# Arbitrary Video Style Transfer via Multi-Channel Correlation

**Yingying Deng[1,2,4], Fan Tang[3*], Weiming Dong[1,2,4*], Haibin Huang[5], Chongyang Ma[5] and Changsheng Xu[1,2,4]**

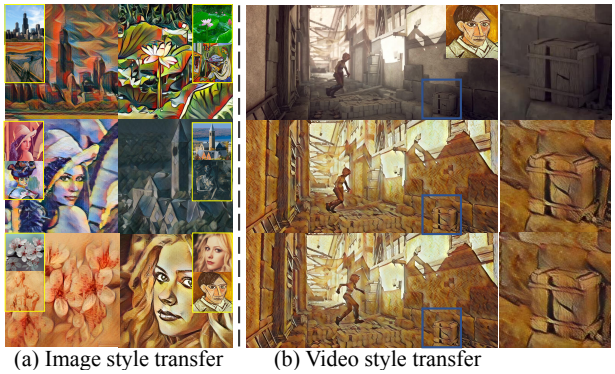
[1] School of Artificial Intelligence, University of Chinese Academy of Sciences,
[2] NLPR, Institute of Automation, Chinese Academy of Sciences, [3] School of Artificial Intelligence, Jilin University,
[4] CASIA-LLvision Joint Lab, [5] Kuaishou Technology
{dengyingying2017, weiming.dong, changsheng.xu}@ia.ac.cn, tangfan@jlu.edu.cn,
{huanghaibin03, chongyangma}@kuaishou.com

## Abstract

Video style transfer is attracting increasing attention from the artificial intelligence community because of its numerous applications, such as augmented reality and animation production. Relative to traditional image style transfer, video style transfer presents new challenges, including how to effectively generate satisfactory stylized results for any specified style while maintaining temporal coherence across frames. Towards this end, we propose a Multi-Channel Correlation network (MCCNet), which can be trained to fuse exemplar style features and input content features for efficient style transfer while naturally maintaining the coherence of input videos to output videos. Specifically, MCCNet works directly on the feature space of style and content domain where it learns to rearrange and fuse style features on the basis of their similarity to content features. The outputs generated by MCC are features containing the desired style patterns that can further be decoded into images with vivid style textures. Moreover, MCCNet is also designed to explicitly align the features to input and thereby ensure that the outputs maintain the content structures and the temporal continuity. To further improve the performance of MCCNet under complex light conditions, we also introduce illumination loss during training. Qualitative and quantitative evaluations demonstrate that MCCNet performs well in arbitrary video and image style transfer tasks. Code is available at https://github.com/diyiiyiii/MCCNet.

## Introduction

Style transfer is a significant topic in the industrial community and research area of artificial intelligence. Given a content image and an art painting, a desired style transfer method can render the content image into the artistic style referenced by the art painting. Traditional style transfer methods based on stroke rendering, image analogy, or image filtering (Efros and Freeman 2001; Bruckner and Gröller 2007; Strothotte and Schlechtweg 2002) only use low-level features for texture transfer (Gatys, Ecker, and Bethge 2016; Doyle et al. 2019). Recently, deep convolutional neural networks (CNNs) have been widely studied for artistic image generation and translation (Gatys, Ecker, and Bethge 2016; Johnson, Alahi, and Fei-Fei 2016; Zhu et al. 2017; Huang and Serge 2017; Huang et al. 2018; Jing et al. 2020).



(a) Image style transfer (b) Video style transfer

Figure 1: The proposed method can be used for stable video style transfer with high-quality stylization effect: (a) image style transfer results using images in different domains; (b) video style transfer results using different frames from *Sintel*. The stylization results of the same object in different frames are the same. Hence, the rendered effects for the same content are stable between different frames.

Although existing methods can generate satisfactory results for still images, they lead to flickering effects between adjacent frames when applied to videos directly (Ruder, Dosovitskiy, and Brox 2016; Chen et al. 2017). Video style transfer is a challenging task that needs to not only generate good stylization effects per frame but also consider the continuity between adjacent frames of the video. Ruder, Dosovitskiy, and Brox (2016) added temporal consistency loss on the basis of the approach proposed in Gatys, Ecker, and Bethge (2016) to maintain a smooth transition between video frames. Chen et al. (2017) proposed an end-to-end framework for online video style transfer through a real-time optical flow and mask estimation network. Gao et al. (2020) proposed a multistyle video transfer model, which estimates the light flow and introduces temporal constraint. However, these methods highly depend on the accuracy of optical flow calculation, and the introduction of temporal constraint loss reduces the stylization quality of individual frames. Moreover, adding optical flow to constrain the coherence of stylized videos makes the network difficult to train when applied to an arbitrary style transfer model.

