publicly available at http://vllab.ucmerced.edu/wlai24/video_consistency.

4.1 Datasets

We use the DAVIS-2017 dataset [32], which is designed for video segmentation and contains a variety of moving objects and motion types. The DAVIS dataset has 60 videos for training and 30 videos for validation. However, the lengths of the videos in the DAVIS dataset are usually short (less than 3 s) with 4,209 training frames in total. Therefore, we collect additional 100 high-quality videos from Videvo.net [35], where 80 videos are used for training and 20 videos for testing. We scale the height of video frames to 480 and keep the aspect ratio. We use both the DAVIS and VIDEVO training sets, which contains a total of 25,735 frames, to train our network.

4.2 Applications

As we do not make any assumptions on the underlying image-based algorithms, our method is applicable for handling a wide variety of applications.

Artistic Style Transfer. The tasks of image style transfer have been shown to be sensitive to minor changes in content images due to the non-convexity of the Gram matrix matching objective [12]. We apply our method to the results from the state-of-the-art style transfer approaches [23,29].

Colorization. Single image colorization aims to hallucinate plausible colors from a given grayscale input image. Recent algorithms [19,41] learn deep CNNs from millions of natural images. When applying colorization methods to a video frame-by-frame, those approaches typically produce low-frequency flickering.

Image Enhancement. Gharbi et al. [10] train deep networks to learn the user-created action scripts of Adobe Photoshop for enhancing images. Their models produce high-frequency flickering on most of the videos.

Intrinsic Image Decomposition. Intrinsic image decomposition aims to decompose an image into a reflectance and a shading layer. The problem is highly ill-posed due to the scale ambiguity. We apply the approach of Bell et al. [3] to our test videos. As expected, the image-based algorithm produces serious temporal flickering artifacts when applied to each frame in the video independently.

Image-to-Image Translation. In recent years, the image-to-image translation tasks attract considerable attention due to the success of the Generative Adversarial Networks (GAN) [11]. The CycleGAN model [43] aims to learn mappings from one image domain to another domain without using paired training data. When the transformations generate a new texture on images (e.g., photo $\rightarrow$ painting, horse $\rightarrow$ zebra) or the mapping contains multiple plausible solutions (e.g., gray $\rightarrow$ RGB), the resulting videos inevitably suffer from temporal flickering artifacts.

The above algorithms are general and can be applied to any type of videos. When applied, they produce temporal flickering artifacts on most videos in our test sets. We use the WCT [29] style transfer algorithm with three style images, one of the enhancement models of Gharbi et al. [10], the colorization method of

Zhang et al. [41] and the shading layer of Bell et al. [3] as our training tasks, with the rest of the tasks being used for testing purposes. We demonstrate that the proposed method learns a *single* model for multiple applications and also generalizes to *unseen* tasks.

4.3 Evaluation Metrics

Our goal is to generate a temporally smooth video while maintaining the perceptual similarity with the per-frame processed video. We use the following metrics to measure the temporal stability and perceptual similarity on the output videos.

Temporal Stability. We measure the temporal stability of a video based on the flow warping error between two frames:

$$E_{\text{warp}}(V_t, V_{t+1}) = \frac{1}{\sum_{i=1}^{N} M_t^{(i)}} \sum_{i=1}^{N} M_t^{(i)} \|V_t^{(i)} - \hat{V}_{t+1}^{(i)}\|_2^2, \tag{5}$$

where $\hat{V}_{t+1}$ is the warped frame V_{t+1} and $M_t \in \{0, 1\}$ is a non-occlusion mask indicating non-occluded regions. We use the occlusion detection method in [33] to estimate the mask M_t. The warping error of a video is calculated as:

$$E_{\text{warp}}(V) = \frac{1}{T-1} \sum_{t=1}^{T-1} E_{\text{warp}}(V_t, V_{t+1}), \tag{6}$$

which is the average warping error over the entire sequence.

Perceptual Similarity. Recently, the features of the pre-trained VGG network [34] have been shown effective as a training loss to generate realistic images in several vision tasks [8,26,31]. Zhang et al. [42] further propose a perceptual metric by calibrating the deep features of ImageNet classification networks. We adopt the calibrated model of the SqueezeNet [18] (denote as $\mathcal{G}$) to measure the perceptual distance of the processed video P and output video O:

$$D_{\text{perceptual}}(P, O) = \frac{1}{T-1} \sum_{t=2}^{T} \mathcal{G}(O_t, P_t). \tag{7}$$

We note that the first frame is fixed as a reference in both Bonneel et al. [6] and our algorithm. Therefore, we exclude the first frame from computing the perceptual distance in Eq. 7.

