Temporally Consistent Video Colorization with Deep Feature Propagation and Self-regularization Learning

Yihao Liu*, Hengyuan Zhao*, Kelvin C.K. Chan, Xintao Wang, Chen Change Loy, Senior Member, IEEE, Yu Qiao, Senior Member, IEEE, Chao Dong

Abstract—Video colorization is a challenging and highly ill-posed problem. Although recent years have witnessed remarkable progress in single image colorization, there is relatively less research effort on video colorization and existing methods always suffer from severe flickering artifacts (temporal inconsistency) or unsatisfying colorization performance. We address this problem from a new perspective, by jointly considering colorization and temporal consistency in a unified framework. Specifically, we propose a novel temporally consistent video colorization framework (TCVC). TCVC effectively propagates frame-level deep features in a bidirectional way to enhance the temporal consistency of colorization. Furthermore, TCVC introduces a self-regularization learning (SRL) scheme to minimize the prediction difference obtained with different time steps. SRL does not require any ground-truth color videos for training and can further improve temporal consistency. Experiments demonstrate that our method can not only obtain visually pleasing colorized video, but also achieve clearly better temporal consistency than state-of-the-art methods. Codes will be available. A Video demo is provided here.

Index Terms—Video colorization, temporal consistency, feature propagation, self-regularization.

1 INTRODUCTION

Video colorization aims to generate a fully colored video from its monochrome version. This topic is attractive with wide applications, since there are numerous legacy black-and-white movies produced in the past ages. Colorization can also assist other computer vision tasks such as detection [1], [2], tracking [3], [4] and video action recognition [5].

Colorization is a challenging problem due to its highly ill-posed and ambiguous nature. In recent years, plenty of single image colorization methods are proposed and have achieved remarkable progress [6], [7], [10], [11], [12]. Compared with image colorization, video colorization [8], [13], [14] is more complex, and receives relatively less attention. It requires not only satisfactory colorization performance but also good temporal consistency, as evaluated in Figure 1. A simple way to realize this task is to treat a video sequence as a series of frames and to process each frame independently using an image-based colorization model. In practice, however, when colorizing consecutive sequences, this naive solution tends to produce results suffering from flickering artifacts (temporal inconsistency). As shown in Figure 2, the results of InsColor [12], a recent state-of-the-art image-based method, are not temporally consistent. Although the colorization effect of each frame is good, the overall results contain unstable flickering, e.g., the colors of the sky and the clothes are inconsistent. This highlights the temporal consistency problem of video colorization.

In general, there are currently two ways to realize temporally consistent video colorization. The first one is to redesign a specialized video colorization model with explicitly considering temporal coherence. It demands tedious domain knowledge to devise the algorithm involving delicate exploration of network structures and loss functions [6], [7]. A recent work FAVC [8] and BTC [9], our method achieves both satisfactory colorization performance and good temporal consistency. b denotes the image-based method backbone.

Fig. 1. Compared with existing algorithms (CIC [6], IDC [7], FAVC [8] and BTC [9]), our method achieves both satisfactory colorization performance and good temporal consistency. b denotes the image-based method backbone.
Fig. 2. Image-based colorization method, e.g., InsColor [12], tends to bring about severe flickering artifacts with inconsistent colors (highlighted in green rectangles). The colorization effect of video-based method FAVC [8] is not satisfactory. The sky is hazy, the grass is not fully colorized and the overall results are grayish. Instead, our method can achieve good temporal consistency while maintaining excellent colorization performance. More comparison results are shown in Section 4.

individual frame is far from satisfactory. As shown in the third row of Figure 2, the sky is grayish and hazy, and the overall results are not vivid. More results of FAVC can be found in Figure 5. Its results are usually unsaturated with grayish or yellowish hue. Without good colorization, the temporal consistency will be of less significance.

Another way is to apply post-processing on the output frames and generate a more temporally consistent video [9], [15], [16], [17]. For instance, Lai et al. [9] present a deep network with ConvLSTM module for blind video temporal consistency (BTC), which minimizes the short-term and long-term temporal loss to constrain the temporal stability. Although such post-processing methods can enhance the temporal consistency, they directly regenerate all the frames of the original video, which largely alter the overall frame contents and increase the potential risk of incorrect modification when outliers occur. Moreover, these methods cannot achieve task-specific processing. Because different videos in various tasks are manipulated by the same operators, which could lead to dramatic quantitative performance drop comparing with the original output (see Figure 1). Further, methods like BTC [9] only consider the information of previous frames in forward propagation direction. It is necessary to integrate bidirectional information in handling consecutive video sequence.

