GSCNet: A Transformer-Based Granular Style Control Network for Artifact-Free Image Style Transfer

Zheng Gao[a,*], Zhicheng Bao[a], Haoran Fan[a], Xiaoyu Li[a], Qipei Nong[b] and Jiaojiao Jiang[a]

[a]School of Computer Science and Engineering, University of New South Wales, Sydney, NSW, 2052, Australia
[b]School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan, Liaoning, China

ARTICLE INFO

Keywords:
Image style transfer
Universal style transfer
Artifact suppression
Feature normalization
Multi-scale representation

ABSTRACT

Transformer-based image style transfer models capture long-range dependencies but often produce visual artifacts due to insufficient granularity control. We propose GSCNet, a granular style control network that suppresses artifacts from three complementary perspectives. For structural artifacts from limited receptive fields in window-based attention, we design a Dual-grained Attention Module (DAM) that uses global context to guide local feature interactions. For pixel-level artifacts such as uneven stylization from abrupt value shifts, a Feature Stabilization Module (FSM) normalizes outlier activations using neighborhood information. For fragmented detail artifacts caused by overfitting, a Detail Engraver Adapter with LoRA enhances fine-grained perception while maintaining coherent style fusion. Extensive experiments demonstrate that GSCNet leads on four of the six evaluation metrics over the strongest baseline (S2WAT), with up to 41% lower feature-level identity loss (Identity Loss 2); as a rough aggregate, this corresponds to a mean improvement of 12% across the six metrics. The gains are concentrated in content preservation, identity consistency, structural similarity, and reconstruction fidelity, while the higher style loss reflects a deliberate content–style trade-off.

1. Introduction

Image style transfer aims to apply the artistic style of one image while preserving the semantic content of another. In this field, approaches have progressed from traditional texture transfer [11], to iterative and feed-forward CNN methods [14, 16, 21, 5], and further to universal style transfer (UST) techniques [46, 25, 32], which significantly improved generalization to arbitrary new styles. Nevertheless, CNN-based approaches often struggle to capture the global arrangement of style elements due to their limited receptive fields. To overcome this limitation, recent studies have introduced Transformers, which excel at modeling long-range dependencies and thus provide stronger global information representation. For instance, StyTr2 [10] has exploited this strength to achieve state-of-the-art performance in style transfer tasks. However, despite their global modeling capabilities, such attention-based models often struggle with fine-grained local details.

To address this challenge, window-based attention mechanisms, such as Swin Transformer [30], aim to improve local detail modeling but often suffer from limited context exchange due to strict spatial confinement. As shown in Figure 1, our experiments demonstrate that using Swin Transformer results in noticeable window-shaped artifacts in regions such as rooftops and bridge tips.

To mitigate these window-shaped artifacts, S2WAT [44] attempts to overcome window artifacts via stripes window attention. However, its fixed geometric constraints and fusion strategy still lead to inefficient context exchange and banded artifacts, as observed in the sky region of the first row in Figure 1. These issues highlight the model's lack of control over global structural granularity. To achieve this control, it is necessary to enrich stripes window attention with global context, thereby overcoming the constraints of a fixed field of view. Based on this idea, we propose the Dual-grained Attention Module in our model.

In addition, we observe pixel-level artifacts in dark-themed images, where excessive discrepancies between style and content cause uneven stylization, indicating insufficient control over local stability granularity, such as the bridge tip in the second row of Figure 1. To resolve this, inspired by the normalization mechanism in neuroscience [4], we adopt a neighborhood-based normalization strategy that balances outlier activations using local context. We implement

*Corresponding author
zheng.gao1@unsw.edu.au (Z. Gao); zhicheng.bao@unsw.edu.au (Z. Bao); fanhaoran68@gmail.com (H. Fan); xiaoyu.li2@student.unsw.edu.au (X. Li); 120213502059@stu.ustl.edu.cn (Q. Nong); jiaojiao.jiang@unsw.edu.au (J. Jiang)

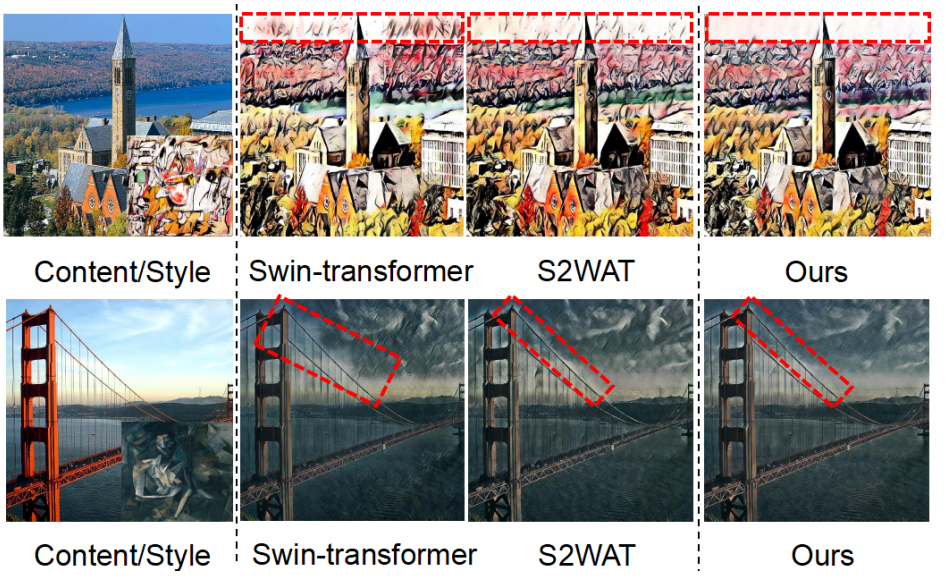

Figure 1: Common issues in Transformer-based style transfer: *(Top)* striped artifacts in flat regions (e.g., sky); *(Bottom)* uneven stylization and brightness across spatial areas (e.g., bridge).

this idea in a Feature Stabilization Module. As shown in Figure 1, our model enables smoother and more uniform stylization at the bridge tip.

In addition, we identify a distinct type of artifact caused by inconsistent style fusion, which manifests as fragmented and incoherent details. This issue arises when the model overly emphasizes global style alignment while overlooking fine-grained control over image details. To address this, we leverage LoRA's orthogonality [41] to guide complementary style capture and propose a Detail Engraver Adapter, a multi-scale module for preserving fine details.

Building upon the above ideas, we propose a Transformer-Based **G**ranular **S**tyle **C**ontrol **Net**work for Artifact-Free Image Style Transfer (**GSCNet**), which addresses artifacts by integrating structural, pixel-level, and detail-aware mechanisms into a cohesive design. Throughout this paper, we use *granularity* to denote the spatial scale at which stylization artifacts arise and are controlled, ranging from coarse global structure (structural granularity), to per-pixel stability (pixel granularity), and to fine local texture (detail granularity); our three modules (DAM, FSM, and DEA) are designed to operate at these three levels, respectively.

Our main contributions are:

1. To reduce artifacts caused by limited receptive fields, we design a Dual-grained Attention Module (DAM), which operates at the structural granularity level by using global context to refine local attention, thereby breaking isolated computation patterns and effectively suppressing structural granular artifacts.

2. To reduce pixel-level artifacts causing uneven stylization, we introduce a Feature Stabilization Module (FSM) that performs neighborhood-aware divisive normalization to clamp outlier activations, yielding uniform illumination without eliminating detail.

3. To suppress detail granular artifacts induced by forced style–content matching, we propose the Detail Engraver Adapter (DEA) to capture complementary fine-grained cues and prevent fragmented or incoherent textures.

4. Extensive experiments show that GSCNet leads on four of the six metrics, with up to 41% lower feature-level identity loss (Identity Loss 2) than the strongest baseline (S2WAT); as a rough aggregate, this corresponds to a mean improvement of 12% across the six metrics.

2. Related Work

Recent advances in style transfer have evolved from traditional patch-based methods such as Image Quilting [11] to more flexible Universal Style Transfer (UST) techniques [46, 25, 32]. Early neural approaches [14, 15, 16, 21, 5]

introduce convolutional neural networks (CNNs) into stylization tasks, significantly improving both visual quality and computational efficiency.

Despite their success, CNN-based methods are limited by local receptive fields, making it challenging to preserve both global structure and fine-grained texture. To address this, many approaches adopt statistical alignment. For example, AdaIN [20] aligns feature distributions via channel-wise mean and variance, offering efficiency but struggling with complex textures. SANet [35] introduces patch-wise attention to better capture local style patterns, though often at the cost of content preservation and global coherence.

The limitations of CNNs in modeling long-range dependencies have motivated the exploration of Transformer-based architectures for style transfer. For example, StyTr2 [10] shows that self-attention can significantly improve stylization quality, but its non-hierarchical design limits multi-scale feature modeling. Hierarchical Transformers such as Swin Transformer [30] introduce shifted-window attention to balance local and global contexts, yet their direct application often results in structural artifacts such as window-shaped distortions.

To address these artifacts, S2WAT [44] proposes striped-window attention and fusion strategies. While this reduces window-related artifacts, residual distortions, such as stripe-like patterns, still persist. These highlight a critical challenge in Transformer-based style transfer, namely, how to effectively model global dependencies without introducing artifacts. Our work directly targets this issue by proposing a novel model that suppresses such artifacts from three different granularity levels.

Beyond the window-attention methods discussed above, two further paradigms have recently been applied to style-related image synthesis. Diffusion-based methods such as InST [45] produce expressive stylizations but rely on iterative multi-step sampling and tend to alter the input content, whereas GSCNet performs arbitrary style transfer in a single feed-forward pass. A separate line of work, mainly in underwater image enhancement, casts the task as unsupervised content–style disentanglement with cycle-consistent translation [47, 48, 29]; although we likewise decouple content and style, GSCNet targets general artistic stylization within a single hierarchical Transformer rather than GAN-based restoration between two fixed domains.

3. Preliminaries

In this section, we introduce the background of the image style transfer task and the universal transfer framework.

3.1. Image Style Transfer

Given a content image I_c and a style image I_s, image style transfer aims to synthesize an output I_{cs} that preserves the content of I_c while adopting the style of I_s. This is typically achieved by minimizing a weighted sum of a content loss $\mathcal{L}_{\text{content}}$ and a style loss $\mathcal{L}_{\text{style}}$ defined as follows.

Let $F^l(I) \in \mathbb{R}^{H_l W_l \times C_l}$ be the feature of an image I extracted from l-th layer of a pretrained VGG-19 [42], where H_l, W_l are the spatial dimensions and C_l is the channel length. The content loss $\mathcal{L}_{\text{content}}$ is defined as

$$\mathcal{L}_{\text{content}}(I_{cs}, I_c) = \sum_{l \in \mathcal{C}} \| F^l(I_{cs}) - F^l(I_c) \|_F^2,$$

where $\mathcal{C}$ denotes a selected set of layers. In this work, we use the relu_{4_1} and relu_{5_1} layers in VGG-19. The rationale for matching features at these deeper layers is that they capture higher-level, object-centric semantics of images [3, 1].

Unlike content, which is captured by object-level semantics, style is characterized by statistical dependencies across feature channels. It can be modeled using maximum mean discrepancy (Gram matrix matching) [27, 15] or Wasserstein distance [34]. In this work, we adopt moment matching by aligning first- and second-order statistics [20, 22].

$$\begin{aligned}\mathcal{L}_{\text{style}}(I_{cs}, I_s) = &\sum_{l \in \mathcal{S}} \| \text{Mean}(F^l(I_{cs})) - \text{Mean}(F^l(I_s)) \|_2^2 \\ &+ \sum_{l \in \mathcal{S}} \| \text{Var}(F^l(I_{cs})) - \text{Var}(F^l(I_s)) \|_2^2,\end{aligned}$$

where Mean and Var are channel-wise mean and variance of features, and $\mathcal{S}$ is a set of relu_{2_1}, relu_{3_1}, relu_{4_1} and relu_{5_1} in the VGG-19 in our implementation.

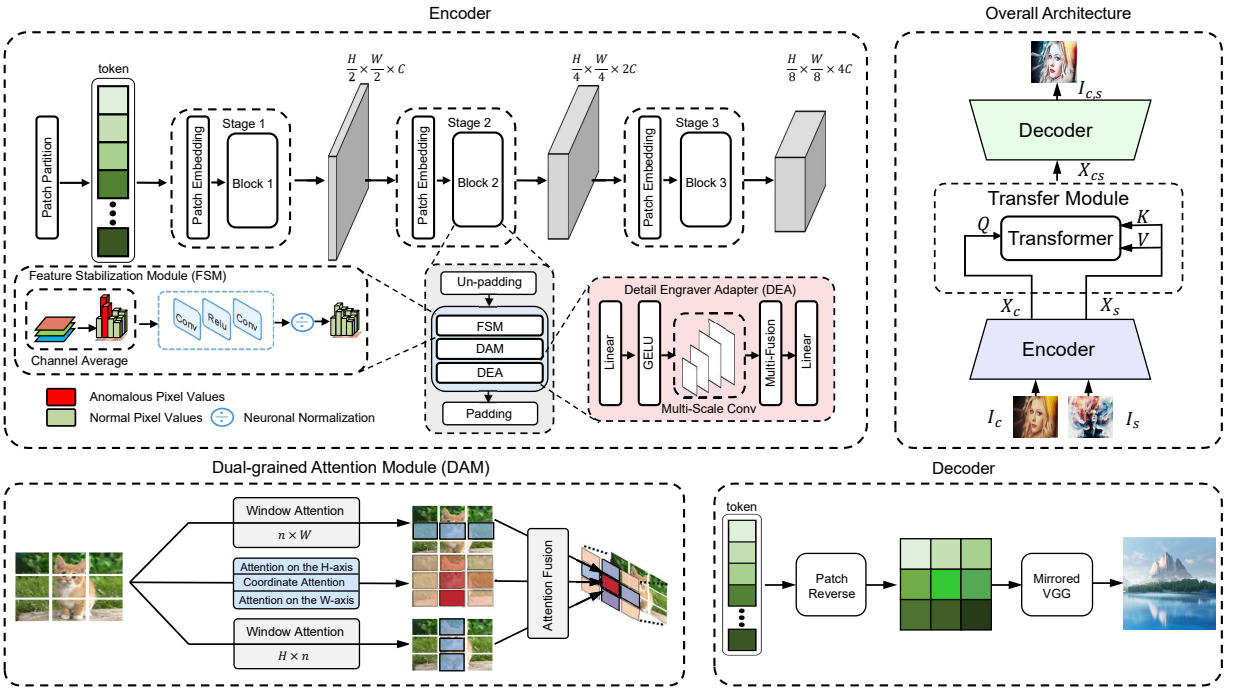

Figure 2: Overview Architecture of **GSCNet**. A hierarchical encoder encodes the content image I_c and the style image I_s into multi-scale features. Each encoder stage contains three modules: a *Feature Stabilization Module* (FSM) that performs neighborhood-aware normalization to suppress pixel-level artifacts; a *Dual-grained Attention Module* (DAM) that fuses stripes window attention with Coordinate Attention to couple local textures with global structure; and a *Detail Engraver Adapter* (DEA), for fine-grained detail enhancement at low overhead. Then, a Transformer-based transfer module takes X_c as queries and X_s as keys/values to produce synthesized features X_{cs}, which are upsampled by a mirrored-VGG decoder to the final stylized image I_{cs}.

3.2. Universal Style Transfer Framework

Our model builds on the Universal Style Transfer (UST) framework [35], which we briefly introduce here. UST is an encoder–transfer–decoder framework that performs arbitrary style transfer by aligning content and style in a shared deep feature space. A fixed convolutional network (e.g., VGG-19) defines this space by mapping images into it and a decoder reconstructs images from it. In this representation, content is largely preserved by the spatial arrangement of features, while style is captured by low-order statistics of those features. The transfer module modifies the encoded content so that its statistics match those of the style, typically by aligning channel-wise means and variances. The transformed representation is then decoded to produce the stylized image. Because the transfer depends only on statistics estimated from the style features, UST supports arbitrary styles at test time without per-style training by applying the procedure across multiple layers in a coarse-to-fine manner, which further improves perceptual fidelity while preserving structure. This process was formalized using the language of optimal transport by [34], and we refer the interested readers to Appendix B.

4. Method

In this section, we first present our network architecture, and then provide detailed explanations of the three core modules along with the associated training losses.

4.1. Overall Architecture

Figure 2 illustrates the overall Encoder-Transfer-Decoder architecture of GSCNet, which we describe in detail below.

Encoder. The encoder takes the content image I_c and the style image I_s as inputs and outputs the corresponding content and style features X_c and X_s. Inspired by Swin Transformer [30], our encoder first divides the content image and style image into non-overlapping patches via the Patch Partition module. These patches are then processed through three consecutive stages with identical structures, where the spatial resolution is progressively reduced. This hierarchical design enables the encoder to capture fine-grained details at early high-resolution stages while gradually focusing on global structural information at later low-resolution stages.

Each stage begins with a Patch Embedding layer that projects the input patches into the required dimension, followed by three customized modules. **(i) Feature Stabilization Module (FSM)** mitigates inconsistencies in style transfer (e.g., brightness imbalance), ensuring stable inputs for the next module. **(ii) Dual-grained Attention Module (DAM)** integrates fine-grained local textures and coarse-grained global structures to enhance stylistic representation and content alignment. **(iii) Detail Engraver Adapter (DEA)** refines multi-granularity texture details that may be overlooked during the previous attention module. These modules produce a compact multi-scale feature representation for subsequent decoding.

Transfer Module. The transfer module takes content and style features X_c and X_s as inputs and outputs the synthesized feature X_{cs}. We adopt a multi-layer Transformer decoder for this module, which takes $Q = X_c W_Q$ as queries, $K = X_s W_K$ as keys, and $V = X_s W_V$ as values, where W_Q, W_K, W_V are trainable weights. This setup lets each content query attend over the global style space and aggregate its most pertinent style patterns, resulting in a fusion map that couples structural fidelity with rich stylistic detail.

Decoder. The decoder takes the synthesized feature X_{cs} as input and outputs the synthesized image I_{cs}. Following established UST pipelines [20, 35, 9], we adopt a mirrored VGG-based decoder to progressively restore the spatial resolution of the previous fusion map. This process involves a series of upsampling and convolutional layers, ultimately reconstructing the stylized image at the original resolution.

The implementation details are available in Appendix D.

4.2. Feature Stabilization Module (FSM)

A Feature Stabilization Module (FSM) is introduced to normalize outlier activations based on neighborhood information, ensuring smooth and balanced style coverage. In practice, style transfer with certain styles (e.g., dark-themed images) often leads to uneven stylization, manifested as pixel-granular artifacts where individual pixels deviate from their neighbors. These deviations resemble neuronal activation, which in biology is normalized by the aggregated activity of surrounding neurons to prevent response saturation [4]. Inspired by this mechanism, our FSM exploits neuronal normalization to achieve stable style transfer.

Let $X \in \mathbb{R}^{N \times C}$ denote the input of FSM, either content image feature or style image feature, where $N = HW$ is the spatial dimension and C is the channel length. The FSM learns edges and textures by first computing the channel mean and processing it through a convolutional-ReLU-convolutional neural network Φ. This yields a context-aware signal Z, defined as $Z_i = \Phi(\sum_{c=1}^{C} X_{i,c}/C)$, which represents neighborhood strength. Neuronal normalization is then applied via a division-based modulation operation, computed as:

$$Y_{i,c} = \frac{X_{i,c}}{\text{Sigmoid}(Z_i) \cdot \alpha_c + \epsilon},$$

where $\alpha = (\alpha_c)_{c=1}^{C}$ sets a learnable operating range, and ϵ ensures numerical stability. In this way, through neighborhood information, abnormal pixels are suppressed, obtaining a modulation signal Y that stabilizes style exposure.

While neuronal normalization stabilizes style exposure, it also suppresses fine details. To address this, we extract and retain details using a depthwise-separable convolution [8] followed by GELU, yielding a corrected detail map $\widetilde{Y}$. We then estimate a spatially varying gain map M that leverages neighborhood intensity, where a lightweight convolutional refiner processes $\widetilde{Y}$ and its output is squashed to $[0, 1]$ by a Sigmoid. Finally, we fuse the corrected details with a gated residual of the original input X. This process can be formalized as:

$$\begin{aligned} \widetilde{Y} &= \text{GELU}(\text{SepConv}(X)) \\ M &= \text{Sigmoid}(\text{Conv}(\widetilde{Y})) \\ X^{\text{enh}} &= \kappa \cdot \widetilde{Y} + (\mathbf{1}_{N \times C} + M) \odot X \end{aligned}$$

where $\odot$ is Hadamard product, $\mathbf{1}_{N \times C}$ is an all-ones matrix of size $N \times C$, and κ is a trainable parameter.

4.3. Dual-grained Attention Module (DAM)

Existing works using stripes window attention can capture local textures but, by confining computation to geometrically fixed windows, lack global context and often produce pronounced structural granular artifacts. To mitigate this problem, we augment the coarse-grained global context and fuse it with fine-grained local details, thereby reducing artifacts arising from locality. This mechanism is realized through our Dual-grained Attention Module (DAM).

In DAM, fine-grained local features are extracted using the efficient stripes window attention mechanism of Zhang et al. [44], which enhances stylistic detail in feature maps. In our module, it captures high-frequency details and intricate textures within horizontal ($n \times W$) and vertical ($H \times n$) bands as the fine-grained local feature extraction, where n is a hyperparameter.

For the coarse-grained global context, we employ Coordinate Attention [19]. Unlike stripes window attention, it aggregates information along the entire horizontal and vertical directions, producing compact axis-specific descriptors while capturing long-range dependencies. This directional aggregation provides a global yet coarse representation of the feature map, which we use to modulate local features.

In DAM, the Coordinate Attention proceeds in several steps. First, we generate a comprehensive global summary by performing pooling along both horizontal and vertical axes. While standard Coordinate Attention relies solely on average pooling to capture overall feature statistics, we further introduce an additional max pooling branch. We denote the features obtained by these pooling operators by $X^w_{\text{avg}}, X^w_{\text{max}}, X^h_{\text{avg}}, X^h_{\text{max}}$. This unique design complements average pooling by emphasizing the most salient responses, thereby retaining key structural details that would otherwise be smoothed out. As a result, four distinct feature vectors are obtained and concatenated for fusion:

$$X^{\text{pool}} = \text{Concat}(X^w_{\text{avg}}, X^w_{\text{max}}, X^h_{\text{avg}}, X^h_{\text{max}}),$$
$$X^{\text{fused}} = \text{ReLU}(\text{BatchNorm}(\text{Conv}(X^{\text{pool}}))).$$

The fused representation X^{fused} is then decomposed into two parts, i.e., $X^{\text{fused}} = [U^w, U^h]$. Concretely, through separate convolutional layers and Sigmoid functions, these are transformed into two attention maps, $A^w = \text{Conv}(U^w)$ and $A^h = \text{Conv}(U^h)$. These maps serve as gating signals, encoding the global importance of each row and column. Finally, the attention gating signals are employed to reweight the original input feature map X, thereby injecting global awareness into the features and enabling each local feature to recognize its position within the broader image structure. The Coordinate Attention is given by

$$\text{CoordAttn}(X^{\text{enh}}) = (A^h \otimes \mathbf{1}_W) \odot (\mathbf{1}_H \otimes A^w) \odot X^{\text{enh}}.$$

Finally, we adopt the Attention Fusion module from Zhang et al. [44]. This mechanism generates the final output through similarity-weighted summation, enabling the model to dynamically balance local texture detail and global structural coherence at each spatial location.

In Appendix C, we theoretically analyze the modified Coordinate Attention and show that it behaves as a per-channel diagonal operator in space, so it never blends content across locations, whereas window attention mixes positions through a dense attention matrix and channels via a value projection, making it more susceptible to striped and boundary artifacts.

4.4. Detail Engraver Adapter (DEA)

In the late stage of training, overfitting causes the model to forcibly match the style and content images, leading to fragmented artifacts from inconsistent alignment. To address this, drawing inspiration from [41, 24], we leverage the property that LoRA learns low-rank updates whose directions are orthogonal [41] to the singular vectors of the original pre-trained weight matrix. Although LoRA parameters are orthogonal, their framework-imposed constraints hinder the modeling of novel, small-scale nonlinear local image patterns, leaving the previous module unable to capture multi-granularity features. This limitation is critical for artistic style transfer, where styles exhibit multi-scale, multi-granularity textures, motivating the design of a module that jointly captures diverse local details.

To achieve this, we propose a Detail Engraver Adapter (DEA) that preserves global context by modeling long-range dependencies while enriching local representations via multi-scale depthwise convolutions, enabling fine-grained texture modeling across scales. The Adapter first applies a linear projection to reduce channel dimensionality for efficiency:

$$X^{\text{down}} = \text{GELU}(X^{\text{in}} W_{\text{down}} + \mathbf{1}_N b^\top_{\text{down}}),$$

where X^{in} is the output of DAM. $W_{\text{down}} \in \mathbb{R}^{C\times(C/4)}$ and $b \in \mathbb{R}^{C/4}$ are trainable weights. Next, to capture local context at multiple scales, we apply depthwise 2D convolutions with kernel sizes $K = \{3, 5, 7, 9\}$ and average their outputs:

$$X^{\text{mid}} = \frac{1}{4} \sum_{k\in K} \text{DepthWiseConv}_k(X^{\text{down}}).$$

Finally, the features are projected back and fused with the input through a learnable residual scaling:

$$X^{\text{out}} = X^{\text{in}} + \gamma \cdot \left(X^{\text{mid}} W_{\text{up}} + \mathbf{1}_N b_{\text{up}}^{\top}\right),$$

where γ controls the local enhancements, ensuring smooth integration with the backbone's global information. During training, the DEA is trained together with LoRA.

4.5. Network Training

Beyond the content and style losses introduced in Section 3, we adopt identity regularization proposed by [35] to curb excessive content distortion and under-stylization. Let I_{cc} and I_{ss} denote the outputs of our model when the (content, style) pairs are (I_c, I_c) and (I_s, I_s), respectively. Then we can define a pixel-level regularization term

$$\mathcal{L}_{\text{id1}} = \left\|I_{cc} - I_c\right\|_F + \left\|I_{ss} - I_s\right\|_F,$$

and a feature-level regularization term

$$\mathcal{L}_{\text{id2}} = \sum_{l\in\mathcal{N}} \left\|F^l(I_{cc}) - F^l(I_c)\right\|_F + \left\|F^l(I_{ss}) - F^l(I_s)\right\|_F.$$

where $\mathcal{N}$ is a set of the VGG-19 layers, consisting of relu_{2_1}, relu_{3_1}, relu_{4_1} and relu_{5_1}. The total training objective combines all terms:

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

where λ_c, λ_s, λ_{id1} and λ_{id2} weight the respective losses.

5. Experiments

In this section, we evaluate GSCNet's effectiveness in artifact suppression and compare it with several SOTA methods. And, we further conduct ablation studies.

5.1. Experimental Settings

The implementation details are available in Appendix D.

Datasets: We conduct experiments on two widely used style transfer benchmarks: the MS-COCO dataset [28] and the WikiArt dataset [36]. For training, we sample 80,000 content images from MS-COCO and 80,000 style images from WikiArt. For evaluation, we randomly sample 120 content images from MS-COCO and 120 style images from WikiArt.

Baselines: We compare our method against several representative style transfer models: **S2WAT** [44], our direct predecessor, a hierarchical Transformer built on striped-window attention; **StyTr2** [10], a pure-Transformer model that captures long-range dependencies through content-aware positional encoding; **SANET** [35], a style-attentional network that flexibly matches style patterns to content via learnable attention; **IEST** [6], which couples internal–external learning with contrastive learning for more harmonious stylization; **CAST** [46], a contrastive-learning-based arbitrary style transfer method; **AdaIN** [20], which aligns the channel-wise mean and variance of content features to those of the style; and the diffusion-based **InST** [45], which performs inversion-based style transfer with a pretrained diffusion model.

Evaluation Metrics: Following prior studies [2, 20, 10], we adopt six widely used metrics that jointly evaluate content preservation, style transfer quality, identity consistency, and reconstruction fidelity. Among these, the *multi-scale structural similarity index* (MS-SSIM) quantifies the preservation of spatial structure, with higher values reflecting better retention of structural details, and the *peak signal-to-noise ratio* (PSNR) measures reconstruction fidelity, with higher values indicating better quality. The remaining four loss-based metrics (content loss, style loss, and the two identity losses) are defined in Sections 3.1 and 4.5, where lower values indicate better performance.

5.2. Artifact and Stylization Consistency Analysis

A key challenge in style transfer is suppressing artifacts while ensuring consistent stylization. To assess these aspects, we conduct a qualitative comparison across representative baselines and our proposed model. Figure 3 presents results on four randomly selected image pairs, allowing us to examine how well each method preserves structural integrity and achieves coherent style application.

Regarding artifact suppression, we can observe from the first two rows of Figure 3 that our model retains the facial details and preserves the integrity of the woman's face, without producing strong artifacts or noise. In contrast, S2WAT and StyTr2 both produce striped noise and artifacts, and the AdaIN, which relies heavily on mean and variance alignment, even causes collapse of image details, which significantly affects visual perception. For another case in environmental scenes, i.e., the fourth row in Figure 3, our model retains a clean sky and preserves the integrity of image details. In contrast, the small white house in the bottom-right corner of the image generated by StyTr2 is damaged. This indicates that the model failed to effectively preserve the integrity of the content, resulting in artifacts. IEST model generates images with light style color application. This method reduces artifacts in the image but has a negative impact on capturing rich style features. Meanwhile, InST, as a diffusion-based method, applies a strong stylistic appearance but substantially alters the input, with facial structure and content details no longer faithfully preserved.

Regarding uneven stylization, we can observe in the third row of Figure 3 that our model demonstrates a consistent and holistic application of style across the entire image, while maintaining the structural integrity of the male face. In contrast, the male face in the image generated by S2WAT is hardly affected by the stylization at all, which means that this model overly biases towards content preservation, resulting in an incomplete stylization. Similarly, CAST and IEST struggle to capture the essence of the style, failing to reproduce its signature textures and colors. At the other extreme, StyTr2 successfully applies style globally but destroys critical content information, causing severe distortion of facial features in the male face. Furthermore, SANET and AdaIN are lacking in both aspects, resulting in inconsistent style application and severe degradation of content details.

Figure 3: Qualitative comparison on style transfer. Ours achieves more uniform style coverage and preserves facial details, while prior methods exhibit structural granular artifacts and brightness imbalance.

5.3. Frequency-Domain Analysis of Flat-Region Artifacts

The qualitative results in Section 5.2 show that window-based attention leaves periodic stripe artifacts in flat regions. However, the standard metrics used in style transfer, namely the content and style losses, MS-SSIM, and PSNR, assess overall fidelity and perceptual quality and do not isolate this kind of localized, periodic artifact, and to the best of our knowledge no existing metric directly quantifies it. We therefore design a dedicated, reference-free frequency-domain measure for these artifacts: a scalar PP(I) (the periodic-peak score), where a larger value indicates stronger artifacts and a truly smooth region yields a value close to 1. To avoid manual region selection, the flat-region

Table 1
Performance Comparison of Style Transfer Models.

Model	*Content* Loss ↓	*Style* Loss ↓	*Identity* Loss 1 ↓	*Identity* Loss 2 ↓	MS-SSIM ↑	PSNR ↑
S2WAT	1.66	1.74	0.16	1.38	0.651	12.685
StyTr2	1.83	1.52	0.26	3.10	0.605	12.090
CAST	2.07	4.33	1.94	18.72	0.619	12.082
IEST	1.81	2.72	0.91	7.16	0.551	12.326
SANET	2.16	**1.11**	0.81	6.03	0.448	11.125
AdaIN	1.71	3.50	2.54	17.97	0.539	11.268
InST	2.39	9.88	**0.06**	26.96	0.600	9.960
Ours	**1.45**	1.97	0.13	**0.81**	**0.710**	**13.179**

mask Ω is computed once from the content image I_c and shared across all methods: we take the pixels whose Sobel gradient magnitude is below the per-image median and apply a morphological erosion of radius 3. Deriving the mask from I_c rather than from the stylized output I prevents the high-gradient artifact pixels under inspection from being discarded as non-flat. We then slide a window over I and keep only the patches with at least 90% of their pixels inside Ω, denoting this set by $\mathcal{P}(I)$.

For each patch p, we subtract its mean, apply a separable 2D Hann window, take the normalized power spectrum, and average it angularly along the frequency radius to obtain a radial spectrum $P_p(f)$, where $f \in [0, 1]$ and $f = 1$ corresponds to the Nyquist frequency. Since the power spectrum of a flat region roughly follows a power-law decay, we fit a baseline $B_p(f) = 10^{a \log f + b}$ by least squares on $(\log f, \log P_p(f))$ and define

$$E_p(f) = \frac{P_p(f)}{B_p(f)}, \qquad \pi(p) = \max_{f \in [0.15, 0.6]} E_p(f), \qquad \mathrm{PP}(I) = \frac{1}{|\mathcal{P}(I)|} \sum_{p \in \mathcal{P}(I)} \pi(p). \tag{1}$$

A flat region with only the natural $1/f^a$ decay gives $E_p \approx 1$, whereas a periodic stripe raises E_p into a peak above 1 at the corresponding frequency, so a lower PP means fewer artifacts. As the baseline is fitted from each patch itself, this whitening is self-referential and requires no external clean reference. In our experiments the patch size is set to 32×32.

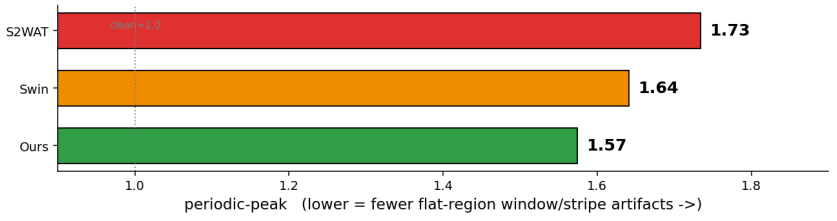

Figure 4: Periodic-peak (PP) scores on flat regions (lower is cleaner). Among window-attention methods, GSCNet (1.57) produces the fewest flat-region window/stripe artifacts, below Swin Transformer (1.64) and S2WAT (1.73).

We evaluate on 100 content-style pairs (5 content images $\times$ 20 style images). As shown in Figure 4, GSCNet attains the lowest score (1.57) among the window-attention methods, below Swin (1.64) and its predecessor S2WAT (1.73). This provides additional quantitative evidence that GSCNet produces the fewest structural artifacts in flat regions, consistent with Section 5.2.

5.4. Comparison with SOTA methods

To evaluate the performance of our proposed model on the image style transfer task, we compare it against all baselines using the six quantitative metrics introduced in Section 5.1, with the results shown in Table 1.

For quantitative evaluation, as shown in Table 1, GSCNet achieves the best result on four of the six metrics. Compared with S2WAT, it attains the lowest content loss (1.45, 12.7% lower), the highest MS-SSIM (0.710, 9.1%

higher), the lowest feature-level identity loss (Identity Loss 2 = 0.81, 41.3% lower) together with a low pixel-level identity loss (Identity Loss 1 = 0.13, 18.8% lower than S2WAT); it also attains the highest PSNR (13.179, 3.9% higher). Among the feed-forward baselines, the only metric on which GSCNet does not rank first is style loss (1.97), which is higher than S2WAT (1.74) and SANET (1.11); the diffusion baseline InST attains a lower pixel-level identity loss (Identity Loss 1 = 0.06) but is markedly worse on all other metrics. We regard this as a deliberate content–style trade-off: GSCNet prioritizes content and identity preservation, and the qualitative examples in Figure 3 suggest that the resulting stylizations remain visually coherent. Averaged over the six metrics, including the 13.2% style-loss regression, GSCNet still improves by 12% over S2WAT, reflecting a better overall balance between content fidelity and faithful style rendering.

5.5. Ablation Study

5.5.1. Dual-grained Attention Module

To assess the contribution of the DAM, we perform an ablation experiment by replacing Coordinate Attention with the window attention, denoted as w/o CA, and compare it with our proposed model. The results are shown in Figure 5.

As shown in Figure 5, the w/o CA setting introduces severe striping artifacts in the sky regions, a typical issue of window-based attention-driven strip merging. In contrast, our proposed model with CA produces images with clearer structural details and significantly reduced striped artifacts, demonstrating the effectiveness of the proposed module in mitigating artifacts and enhancing visual fidelity.

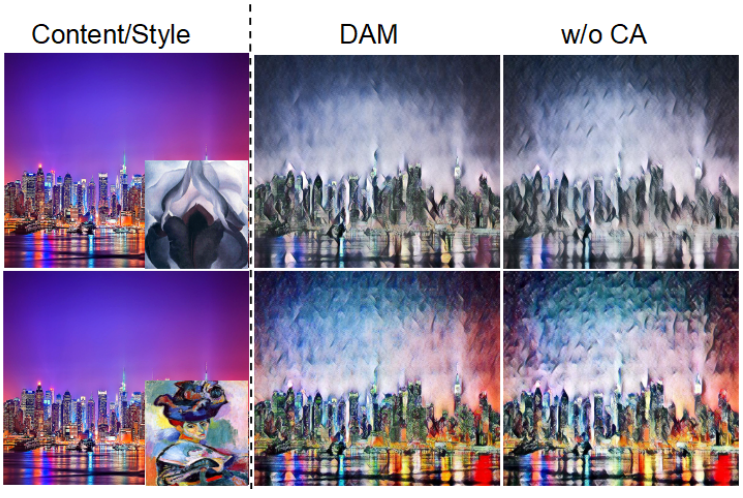

Figure 5: Ablation Experiments on Coordinate Attention within DAM.

5.5.2. Feature Stabilization Module

To validate the effectiveness of the proposed FSM, we conduct an ablation experiment comparing four settings: S2WAT (w/o FSM), S2WAT(w FSM), Ours (w/o FSM), and Ours (Full). The qualitative results are shown in Figure 6, and the corresponding quantitative evaluation is reported in Table 2.

As shown in Figure 6, the models w/o FSM suffer from uneven style transfer and, in some cases, localized overexposure (e.g., the excessively bright contours of the person in S2WAT (w/o FSM)). In contrast, incorporating FSM (S2WAT(w FSM) and Ours (Full)) yields more balanced and natural style transfer, indicating that FSM helps stabilize local brightness and reduce exposure artifacts.

To quantify the plug-in benefit of FSM, Table 2 reports the metrics under four settings. On both backbones, FSM consistently improves identity preservation, structural similarity, and reconstruction fidelity. When added to S2WAT, FSM reduces Identity Loss 2 from 1.38 to 1.01 and raises MS-SSIM (0.651 to 0.678) and PSNR (12.685 to 12.736), at the cost of a marginal increase in content and style loss, reflecting a similar trade-off in which improved stability and identity preservation come at a small cost in content and style loss. When integrated into our full model, FSM improves all six metrics, most notably Identity Loss 2 (1.00 to 0.81), MS-SSIM (0.701 to 0.710), and PSNR (13.005

Figure 6: Ablation Experiments on Feature Stabilization Module (FSM).

Table 2
Quantitative ablation of the Feature Stabilization Module (FSM), evaluated as a plug-in on both S2WAT and our model. Best result per column in bold.

Setting	*Content* Loss ↓	*Style* Loss ↓	*Identity* Loss 1 ↓	*Identity* Loss 2 ↓	MS-SSIM ↑	PSNR ↑
S2WAT (w/o FSM)	1.66	**1.74**	0.16	1.38	0.651	12.685
S2WAT(w FSM)	1.69	1.80	0.15	1.01	0.678	12.736
Ours (w/o FSM)	1.46	1.99	0.15	1.00	0.701	13.005
Ours (Full)	**1.45**	1.97	**0.13**	**0.81**	**0.710**	**13.179**

to 13.179). These quantitative gains are consistent with the qualitative results in Figure 6, where FSM removes the localized overexposure and uneven brightness present in the settings without it.

5.5.3. Detail Engraver Adapter

To validate the effectiveness of the DEA, we compare two settings: our Model (w/o DEA) and our Full Model. The results are shown in Figure 7.

As shown in Figure 7, the model (w/o DEA) does not exhibit large structural artifacts on the generated face but still suffers from noticeable point-like artifacts. In contrast, applying our DEA substantially reduces facial artifacts while significantly decreasing the number of trainable parameters, thereby achieving effective model lightweighting without compromising visual quality. A detailed efficiency comparison of total parameters, trainable parameters, GFLOPs, and inference latency against representative baselines is provided in Appendix F.

Figure 7: Ablation Experiments on the Detail Engraver Adapter with LoRA.

6. Conclusion

We propose a novel style transfer model, **GSCNet**, which effectively suppresses artifacts from three levels of granularity: **(i) Structural granularity:** we propose the Dual-grained Attention Module to integrate global context with local stripes window attention, addressing window-shaped artifacts. **(ii) Pixel granularity:** we propose Feature Stabilization Module to suppress abnormal pixel activations, mitigating pixel-level artifacts. **(iii) Detail granularity:** Through our proposed Detail Engraver Adapter, the model is guided to capture complementary multi-scale stylistic cues, alleviating fragmented detail artifacts. Extensive experiments demonstrate that our model can effectively suppress artifacts, achieving high-quality style transfer with structural coherence and detail preservation.

6.1. Limitations and Future Work

GSCNet's inference latency is higher than that of purely attention-based baselines, because the multi-scale depthwise convolutions in DEA are memory-bandwidth-bound and achieve lower GPU utilization than the large matrix multiplications that dominate attention-heavy architectures; this is a deliberate quality–efficiency trade-off for consistent artifact suppression. In addition, the model intentionally prioritizes content and structural preservation, which results in a slightly higher style loss and may lead to mild under-stylization for styles that demand strong texture dominance. Regarding evaluation, the periodic-peak (PP) metric proposed in this work specifically quantifies periodic stripe artifacts in flat regions (structural granularity), whereas pixel-granularity artifacts (e.g., brightness imbalance) and detail-granularity artifacts (e.g., texture fragmentation) are assessed primarily through the FSM ablation in Table 2 and qualitative comparisons; a unified artifact metric across structural, pixel-level, and detail-level artifacts, as well as a formal user study, remains future work. Future work will explore operator fusion and kernel optimization to reduce inference latency, introduce a tunable content–style trade-off mechanism to accommodate a wider range of stylistic preferences, and incorporate user studies together with more comprehensive artifact quantification metrics.

Funding

This work was supported by the Australian Research Council (ARC) under the DECRA Fellowship DE240101049.

Declaration of competing interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The datasets used in this work are publicly available. Content images are from MS-COCO [28] and style images are from WikiArt [36]. The code and trained models will be made publicly available upon acceptance.

CRediT authorship contribution statement

Zheng Gao: Conceptualization, Methodology, Software, Investigation, Validation, Formal analysis, Writing – original draft. **Zhicheng Bao:** Methodology, Visualization, Writing – review & editing. **Haoran Fan:** Methodology, Visualization, Writing – review & editing. **Xiaoyu Li:** Formal analysis, Writing – original draft, Writing – review & editing. **Qipei Nong:** Software, