Two Birds, One Stone: A Unified Framework for Joint Learning of Image and Video Style Transfers

Bohai Gu\textsuperscript{1,3}, Heng Fan\textsuperscript{2}, Libo Zhang\textsuperscript{1,3}†

\textsuperscript{1} Institute of Software Chinese Academy of Sciences, Beijing, China
\textsuperscript{2} Department of Computer Science and Engineering, University of North Texas, Denton TX, USA
\textsuperscript{3} University of Chinese Academy of Sciences, Beijing, China

Abstract

Current arbitrary style transfer models are limited to either image or video domains. In order to achieve satisfying image and video style transfers, two different models are inevitably required with separate training processes on image and video domains, respectively. In this paper, we show that this can be precluded by introducing UniST, a Unified Style Transfer framework for both images and videos. At the core of UniST is a domain interaction transformer (DIT), which first explores context information within the specific domain and then interacts contextualized domain information for joint learning. In particular, DIT enables exploration of temporal information from videos for the image style transfer task and meanwhile allows rich appearance texture from images for video style transfer, thus leading to mutual benefits. Considering heavy computation of traditional multi-head self-attention, we present a simple yet effective axial multi-head self-attention (AMSA) for DIT, which improves computational efficiency while maintains style transfer performance. To verify the effectiveness of UniST, we conduct extensive experiments on both image and video style transfer tasks and show that UniST performs favorably against state-of-the-art approaches on both tasks. Code is available at \url{https://github.com/NevSNev/UniST}.

1. Introduction

Artistic image style transfer \cite{9} aims at migrating a desirable style pattern from an inference image to the origin image while preserving the original content structures. Although CNNs based methods have been well studied in this field \cite{16, 21, 22, 29}, they fail to capture the long-range interaction between the style and content domains, which may result in suboptimal results.

\textsuperscript{†}Corresponding author: Libo Zhang (libo@iscas.ac.cn).

Figure 1: Comparison between single domain (image or video) style transfer and our joint style transfer.

Recently, owing to the ability to model long-range dependencies, Transformers \cite{31} have shown excellent performance in a wide range of tasks including style transfer. For example, Stytr\textsuperscript{2} \cite{9} introduces a pure transformer network to deal with image style transfer. However, pixel-level self-attention brings additional computational complexity, resulting in lower efficiency.

Unlike image style transfer, video style transfer brings in new challenges of preserving temporal consistency between stylized video frames. To achieve style transfer on the video domain, a feasible solution is to adapt existing image-based style transfer methods (\textit{e.g.} \cite{21, 25}) by retraining them with modification in architecture and/or loss functions. Despite simplicity, this domain adaption requires another repetitive and tedious training process, resulting in resource waste to some extent. Some other methods (\textit{e.g.} \cite{7, 38}) directly adopt the same model from video to image, but the results are somewhat visually flawed.

To solve the above issues, we present a Unified Style
Transfer framework, termed UniST, for both images and videos. The proposed network leverages the local and long-range dependencies jointly. More specifically, UniST first applies CNNs to generate tokens, and then models long-range dependencies to excavate domain-specific information with domain interaction transformer (DIT). Afterwards, DIT sequentially interacts contextualized domain information for joint learning. Considering that the vanilla self-attention suffers from a heavy computational burden, we are inspired by axial attention [32] and develop the Axial Multi-head Self-Attention (AMSA) mechanism to calculate attention efficiently for either images or video input. To our best knowledge, our approach is the first unified solution to handle both image and video style transfers simultaneously.

To verify the effectiveness of our approaches, we carry out extensive experiments on ImageNet [6] and MPI [2] for image and video field respectively. The results demonstrate that our unified solution can achieve better performance than current state-of-the-art image-based and video-based algorithms, evidencing its superiority and efficiency.

In summary, we make the following contributions in this work: (1) We propose a new joint learning framework for arbitrary image and video style transfers, in which two tasks can benefit from each other to improve the performance. To our best knowledge, this is the first work towards a unified solution with joint interaction. (2) We develop the Axial Multi-head Self-Attention mechanism to address computational complexity and adapt to tokens from image and video input. (3) Extensive experiments on both image and video style transfer tasks demonstrate the effectiveness of our approach compared with state-of-the-art methods.