Unlike the aforementioned approaches, we tackle video colorization from a new perspective. Rather than designing a complicated and specialized model, we jointly account for both frame-level colorization and temporally consistent constraints in a unified deep architecture. Specifically, we propose a novel Temporally Consistent Video Colorization framework (TCVC) that leverages deep features extracted from image-based model $G$ to generate contiguous adjacent features by bidirectional feature propagation. We only utilize $G$ to extract several anchor frame features while the remaining internal frame features are all generated from anchor frames. Eventually, the colorization performance of our method surpasses that of image-based model $G$, and the temporal consistency is largely improved as well. Throughout the process, we formulate the spatial-temporal alignment and propagation in high-dimensional feature space rather than image space. Differing from conventional supervised learning, we do not employ any explicit loss with the ground-truth color video, but only adopt the temporal warping loss for self-regularization. As a result, our method is label-free and data-independent. Such self-regularization mechanism also makes the training procedure very efficient. Experiments demonstrate that the proposed framework can favorably preserve the colorization performance of image-based method while simultaneously achieving state-of-the-art temporal consistency for video colorization (as compared in Figure 1).

2 Related Work

Image and video colorization. Conventional colorization methods resort to additional information provided by user scribbles [18], [19], [20], [21], [22] or example images [23], [24], [24], [25]. These methods treat colorization as a constrained optimization problem, e.g., Levin et al. [18] proposed an interactive colorization technique that propagated colors from scribbles to neighboring similar pixels. Recently, deep learning techniques have been applied to colorization [5], [7], [10], [11], [12], [26], [27]. Iizuka et al. [10] devised a two-branch network for jointly learning colorization and classification. Zhang et al. [6] modeled the colorization as a classification problem to predict the distribution of possible
colors for each pixel. Su et al. [12] proposed an instance-aware image colorization model which integrated object detection and colorization together. Aforementioned works have achieved impressive performance on single images but heavily suffer from flickering artifacts when tested on video. Video colorization [8], [13], [14], [28], [29] needs to consider both colorization performance and temporal consistency. Recently, a pioneer deep-learning-based work FAVC [8] was proposed for automatic video colorization, which is the most relevant work to ours. FAVC regularized its model with KNN graph built on the ground-truth color video and simultaneously posed a temporal loss term for constraining temporal consistency. However, the colorization performance of FAVC is not satisfactory.

**Video temporal consistency.** The temporal consistency problem is addressed on a diverse type of applications, such as artistic style transfer [30], [31], [32], [33], [34], image enhancement [35], [36], [37], [38] and colorization [8], [9]. Bonneel et al. [15] proposed a gradient-domain technique to infer the temporal regularity from the original unprocessed video. Yao et al. [16] developed an online keyframe strategy to keep track of the dynamic objects and to handle occlusions. Lai et al. [9] presented a ConvLSTM-based method which took advantage of deep recurrent network and perceptual similarity [39]. These video temporal consistency algorithms are usually post-processing methods, which modify each frame of the input video and produce a new output video. Instead of applying post-processing, our work addresses the temporal consistency and video colorization in a unified framework.

### 3 Methodology

#### 3.1 Temporally Consistent Video Colorization

Given an input grayscale video, the objective of video colorization is to obtain its corresponding colored version. Following previous works [6], [7], [10], [12], we perform this task in CIE Lab color space and predict two associated chrominance channels of a grayscale image, i.e., a channel and b channel.

**3.1.1 Overview**

For a long input grayscale video sequence, we can decompose it into several intervals. Assume each interval sequence contains N consecutive grayscale frames $X = \{x_1, x_2, \cdots, x_N\}$, we denote the start frame $x_1$ and the last frame $x_N$ as anchor frames, and the remaining $N - 2$ frames as internal frames. Thereupon, the input sequence is divided with several anchor frames and internal frames in between. Our method works on each interval sequence and we consider the input sequence as a continuum with continual camera and object motions.

The proposed TCVC framework leverages the temporal and motional information of consecutive frames in high-dimensional feature space. For any image-based colorization model $\mathcal{G}$, it can be naturally separated into two parts: feature extraction module $\mathcal{G}_E$ and color mapping module $\mathcal{G}_C$. Generally, the color mapping module corresponds to the last output layer of $\mathcal{G}$, while the feature extraction module includes all the layers before the output layer. As illustrated in Figure 3, firstly, the deep features of the two anchor frames are extracted through the feature extraction module $\mathcal{G}_E$. Then, the features are sequentially propagated in forward and backward directions frame by frame. They contain essential information for colorization. Finally, at each frame step, the associated deep features are fed into the shared color mapping module $\mathcal{G}_C$ to obtain the predicted color chrominance channels $y_i$.

#### 3.1.2 Anchor Frame Processing

Given the anchor frames at both ends of the interval sequence, our goal is to generate the color channels of each internal frame by propagating the features of the anchor frames. The anchor frames are directly processed by colorization backbone $\mathcal{G}$. As shown in Figure 3, in each interval sequence, the anchor frame branch colorizes the two anchor frames and extracts the deep features for propagation:

\[
\mathcal{F}^f_i = \mathcal{G}_E(x_1), \quad \hat{y}_1 = \mathcal{G}_C(\mathcal{F}^f_i),
\]

\[
\mathcal{F}^b_N = \mathcal{G}_E(x_N), \quad \hat{y}_N = \mathcal{G}_C(\mathcal{F}^b_N).
\] (1)

The superscripts $f$ and $b$ represent forward and backward directions, respectively. As described in Equation (1), the extracted features here are the output of the penultimate layer of model $\mathcal{G}$, which properly matches with the color mapping module $\mathcal{G}_C$. The color mapping module $\mathcal{G}_C$ accepts these features and outputs the predicted color channels. Remarkably, $\mathcal{G}$ can be any CNN-based algorithm, such as CIC [6], IDC [7], etc., making it a plug-and-play framework. Note that, in TCVC framework, the colorization backbone $\mathcal{G}$ is fixed without training. We only adopt its $\mathcal{G}_E$ to extract anchor frame features and $\mathcal{G}_C$ to predict $y_i$. Since the anchor frame branch applies model $\mathcal{G}$ on the anchor frames straightforward, it will not change any of the colorization style and performance of $\mathcal{G}$. With the initial deep features extracted from anchor frames, the features of each internal frame are progressively generated by bidirectional propagation. The forward feature propagation is initiated at the start anchor frame $x_1$ and the backward feature propagation is initiated at the last anchor frame $x_N$.

#### 3.1.3 Bidirectional Deep Feature Propagation

For internal frames, we make use of the temporal and motional characteristics of video sequence to generate the associated features from anchor frame features. This procedure is carried out by backward propagation and forward propagation sequentially. We adopt bidirectional feature propagation since all the information needed to generate the internal frame features is encoded and contained in the forward and backward directions. The feature propagation initiates from the backward direction.

**Backward propagation.** As depicted in Figure 3, the feature propagation begins at backward direction. We first estimate the optical flow between two adjacent frames. Based on the estimated motion fields, we can obtain coarsely warped internal frame features in backward direction:

\[
\mathcal{F}^b_i = \text{warp}(\mathcal{F}^b_{i+1}, f_{i+1 \rightarrow i}),
\] (2)

where $\mathcal{F}^b_i$ is the backward warped feature at the $i$-th frame, $i = N - 1, N - 2, \cdots, 2$. $\text{warp}(\cdot, \cdot)$ denotes the warping function, which can be implemented using bilinear interpolation [40] and $f_{i+1 \rightarrow i}$ represents the optical flow from frame $x_{i+1}$...
to \( x_1 \). We adopt FlowNet2 [41] to compute the flow, due to its effectiveness in relative tasks. The backward propagation starts from \( F^b_i \) and generates the features of each internal frame \( F^b_i \). However, only backward propagation is insufficient. Without complementary information from the opposite direction, the warping operation will cause the features to continuously shift in one direction, resulting in information loss. Therefore, after the backward propagation, the forward propagation will start from the other direction.

**Forward propagation.** The forward propagation starts from the first frame \( x_1 \). Similar to the backward propagation, the forward propagation first obtains a coarsely warped internal frame feature based on the estimated optical flow. Furthermore, forward propagation is obligated for more functions, including integrating backward and forward features, and generating color channels with the fused features. To integrate the features propagated in backward and forward directions, we devise an effective frame-specific feature fusion module (FFM). The details of FFM will be described later. We denote the output feature of FFM in the forward propagation as \( \tilde{F}^f_{i+1} \), which combines fine bidirectional information for subsequent colorization. Note that, except the first forward feature \( F^f_1 \) is directly delivered to the next frame, the other forward features to be propagated are the features after fusion, i.e., \( F^f_{i+1} \):

\[
F^f_{i+1} = \text{warp}(\tilde{F}^f_i, f_{i\rightarrow i+1}),
\]

where \( i = 1, 2, \cdots, N - 2 \). After feeding \( \tilde{F}^f_i \) into the shared color mapping module \( G_C \), the predicted color channels of internal grayscale frame \( x_i \) is obtained:

\[
\hat{y}_i = G_C(\tilde{F}^f_i),
\]

where \( i = 2, 3, \cdots, N - 1 \).

Fig. 3. The proposed TCVC framework (take \( N = 4 \) for example). The anchor frame branch colorizes the two anchor frames and extracts the deep features for propagation. With bidirectional deep feature propagation, the internal frame features are all generated from anchor frames, which ensures the temporal consistency in high-dimensional feature space.

Fig. 4. The structure of feature fusion module (FFM), which contains a weighting network and a feature refine network.

**Feature fusion module.** The structure of the proposed feature fusion module is detailed in Figure 4. It contains a weighting network (WN) and a feature refine network (FRN), which are both three-layer plain CNNs. In FFM, three consecutive images \( x_{i-1}, x_i, x_{i+1} \) are first fed into \( G_E \) to obtain the corresponding features, which are then concatenated together with other inputs to feed in the WN and FRN.

Intuitively, the warped backward feature \( F^b_i \) and forward feature \( F^f_i \) are both coarsely aligned with current frame \( x_i \). However, due to different propagation directions, there are complementary and redundant parts between them in different pixel locations. The weighting network predicts a weighting map \( W \in \mathbb{R}^{H \times W \times 1} \) ranged in \([0, 1] \). Then, the forward and backward features are fused by a simple linear interpolation: \( F^b_{i+1} = W \odot F^f_{i+1} + (1 - W) \odot F^b_i \), where \( \odot \) denotes the element-wise multiplication operation.
$F_i^{fb}$ contains the information of both forward and backward features. Due to the inaccurate flow estimation and the information loss caused by warping operation, errors will accumulate in the propagation process. Therefore, we further refine the feature according to the adjacent spatiotemporal information.

As shown in Figure 4, the feature refine network accepts the roughly fused feature $F_i^{fb}$ and generates a refining residual $F_i^{res}$. Specially, the feature refine network additionally takes into account the backward feature $F_{i+1}^{b}$ of the latter frame and the forward feature $F_{i-1}^{f}$ of the previous frame. The reason for such design is that $F_{i+1}^{b}$ and $F_{i-1}^{f}$ implicitly encode and contain all the information needed to obtain the aligned feature at current $i$-th frame. $1 \times 1$ convolutions are used to reduce and unify the dimensionality. The final refined feature at $i$-th frame is obtained as:

$$F_i^f = F_i^{fb} + F_i^{res}$$

$F_i^f$ will be propagated to the next frame. By utilizing the information of current frame and adjacent frames, FFM can achieve frame-specific feature fusion in a coarse-to-fine manner. With bidirectional feature propagation, the internal frame features are all generated from anchor frame features.

### 3.2 Self-regularization Learning

One unique characteristic of the proposed framework is that it utilizes a self-regularization learning scheme without relying on ground-truth color videos. Here the self-regularization learning means that we do not employ any explicit loss with the ground-truth color video, which is different from the conventional supervised learning. In TCVC, we do not need to train the colorization backbone $G$. To let the network learn temporal consistency, we adopt the temporal warping loss as follows:

$$L_{TW} = \sum_{d=1,2} \sum_{i=1}^{N-d} \| M_{i+d \rightarrow i} \odot (\hat{y}_i - y_i^{\warp}) \|_2^2$$

where $y_i^{\warp} = \warp(y_{i+d}, f_{i+d \rightarrow i})$, and $d$ represents the time interval for temporal warping. $M_{i+d \rightarrow i} = \exp(-\alpha\|y_i - y_i^{\warp}\|_2^2)$ is the visibility mask. Following [2], we set $\alpha = 50$. This loss function explicitly poses a penalty to the temporal consistency between adjacent frames. It is noteworthy that there is no ground-truth color video used during training. A consecutive grayscale input video is all we need. More discussions can be found in the supplementary file.

Further, the self-regularization learning also makes our framework free from the influence of training and testing data, i.e., the proposed method is data-independent and label-free. As long as the input video contains consecutive motional frames, it can be adopted as our training set. Another advantage of the proposed self-regularization is that it does not require magnanimous training data and it has few trainable parameters. Thus, the training procedure can be very efficient (about two days).

### 3.3 Multiple Anchor Frame Sampling

For a long input sequence (over dozens of frames), we first divide it into several intervals by uniformly sampling anchor frames or specifying the interval length $N$. Once the interval length $N$ is determined, the anchor frames are also determined during inference phase. Empirically, this scheme has already worked fine. However, we also provide an optional post-processing scheme to further enhance the performance. Specifically, we sample the anchor frames multiple times (choose different $N$), and then average the results of each output. This procedure can be regarded as an ensemble method during testing. It could eliminate the uncertainty and inconsistency of anchor frames to some extent and achieve better temporal consistency. In the experiments, we adopt $N = 15$ and $N = 17$ for ensembling.

### 3.4 Discussions

#### 3.4.1 Differences with other methods

The proposed TCVC framework is conceptually different from previous works on video colorization and video temporal consistency in motivation and methodology. We address the video colorization problem from a new perspective. In summary, the proposed method is different from previous solutions in three aspects: i) The proposed framework takes advantage of an ingenious image-based model and focuses on the temporally consistent constraints. It can favorably achieve both good colorization performance and satisfactory temporal consistency. ii) We formulate the spatial-temporal alignment and propagation in high-dimensional feature space. iii) Different from conventional supervised learning, it adopts a self-regularization learning scheme without relying on ground-truth color videos.