In the current work, we revisit the basic operations in

*Co-corresponding authors

state-of-the-art image stylization approaches and propose a frame-based *Multi-Channel Correlation* network (MCCNet) for temporally coherent video style transfer that does not involve the calculation of optical flow. Our network adaptively rearranges style representations on the basis of content representations by considering the multi-channel correlation of these representations. Through this approach, MCCNet is able to make style patterns suitable for content structures. By further merging the rearranged style and content representations, we generate features that can be decoded into stylized results with clear content structures and vivid style patterns. MCCNet aligns the generated features to content features, and thus, slight changes in adjacent frames will not cause flickering in the stylized video. Furthermore, the illumination variation among consecutive frames influences the stability of video style transfer. Thus, we add random Gaussian noise to the content image to simulate illumination varieties and propose an illumination loss to make the model stable and avoid flickering. As shown in Figure 1, our method is suitable for stable video style transfer, and it can generate single-image style transfer results with well-preserved content structures and vivid style patterns. In summary, our main contributions are as follows:

- We propose MCCNet for framed-based video style transfer by aligning cross-domain features with input videos to render coherent results.
- We calculate the multi-channel correlation across content and style features to generate stylized images with clear content structures and vivid style patterns.
- We propose an illumination loss to make the style transfer process increasingly stable so that our model can be flexibly applied to videos with complex light conditions.

## Related Work

**Image style transfer.** Image style transfer has been widely studied in recent years. Essentially, it enables the generation of artistic paintings without the expertise of a professional painter. Gatys, Ecker, and Bethge (2016) found that the inner products of the feature maps in CNNs can be used to represent style and proposed a neural style transfer (NST) method through continuous optimization iterations. However, the optimization process is time consuming and cannot be widely used. Johnson, Alahi, and Fei-Fei (2016) put forward a real-time style transfer method to dispose of a specific style in one model. Dumoulin, Shlens, and Kudlur (2016) proposed conditional instance normalization (CIN), which allows the learning of multiple styles in one model by reducing a style image into a point in the embedded space. A number of methods achieve arbitrary style transfer by aligning the second-order statistics of style and content images (Huang and Serge 2017; Li et al. 2017; Wang et al. 2020b). Huang and Serge (2017) proposed an arbitrary style transfer method by adopting adaptive instance normalization (AdaIN), which normalizes content features using the mean and variance of style features. Li et al. (2017) used whitening and coloring transformation (WCT) to render content images with style patterns. Wang et al. (2020b) adopted deep feature perturbation (DFP) in a WCT-based model to achieve diversified arbitrary style transfer. However, these holistic transformations lead to unsatisfactory results. Park and Lee (2019) proposed a style-attention network (SANet) to obtain abundant style patterns in generated results but failed to maintain distinct content structures. Yao et al. (2019) proposed an attention-aware multi-stroke style transfer (AAMS) model by adopting self-attention to a style swap-based image transfer method, which highly relies on the accuracy of the attention map used. Deng et al. (2020) proposed a multi-adaptation style transfer (MAST) method to disentangle content and style features and combine them adaptively. However, some results are rendered with uncontrollable style patterns.

In the current work, we propose an arbitrary style transfer approach, which can be applied to video transfer with better stylized results than other state-of-the-art methods.

**Video style transfer.** Most video style transfer methods rely on existing image style transfer methods (Ruder, Dosovitskiy, and Brox 2016; Chen et al. 2017; Gao et al. 2020). Ruder, Dosovitskiy, and Brox (2016) built on NST and added a temporal constraint to avoid flickering. However, the optimization-based method is inefficient for video style transfer. Chen et al. (2017) proposed a feed-forward network for fast video style transfer by incorporating temporal information. Chen et al. (2020) distilled knowledge from the video style transfer network with optical flow to a student network to avoid optical flow estimation in the test stage. Gao et al. (2020) adopted CIN for multi-style video transfer, and the approach incorporated one FlowNet and two ConvLSTM modules to estimate light flow and introduce a temporal constraint. The temporal constraint aforementioned is achieved by calculating optical flow, and the accuracy of optical flow estimation affects the coherence of the stylized video. Moreover, the diversity of styles is limited because of the used basic image style transfer methods used. Wang et al. (2020a) proposed a novel interpretation of temporal consistency without optical flow estimation for efficient zero-shot style transfer. Li et al. (2019) learned a transformation matrix for arbitrary style transfer. They found that the normalized affinity for generated features are the same as that for content features, and is thus suitable for frame-based video style transfer. However, the style patterns in their stylized results are not clearly noticeable.

To avoid using optical flow estimation while maintaining video continuity, we aim to design an alignment transform operation to achieve stable video style transfer with vivid style patterns.

## Methodology

As shown in Figure 2, the proposed MCCNet adopts an encoder-decoder architecture. Given a content image $I_c$ and a style image $I_s$, we can obtain corresponding feature maps $f_c = E(I_c)$ and $f_s = E(I_s)$ through the encoder. Through MCC calculation, we generate $f_{cs}$ that can be decoded into stylized image $I_{cs}$.

We first formulate and analyze the proposed multi-channel correlation in Sections  and  and then introduce the

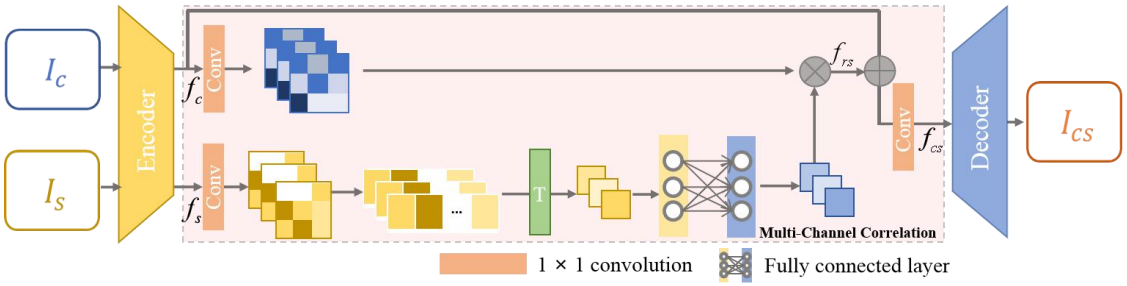



Figure 2: Overall structure of MCCNet. The green block represents the operation of the feature vector multiplied by the transpose of the vector. $\otimes$ and $\oplus$ represent the matrix multiplication and addition. $I_c$, $I_s$ and $I_{cs}$ refer to the content, style and generated images, respectively.

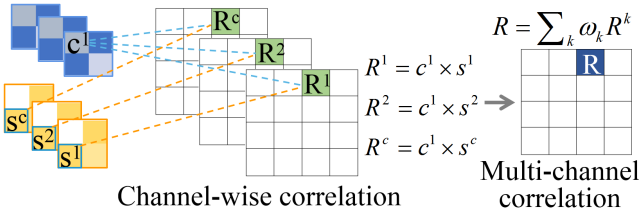
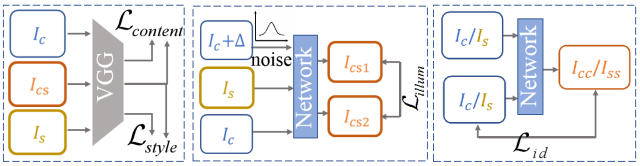



Figure 3: Correlation matrix CO. $R$ is the element of CO.




(a) Perceptual loss (b) Illumination loss (c) Identity loss

Figure 4: Loss function.

configuration of MCCNet involving feature alignment and fusion in Section .

## Multi-Channel Correlation

Cross-domain feature correlation has been studied for image stylization (Park and Lee 2019; Deng et al. 2020). It fuses the multiple content and style features by using several adaption/attention modules without considering the inter-channel relationship of the content features. In SANet (Park and Lee 2019), the generated features can be formulated as

$$f_{cs} = F(f_c, f_s) \propto exp(f(f_c), g(f_s))h(f_s). \quad (1)$$

Slight changes in input content features can lead to a large-scale variation in output features. MAST (Deng et al. 2020) has the same issue. Thus the coherence of input content features cannot be migrated to generated features, and the stylized videos will present flickering artifacts.

In this work, we propose the multi-channel correlation for frame-based video stylization, as illustrated in Figure 3. For channel $i$, the content and style features are $f_c^i \in \mathbb{R}^{H\times W}$ and $f_s^i \in \mathbb{R}^{H\times W}$, respectively. We reshape them to $f_c^i \in \mathbb{R}^{1\times N}$, $f_c^i = [c_1, c_2, \cdots, c_N]$ and $f_s^i \in \mathbb{R}^{1\times N}$, $f_s^i = [s_1, s_2, \cdots, s_N]$, where $N = H \times W$. The channel-wise correlation matrix between content and style is calculated by

$$CO^i = f_c^{iT} \otimes f_s^i. \quad (2)$$

Then, we rearrange the style features by

$$f_{rs}^i = f_s^i \otimes CO^{iT} = \|f_s^i\|_2 f_c^i, \quad (3)$$

where $\|f_s^i\|_2 = \sum_{j=1}^N s_j^2$. Finally, the channel-wise generated features are

$$f_{cs}^i = f_c^i + f_{rs}^i = (1 + \|f_s^i\|_2) f_c^i. \quad (4)$$

However, the multi-channel correlation in style features is also important to represent style patterns (e.g., texture and color). We calculate the correlation between each content channel and every style channel. The $i$-th channel for generated features can be rewritten as

$$f_{cs}^i = f_c^i + f_{rs}^i = (1 + \sum_{k=1}^{C} w_k \|f_s^k\|_2) f_c^i, \quad (5)$$

where $C$ is the number of channels and $w_k$ represents the weights of the $k$-th style channel. Finally, the generated features $f_{cs}$ can be obtained by

$$f_{cs} = K f_c, \quad (6)$$

where $K$ is the style information learned by training.