2. Related Work

Image Style Transfer CNNs based style transfer models are widely applied in the image field. Gatys et al. [13] apply the CNN model to iteratively generate stylized outputs. Johnson et al. [18] adopt an end-to-end model to accomplish real-time style transfer for the specific style. More generally, fast arbitrary style transfer is attracting enormous attention. Therefore, Huang et al. [16] achieve arbitrary style transfer by adaptively applying mean and standard deviation of style to that of content (AdaIN), which is widely adopted in image generation tasks [19, 23] for better feature fusion. Similarly, Li et al. [22] accomplish style transfer with two transformation steps including whitening and coloring. Then, Sheng et al. [29] design a multi-scale model combined with AdaIN and style-swap.

Recently, [4, 7, 8, 25–27, 37] introduce the self-attention mechanism to the encoder-transfer-decoder framework for better style transfer. Moreover, Deng et al. [9] take advantage of the Transformer’s long-range dependencies while refusing to adopt the CNN’s local dependencies. Meanwhile, the pure transformer is hugely computational and the position encoding needs to be presented specially. All of the above lead to slow inference speed. In contrast, UniST leverages both the Transformer’s long-range dependencies and CNN’s locality dependencies to build a unified framework for joint learning of image and video style transfers. After one pass of training, our framework can generate vivid image and video stylization results in real-time applications.

Video Style Transfer In addition to image style transfer, video style transfer presents new challenges, including both vivid stylization results and well-maintained temporal consistency. To this end, many previous works [3, 12, 15, 39] directly add optical flows consistency constraint to image style transfer solutions to enhance the inter-frame correlation. However, the optical flow requires extra complex computation, making it impractical to process high-resolution or long videos. So there emerge some works that addresses the stability issue with other approaches instead of optical flow warping. Li et al. [21] present a linear transformation module which based on content and style features. And Wu et al. [36] propose a SANet based framework that addresses the temporal consistency with a SSIM consistency...
constraint. Moreover, Deng et al. [7] learn the per-channel correlation via a transformed self-attention. On this routine, Liu et al. [25] improve the temporal consistency via a modified self-attention and a cross-image similarity loss. Similarly, Wu et al. [38] devise a generic contrastive coherence preserving loss applied to local patches. Despite meeting multi-task domains, the results are somewhat flawed.

In this work, without using any inter-frame information like optical flows, UniST takes advantage of the unified image-video joint learning style transfer framework to facilitate video stylization effects. And the experiments demonstrate that it also achieves great temporal consistency.

3. Methodology

Overall Framework. Given a style image \(I_s \in R^{H \times W \times 3}\) and the content sequence \(Seq_c \in R^{T \times H \times W \times 3}\), which is the concatenation of image and video. Our framework eventually synthesizes the stylized sequence \(Seq_{cs} \in R^{T \times H \times W \times 3}\). As in Figure 2, our framework leverages local and long-range dependencies jointly. Notably, the CNN encoder is not only used for local spatial information extraction, but also for tokenization. Then we use \(DIT\) to accomplish two types of style transfer task jointly, which first explores context information within the content and style domains and then interacts contextualized domain information for joint learning. Below, we will detail our framework.

3.1. Tokenization

As discussed above, Deng et al. [9] split the input images into patches directly for transformer input with the necessary position encoding, resulting in low efficiency. In this work, we take advantage of CNNs to strengthen locality and improve the efficiency. Similar to [16], we use the pre-trained VGG-19 network [30] to extract feature maps of input images. Then, the 512-dim vector at each pixel in the \(relu_{4,1}\) layer is treated as a token for further transformer encoders. In this way, we combine CNNs and transformer to exploit both local and long-range dependencies. Meanwhile, there is no need to maintain the position encoding.

3.2. Domain Interaction Transformer

As in Figure 2, \(DIT\) consists of three modules in turn, namely intra-domain extraction, video-image and style-content interaction. Specifically, the domain-specific extraction module is stacked with \(N_c\) \(N_s\) encoder blocks for the \(Seq_c\) and \(I_s\) respectively. While content-style interaction is stacked with \(N_t\) transformer decoder blocks.

