S2WAT: Image Style Transfer via Hierarchical Vision Transformer Using Strips Window Attention

Chiyu Zhang[1,2], **Xiaogang Xu**[3,4,*], **Lei Wang**[1], **Zaiyan Dai**[1], **Jun Yang**[1,5,*]
[1]Sichuan Normal University
[2]Nanjing University of Aeronautics and Astronautics
[3]Zhejiang Lab
[4]Zhejiang University
[5]Visual Computing and Virtual Reality Key Laboratory of Sichuan Provience
{alienzhang19961005, xiaogangxu00, londmi9, zaiyan.dai}@gmail.com, jkxy_yjun@sicnu.edu.cn

Abstract

Transformer's recent integration into style transfer leverages its proficiency in establishing long-range dependencies, albeit at the expense of attenuated local modeling. This paper introduces Strips Window Attention Transformer (S2WAT), a novel hierarchical vision transformer designed for style transfer. S2WAT employs attention computation in diverse window shapes to capture both short- and long-range dependencies. The merged dependencies utilize the "Attn Merge" strategy, which adaptively determines spatial weights based on their relevance to the target. Extensive experiments on representative datasets show the proposed method's effectiveness compared to state-of-the-art (SOTA) transformer-based and other approaches. The code and pre-trained models are available at https://github.com/AlienZhang1996/S2WAT.

Introduction

Background. Image style transfer imparts artistic characteristics from a style image to a content image, evolving from traditional (Efros and Freeman 2001) to iterative (Gatys, Ecker, and Bethge 2015, 2016) and feed-forward methods (Johnson, Alahi, and Fei-Fei 2016; Chen et al. 2017). Handling multiple styles concurrently remains a challenge, addressed by Universal Style Transfer (UST) (Park and Lee 2019; Kong et al. 2023; Li et al. 2022). This sparks innovative approaches like attention mechanisms for feature stylization (Yao et al. 2019; Deng et al. 2020; Chen et al. 2021), the Flow-based method (An et al. 2021) for content leakage, and Stable Diffusion Models (SDM) for creative outcomes (Zhang et al. 2023). New neural architectures, notably the transformer, show remarkable potential. (Deng et al. 2022) introduces StyTr2, leveraging transformers for SOTA performance. However, StyTr2's encoder risks losing information due to one-time downsampling, impacting local details with global MSA (multi-head self-attention).

Challenge. To enhance the transformer's local modeling capability, recent advancements propose the use of window-based attention computation, exemplified by hierarchical structures like Swin-Transformer (Liu et al. 2021). However, applying window-based transformers directly for feature extraction in style transfer can lead to grid-like patterns, as depicted in Fig. 1 (c). This arises due to the localized nature of window attention, termed the *locality problem*. While window shift can capture long-range dependencies (Liu et al. 2021), it necessitates deep layer stacks, introducing substantial model complexity for style transfer, particularly with high-resolution samples.

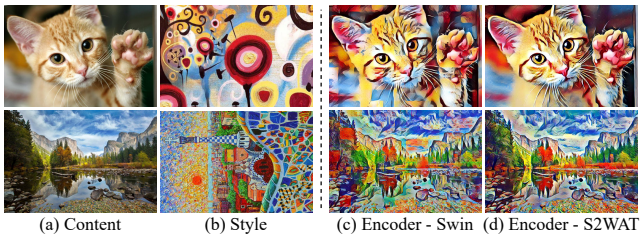

(a) Content (b) Style (c) Encoder - Swin (d) Encoder - S2WAT

Figure 1: Illustration for locality problem. (c) Results of the Swin-based model. (d) Results from our S2WAT.

Motivation and Technical Novelty.Diverging from current transformer-based approaches, we introduce a novel hierarchical transformer framework for image style transfer, referred to as S2WAT (**S**trip**s** **W**indow **A**ttention **T**ransformer). This structure meticulously captures both local and global feature extraction, inheriting the efficiency of window-based attention. In detail, we introduce a distinct attention mechanism (**S**tri**p**s **W**indow Attention, SpW Attention) that amalgamates outputs from multiple window attentions of varying shapes. These diverse window shapes enhance the equilibrium between modeling short- and long-range dependencies, and their integration is facilitated through our devised "Attn Merge" technique.

In this paper, we formulate the SpW Attention in a simple while effective compound mode, which encompasses three window types: horizontal strip-like, vertical strip-like, and square windows. The attention computations derived from strip windows emphasize long-range modeling for extracting non-local features, while the square window attention focuses on short-range modeling for capturing local features.

Furthermore, the "Attn Merge" method combines attention outputs from various windows by computing spatial correlations between them and the input. These calculated correlation scores serve as merging weights. In contrast to

*Corresponding authors

static merge strategies like summation and concatenation, "Attn Merge" dynamically determines the significance of different window attentions, thus enhancing transfer effect.

Contributions. Extensive quantitative and qualitative experiments are conducted to prove the effectiveness of the proposed framework, including a large-scale user study. The main contributions of our work include:

- We introduce a pioneering image style transfer framework, S2WAT, founded on a hierarchical transformer. This framework adeptly undertakes both short- and long-range modeling concurrently, effectively mitigating the challenge of locality issues.
- We devise a novel attention computation within the transformer for style transfer, termed SpW Attention. This mechanism intelligently merges outputs from diverse window attentions using the "Attn Merge" approach.
- We extensively evaluate our proposed S2WAT on well-established public datasets, demonstrating its state-of-the-art performance for the style transfer task.