From Equation (6), we can conclude that MCCNet can help to generate stylized features that are strictly aligned to content features. Therefore, the coherence of input videos is naturally maintained to output videos and slight changes (e.g., objects motions) in adjacent frames cannot lead to violent changes in the stylized video. Moreover, through the correlation calculation between content and style representations, the style patterns are adjusted according to content distribution. The adjustment helps generate stylized results with stable content structures and appropriate style patterns.

## Coherence Analysis

A stable style transfer requires that the output stylized video to be as coherent as the input video. As described in Wang et al. (2020a), a coherent video should satisfy the following constraint:

$$\|X^m - W_{X^n \to X^m} X^n\| < \delta, \quad (7)$$

where $X^m, X^n$ are the $m$-th and $n$-th frames, respectively; and $W_{X^n \to X^m}$ is the warping matrix from $X^m$ to $X^n$. $\delta$ is a

minimum so that humans are not aware of the minor flickering artifacts. When the input video is coherent, we obtain the content features $f_c$ of each frame. And the content features of the $m$-th and $n$-th frame satisfy

$$\|f_c^m - W f_c^n\| < \delta. \tag{8}$$

For corresponding output features $f_{cs}^m$ and $f_{cs}^n$:

$$\begin{aligned} \|f_{cs}^m - W f_{cs}^n\| &= \|K f_c^m - W K f_c^n\| \\ &= |K| \cdot \|f_c^m - W f_c^n\| < |K| \cdot \delta. \end{aligned} \tag{9}$$

Then, the output video also satisfies:

$$\|f_{cs}^m - W f_{cs}^n\| < \gamma, \tag{10}$$

where $\gamma$ is a minimum. Therefore, our network can migrate the coherence of the input video frame features to the stylized frame features without additional temporal constraints. We further demonstrate that the coherence of stylized frame features can be well-transited to the generated video despite the convolutional operation of the decoder in Section . Such observations prove that the proposed MCCNet is suitable for video style transfer tasks.

## Network Structure and Training

In this section, we introduce how to involve MCC in the encoder-decoder based stylization framework (Figure 2). Given $f_c$ and $f_s$, we first normalize them and feed them to the $1 \times 1$ convolution layer. We then stretch $f_s \in \mathbb{R}^{C \times H \times W}$ to $f_s \in \mathbb{R}^{C \times 1 \times N}$, and $f_s f_s^T = [\|f_s^1\|_2, \|f_s^2\|_2, ..., \|f_s^C\|_2]$ is obtained through covariance matrix calculation. Next, we add a fully connected layer to weigh the different channels in $f_s f_s^T$. Through matrix multiplication, we weigh each content channel with a compound of different channels in $f_s f_s^T$ to obtain $f_{rs}$. Then, we add $f_c$ and $f_{rs}$ to obtain $f_{cs}$ defined in Equation 5. Finally, we feed $f_{cs}$ to a $1 \times 1$ convolutional layer and decode it to obtain the generated image $I_{cs}$.

As shown in Figure 4, our network is trained by minimizing the loss function defined as

$$\begin{aligned} \mathcal{L} = \lambda_{content}\mathcal{L}_{content} + \lambda_{style}\mathcal{L}_{style} \\ + \lambda_{id}\mathcal{L}_{id} + \lambda_{illum}\mathcal{L}_{illum}. \end{aligned} \tag{11}$$

The total loss function includes perceptual loss $\mathcal{L}_{content}$ and $\mathcal{L}_{style}$, identity loss $\mathcal{L}_{id}$ and illumination loss $\mathcal{L}_{illum}$ in the training procedure. The weights $\lambda_{content}$, $\lambda_{style}$, $\lambda_{id}$, and $\lambda_{illum}$ are set to $4$, $15$, $70$, and $3,000$ to eliminate the impact of magnitude differences.

**Perceptual loss.** We use a pretrained VGG19 to extract content and style feature maps and compute the content and style perceptual loss similar to AdaIN (Huang and Serge 2017). In our model, we use layer $conv4\_1$ to calculate the content perceptual loss and layers $conv1\_1$, $conv2\_1$, $conv3\_1$, and $conv4\_1$ to calculate the style perceptual loss. The content perceptual loss $\mathcal{L}_{content}$ is used to minimize the content differences between generated images and content images, where

$$\mathcal{L}_{content} = \|\phi_i(I_{cs}) - \phi_i(I_c)\|_2. \tag{12}$$

| Image size | 256 | 512 | 1024 |
|---|---|---|---|
| Ours | 0.013 | 0.015 | 0.019 |
| MAST | 0.030 | 0.096 | 0.506 |
| CompoundVST | 0.049 | 0.098 | 0.285 |
| DFP | 0.563 | 0.724 | 1.260 |
| Linear | 0.010 | 0.013 | 0.022 |
| SANet | 0.015 | 0.019 | 0.021 |
| AAMS | 2.074 | 2.173 | 2.456 |
| AdaIN | 0.007 | 0.008 | 0.009 |
| WCT | 0.451 | 0.579 | 1.008 |
| NST | 19.528 | 37.211 | 106.372 |

Table 1: Inference time of different methods.

The style perceptual loss $\mathcal{L}_{style}$ is used to minimize the style differences between generated images and style images:

$$\mathcal{L}_{style} = \sum_{i=1}^{L} \mathcal{L}_{style}^i, \tag{13}$$

$$\begin{aligned} \mathcal{L}_{style}^i = \|\mu(\phi_i(I_{cs})) - \mu(\phi_i(I_s))\|_2 \\ + \|\sigma(\phi_i(I_{cs})) - \sigma(\phi_i(I_s))\|_2, \end{aligned} \tag{14}$$

where $\phi_i(\cdot)$ denotes the features extracted from the $i$-th layer in a pretrained VGG19, $\mu(\cdot)$ denotes the mean of features, and $\sigma(\cdot)$ denotes the variance of features.

**Identity loss.** We adopt the identity loss to constrain the mapping relation between style features and content features, and help our model to maintain the content structure without losing the richness of the style patterns. The identity loss $\mathcal{L}_{id}$ is defined as:

$$\mathcal{L}_{id} = \|I_{cc} - I_c\|_2 + \|I_{ss} - I_s\|_2, \tag{15}$$

where $I_{cc}$ denotes the generated results using a common natural image as content image and style image and $I_{ss}$ denotes the generated results using a common painting as content image and style image.

**Illumination loss.** For a video sequence, the illumination may change slightly that is difficult to be discovered by humans. The illumination variation in video frames could influence the final transfer results and result in flicking. Therefore, we add random Gaussian noise to the content images to simulate light. The illumination loss is formulated as

$$\mathcal{L}_{Illum} = \|G(I_c, I_s) - G(I_c + \Delta, I_s)\|_2, \tag{16}$$

where $G(\cdot)$ is our generation function, $\Delta \sim \mathcal{N}(0, \sigma^2 I)$. With illumination loss, our method can be robust to complex light conditions in input videos.

# Experiments

Typical video stylization methods use temporal constraint and optical flow to avoid flickering in generated videos. Our method focuses on promoting the stability of the transform operation in the arbitrary style transfer model on


Figure 5: Comparison of image style transfer results. The first column shows style images, the second column shows content images. The remaining columns are stylized results of different methods.

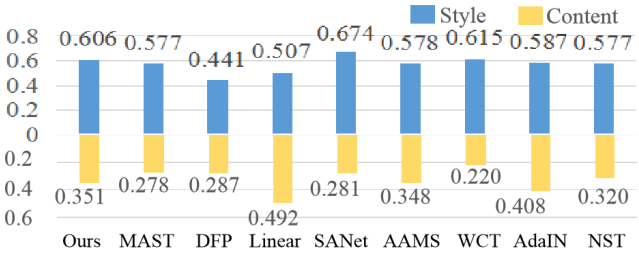



Figure 6: Classification accuracy.

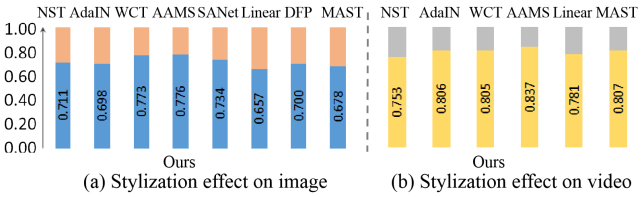



Figure 7: User study results.

the basis of single frame. Thus, the following frame-based SOTA stylization methods are selected for comparison: MAST (Deng et al. 2020), CompoundVST (Wang et al. 2020a), DFP (Wang et al. 2020b), Linear (Li et al. 2019), SANet (Park and Lee 2019), AAMS (Yao et al. 2019), WCT (Li et al. 2017), AdaIN (Huang and Serge 2017), and NST (Gatys, Ecker, and Bethge 2016).

In this section, we start from the training details of the proposed approach and then move on to the evaluation of image (frame) stylization and the analysis of rendered videos.

## Implementation Details and Statistics

We use MS-COCO (Lin et al. 2014) and WikiArt (Phillips and Mackintosh 2011) as the content and style image datasets for network training. At the training stage, the images are randomly cropped to $256 \times 256$ pixels. At the inference stage, images in arbitrary size are acceptable. The encoder is a pretrained VGG19 network. The decoder is a mirror version of the encoder, except for the parameters that need to be trained. The training batch size is 8, and the whole model is trained through $160,000$ steps.

**Timing information.** We measure our inference time for the generation of an output image and compare the result with those of SOTA methods using 16G TitanX GPU. The optimization-based method NST (Gatys, Ecker, and Bethge 2016) is trained for 30 epochs. Table 1 shows the inference times of different methods using three scales of image size. The inference speed of our network is much faster than that in (Gatys, Ecker, and Bethge 2016; Li et al. 2017; Yao et al. 2019; Wang et al. 2020b,a; Deng et al. 2020). Our method can achieve a real-time transfer speed that is comparable to that of (Huang and Serge 2017; Li et al. 2019; Park and Lee 2019) for efficient video style transfer.

## Image Style Transfer Results

**Qualitative analysis.** CompoundVST is not selected for image stylization comparison because it is only used for video style transfer. The comparisons of image stylizations are shown in Figure 5. On the basis of the optimized training mechanism, NST may introduce failure results (the second row), and it cannot easily achieve a trade-off between the content structure and style patterns in the rendered images.

In addition to crack effects caused by over-simplified calculation in AdaIN, the absence of correlation between dif-

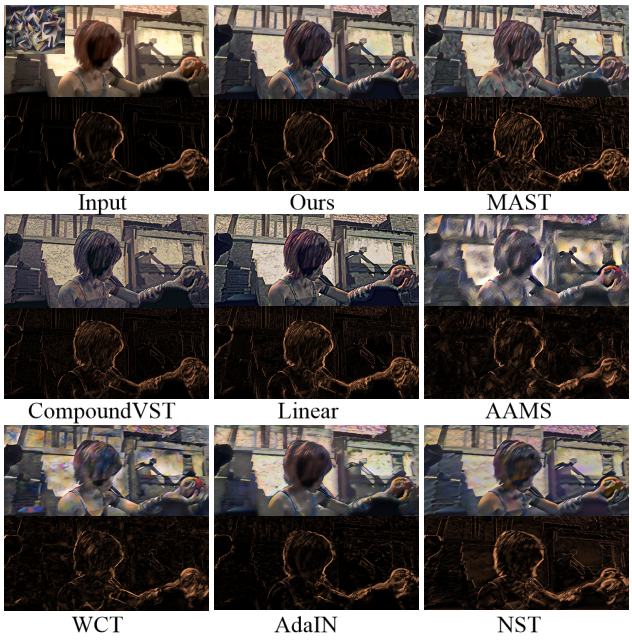



Figure 8: Visual comparisons of video style transfer results. The first row shows the video frame stylized results. The second row shows the heat maps which are used to visualize the differences between two adjacent video frame.

ferent channels causes a poor transfer of style color distribution badly transferred (the 4th row). As for WCT, the local content structures of generated results are damaged due to the global parameter transfer method. Although the attention map in AAMS helps to make the main structure exact, the patch splicing trace affects the overall generated effect. Some style image patches are transferred into the content image directly (the first row) by SANet and damage the content structures (the 4th row). By learning a transformation matrix for style transfer, the process of Linear is too simple to acquire adequate style textural patterns for rendering. DFP focuses on generating diversified style transfer results, but it may lead to failures similar to WCT. MAST may introduce unexpected style patterns in rendered results (the 4th and the 5th rows).

In our network, the rearranged style features fit the original content features, and the fused generated features consist of clear content structures and controllable style pattern information. Therefore, our method can achieve satisfactory stylized results.

**Quantitative analysis.** Two classification models are trained to assess the performance of different style transfer networks by considering their ability to maintain content structure and style migration. We generate several stylized images by using different style transfer methods aforementioned. Then we input the stylized images generated by different methods into the style and content classification models. High-accuracy style classification indicates that the style transfer network can easily learn to obtain effective style information while high-accuracy content classification indicates that the style transfer network can easily learn to maintain the original content information. From Figure 6, we can conclude that AAMS, AdaIN, and MCCNet can establish a balance between content and style. However, our network is superior to the two methods in terms of visual effects.

**User study** We conducted user studies to compare the stylization effect of our method with those of the aforementioned SOTA methods. We selected 20 content images and 20 style images to generate 400 stylized images using different methods. First, we showed participants a pair of content and style images. Second, we showed them two stylized results generated by our method and a random contrast method. Finally, we asked the participants which stylized result has the best rendered effects by considering the integrity of the content structures and the visibility of the style patterns. We collected $2,500$ votes from 50 participants and present the voting results in Figure 7(a). Overall, our method can achieve the best image stylization effect.

## Video Style Transfer Results

Considering the size limitation for input images of SANet and generation diversity of DFP, we do not include SANet and DFP for video stylization comparisons. We synthesize 14 stylized video clips by using the other methods and measure the coherence of the rendered videos by calculating the differences in the adjacent frames. As shown in Figure 8, the heat maps in the second row visualize the differences between two adjacent frames of the input or stylized videos. Our method can highly promote the stability of image style transfer. The differences of our results are closest to those of the input frames without damaging the stylization effect. MAST, Linear, AAMS, WCT, AdaIN, and NST fail to retain the coherence of the input videos. Linear can also generate a relatively coherent video, but the result continuity is influenced by nonlinear operation in deep CNNs.

Given two adjacent frames $F_t$ and $F_{t-1}$ in a T-frame rendered clip, we define $Diff_{F(t)} = ||F_t - F_{t-1}||$ and calculate the mean ($mean_{Diff}$) and variance ($var_{Diff}$) of $Diff_{F(t)}$. As shown in Table 2, we can conclude that our method can yield the best video results with high consistency.

**User study.** The global feature sharing used in CompoundVST increases the complexity of the model and limits the length of video clips that can be processed. Therefore, CompoundVST is not selected for comparison in this section. Then we conducted user studies to compare the video stylization effects of our method with those of the aforementioned SOTAs. First, we showed the participants an input video clip and a style image. Second, we showed them two stylized video clips generated by our method and a random contrast method. Finally, we asked the participants which stylized video clip is the most satisfying by considering the stylized effect and the stability of the videos. We collected 700 votes from 50 participants and present the the voting

| | Inputs | Ours | MAST | CompoundVST | Linear | AAMS | WCT | AdaIN | NST |
|---|---|---|---|---|---|---|---|---|---|
| Mean | 0.0143 | **0.0297** | 0.0548 | 0.0438 | 0.0314 | 0.0546 | 0.0498 | 0.0486 | 0.0417 |
| Variance | 0.0022 | **0.0054** | 0.0073 | 0.0054 | 0.0061 | 0.0067 | 0.0061 | 0.0067 | 0.0065 |

Table 2: Average $mean_{Diff}$ and $var_{Diff}$ of inputs and 14 rendered clips.

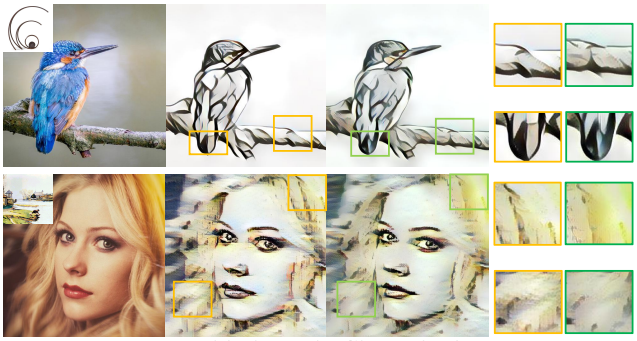



Figure 9: Ablation study of channel correlation. Without multi-channel correlation, the stylized results present less style patterns (e.g., hair of the woman) and may maintain original color distribution (the blue color in the bird's tail).


Figure 10: Ablation study of removing the illumination loss. The bottom row is heat maps used to visualize the differences between the above two video frames.

results in Figure 7(b). Overall, our method can achieve the best video stylization effect.

## Ablation Study

**Channel correlation.** Our network is based on multi-channel correlation calculation shown in Figure 3. To analyze the influence of multi-channel correlation on stylization effects, we change the model to calculate channel-wise correlation without considering the relationship between style channels. Figure 9 shows the results. Through channel-wise calculation, the stylized results show few style patterns (e.g., hair of the woman) and may maintain the original color distribution (the blue color in the bird's tail). Meanwhile, the style patterns are effectively transferred by considering the multi-channel information in the style features.

**Illumination loss.** The illumination loss is proposed to eliminate the impact of video illumination variation. We remove the illumination loss in the training stage and compare the results with ours in Figure 10. Without illumination loss,


Figure 11: Ablation study of network depth. Compared to the shallower network, our method generates artistic style transfer results with more stylized patterns.


Figure 12: Ablation study of network depth. The heat maps in the second row show the differences between two video frames in the first row.

the differences between the two video frames increase, with the mean value being 0.0317.

**Network depth.** We use a shallow auto-encoder up to $relu3$-1 instead of $relu4$-1 to discuss the effects of the convolution operation of the decoder on our model. As shown in Figure 11, for image style transfer, the shallower network can not generate results with vivid style patterns (e.g., the circle element in the first row of our results). As shown in Figure 12, the depth of the network exerts little impact on the coherence of the stylized video. This phenomenon suggests that the coherence of stylized frames features can be well-transited to the generated video despite the convolution operation of the decoder.

## Conclusion

In this work, we propose MCCNet for stable arbitrary video style transfer. The proposed network can migrate the coherence of input videos to stylized videos and thereby guarantee the stability of rendered videos. Meanwhile, MCCNet can generate stylized results with vivid style patterns and detailed content structures by analyzing the multi-channel correlation between content and style features. Moreover, the illumination loss improves the stability of generated video under complex light conditions.

## Acknowledgements

This work was supported by National Key R&D Program of China under no. 2020AAA0106200, and by National Natural Science Foundation of China under nos. 61832016, U20B2070 and 61672520.

## References

Bruckner, S.; and Gröller, M. E. 2007. Style transfer functions for illustrative volume rendering. *Computer Graphics Forum* 26(3): 715–724.

Chen, D.; Liao, J.; Yuan, L.; Yu, N.; and Hua, G. 2017. Coherent online video style transfer. In *IEEE International Conference on Computer Vision (ICCV)*, 1105–1114.

Chen, X.; Zhang, Y.; Wang, Y.; Shu, H.; Xu, C.; and Xu, C. 2020. Optical Flow Distillation: Towards Efficient and Stable Video Style Transfer. In *European Conference on Computer Vision (ECCV)*. Springer.

Deng, Y.; Tang, F.; Dong, W.; Sun, W.; Huang, F.; and Xu, C. 2020. Arbitrary Style Transfer via Multi-Adaptation Network. In *ACM International Conference on Multimedia*, 2719–2727. New York, NY, USA: Association for Computing Machinery.

Doyle, L.; Anderson, F.; Choy, E.; and Mould, D. 2019. Automated pebble mosaic stylization of images. *Computational Visual Media* 5(1): 33–44.

Dumoulin, V.; Shlens, J.; and Kudlur, M. 2016. A learned representation for artistic style. *arXiv preprint arXiv:1610.07629* .

Efros, A. A.; and Freeman, W. T. 2001. Image quilting for texture synthesis and transfer. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, 341–346.

Gao, W.; Li, Y.; Yin, Y.; and Yang, M.-H. 2020. Fast Video Multi-Style Transfer. In *The IEEE Winter Conference on Applications of Computer Vision*, 3222–3230.

Gatys, L. A.; Ecker, A. S.; and Bethge, M. 2016. Image style transfer using convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2414–2423. IEEE.

Huang, X.; Liu, M.-Y.; Belongie, S.; and Kautz, J. 2018. Multimodal unsupervised image-to-image translation. In *European Conference on Computer Vision (ECCV)*, 172–189. Springer.

Huang, X.; and Serge, B. 2017. Arbitrary style transfer in real-time with adaptive instance normalization. In *IEEE International Conference on Computer Vision (ICCV)*, 1501–1510. IEEE.

Jing, Y.; Liu, X.; Ding, Y.; Wang, X.; Ding, E.; Song, M.; and Wen, S. 2020. Dynamic Instance Normalization for Arbitrary Style Transfer. In *Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI)*, 4369–4376. AAAI Press.

Johnson, J.; Alahi, A.; and Fei-Fei, L. 2016. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision (ECCV)*, 694–711. Springer.

Li, X.; Liu, S.; Kautz, J.; and Yang, M.-H. 2019. Learning linear transformations for fast image and video style transfer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 3809–3817.

Li, Y.; Fang, C.; Yang, J.; Wang, Z.; Lu, X.; and Yang, M.-H. 2017. Universal style transfer via feature transforms. In *International Conference on Neural Information Processing Systems (NIPS)*, 386–396.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, 740–755. Springer.

Park, D. Y.; and Lee, K. H. 2019. Arbitrary Style Transfer With Style-Attentional Networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 5880–5888. IEEE.

Phillips, F.; and Mackintosh, B. 2011. Wiki Art Gallery, Inc.: A case for critical thinking. *Issues in Accounting Education* 26(3): 593–608.

Ruder, M.; Dosovitskiy, A.; and Brox, T. 2016. Artistic style transfer for videos. In *German Conference on Pattern Recognition*, 26–36. Springer.

Strothotte, T.; and Schlechtweg, S. 2002. *Non-photorealistic computer graphics: modeling, rendering, and animation*. Morgan Kaufmann.

Wang, W.; Xu, J.; Zhang, L.; Wang, Y.; and Liu, J. 2020a. Consistent Video Style Transfer via Compound Regularization. In *Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI)*, 12233–12240. AAAI Press.

Wang, Z.; Zhao, L.; Chen, H.; Qiu, L.; Mo, Q.; Lin, S.; Xing, W.; and Lu, D. 2020b. Diversified Arbitrary Style Transfer via Deep Feature Perturbation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7789–7798.

Yao, Y.; Ren, J.; Xie, X.; Liu, W.; Liu, Y.-J.; and Wang, J. 2019. Attention-aware multi-stroke style transfer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1467–1475. IEEE.

Zhu, J.-Y.; Park, T.; Isola, P.; and Efros, A. A. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision (CVPR)*, 2223–2232. IEEE.