#### 3.4.2 Uniqueness of TCVC

Our goal is to improve the temporal consistency based on image-based model $G$ without retraining it. The unique feature of TCVC is that it requires no GT color videos during training. This unique trait is a concomitant by-product, introduced by the method itself with feature propagation and self-regularization learning. Specifically, TCVC first leverages a pretrained image-based model $G$ to extract anchor frame features containing color information. Then, TCVC propagates the color information from anchor frames to the remaining internal frames. The color information is inherited from $G$, and TCVC focuses on the propagation of such sparse information with explicitly considering the temporal consistency. The temporal warping loss poses a penalty to implicitly guide the network to better propagate the color information in a self-regularization and self-supervised manner, with explicitly enhancing the temporal consistency between adjacent frames. Further, we have also conducted experiments to demonstrate the effectiveness of such learning scheme.

With such a unique characteristic, TCVC obtains state-of-the-art results on temporally consistent video colorization. As shown in Figure 2, existing image-based method like InsColor [12] tends to produce severe flickering artifacts with inconsistent colorization. The colorization performance of video-based method FAVC [8] is not satisfactory, which will produce grayish and unsaturated hue. Instead, the colorization effect of TCVC framework is temporally consistent and visually pleasing, due to bidirectional feature propagation and self-regularization learning.
Fig. 5. Visual comparison with state-of-the-art methods. Image-based methods [7], [12] are prone to produce severe flickering artifacts. Post-processing method BTC [9] cannot achieve long-term temporal consistency well and cannot handle outliers. The results of FAVC [8] are usually unsaturated and sometimes contain strange greenish hue, e.g., there are strange greenish regions on the gun in the upper sequence. Please zoom in for best view.
4 Experiments

Datasets. Following previous works [8], [9], we adopt DAVIS dataset [42] and Videvo dataset [9] for training and testing. The DAVIS dataset is designed for video segmentation, which includes a variety of moving objects and motion types. It has 60 videos for training and 30 videos for testing. The Videvo dataset contains 80 videos for training and 20 videos for testing. The training videos are all resized to $300 \times 300$. We mix the DAVIS and Videvo training sets to conduct self-regularization learning as in Section 3.2.

Metrics. We evaluate the results in two facets: colorization performance and video temporal consistency. The colorization performance is paramount for colorization task. Without good colorization, the temporal consistency will be less of significance. For example, unsaturated images with few colors could result in better consistency. However, such neutral results cannot meet the requirements of good colorization. To measure the colorization performance, we adopt PSNR and $L_2$ error in Lab color space. Moreover, we also utilize the colorfulness measurement proposed by Hasler and Suesstrunk [43], to roughly evaluate the color
diversity of the resulting images produced by different methods. For temporal consistency, we adopt warp error proposed in [9]. However, warp error is uncorrelated with the video color and is easily affected by the performance of flow estimation module used in the measurement. Therefore, we propose a more suitable Color Distribution Consistency index (CDC) to further measure the temporal consistency, which is specially devised for video colorization task. Specifically, it computes the Jensen-Shannon (JS) divergence of the color distribution between consecutive frames:

$$CDC_t = \frac{1}{3 \times (N - b)} \sum_{c \in \{r, g, b\}} \sum_{i=1}^{N-t} JS(P_c(I^t), P_c(I^{t+i})),$$

where $N$ is the video sequence length and $P_c(I^t)$ is the normalized probability distribution of color image $I$ across $c$ channel, which can be calculated from the image histogram. $t$ denotes the time step. A smaller $t$ indicates short-term temporal consistency, while larger $t$ indicates long-term temporal consistency. The JS divergence can measure the similarity between two color probability distributions. Considering the long-term and short-term temporal consistency together, we propose the following index:

$$CDC = \frac{1}{3} (CDC_1 + CDC_2 + CDC_4).$$

It takes $t = 1$, $t = 2$ and $t = 4$ into account, which can appropriately reflect the temporal consistency for color distribution. Too large $t$ will lead to much difference in content between the two frames, causing the color distribution to change rapidly. Moreover, we also conducted a user study for subjective evaluation.

### 4.1 Implementation Details

In our implementation, we adopt CIC [6] and IDC [7] as the image-based colorization backbone $G$. Note that we do not need to train $G$. When training, the input interval length $N = 10$, while $N$ is set to 17 when testing. The batch size is 4 and the patch size of input frames is 256 $\times$ 256. The learning rate is initialized to $5e^{-5}$ and is decayed by half every 10,000 iterations. The Adam optimizer [44] is adopted. We use PyTorch framework and train all models using four GTX 2080Ti GPUs.

### 4.2 Comparison with State-of-the-art Methods

Since this paper focuses on temporally consistent video colorization, FAVC [8] is the main competitor. FAVC is the newest and the first learning-based fully automatic video colorization method. Unfortunately, there is only FAVC published in the top conference or journal. We follow FAVC and conduct sufficient comparisons with image-based, video-based and post-processing methods. Specifically, we compare our method with representative single image colorization methods [6], [7], [10], [12] and video colorization method FAVC [8]. In addition, we apply the blind temporal consistency methods BTC [9] and DVP [17] on [6] and [7] to form another two groups of comparison methods.

#### Quantitative comparison

Table 1 summarizes the quantitative results and Figure 1. Image-based methods [7], [12] can achieve relatively higher PSNR, while their temporal consistency is poor. Video-based method FAVC [8] slightly improves the temporal consistency but its colorization performance is not satisfactory, as shown in Figure 5. Quantitatively, FAVC yields the lowest colorfulness value among all the methods. BTC [9] and DVP [17] can largely enhance the temporal consistency, but the cost is that the PSNR values decrease dramatically compared to original outputs of [6], [7]. Further, BTC is vulnerable to be affected by outliers and DVP leans to produce colorless results (see Figure 8). Moreover, DVP [17] is an image-specific one-shot algorithm which requires independent training during testing. Thus, it is time-consuming to conduct inference, making it impractical for real-time or high-speed applications.

For comparison, we adopt [6], [7] as our backbones. After integrated in TCVC, the temporal consistency gets improved, validating the effectiveness of TCVC. Moreover, TCVC can achieve impressive colorization performance with high PSNR values. TCVC can even slightly boost the PSNR values and reduce the $L_2$ error in Lab space. TCVC can also perfectly remain the colorfulness, while BTC and...
DVP could lower the resulting colorfulness values. Note that, for fairness, we do not use scene cut techniques on the test datasets, but we still achieve the best results. For very long videos, some simple techniques can be used, like histogram/block matching, which are easy to be incorporated with TCVC. With scene cut techniques, the performance of TCVC is supposed to be further improved.

**Qualitative comparison.** Visual comparisons are shown in Figure 2, 5, 6, 7 and 8. Image-based methods [7], [12] are prone to produce severe flickering artifacts. Their predicted color of one object differs in consecutive frames. For example, in Figure 6, the car is colorized in red by InsColor [12] in the first four frames, while it is painted bluish in the last frame; The dancer’s clothes are colorized in brighter red by IDC [7] in the first and fourth frames, while in the other frames, the color of the clothes becomes lighter and less saturated. After applying post-processing method BTC [9], the results can become more temporally consistent. However, BTC modifies all the frames of the original output video, which could immensely decrease the PSNR values as discussed before. Further, BTC is susceptible to outliers and cannot deal with the extreme outliers properly and thoroughly. As shown in the lower part of Figure 6, BTC fails to achieve temporal consistency in this consecutive sequence: the outlier red region stays unchanged after applying BTC. As shown in Figure 8, DVP [17] could remove the color of the original images or produce results with weird green tone. Further, the results of DVP are likely to contain color contaminations. Compared with state-of-the-art image-based methods, the results of FAVC [8] are
usually not vivid with unsaturated and grayish hue. FAVC [8] sometimes could even produce strange greenish color in objects (see the lower part of Figure 6). Compared with previous works, our method can achieve both good colorization performance and temporal consistency. Particularly, TCVC can produce colorized results with long-term temporal consistency, since all the internal frames are generated by continual feature propagation. Thus, different from BTC, TCVC can handle outliers and achieve impressive quantitative performance.

**Results on legacy black-and-white movies.** Additionally, we display several visual results on legacy black-and-white movies to demonstrates the good generalization ability of our method. It is an attractive application of video colorization. As shown in Figure 7, one can see that our model is able to produce good colorization results on legacy grayscale films.

**User study.** We also conducted a user study with 20 participants for subjective evaluation. 15 videos are randomly selected from the test datasets. We compare our method with video colorization methods FAVC [8], CIC [6]+BTC and IDC [7]+BTC in pairwise manner. The participants are asked to choose the better one by colorization performance and the temporal consistency. As the results shown in Figure 9, the proposed framework TCVC surpasses all other methods by a large margin. More than 75.0% (225) of users’ choices favor our results.