Related Work

Image Style Transfer. Style transfer methods can fall into single-style (Ulyanov, Vedaldi, and Lempitsky 2016), multiple-style (Chen et al. 2017), and arbitrary-style (UST) (Zhang et al. 2022; Kong et al. 2023; Ma et al. 2023) categories based on their generalization capabilities. Besides models based on CNNs, recent works include Flow-based ArtFlow (An et al. 2021), transformer-based StyTr2 (Deng et al. 2022), and SDM-based InST (Zhang et al. 2023). ArtFlow, with Projection Flow Networks (PFN), achieves content-unbiased results, while IEST (Chen et al. 2021) and CAST (Zhang et al. 2022) use contrastive learning for appealing effects. InST achieves creativity through SDM. Models like (Wu et al. 2021b; Zhu et al. 2023; Hong et al. 2023) use transformers to fuse image features, and (Liu et al. 2022; Bai et al. 2023) encode text prompts for text-driven style transfer. StyTr2 leverages transformers as the backbone for pleasing outcomes. Yet, hierarchical transformers remain unexplored in style transfer.

Hierarchical Vison Transformer. Lately, there has been a resurgence of interest in hierarchical architectures within the realm of transformers. Examples include LeViT (Graham et al. 2021) & CvT (Wu et al. 2021a), which employ global MSA; PVT (Wang et al. 2021) & MViT (Fan et al. 2021), which compress the resolution of K & V. However, in these approaches, local information is not adequately modeled. While Swin effectively captures local information through shifted windows, it still gives rise to the locality problem when applied to style transfer (see Fig. 1). Intuitive attempts, such as inserting global MSA (see Section Pre-Analysis) or introducing Mix-FFN (Xie et al. 2021) by convolutions (see appendix), are powerless for locality problem. In the context of style transfer, a promising avenue involves advancing further with a new transformer architecture that encompasses both short- and long-range dependency awareness and possesses the capability to mitigate the locality problem.

Differences with Other Methods While the attention mechanism in certain prior methods may share similarities with the proposed SpW Attention, several key distinctions exist. 1) The fusion strategy stands out: our proposed "Attn Merge" demonstrates remarkable superiority in image style transfer. 2) In our approach, all three window shapes shift based on the computation point, and their sizes dynamically adapt to variations in input sizes. Detailed differentiations from previous methods, such as CSWin, Longformer, and iLAT have been outlined in the Appendix.

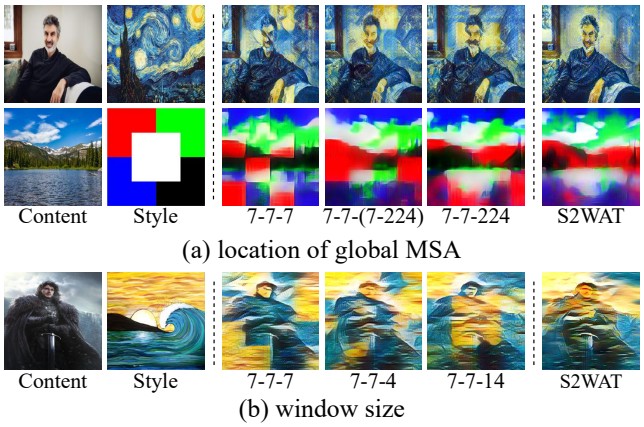


Figure 2: Results of the Swin-based encoder experiments. *7-7-7* means the Swin used has 3 stages (each stage with 2 layers) and 7 is the window size of each layer. *7-7-(7-224)* denotes the window size of the last layer in the last stage is 224 which represents global MSA. (a) Results of experiments inserting global MSA in certain layers. (b) Results of experiments changing the window size.

Pre-Analysis

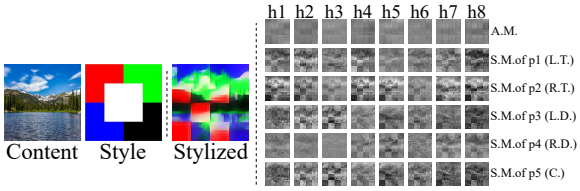
Our preliminary analysis aims to unveil the underlying causes of grid-like outcomes (locality problem) that arise when directly employing Swin for style transfer. Our hypothesis points towards the localized nature of window attention as the primary factor. To validate this hypothesis, we undertake experiments across four distinct facets as discussed in this Section. The details of the models tested in this part can be found in Appendix.

Global MSA for Locality Problem

The locality problem should be relieved or excluded when applying global MSA instead of window or shifted window attention, if the locality of window attention is the culprit. In the Swin-based encoder, we substitute the last one or two window attentions with global MSA, configuring the window size for target layers at 224 (matching input resolution). Fig. 2 (a) presents the experiment results, highlighting grid-like textures at a window size of 7 (column 3) and block-like formations when the last window attention is swapped with global MSA (column 4). While replacing the last two window attentions with global MSA effectively alleviates grid-like textures, complete exclusion remains a challenge. This series of experiments substantiates that the locality problem indeed stems from the characteristics of window attention.

(c) feature maps

Figure 3: The last feature maps from Swin-based encoder.



(d) attention maps

Figure 4: Attention maps (row 1) and similarity maps (rows 2-6) of the five points. Attention and similarity maps differ in shape. They are scaled for easy observation. And hi denotes i-th attention head. S/A.M. is short for "similarity/attention maps" and L/R/T/D/C for "left/right/top/down/center".

Influence of Window Size for Locality Problem

The window size in window attention, akin to the receptive field in CNN, delineates the computational scope. To examine the impact of window size, assuming the locality of window attention causes the locality problem, we investigate three scenarios: window sizes of 4, 7, and 14 for the last stage. The outcomes of these experiments are depicted in Fig. 2 (b). Notably, relatively small blocks emerge with a window size of 4 (column 4), while a shifted window's rough outline materializes with a window size of 14 (column 5). This series of experiments underscores the pivotal role of window size in the locality problem.

Locality Phenomenon in Feature Maps

In previous parts, we discuss the changes in external factors, and we will give a shot at internal factors in the following parts. Since the basis of the transformer-based transfer module is the similarity between content and style features, the content features should leave clues if the stylized images are grid-like. For this reason, we check out all the feature maps from the last layer of the encoder and list some of them (see Fig. 3), which are convincing evidence to prove that features from window attention have strongly localized.

Locality Phenomenon in Attention Maps

To highlight the adverse impact of content feature locality on stylization, we analyze attention maps from the first inter-attention (Cheng, Dong, and Lapata 2016) in the transfer module (see Fig. 4). Five points, representing corners (p1: top-left in red, p2: top-right in green, p3: bottom-left in blue, p4: bottom-right in black), and the central point (p5: white) are selected from style features to gauge their similarity with content features. These points, extracted from specific columns of attention maps and reshaped into squares, mirror content feature shapes. The similarity map of p1 reveals pronounced responses aligned with red blocks in the stylized image. Conversely, p2, p3, and p5 exhibit robust responses in areas devoid of red blocks. As for p4's similarity map, responses are distributed widely. These outcomes underline the propagation of window attention's locality from content features within the encoder to the attention maps of the transfer module. This influence significantly disrupts the stylization process, ultimately culminating in the locality problem. To address this issue, we present the SpW Attention and S2WAT solutions.

Method

Fig. 5 (c) presents the workflow of proposed S2WAT.

Strips Window Attention

As illustrated in Fig. 5 (b), SpW Attention comprises two distinct phases: a window attention phase and a fusion phase.

Window Attention. Assuming input features possess a shape of $C \times H \times W$ and n denotes the strip width, the first phase involves three distinct window attentions: a horizontal strip-like window attention with a window size of $n \times W$, a vertical strip-like window attention with a window size of $H \times n$, and a square window attention with a window size of $M \times M$ (where $M = 2n$). A single strip-like window attention captures local information along one axis while accounting for long-range dependencies along the other. In contrast, the square window attention focuses on the surrounding information. Combining the outputs of these window attentions results in outputs that consider both local information and long-range dependencies. Illustrated in Fig. 6, the merged approach gathers information from a broader range of targets, striking a favorable balance between computational complexity and the ability to sense global information.

In computing square window attention, we follow (Liu et al. 2021) to include relative position bias $B \in \mathbb{R}^{M^2 \times M^2}$ to each head in computing the attention map, as

$$\text{W-MSA}_{M\times M}(Q,K,V) = \text{Softmax}(\frac{QK^T}{\sqrt{d}} + B)V, \quad (1)$$

where $Q, K, V \in \mathbb{R}^{M^2 \times d}$ are the *query*, *key*, and *value* matrices; d is the dimension of *query*/*key*, M^2 is the number of patches in the window, and $\text{W-MSA}_{M\times M}$ denotes multi-head self-attention using window in shape of $M \times M$. We exclusively apply relative position bias to square window attention, as introducing it to strip-like window attention did not yield discernible enhancements.

Attn Merge. Following the completion of the window attention phase, a fusion module named "Attn Merge" is engaged to consolidate the outcomes with the inputs. Illustrated in Fig. 7, "Attn Merge" comprises three core steps: first, tensor stacking; second, similarity computation between the first tensor and the rest at every spatial location; third, weighted summation based on similarity. The compu-

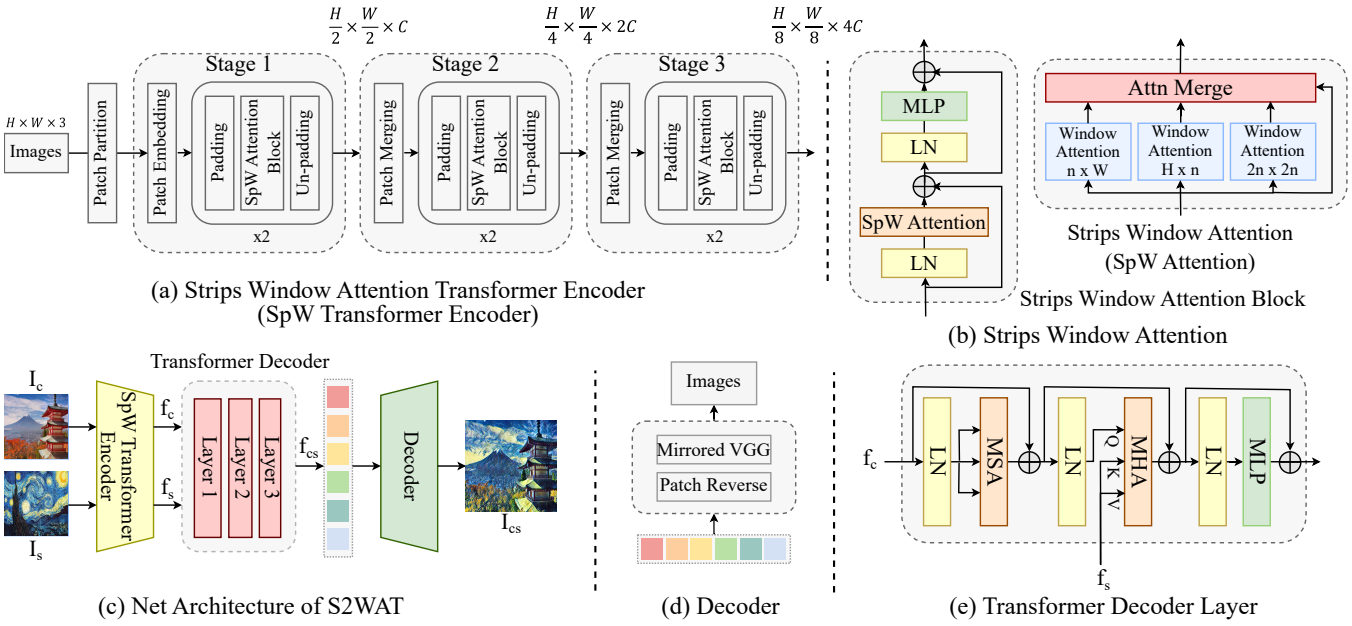


Figure 5: Overall pipeline of the proposed S2WAT. Given a content image I_c and a style image I_s, the encoder produces corresponding features f_c and f_s. These features undergo style transfer from f_s to f_c within the transfer module, yielding stylized features f_{cs}. Subsequently, stylized features are decoded in the decoder to generate the stylized image I_{cs}.

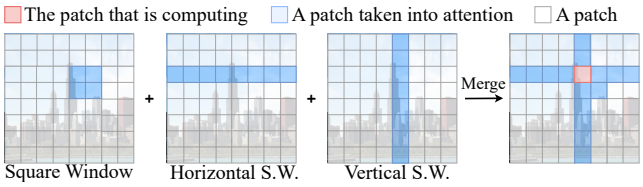


Figure 6: Receptive field of Strips Window Attention. Single strip-like window attention or square window attention can only glean information from limited targets in the image, while the merged one enlarges the receptive region to multiple directions. S.W. denotes "strip window".

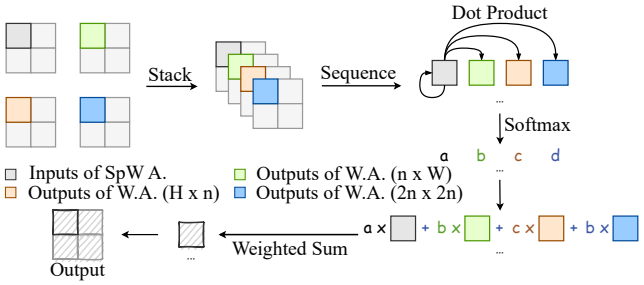


Figure 7: Workflow of "Attn Merge". W./A. denotes "Window/Attention".

tational efficiency of "Attn Merge" is noteworthy, as

$$\begin{aligned} Y &= \text{Stack}(x, a, b, c), \quad Y \in \mathbb{R}^{n\times 4\times d}, \\ x' &= \text{Unsqueeze}(x), \quad x \in \mathbb{R}^{n\times 1\times d}, \\ Z &= x' Y^T Y, \quad Z \in \mathbb{R}^{n\times 1\times d}, \\ z &= \text{Squeeze}(Z), \quad z \in \mathbb{R}^{n\times d}, \end{aligned} \tag{2}$$

where $x, a, b, c \in \mathbb{R}^{n\times d}$ are input tensors and z is the outputs; Stack denotes the operation to collect tensors in a new dimension and Unsqueeze / Squeeze represents the operation to add or subtract a dimension of tensor.

Strips Window Attention Block. We now provide an overview of the comprehensive workflow of the SpW Attention block. The structure of the SpW Attention block mirrors that of a standard transformer block, except for the substitution of MSA with a SpW Attention (SpW-MSA) module. As depicted in Fig. 5 (b), a SpW Attention block comprises a SpW-MSA module, succeeded by a two-layer MLP featuring GLUE as the non-linear activation in between. Preceding each SpW-MSA module and MLP, a LayerNorm (LN) operation is applied, and a residual connection is integrated after each module. The computation process of a SpW Attention block unfolds as follows:

$$\begin{aligned} \hat{z}^l_{n\times W} &= \text{W-MSA}_{n\times W}(\text{LN}(z^{l-1})), \\ \hat{z}^l_{H\times n} &= \text{W-MSA}_{H\times n}(\text{LN}(z^{l-1})), \\ \hat{z}^l_{2n\times 2n} &= \text{W-MSA}_{2n\times 2n}(\text{LN}(z^{l-1})), \\ \tilde{z}^l &= \mathcal{A}(\text{LN}(z^{l-1}), \hat{z}^l_{n\times W}, \hat{z}^l_{H\times n}, \hat{z}^l_{2n\times 2n}) + z^{l-1}, \\ z^l &= \text{MLP}(\text{LN}(\hat{z}^l)) + \tilde{z}^l, \end{aligned} \tag{3}$$

where "$\mathcal{A}$" means "Attn Merge", z^l , $\tilde{z}^l$ and $\hat{z}^l$ denote the outputs of MLP, "Attn Merge", and W-MSA for block l, respectively; W-MSA$_{n\times m}$ denotes multi-head self-attention using window in shape of $n \times m$. As shown in (3), the SpW Attention block primarily consists of two parts: SpW Atten-

tion (comprising W-MSA and "Attn Merge") and an MLP.

Computational Complexity. To make the cost of computation in SpW Attention clear, we compare the computational complexity of MSA, W-MSA, and the proposed SpW-MSA. Supposing the window size of W-MSA and the strip width of SpW-MSA are equal to M and C is the dimension of inputs, the computational complexity of a global MSA, a square window based one, and a Strips Window based one on an image of $h \times w$ patches are:

$$\Omega(\text{MSA}) = 2(wh)^2C + 4whC^2, \tag{4}$$

$$\Omega(\text{W - MSA}) = 2M^2whC + 4whC^2, \tag{5}$$

$$\Omega(\text{SpW - MSA}) = 2M(w^2h + wh^2 + 4Mwh)C + 12whC^2 + 8whC. \tag{6}$$

As shown in Eqs. (4)-(6), MSA is quadratic to the patch number hw, and W-MSA is linear when M is fixed. And the proposed SpW-MSA is something in the middle.