4.4 Analysis and Discussions

An extremely blurred video may have high temporal stability but with low perceptual similarity; in contrast, the processed video itself has perfect perceptual similarity but is temporally unstable. Due to the trade-off between the temporal stability and perceptual similarity, it is important to balance these two properties and produce visually pleasing results.

To understand the relationship between the temporal and content losses, we train models with the several combinations of λ_p and λ_t ($= \lambda_{st} = \lambda_{lt}$). We use one of the styles (i.e., udnie) from the fast neural style transfer method [23] for evaluation. We show the quantitative evaluation on the DAVIS test set in Fig. 5. We observe that the ratio $r = \lambda_t/\lambda_p$ plays an important role in balancing the temporal stability and perceptual similarity. When the ratio $r < 10$, the perceptual loss dominates the optimization of the network, and the temporal flickering remains in the output videos. When the ratio $r > 10$, the output videos become overly blurred and therefore have a large perceptual distance to the processed videos. When λ_t is sufficiently large (i.e., $\lambda_t \geq 100$), the setting $r = 10$ strikes a good balance to reduce temporal flickering while maintaining small perceptual distance. Our results find similar observation on other applications as well.

λ_t	λ_p	$r = \frac{\lambda_t}{\lambda_p}$	E_{warp}	$D_{\text{perceptual}}$
10	0.01	1000	0.0279	0.1744
10	0.1	100	0.0265	0.1354
10	1	10	0.0615	0.0071
10	10	1	0.0621	0.0072
100	1	100	0.0277	0.1324
100	10	10	0.0442	0.0170
100	100	1	0.0621	0.0072
1000	1	1000	0.0262	0.1848
1000	10	100	0.0275	0.1341
1000	100	10	0.0453	0.0158
1000	1000	1	0.0621	0.0072

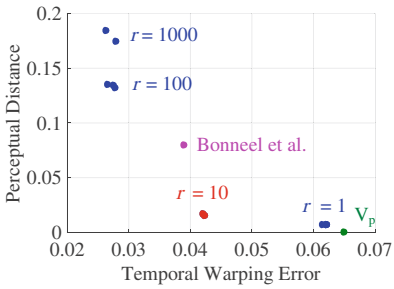


Fig. 5. Analysis of parameters. (Left) When λ_t is large enough, choosing $r = 10$ (shown in red) achieves a good balance between reducing temporal warping error as well as perceptual distance. (Right) The trade off between perceptual similarity and temporal warping with different ratios r, as compared to Bonneel et al. [6], and the original processed video, V_p.

(a) Original frames (b) Processed frames (c) Bonneel et al. [6] (d) Ours

Fig. 6. Visual comparisons on style transfer. We compare the proposed method with Bonneel et al. [6] on smoothing the results of WCT [29]. Our approach maintains the stylized effect of processed video and reduce the temporal flickering.

4.5 Comparison with State-of-the-Art Methods

We evaluate the temporal warping error Eq. 6 and perceptual distance Eq. 7 on the two video test sets. We compare the proposed method with Bonneel et al. [6] on 16 applications: 2 styles of Johnson et al. [23], 6 styles of WCT [29], 2 enhancement models of Gharbi et al. [10], reflectance and shading layers of Bell et al. [3], 2 photo-to-painting models of CycleGAN [43] and 2 colorization algorithms [19,41]. We provide the average temporal warping error and perceptual distance in Tables 2 and 3, respectively. In general, our results achieves lower perceptual distance while maintains comparable temporal warping error with the results of Bonneel et al. [6].

We show visual comparisons with Bonneel et al. [6] in Figs. 6 and 7. Although the method of Bonneel et al. [6] produces temporally stable results, the assumption of identical gradients in the processed and original video leads to overly smoothed contents, for example from stylization effects. Furthermore, when the occlusion occurs in a large region, their method fails due to the lack of a long-term temporal constraint. In contrast, the proposed method dramatically reduces the temporal flickering while maintaining the perceptual similarity with the

Table 2. Quantitative evaluation on temporal warping error. The "Trained" column indicates the applications used for training our model. Our method achieves a similarly reduced temporal warping error as Bonneel et al. [6], which is significantly less than the original processed video (V_p).

Task	Trained	DAVIS			VIDEVO		
		V_p	[6]	Ours	V_p	[6]	Ours
WCT [29]/antimono	✓	0.054	**0.031**	0.035	0.025	0.014	**0.013**
WCT [29]/asheville		0.088	**0.047**	0.055	0.045	0.025	**0.023**
WCT [29]/candy	✓	0.069	**0.037**	0.045	0.034	**0.018**	**0.018**
WCT [29]/feathers		0.052	**0.029**	**0.029**	0.027	0.016	**0.012**
WCT [29]/sketch	✓	0.046	0.028	**0.023**	0.022	0.015	**0.009**
WCT [29]/wave		0.049	0.030	**0.027**	0.026	0.015	**0.011**
Fast-neural-style [23]/princess		0.073	0.048	**0.047**	0.039	0.023	**0.021**
Fast-neural-style [23]/udnie		0.065	**0.039**	0.042	0.028	0.017	**0.015**
DBL [10]/expertA	✓	0.039	0.035	**0.028**	0.018	0.016	**0.010**
DBL [10]/expertB		0.034	0.031	**0.025**	0.015	0.014	**0.008**
Intrinsic [3]/reflectance		0.024	0.020	**0.015**	0.012	0.008	**0.005**
Intrinsic [3]/shading	✓	0.016	0.012	**0.009**	0.008	0.006	**0.003**
CycleGAN [43]/photo2ukiyoe		0.037	0.030	**0.026**	0.019	0.016	**0.010**
CycleGAN [43]/photo2vangogh		0.040	0.032	**0.029**	0.021	0.017	**0.013**
Colorization [41]	✓	0.030	0.028	**0.024**	0.012	0.011	**0.008**
Colorization [19]		0.030	0.028	**0.023**	0.012	0.011	**0.008**
Average		0.047	0.032	**0.030**	0.023	0.015	**0.012**

Table 3. Quantitative evaluation on perceptual distance. Our method has lower perceptual distance than Bonneel et al. [6].

Task	Trained	DAVIS		VIDEVO	
		[6]	Ours	[6]	Ours
WCT [29]/antimono	✓	0.098	**0.019**	0.106	**0.016**
WCT [29]/asheville		0.090	**0.019**	0.098	**0.015**
WCT [29]/candy	✓	0.133	**0.023**	0.139	**0.018**
WCT [29]/feathers		0.093	**0.016**	0.100	**0.011**
WCT [29]/sketch	✓	0.042	**0.021**	0.046	**0.014**
WCT [29]/wave		0.065	**0.015**	0.072	**0.013**
Fast-neural-style [23]/princess		0.143	**0.029**	0.165	**0.018**
Fast-neural-style [23]/udnie		0.070	**0.017**	0.076	**0.014**
DBL [10]/expertA	✓	0.026	**0.011**	0.033	**0.007**
DBL [10]/expertB		0.023	**0.011**	0.030	**0.007**
Intrinsic [3]/reflectance		0.044	**0.013**	0.056	**0.008**
Intrinsic [3]/shading	✓	0.029	**0.017**	0.032	**0.009**
CycleGAN [43]/photo2ukiyoe		0.042	**0.012**	0.054	**0.007**
CycleGAN [43]/photo2vangogh		0.067	**0.016**	0.079	**0.011**
Colorization [41]	✓	0.062	**0.013**	0.055	**0.009**
Colorization [19]		0.033	**0.011**	0.034	**0.008**
Average		0.088	**0.017**	0.073	**0.012**

processed videos. We note that our approach is not limited to the above applications but can also be applied to tasks such as automatic white balancing [14], image harmonization [4] and image dehazing [13]. Due to the space limit, we provide more results and videos on our project website.

4.6 Subjective Evaluation

We conduct a user study to measure user preference on the quality of videos. We adopt the pairwise comparison, i.e., we ask participants to choose from a pair of videos. In each test, we provide the original and processed videos as references and show two results (Bonneel et al. [6] and ours) for comparisons. We randomize the presenting order of the result videos in each test. In addition, we ask participants to provide the reasons that they prefer the selected video from the following options: (1) The video is less flickering. (2) The video preserves the effect of the processed video well.