### 4.3 Advantages of Adopting Anchor Frames

In TCVC framework, the anchor frames are directly processed by the well-performed image-based model $G$, and the internal frames are generated by bidirectional propagation from anchor frames. We demonstrate the advantages of adopting anchor frames by statistical analysis. Specifically, we aim to answer the following questions: 1) Since all the anchor frames are the same with that of $G$, what is the influence of sampling anchor frames with different interval length $N$? 2) What is the effect of adopting deep feature propagation to generate internal frames? 3) What are the advantages of TCVC compared with post-processing method BTC [9]? To answer these questions, we have calculated the PSNR values of the anchor frames and the internal frames produced by TCVC under different interval length $N$ on DAVIS [42] dataset. Then, we compare the corresponding PSNR values with the backbone model IDC [7] and post-processing method BTC [9].

| $N$ | anchor | #anchor | internal | #internal | all  |
|-----|--------|---------|----------|-----------|------|
| 5   | 25.47  | 372     | 25.40    | 1627      | 25.42|
| 9   | 25.44  | 235     | 25.42    | 1764      | 25.42|
| 13  | 25.43  | 184     | 25.43    | 1815      | 25.42|
| 17  | 25.43  | 153     | 25.41    | 1846      | 25.42|
| 21  | 25.42  | 136     | 25.41    | 1863      | 25.42|

**TABLE 2**

Quantitative analysis of image-based colorization backbone IDC [7]. We calculate the corresponding PSNR values of anchor frames and internal frames for various $N$. Different $N$ will lead to different separated sets for anchor and internal frames.

Larger interval length $N$ connotes that fewer anchor frames will be sampled. One may concern that TCVC could sample anomalous anchor frames (outliers), resulting in the accumulation of errors throughout the feature propagation. From Table 2, we observe that the PSNR values of anchor frames for different samplings are relatively stable. Statistically, there are few outliers in a sequence. Thus,
the probability of sampling anomalous anchor frames is marginal. Further, with the increase of $N$, the number of sampled anchor frames will be reduced, and more outlier frames will fall in internal frames to be regenerated. In such cases, compared with post-processing method like BTC [9], TCVC can better get rid of the influence of outliers. As shown in Table 3, although BTC could enhance the video temporal consistency, the PSNR is significantly reduced. For TCVC, since the anchor frames are directly processed by image-based method, the PSNR of anchor frames is the same as IDC [7], while the PSNR of internal frames is further improved. This is because TCVC can avoid the influence of anomalous internal frames with low PSNR values, since all the internal frames are regenerated by feature propagation from anchor frames. Hence, TCVC can successfully achieve satisfactory temporal consistency while maintaining good colorization performance.

5 Ablation Study
We further conduct ablation studies to demonstrate the effectiveness of the proposed FFM, bidirectional propagation and self-regularization learning. We test the models with interval length $N = 11$ on DAVIS dataset.

5.1 Effectiveness of Feature Fusion Module
The purpose of the feature fusion module (FFM) is to integrate the backward and forward features in a dedicated coarse-to-fine manner. It leverages the information of current frame and adjacent frames to achieve frame-specific feature fusion. To demonstrate its effectiveness, we replace the FFM with plain convolutional networks to fuse the bidirectional features. The experimental results are shown in the second and third rows of Table 4. By adopting FFM, the temporal consistency is further improved from 0.004003 to 0.003874, which validates the effectiveness of FFM.

5.2 Effectiveness of Bidirectional Propagation
In this paper, we propose bidirectional propagation to generate the consecutive internal features from anchor features. If we only conduct unidirectional propagation without complementary information from the opposite direction, the warping operation will cause the features continuously to shift in one direction, leading to information loss. We conduct an ablation study to validate the effectiveness of bidirectional propagation. For unidirectional propagation, we do not need to fuse the forward and backward features with FFM, so we replace FFM with a plain network. As shown in the first and second rows of Table 4, TCVC model with one direction is much inferior to that with two directions. By the utilization of bidirectional propagation, both the PSNR and temporal consistency are largely improved.

5.3 Effectiveness of Self-regularization Learning
We compare conventional supervised-learning and the proposed self-regularization in the same of TCVC framework. In particular, we train the proposed framework using different regularization terms. 1) Only adopting $L_2$ loss with ground-truth color videos, 2) Only adopting temporal warping loss for self-regularization, 3) Adopting both $L_2$ and temporal warping losses simultaneously. As shown in Table 5, interestingly, the performance of adopting only $L_2$ loss is much inferior to that of adopting temporal warping loss. It fails to achieve satisfactory colorization performance nor temporal consistency. As shown in Figure 11, TCVC model with only adopting $L_2$ loss produces results with severe visual artifacts. This is because $L_2$ loss cannot regularize the procedure of feature warping and fusion in TCVC framework. In addition, $L_2$ loss is not robust to the intrinsic ill-posed nature of colorization problem, which is also addressed in [6]. When adopting both $L_2$ loss and temporal warping loss, the results are better than only adopting $L_2$ loss but still inferior to only adopting temporal warping loss. Adopting $L_2$ loss on GT will degrade the colorization performance, since we do not retrain $G$. Avoiding using GT color makes the framework concentrate on reorganizing the consecutive features. This experiment validates the effectiveness of the self-regularization learning for TCVC. It also stresses the difficulties for achieving both good colorization effect and temporal consistency. Therefore, different from conventional supervised learning, we design such an elaborate mechanism, making it efficient and unique.

5.4 Exploration on Interval Length
To further explore the influence of interval length $N$, we conduct experiments with different $N$ during training and testing phases. In particular, we adopt $N = 5$, $N = 7$ and $N = 10$ for training while $N = 3$, $N = 5$, $N = 9$, $N = 11$, $N = 17$ and $N = 19$ for testing. The experimental results are listed in Table 6. It can be seen that the interval length $N$ mainly affects the temporal consistency while it has marginal impact on PSNR values. Adopting more internal frames for training and testing can achieve better temporal consistency. Specifically, when the testing interval length is

| $N$ | PSNR↑ | CDC↓ |
|-----|-------|-------|
| 3   | 25.35 | 0.004003 |
| 5   | 25.35 | 0.004003 |
| 7   | 25.35 | 0.004003 |
| 10  | 25.35 | 0.004003 |
| 17  | 25.35 | 0.004003 |
| 19  | 25.35 | 0.004003 |

**Table 4** Effectiveness of FFM and bidirectional propagation.
fixed, the more frames adopted in training, the better the consistency performance. This is because our framework propagates the features in intervals, longer intervals could achieve longer temporal consistency. In addition, adopting more consecutive frames for training can make the model have larger temporal receptive field and learn more motion patterns. Nevertheless, too large interval length $N$ for training will cost more GPU memory and computational resources. Too large $N$ for testing will result in PSNR drop, because the difficulty of optical flow estimation and feature fusion has increased as well. Thus, we adopt moderate $N = 10$ for training and $N = 17$ for testing in the main experiment.

| Train $N$ | Test $N$ | PSNR↑ | CDC↓ |
|----------|----------|------|------|
| 3        | 3        | 25.47 | 0.004324 |
| 5        | 5        | 25.49 | 0.004103 |
| 9        | 9        | 25.46 | 0.003949 |
| 11       | 11       | 25.47 | 0.003877 |
| 17       | 17       | 25.48 | 0.003838 |
| 19       | 19       | 25.39 | 0.003885 |

| 7        | 3        | 25.47 | 0.004405 |
| 5        | 5        | 25.50 | 0.004149 |
| 9        | 9        | 25.47 | 0.003981 |
| 11       | 11       | 25.47 | 0.003865 |
| 17       | 17       | 25.46 | 0.003767 |
| 19       | 19       | 25.38 | 0.003754 |

| 10       | 3        | 25.47 | 0.004479 |
| 5        | 5        | 25.50 | 0.004227 |
| 9        | 9        | 25.49 | 0.003963 |
| 11       | 11       | 25.50 | 0.003874 |
| 17       | 17       | 25.49 | 0.003771 |
| 19       | 19       | 25.40 | 0.003749 |

**TABLE 6**

Exploration on interval length $N$. More internal frames could benefit temporal consistency.

### 6 Failure Cases

We here show several failure cases of TCVC. As shown in Figure 10, in some cases, TCVC could produce results with ghost artifacts or color contamination. This is mainly due to the inaccurate optical flow estimation, especially when large motions or severe occlusions occur. The estimation of optical flow (OF) and occlusion (OCC) is crucial for most video-relevant tasks, e.g., video super-resolution [45], video frame interpolation [46] and video compression [47]. However, we have tested TCVC on a large number of videos. It outperforms all other works qualitatively and yields the best quantitative evaluations on average. In the Appendix, we provide the detailed evaluation of each test video. The performance is stable and robust. Certainly, there is room for improving our method. With better OF/OCC estimation, TCVC can continuously be promoted. Research effort on better optical flow and occlusion estimation will largely contribute to lots of computer vision tasks.

### 7 Conclusion

We propose a temporally consistent video colorization framework (TCVC) with deep feature propagation and self-regularization learning. TCVC generates contiguous adjacent features for colorizing video. It adopts a self-regularization learning scheme and does not require any ground-truth color video for training. TCVC can achieve both good colorization effect and temporal consistency.

### References

[1] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” in *Advances in neural information processing systems*, 2015, pp. 91–99.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[3] C. Vondrick, A. Shrivastava, A. Fathi, S. Guadarrama, and K. Murphy, “Tracking emerges by colorizing videos,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[4] Z. Zhang and H. Peng, “Deeper and wider siamese networks for real-time visual tracking,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4591–4600.

[5] G. Larsson, M. Maire, and G. Shakhnarovich, “Colorization as a proxy task for visual understanding,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[6] R. Zhang, P. Isola, and A. A. Efros, “Colorful image colorization,” in *European conference on computer vision*. Springer, 2016, pp. 649–666.