Overall Architecture

In contrast to StyTr2 (Deng et al. 2022), which employs separate encoders for different input domains, we adhere to the conventional encoder-transfer-decoder design of UST. This architecture encodes content and style images using a single encoder. An overview is depicted in Fig. 5.

Encoder. Like Swin, S2WAT's encoder initially divides content and style images into non-overlapping patches using a patch partition module. These patches serve as "tokens" in transformers. We configure the patch size as 2×2, resulting in patch dimensions of $2 \times 2 \times 3 = 12$. Subsequently, a linear embedding layer transforms the patches into a user-defined dimension (C).

After embedding, the patches proceed through a series of consecutive SpW Attention blocks, nestled between padding and un-padding operations. Patches are padded to achieve divisibility by twice the strip width and cropped (un-padded) after SpW Attention blocks, preserving the patch count. Notably, patch padding employs reflection to mitigate potential light-edge artifacts that can arise when using constant 0 padding. These SpW Attention blocks uphold the patch count ($\frac{H}{2} \times \frac{W}{2}$) and, in conjunction with the patch embedding layer and padding/un-padding operations, constitute "Stage 1".

To achieve multi-scale features, gradual reduction of the patch count is necessary as the network deepens. Swin introduces a patch merging layer as a down-sample module, extracting elements with a two-step interval along the horizontal and vertical axes. By concatenating 2×2 groups of these features in the channel dimension and reducing channels from $4C$ to $2C$ through linear projection, a 2x downsampling result is obtained. Subsequent application of SpW Attention blocks, flanked by padding and un-padding operations, transforms the features while preserving a resolution of $\frac{H}{4} \times \frac{W}{4}$. This combined process is designated as "Stage 2". This sequence is reiterated for "Stage 3", yielding an output resolution of $\frac{H}{8} \times \frac{W}{8}$. Consequently, the encoder's hierarchical features in S2WAT can readily be employed with techniques like FPN or U-Net.

Transfer Module. A multi-layer transformer decoder replaces the transfer module, similar to StyTr2 (Deng et al. 2022). In our implementation, we maintain a close resemblance to the original transformer decoder (Vaswani et al. 2017), with two key distinctions from StyTr2: a) The initial attention module of each transformer decoder layer is MSA, whereas StyTr2 employs MHA (multi-head attention); b) LayerNorm precedes the attention module and MLP, rather than following them. The structure is presented in Fig. 5 (e) and more details can be found in codes.

Decoder. In line with prior research (Huang and Belongie 2017; Park and Lee 2019; Deng et al. 2021), we utilize a mirrored VGG for decoding stylized features. Detailed implements are available in codes.

Network Optimization

Similar to (Huang and Belongie 2017), we formulate two distinct perceptual losses for gauging the content dissimilarity between stylized images I_{cs} and content images I_c, along with the style dissimilarity between stylized images I_{cs} and style images I_s. The content perceptual loss is defined as:

$$\mathcal{L}_{content} = \sum_{l \in \mathcal{C}} \|\overline{\phi_l(I_{cs})} - \overline{\phi_l(I_c)}\|_2, \tag{7}$$

where the overline denotes mean-variance channel-wise normalization; $\phi_l(\cdot)$ represents extracting features of layer l from a pre-trained VGG19 model; $\mathcal{C}$ is a set consisting of $relu4\_1$ and $relu5\_1$ in the VGG19. The style perceptual loss is defined as:

$$\begin{aligned}\mathcal{L}_{style} = &\sum_{l \in \mathcal{S}} \|\mu(\phi_l(I_{cs})) - \mu(\phi_l(I_s))\|_2 \\ &+ \|\sigma(\phi_l(I_{cs})) - \sigma(\phi_l(I_s))\|_2,\end{aligned} \tag{8}$$

where $\mu(\cdot)$ and $\sigma(\cdot)$ denote mean and variance of features, respectively; and $\mathcal{S}$ is a set consisting of $relu2\_1$, $relu3\_1$, $relu4\_1$ and $relu5\_1$ in the VGG19.

We also adopt identity losses (Park and Lee 2019) to further maintain the structure of the content image and the style characteristics of the style image. The two different identity losses are defined as:

$$\mathcal{L}_{id1} = \|I_{cc} - I_c\|_2 + \|I_{ss} - I_s\|_2, \tag{9}$$

$$\mathcal{L}_{id2} = \textstyle\sum_{l \in \mathcal{N}} \|\phi_l(I_{cc}) - \phi_l(I_c)\|_2 + \|\phi_l(I_{ss}) - \phi_l(I_s)\|_2, \tag{10}$$

where I_{cc} (or I_{ss}) denotes the output image stylized from two same content (or style) images and $\mathcal{N}$ is a set consisting of $relu2\_1$, $relu3\_1$, $relu4\_1$ and $relu5\_1$ in the VGG19. Finally, our network is trained by minimizing the loss function defined as:

$$\mathcal{L}_{total} = \lambda_c\mathcal{L}_{content} + \lambda_s\mathcal{L}_{style} + \lambda_{id1}\mathcal{L}_{id1} + \lambda_{id2}\mathcal{L}_{id2}\ , \tag{11}$$

where λ_c, λ_s, λ_{id1}, and λ_{id2} are the weights of different losses. We set the weights to 2, 3, 50, and 1, relieving the impact of magnitude differences.

Experiments

MS-COCO (Lin et al. 2014) and WikiArt (Phillips and Mackintosh 2011) are used as the content dataset and the style dataset respectively. Other implementation details are available in Appendix and codes.

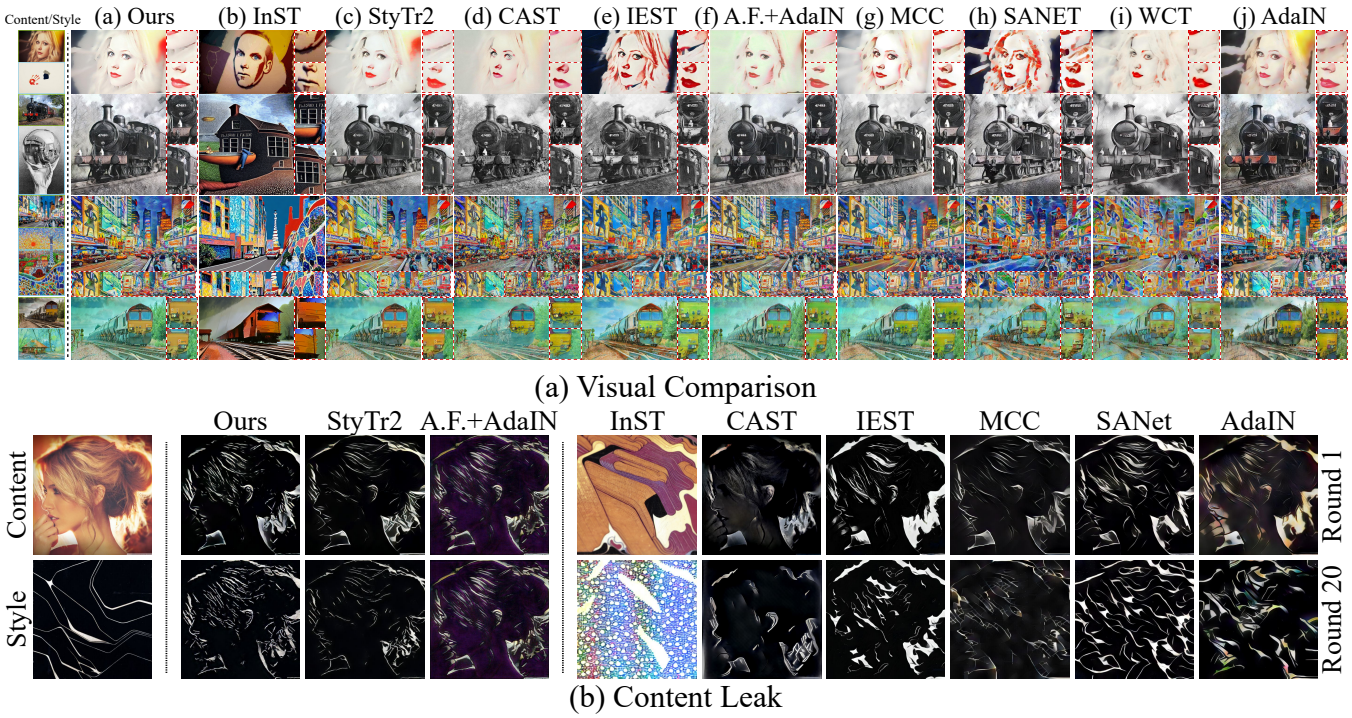


Figure 8: Visual comparison of the results from SOTA methods and visualization of content leak. A.F. denotes "ArtFlow".

Style Transfer Results

In this Section, we compare the results between the proposed S2WAT and previous SOTAs, including AdaIN (Huang and Belongie 2017), WCT (Li et al. 2017), SANet (Park and Lee 2019), MCC (Deng et al. 2021), ArtFlow (An et al. 2021), IEST (Chen et al. 2021), CAST (Zhang et al. 2022), StyTr2 (Deng et al. 2022) and InST (Zhang et al. 2023).

Qualitative Comparison. In Fig. 8 (a), we present visual outcomes of the compared algorithms. AdaIN, relying on mean and variance alignment, fails to capture intricate style patterns. While WCT achieves multi-level stylization, it compromises content details. SANet, leveraging attention mechanisms, enhances style capture but may sacrifice content details. MCC, lacking non-linear operations, faces overflow issues. Flow-based ArtFlow produces content-unbiased outcomes but may exhibit undesired patterns at borders. CAST retains content structure through contrastive methods but may compromise style. InST's diffusion models yield creative results but occasionally sacrifice consistency. StyTr2 and proposed S2WAT strike a superior balance, with S2WAT excelling in preserving content details (e.g., numbers on the train, the woman's glossy lips, and letters on billboards), as highlighted in dashed boxes in Fig. 8 (a). Additional results are available in the Appendix.

Quantitative Comparison. In this section, we follow a methodology akin to (Huang and Belongie 2017; An et al. 2021; Deng et al. 2022) utilizing losses as indirect metrics. Style, content, and identity losses serve as metrics, evaluating style quality, content quality, and input information retention, respectively. Additionally, inspired by (An et al. 2021), the Structural Similarity Index (SSIM) is included to gauge structure preservation. As shown in Table 1, S2WAT achieves the lowest content and identity losses, while SANet exhibits the lowest style loss. StyTr2 and S2WAT show comparable loss performance, emphasizing style and content, respectively. Due to its content-unbiased nature, ArtFlow registers identity losses of 0, signaling an unbiased approach. While ArtFlow is unbiased, S2WAT outperforms it in style and SSIM. S2WAT attains the highest SSIM, indicating superior content structure retention. It excels in preserving both content input structures and artistic style characteristics simultaneously.

Content Leak

Content leak problem occur when applying the same style image to a content image repeatedly, especially if the model struggles to preserve content details impeccably. Following (An et al. 2021; Deng et al. 2022), We investigate content leakage in the stylization process, focusing on S2WAT and comparing it to ArtFlow, StyTr2, CNN-based, and SDM-based methods. Our experiments, detailed in Fig. 8 (b), reveal S2WAT and StyTr2, both transformer-based, exhibit minimal content detail loss over 20 iterations, surpassing CNN and SDM methods known for noticeable blurriness. While CAST alleviates content leak partially, the stylized effect remains suboptimal. In summary, S2WAT effectively mitigates the content leak issue.

InST occasionally underperforms, especially when content and style input styles differ significantly, potentially due to overfitting in the Textual Inversion module during single-image training. More details are available in the Appendix.

Method	Ours	InST	StyTr2	CAST	IEST	ArtFlow-AdaIN	ArtFlow-WCT	MCC	SANet	WCT	AdaIN
Content Loss ↓	**1.66**	3.73	1.83	2.07	1.81	1.93	1.73	1.92	2.16	2.56	1.71
Style Loss ↓	1.74	29.98	1.52	4.33	2.72	1.90	1.89	1.70	**1.11**	2.23	3.50
Identity Loss 1 ↓	**0.16**	0.71	0.26	1.94	0.91	0.00	0.00	1.07	0.81	3.01	2.54
Identity Loss 2 ↓	**1.38**	134.23	3.10	18.72	7.16	0.00	0.00	7.72	6.03	21.88	17.97
SSIM ↑	**0.651**	0.401	0.605	0.619	0.551	0.578	0.612	0.578	0.448	0.364	0.539

Table 1: Quantitative evaluation results of different style transfer methods. The losses above are average values from 400 random samples, while SSIMs are computed average from 100 pieces. For a fair comparison, we take $relu1\_1$ into consideration in computing style loss and identity loss 2 while not in the training of S2WAT. The optimal results are highlighted in bold, the second-best results are underlined, and instances with a value of 0 are derived from unbiased methods.

Figure 9: Visual comparisons when utilizing different fusion strategies for attention outputs from multiple windows.

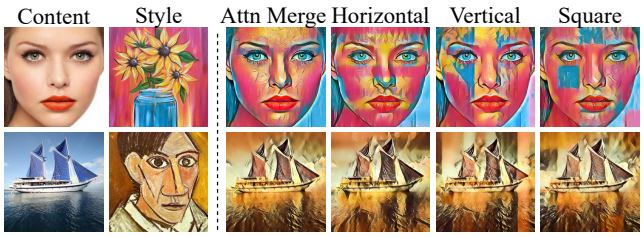


Figure 10: Visual comparisons for the ablation study when employing different window attention mechanisms.

Method	Ours	StyTr2	ArtFlow	MCC	SANet
Percent(%)	**25.4**	23.6	13.3	19.4	18.3

Table 2: Percentage of votes in the user study.

Ablation Study

Attn Merge. In order to showcase the effectiveness and superiority of "Attn Merge", we undertake experiments where "Attn Merge" is replaced by fusion strategies such as the concatenation operation (as employed by CSWin) or the sum operation. The outcomes are depicted in Fig. 9. Stylized images generated using the sum operation are extensively corrupted, indicating a failure in model optimization. On the other hand, outputs obtained through concatenation relinquish a substantial portion of information from input images, particularly the style images. An intuitive rationale for this phenomenon lies in the optimization challenges posed by straightforward fusion operations. Comprehensive explanations are available in the Appendix. The proposed "Attn Merge", however, facilitates smooth information transmission, allowing the model to undergo normal training.

Strips Window. To verify the demand to fuse outputs from window attention of various sizes, we carry out experiments employing window attention with distinct window sizes independently. As illustrated in Fig. 10, the utilization of horizontal or vertical strip-like windows in isolation yields corresponding patterns. Applying square windows alone results in grid-like patterns. However, the incorporation of "Attn Merge" to fuse outcomes leads to pleasing stylized images, surpassing the results obtained solely from window attention. Further details regarding the ablation study for Swin and Swin with Mix-FFN can be found in the Appendix.

User Study

In comparing virtual stylization effects between S2WAT and the aforementioned SOTAs like StyTr2, ArtFlow, MCC, and SANet, user studies were conducted. Using a widely-employed online questionnaire platform, we created a dataset comprising 528 stylized images from 24 content images and 22 style images. Participants, briefed on image style transfer and provided with evaluation criteria, assessed 31 randomly selected content and style combinations. Criteria emphasized preserving content details and embodying artistic attributes. With 3002 valid votes from 72 participants representing diverse backgrounds, including high school students and professionals in computer science, art, and photography, our method achieved a marginal victory in the user study, as reflected in Table 2. Additional details including an example questionnaire page can be found in the Appendix.

Conclusion

In this paper, we introduce S2WAT, a pioneering image style transfer framework founded upon a hierarchical vision transformer architecture. S2WAT's prowess lies in its capacity to simultaneously capture local and global information through SpW Attention. The SpW Attention mechanism, featuring diverse window attention shapes, ensures an optimal equilibrium between short- and long-range dependency modeling, further enhanced by our proprietary "Attn Merge". This adaptive merging technique efficiently gauges the significance of various window attentions based on target similarity. Furthermore, S2WAT mitigates the content leak predicament, yielding stylized images endowed with vibrant style attributes and intricate content intricacies.

Acknowledgements

This work was supported by the National Key R&D Program of China (2021YFB3100700), the National Natural Science Foundation of China (62076125, U20B2049, U22B2029, 62272228), and Shenzhen Science and Technology Program (JCYJ20210324134408023, JCYJ20210324134810028).

References

An, J.; Huang, S.; Song, Y.; Dou, D.; Liu, W.; and Luo, J. 2021. Artflow: Unbiased image style transfer via reversible neural flows. In *CVPR*.

Bai, Y.; Liu, J.; Dong, C.; and Yuan, C. 2023. ITstyler: Image-optimized Text-based Style Transfer. *arXiv*.

Chen, D.; Yuan, L.; Liao, J.; Yu, N.; and Hua, G. 2017. Stylebank: An explicit representation for neural image style transfer. In *CVPR*.

Chen, H.; Wang, Z.; Zhang, H.; Zuo, Z.; Li, A.; Xing, W.; Lu, D.; et al. 2021. Artistic style transfer with internal-external learning and contrastive learning. In *NeurIPS*.

Cheng, J.; Dong, L.; and Lapata, M. 2016. Long short-term memory-networks for machine reading. In *EMNLP*.

Deng, Y.; Tang, F.; Dong, W.; Huang, H.; Ma, C.; and Xu, C. 2021. Arbitrary video style transfer via multi-channel correlation. In *AAAI*.

Deng, Y.; Tang, F.; Dong, W.; Ma, C.; Pan, X.; Wang, L.; and Xu, C. 2022. StyTr2: Image Style Transfer with Transformers. In *CVPR*.

Deng, Y.; Tang, F.; Dong, W.; Sun, W.; Huang, F.; and Xu, C. 2020. Arbitrary style transfer via multi-adaptation network. In *ACM MM*.

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*.

Fan, H.; Xiong, B.; Mangalam, K.; Li, Y.; Yan, Z.; Malik, J.; and Feichtenhofer, C. 2021. Multiscale vision transformers. In *ICCV*.

Gatys, L.; Ecker, A. S.; and Bethge, M. 2015. Texture synthesis using convolutional neural networks. In *NeurIPS*.

Gatys, L. A.; Ecker, A. S.; and Bethge, M. 2016. A neural algorithm of artistic style. In *Vision Sciences Society*.

Graham, B.; El-Nouby, A.; Touvron, H.; Stock, P.; Joulin, A.; Jégou, H.; and Douze, M. 2021. Levit: a vision transformer in convnet's clothing for faster inference. In *ICCV*.

Hong, K.; Jeon, S.; Lee, J.; Ahn, N.; Kim, K.; Lee, P.; Kim, D.; Uh, Y.; and Byun, H. 2023. AesPA-Net: Aesthetic Pattern-Aware Style Transfer Networks. In *ICCV*.

Huang, X.; and Belongie, S. 2017. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*.

Johnson, J.; Alahi, A.; and Fei-Fei, L. 2016. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*.

Kong, X.; Deng, Y.; Tang, F.; Dong, W.; Ma, C.; Chen, Y.; He, Z.; and Xu, C. 2023. Exploring the Temporal Consistency of Arbitrary Style Transfer: A Channelwise Perspective. *IEEE Transactions on Neural Networks and Learning Systems*.

Li, G.; Cheng, B.; Cheng, L.; Xu, C.; Sun, X.; Ren, P.; Yang, Y.; and Chen, Q. 2022. Arbitrary Style Transfer with Semantic Content Enhancement. In *The 18th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and its Applications in Industry*.

Li, Y.; Fang, C.; Yang, J.; Wang, Z.; Lu, X.; and Yang, M.-H. 2017. Universal style transfer via feature transforms. In *NeurIPS*.

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 *ECCV*.

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*.

Liu, Z.-S.; Wang, L.-W.; Siu, W.-C.; and Kalogeiton, V. 2022. Name your style: An arbitrary artist-aware image style transfer. *arXiv*.

Ma, Y.; Zhao, C.; Li, X.; and Basu, A. 2023. RAST: Restorable Arbitrary Style Transfer via Multi-Restoration. In *WACV*.

Park, D. Y.; and Lee, K. H. 2019. Arbitrary style transfer with style-attentional networks. In *CVPR*.

Phillips, F.; and Mackintosh, B. 2011. Wiki Art Gallery, Inc.: A case for critical thinking. *Issues in Accounting Education*.

Ulyanov, D.; Vedaldi, A.; and Lempitsky, V. 2016. Instance normalization: The missing ingredient for fast stylization. *arXiv*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *NeurIPS*.

Wang, W.; Xie, E.; Li, X.; Fan, D.-P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; and Shao, L. 2021. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *ICCV*.

Wu, H.; Xiao, B.; Codella, N.; Liu, M.; Dai, X.; Yuan, L.; and Zhang, L. 2021a. Cvt: Introducing convolutions to vision transformers. In *ICCV*.

Wu, X.; Hu, Z.; Sheng, L.; and Xu, D. 2021b. Styleformer: Real-time arbitrary style transfer via parametric style composition. In *ICCV*.

Xie, E.; Wang, W.; Yu, Z.; Anandkumar, A.; Alvarez, J. M.; and Luo, P. 2021. SegFormer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, 34: 12077–12090.

Yao, Y.; Ren, J.; Xie, X.; Liu, W.; Liu, Y.-J.; and Wang, J. 2019. Attention-aware multi-stroke style transfer. In *CVPR*.

Zhang, Y.; Huang, N.; Tang, F.; Huang, H.; Ma, C.; Dong, W.; and Xu, C. 2023. Inversion-based style transfer with diffusion models. In *CVPR*, 10146–10156.

Zhang, Y.; Tang, F.; Dong, W.; Huang, H.; Ma, C.; Lee, T.-Y.; and Xu, C. 2022. Domain Enhanced Arbitrary Image Style Transfer via Contrastive Learning. In *SIGGRAPH*.

Zhu, M.; He, X.; Wang, N.; Wang, X.; and Gao, X. 2023. All-to-key Attention for Arbitrary Style Transfer. In *ICCV*.