We evaluate all 50 test videos with the 10 test applications that were held out during training. We ask each user to compare 20 video pairs and obtain results from a total of 60 subjects. Fig. 8(a) shows the percentage of obtained votes, where our approach is preferred on all 5 applications. In Fig. 8(b), we show the

reasons when a method is selected. The results of Bonneel et al. [6] are selected due to temporal stability, while users prefer our results as we preserve the effect of the processed video well. The observation in the user study basically follows the quantitative evaluation in Sect. 4.5.

4.7 Execution Time

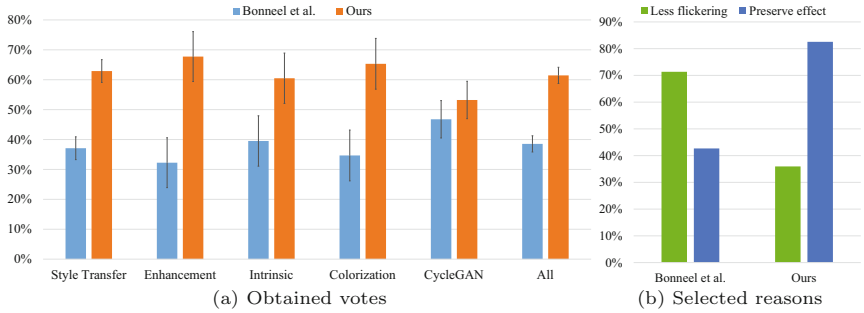
We evaluate the execution time of the proposed method and Bonneel et al. [6] on a machine with a 3.4 GHz Intel i7 CPU (64G RAM) and an Nvidia Titan X GPU. As the proposed method does not require computing optical flow at test time, the execution speed achieves 418 FPS on GPU for videos with a resolution of 1280×720. In contrast, the speed of Bonneel et al. [6] is 0.25 FPS on CPU.

4.8 Limitations and Discussion

Our approach is not able to handle applications that generate entirely different image content on each frame, e.g., image completion [31] or synthesis [8]. Extending those methods to videos would require incorporating strong video

(a) Original frames (b) Processed frames (c) Bonneel et al. [6] (d) Ours

Fig. 7. Visual comparisons on colorization. We compare the proposed method with Bonneel et al. [6] on smoothing the results of image colorization [19]. The method of Bonneel et al. [6] cannot preserve the colorized effect when occlusion occurs.



(a) Obtained votes (b) Selected reasons

Fig. 8. Subjective evaluation. On average, our method is preferred by 62% users. The error bars show the 95% confidence interval.

priors or temporal constraints, most likely into the design of the specific algorithms themselves.

In addition, in the way the task is formulated there is always a trade-off between being temporally coherent or perceptually similar to the processed video. Depending on the specific effect applied, there will be cases where flicker (temporal instability) is preferable to blur, and vice versa. In our current method, the user can choose a model based on their preference for flicker or blur, but an interesting area for future work would be to investigate perceptual models for what is considered acceptable flicker and acceptable blur. Nonetheless, we use the same trained model (same parameters) for all our results and showed clear viewer preference over prior methods for blind temporal stability.

5 Conclusions

In this work, we propose a deep recurrent neural network to reduce the temporal flickering problem in per-frame processed videos. We optimize both short-term and long-term temporal loss as well as a perceptual loss to reduce temporal instability while preserving the perceptual similarity to the processed videos. Our approach is agnostic to the underlying image-based algorithms applied to the video and generalize to a wide range of unseen applications. We demonstrate that the proposed algorithm performs favorably against existing blind temporal consistency method on a diverse set of applications and various types of videos.

Acknowledgments. This work is supported in part by the NSF CAREER Grant #1149783, NSF Grant No. # 1755785, and gifts from Adobe and Nvidia.

References

1. Aydin, T.O., Stefanoski, N., Croci, S., Gross, M., Smolic, A.: Temporally coherent local tone mapping of HDR video. ACM TOG (2014)
2. Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D.B.: PatchMatch: a randomized correspondence algorithm for structural image editing. ACM TOG (2009)
3. Bell, S., Bala, K., Snavely, N.: Intrinsic images in the wild. ACM TOG (2014)
4. Bonneel, N., Rabin, J., Peyré, G., Pfister, H.: Sliced and radon wasserstein barycenters of measures. J. Math. Imaging Vis. **1**(51), 22–45 (2015)
5. Bonneel, N., Sunkavalli, K., Paris, S., Pfister, H.: Example-based video color grading. ACM TOG (2013)
6. Bonneel, N., Tompkin, J., Sunkavalli, K., Sun, D., Paris, S., Pfister, H.: Blind video temporal consistency. ACM TOG (2015)
7. Chen, D., Liao, J., Yuan, L., Yu, N., Hua, G.: Coherent online video style transfer. In: ICCV (2017)
8. Chen, Q., Koltun, V.: Photographic image synthesis with cascaded refinement networks. In: ICCV (2017)
9. Dong, X., Bonev, B., Zhu, Y., Yuille, A.L.: Region-based temporally consistent video post-processing. In: CVPR (2015)
10. Gharbi, M., Chen, J., Barron, J.T., Hasinoff, S.W., Durand, F.: Deep bilateral learning for real-time image enhancement. ACM TOG (2017)

11. Goodfellow, I. et al.: Generative adversarial nets. In: NIPS (2014)
12. Gupta, A., Johnson, J., Alahi, A., Fei-Fei, L.: Characterizing and improving stability in neural style transfer. In: ICCV (2017)
13. He, K., Sun, J., Tang, X.: Single image haze removal using dark channel prior. In: TPAMI (2011)
14. Hsu, E., Mertens, T., Paris, S., Avidan, S., Durand, F.: Light mixture estimation for spatially varying white balance. ACM TOG (2008)
15. Huang, H. et al.: Real-time neural style transfer for videos. In: CVPR (2017)
16. Huang, J.B., Kang, S.B., Ahuja, N., Kopf, J.: Temporally coherent completion of dynamic video. ACM TOG (2016)
17. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. In: ICCV (2017)
18. Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J., Keutzer, K.: Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <0.5 mb model size (2016)
19. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Let there be color!: joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. ACM TOG (2016)
20. Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., Brox, T.: Flownet 2.0: evolution of optical flow estimation with deep networks. In: CVPR (2017)
21. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: CVPR (2017)
22. Jaderberg, M., Simonyan, K., Zisserman, A., Kavukcuoglu, K.: Spatial transformer networks. In: NIPS (2015)
23. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: ECCV (2016)
24. Lai, W.S., Huang, J.B., Ahuja, N., Yang, M.H.: Deep laplacian pyramid networks for fast and accurate super-resolution. In: CVPR (2017)
25. Lang, M., Wang, O., Aydin, T.O., Smolic, A., Gross, M.H.: Practical temporal consistency for image-based graphics applications. ACM TOG (2012)
26. Ledig, C. et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: CVPR (2017)
27. Lee, H.Y., Tseng, H.Y., Huang, J.B., Singh, M.K.S., Yang, M.H.: Diverse image-to-image translation via disentangled representation. In: ECCV (2018)
28. Levin, A., Lischinski, D., Weiss, Y.: Colorization using optimization. ACM TOG (2004)
29. Li, Y., Fang, C., Yang, J., Wang, Z., Lu, X., Yang, M.H.: Universal style transfer via feature transforms. In: NIPS (2017)
30. Li, Y., Huang, J.B., Narendra, A., Yang, M.H.: Deep joint image filtering. In: ECCV (2016)
31. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: context encoders: feature learning by inpainting. In: CVPR (2016)
32. Perazzi, F., Pont-Tuset, J., McWilliams, B., Van Gool, L., Gross, M., Sorkine-Hornung, A.: A benchmark dataset and evaluation methodology for video object segmentation. In: CVPR (2016)
33. Ruder, M., Dosovitskiy, A., Brox, T.: Artistic style transfer for videos. In: German Conference on Pattern Recognition (2016)
34. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015)
35. Videvo. https://www.videvo.net/

36. Xingjian, S., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.K., Woo, W.C.: Convolutional LSTM network: a machine learning approach for precipitation nowcasting. In: NIPS (2015)
37. Xu, L., Ren, J., Yan, Q., Liao, R., Jia, J.: Deep edge-aware filters. In: ICML (2015)
38. Yan, Z., Zhang, H., Wang, B., Paris, S., Yu, Y.: Automatic photo adjustment using deep neural networks. ACM TOG (2016)
39. Yao, C.H., Chang, C.Y., Chien, S.Y.: Occlusion-aware video temporal consistency. In: ACM MM (2017)
40. Ye, G., Garces, E., Liu, Y., Dai, Q., Gutierrez, D.: Intrinsic video and applications. ACM TOG (2014)
41. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: ECCV (2016)
42. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: CVPR (2018)
43. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: ICCV (2017)