In-Domain Extraction Based on the tokenization, the local context has already been extracted. So \(DIT\) further exploits domain-specific information with a number of consecutively stacked transformer encoder blocks in Figure 2(b). Taking either \(Seq_c\) or \(I_s\) as input, the in-domain extraction module simultaneously capture the long-range information within the content and style domains. Normally, the transformer encoder block consists of a multi-head self-attention (MSA) layer and a feed-forward network (FFN). For the better efficiency, we develop the Axial Multi-head Self-Attention (AMSA) mechanism to replace MSA, combined with a \(1 \times 1\) convolutional layer for locality strengthening. And this mechanism is applied to all of the transformer blocks mentioned below. In addition, residual connections and layer normalization are deployed after each layer. The transformer encoder block is defined as:

\[
S' = AMSA(Conv(Q), Conv(K), Conv(V)) + Q, \quad (1)
\]

\[
S = FFN(S') + S', \quad (2)
\]

where \(S\) is the output sequence.

Video-Image Interaction. After domain-specific excava-
tion, \(DIT\) presents a symmetric module based on two transformer encoder blocks to interact contextual information between two types of content modalities. Notably, we split the input content sequence into two parts: one half is video \(Seq_v\) and the other half is image \(Seq_l\). As illustrated in Figure 2, one of the blocks takes the \(Seq_v\) as \(Q\), the \(Seq_l\) as \(K, V\), and the other one does the opposite. In this way, joint learning is performed for style transfer. And two types of content sequence are concatenated after interaction.

Content-Style Interaction. To finally capture the rele-
vance between the content and the style domain, this mod-
ule consists of a set of stacked transformer decoder blocks. As in Figure 2(c), each transformer decoder block takes the content as \(Q\) while the style as the \(K\) and \(V\), alternately containing two AMSA layers and one FFN layer. Similarly, layer normalization and residual connections are applied after each layer. The transformer decoder block is defined as:

\[
S'' = AMSA(Conv(Q), Conv(K), Conv(V)) + Q, \quad (3)
\]

\[
S' = AMSA(Conv(S''), Conv(K), Conv(V)) + S'', \quad (4)
\]

\[
S = FFN(S') + S'. \quad (5)
\]

3.3. Axial Multi-head Self-Attention.

Inspired by [32], \(DIT\) computes self-attention along a separate axis, rather than in feature maps like other trans-
former models [9,11,14,35,40]. The improved AMSA layer is specialized for style transfer without position encoding.

An AMSA layer consists of two MSA [31] layers in total, operating sequentially along the height and width axes. As in Figure 3, we first reshape Q, K, V to separate the width dimension information, then use the first MSA layer to calculate self-attention along the height axis, and the output f is normalized using the layer normalization (LN) layer. Following almost the same process as before, except that what we separate in the second MSA layer is the height dimension. Notably, we use the previous output f as Q and V, while keeping K unchanged. Experiments in Table 4 verify this dedicated AMSA layer reduces the memory consumption and meanwhile improves inference efficiency.

The reasons behind our AMSA are two-fold. First, the information contained in the previous K, V is crucial for the second MSA layer to learn contextual dependencies either within or across domains. Besides, the V of the second MSA contains the last axial information, which is critical to ensure the stability of stylization. Experiments in Figure 10 show that our design effectively prevents artifacts caused by side effects of axial attention in style transfer.

3.4. Unimodal input for inference.

In inference, UniST takes bimodal content input and outputs both style transfer results simultaneously, achieving two birds with one stone. Since the model has learned complementary content knowledge, UniST could also take unimodal content input into account by replacing the cross-attention with self-attention in the video-image interaction without compromising the quality of results. As in Table 6, unimodality obtains the consistent scores with bimodality.

3.5. Loss functions.

The overall loss function is the weighted summation of the content loss $L_c$, style loss $L_s$, identity loss $L_{id}$ and temporal loss $L_t$:

$$L = \lambda_c L_c + \lambda_s L_s + L_{id} + \lambda_t L_t,$$

where $\lambda_c$, $\lambda_s$ and $\lambda_t$ are balancing factors. UniST uses the pre-trained VGG-19 to extract feature maps as $\phi = \{\text{relu}1_1, \text{relu}2_1, \text{relu}3_1, \text{relu}4_1\}$. In our experiment, we use the above four layers with equal weights to calculate the loss below.

We use Euclidean distance [10] to compute content loss $L_c$:

$$L_c = \sum_{i=1}^{4} \|\phi_i(Seq_c) - \phi_i(Seq_i)\|_2.$$  \hfill(7)

Following [16], the style loss $L_s$ is defined as:

$$L_s = \sum_{i=1}^{4} (\|\mu(\phi_i(Seq_s)) - \mu(\phi_i(I_s))\|_2 + \|\sigma(\phi_i(Seq_s)) - \sigma(\phi_i(I_s))\|_2),$$  \hfill(8)

where $\mu(\cdot)$ and $\sigma(\cdot)$ denote the mean and variance of features separately. We further use the identity loss $L_{id}$ [27] to promote more accurate content and style representation:

$$L_{id} = \lambda_{id1}(\|Seq_c - Seq_i\|_2 + \|I_{ss} - I_s\|_2) + \lambda_{id2}(\sum_{i=1}^{4} \|\phi_i(Seq_c) - \phi_i(Seq_i)\|_2 + \|\phi_i(I_{ss}) - \phi_i(I_s)\|_2),$$  \hfill(9)

where $\lambda_{id1}$ and $\lambda_{id2}$ both are balancing factors. $Seq_c$, $I_{ss}$ denote the results synthesized from two identical content sequences or style images. Figure 4 illustrates the identity loss for better understanding.

Following the recent work AdaAttN [25], we preserve the temporal consistency via a cross-image temporal loss. This loss promotes the cosine distance $D_{sw}$ between adjacent stylization frames to be closer to the cosine distance $D_c$ between adjacent origin frames.

$$L_{id} = \lambda_{id1}(\|Seq_c - Seq_i\|_2 + \|I_{ss} - I_s\|_2),$$  \hfill(6)

where $D_{sw} = 1 - \frac{F_{m,n,x}^{sw}}{\|F_m\| \times \|F_n\|}$. $N$ is the spatial dimension of the current feature map. $F_{m,n,x}^{sw}$ measures cosine distance, and $F_k$ represents the feature vector of the k-th entry. Note, we only adopt layer relu2_1 and relu4_1 to calculate $L_t$. Meanwhile, in each training iteration, we compute the temporal loss between the consecutive video frames. Noted that temporal information is implicitly encoded in this way.

4. Experiment

4.1. Implementing Details

UniST is trained with MS-COCO [24] (image), MPI [2] (video) as the content datasets and WikiArt [28] as the style datasets. For $Seq_c$, the ratio of the two types of content is 1 : 1, and the total number is 6. In the training phase, all images are loaded with the resolution of 256 x 256. While in the inference phase, UniST can be applied to any resolutions. Therefore, we manage to extend the UniST to multi-granularity style transfer (see supplementary material). In experiments, we adopt the resolutions of 1024 x 1024 and 2048 x 2048 for image and video respectively. For the loss function, the balancing factors $\lambda_c$, $\lambda_s$, $\lambda_t$, $\lambda_{id1}$, $\lambda_{id2}$ are set as 0.1, 1.5, 90, 0.1, 0.5, empirically. The number of transformer blocks $N_c$, $N_s$, $N_t$ are empirically set as 2, 1, 3. Our
network is trained for 50K iterations on a NVIDIA Tesla V100 GPU with a batch size of 3. We use the Ranger optimizer [34] with initial learning rate of 0.00005.

4.2. Image Style Transfer

We compare our method with 13 state-of-the-art methods, including AdaIN [16], SANet [27], Linear [21], MCCNet [7], MAST [8], Artflow [1], AdaAttN [25], StyleFormer [37], IEST [4], Stytr2 [9], CCPL [38], epsAM [5], MicroAST [33], MCCNetV2 [20].

Quantitative comparison. We randomly collect 17, 124 content images from ImageNet [6] and 17, 124 style images from WikiArt [28] which are separated from the training set to generate 17, 124 stylization results. Similar to Stytr2 [9], we use the mean euclidean distance and the mean instance statistics difference mentioned in section 3.5 as metrics for content preservation and stylization degree. Furthermore, we conduct the color distribution experiments by adopting color loss in DSLR [17] and adopt the Gram matrices [13] for texture difference. As in Table 1, compared with existing methods, UniST achieves the best performance in both content and style differences and obtains promising results on texture and color differences of style aspects.

Qualitative comparison. As in Figure 5, AdaIN [16] transfers the style patterns but losses important content details (1st, 2nd rows). SANet [27] fails to align the distribution of the style patterns, leading to the distorted object boundaries (1st, 3rd, 5th rows) and inconsistent content background structures (2nd, 4th rows). The stylization of Linear [21] is not satisfactory enough, resulting relatively light migration effects such as pink in the 5th row. As with the linear transformation, MCCNet [7] learns the correlation of each channel through the transformed self-attention, slightly improving the transfer effect, but there are serious overflow problems around the object boundaries (1st, 3rd, 4th rows). MAST [8] distorts the content background structure with excessive style transfer (3rd, 4th rows). Based on WCT [22] in our setting, Artflow [1] leads to conspicuous vertical artifacts at the edges of the generated results (3rd, 5th rows). AdaAttN [25] does the decent foreground style transfer, but requires further rendering of the background (3rd, 5th rows). StyleFormer [37] provides vivid style patterns, but the colors of the results are inconsistent with the style reference images (1st, 3rd, 4th rows). On the contrary, IEST [4] presents stylization results with consistent colors, lacking more desirable style patterns (3rd, 4th, 5th rows). Stytr2 [9] fails to preserve the background structure of the content inference image (2nd, 5th rows). CCPL [38] transfers the style patterns with a lot of vertical artifacts (1st, 3rd, 4th rows). In contrast, based on the inductive bias of CNNs, our method captures the relevance between content and style sequences jointly and efficiently. As a result, we provide vivid stylized results with desirable style pattern details, while keeping the content structure well-maintained.

4.3. Video Style Transfer

For video style transfer, we compare our method with four state-of-the-art methods including Linear [21], MCCNet [7], AdaAttN [25], and CCPL [38]. Note, optical flow is not used for stabilization when conducting comparison.

Quantitative comparison. Following Liu et al. [25], we adopt the official optical flow [2] to wrap the output stylized frame and compute the per-pixel difference between warped and stylized frames. Meanwhile, we use the LPIPS [41] to measure the diversity of adjacent stylized frames, with smaller values indicating better consistency.

In practise, the style input is fixed at 512 × 512, and the content input is fixed at 256 × 256. Table 2 presents optical...
Table 1: Quantitative comparison in image style transfer. The best two results are highlighted in bold and underline.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Optical flow error</th>
<th>LPIPS (↓)</th>
<th>ΔD (↓)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours</td>
<td>11.36</td>
<td>10.30</td>
<td>13.13</td>
</tr>
<tr>
<td>AdAttN (CCVT)</td>
<td>9.69</td>
<td>155.54</td>
<td>177.20</td>
</tr>
<tr>
<td>MCCNetv2</td>
<td>9.60</td>
<td>6.83</td>
<td>6.50</td>
</tr>
<tr>
<td>CCPL</td>
<td>9.27</td>
<td>4.25</td>
<td>4.28</td>
</tr>
<tr>
<td>StyleFormer</td>
<td>9.54</td>
<td>7.10</td>
<td>7.08</td>
</tr>
<tr>
<td>IEST</td>
<td>13.57</td>
<td>14.66</td>
<td>16.82</td>
</tr>
<tr>
<td>AdaAttN</td>
<td>13.57</td>
<td>14.66</td>
<td>16.82</td>
</tr>
<tr>
<td>CCPL (CCVT)</td>
<td>13.57</td>
<td>14.66</td>
<td>16.82</td>
</tr>
<tr>
<td>SANet</td>
<td>13.57</td>
<td>14.66</td>
<td>16.82</td>
</tr>
<tr>
<td>Linear</td>
<td>13.57</td>
<td>14.66</td>
<td>16.82</td>
</tr>
<tr>
<td>Ours</td>
<td>13.57</td>
<td>14.66</td>
<td>16.82</td>
</tr>
</tbody>
</table>

Table 2: The optical flow error ($\times 10^{-2}$) and LPIPS of STOs using 20 styles. Smaller values mean better temporal consistency. For clarity, only three styles are presented. Please refer to supplementary material for more details.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Optical flow error</th>
<th>LPIPS (↓)</th>
<th>ΔD (↓)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours</td>
<td>0.46</td>
<td>1.03</td>
<td>0.92</td>
</tr>
<tr>
<td>AdAttN (CCVT)</td>
<td>69.69</td>
<td>155.54</td>
<td>177.20</td>
</tr>
<tr>
<td>MCCNetv2</td>
<td>178.94</td>
<td>16546</td>
<td>21128</td>
</tr>
<tr>
<td>CCPL</td>
<td>20292</td>
<td>21971</td>
<td>18344</td>
</tr>
<tr>
<td>StyleFormer</td>
<td>23953</td>
<td>21543</td>
<td>21739</td>
</tr>
<tr>
<td>IEST</td>
<td>20566</td>
<td>19618</td>
<td>20241</td>
</tr>
<tr>
<td>AdaAttN</td>
<td>19811</td>
<td>20948</td>
<td>21120</td>
</tr>
</tbody>
</table>

4.4. User Study

Furthermore, we carry out the user study experiment for comparison. Specifically, we use 27 content images and 21 style images to synthesize 567 images in total. Given 20 randomly combinations of content and style, the generated results obtained by 12 image-based style transfer methods. Then, we ask 100 participants to select their favorite one from three aspects: content preservation, style transfer, and overall preference. We collect 2,000 votes and show results in Figure 6(a). The results verifies the superiority of our method over other models for image style transfer. Similarly, we take 12 videos of 50 frames and 21 style images to synthesize 252 video stylized clips in total. Given random combinations of video and style, the stylization clips obtained by 5 video-based style transfer solutions. Then, we ask another 50 subjects to select their favorite one from three views: temporal consistency, stylization effect (considering both content preservation and stylization degree), and overall preference. We collect 200 votes and our method is selected as the best as shown in Figure 6(b).

For participants, there are 83 males and 67 and females (55/45 males/females for image, the other 28/22 males/females for video), aged from 23 to 42.

4.5. Efficiency Analysis

In Table 3, we report the inference time of our joint learning method and other approaches. Note that all the methods are run using a single TITAN XP GPU card. Although our joint learning framework consists of many stacked self-attention layers, our model can still achieve 35 FPS at 256px, which is comparable with attention-based methods such as AdaAttN [25] and Stytr2 [9]. The main limitation of our work is the inference speed is limited when applied for high resolution input, which may hinder its usage.

Notably, our AMSA mechanism is crucial for the joint learning framework to address the expensive memory consumption and huge computation complexity. In particular, the computation complexity can be reduced from $O(H^2 \times$...
Figure 7: Qualitative comparison in video style transfer. Column 4 shows the difference map between the adjacent frames at frame 20.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Ours</th>
<th>SANet</th>
<th>Linear</th>
<th>MCCNet</th>
<th>Artflow</th>
<th>Stytr</th>
<th>CCPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>256 × 256</td>
<td>0.028</td>
<td>0.016</td>
<td>0.011</td>
<td>0.087</td>
<td>0.144</td>
<td>0.109</td>
<td>0.019</td>
</tr>
<tr>
<td>512 × 512</td>
<td>0.111</td>
<td>0.053</td>
<td>0.062</td>
<td>0.144</td>
<td>0.760</td>
<td>0.751</td>
<td>0.064</td>
</tr>
</tbody>
</table>

Table 3: Inference time (sec./image) comparison.

Table 4: Efficiency of AMSA by comparison with MSA.

<table>
<thead>
<tr>
<th>Models</th>
<th>Memory(MiB)</th>
<th>FLOPS (G)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSA</td>
<td>1.8×10^4</td>
<td>4.29</td>
</tr>
<tr>
<td>AMSA</td>
<td>1.1×10^4</td>
<td>0.27</td>
</tr>
</tbody>
</table>

4.6. Ablation Study

Video-image interaction module. Since images and videos have different domain features, we design a cross-attention domain-interaction module for joint learning of two domains for better content knowledge, which improves the overall stylization results. We conduct an experiment by removing the cross-attention interaction module. As in the 2nd, 3rd columns of Table 5, both content preservation and style transfer are highly degraded without this module.

To further demonstrate the superiority of UniST, we compare 3 training strategies without video-image interaction module. As shown in the 4th, 5th, 6th columns of Table 5, UniST achieves the best performance compared with training separately and sequentially. Meanwhile, as shown in Figure 8, video style transfer is improved by learning complex appearances and textures from the image domain.
with UniST framework. Similarly, Figure 9 shows UniST enhances image stylization effects with more vivid results.

<table>
<thead>
<tr>
<th>Training with different images</th>
<th>$D_C$</th>
<th>$D_S$</th>
</tr>
</thead>
<tbody>
<tr>
<td>UniST with videos and 60K images</td>
<td>12.36</td>
<td>0.46</td>
</tr>
<tr>
<td>UniST using videos and 60K images without interaction</td>
<td>15.44</td>
<td>0.69</td>
</tr>
<tr>
<td>Image-only UniST with 60K images</td>
<td>19.80</td>
<td>0.73</td>
</tr>
<tr>
<td>Image-only UniST with 70K images</td>
<td>19.80</td>
<td>0.76</td>
</tr>
<tr>
<td>Image-only UniST with 80K images</td>
<td>19.98</td>
<td>0.71</td>
</tr>
</tbody>
</table>

Table 8: Comparison of UniST with image-only UniST using more training images.

Table 5: Effectiveness of joint learning framework. “video only” and “image only” indicate unimodal input for training, and “Sequential learning” means that we first train the network with one modality and then fine-tune it with another. Specifically, except for the UniST, the others all use the version without video-image interaction module.

Table 6: Metrics comparison of different input.

Table 7: Quantitative comparison under consistent training dataset conditions.

**Inconsistent datasets concerns.** In our method, we adopt two datasets from different modalities for learning. To ensure fairness, we have retrained five image style transfer models, while keeping the training dataset consistent with us. We provide the quantitative comparison in Table 7. Meanwhile, we conduct extra experiments by training the image-only UniST with different dataset scales in Table 8. The results show simply using more images for image-only UniST brings no obvious improvements, while gains come from different modalities and beneficial interactions, and further illustrating the superiority of our model.

**Axial Multi-head Self-Attention.** To verify the effectiveness of AMSA layer, we conduct two different design comparisons in Figure 10. First, the information contained

Figure 9: Visualization of image style transfer with different training strategies.

<table>
<thead>
<tr>
<th>Content</th>
<th>Style</th>
<th>UniST</th>
<th>Image Only</th>
<th>Sequential</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Content" /></td>
<td><img src="image2.png" alt="Style" /></td>
<td>UniST <img src="image3.png" alt="UniST" /></td>
<td>Image Only <img src="image4.png" alt="Image Only" /></td>
<td>Sequential <img src="image5.png" alt="Sequential" /></td>
</tr>
</tbody>
</table>

Figure 10: Ablation study of the AMSA layer. “Standard” is the normal version in Figure 3. (a) We uniformly take the output of the first MSA as the $Q$, $K$, $V$ for the second MSA. The results in 4th column show that it is difficult for the model to capture the relevance between content and style domains in this way. Second, the first axial information is necessary for the $V$ of the second MSA to maintain the stable stylization results. To demonstrate this, we use the output of the first MSA as the $Q$ for the second MSA, while keeping the $K$ and $V$ unchanged. As in the 5th column, the style patterns transferred are mixed with bar artifacts. From the results in the 3th column, the original design in Figure 3 is necessary for our model to prevent side effects caused by axial attention.

5. Conclusion

In this work, we propose an unified style transfer framework, dubbed UniST, for image and video. The key is our novel domain interaction transformer that enables effective mutual feature learning from different modalities for enhancements. Besides, an axial multi-head attention is proposed to capture attentions either within or across the field efficiently. Experiment shows the mixing content input effectively improves stylization results via UniST.

Acknowledgement. Libo Zhang was supported by Youth Innovation Promotion Association, CAS (2020111). Heng Fan and his employer received no financial support for research, authorship, and/or publication of this article.
References


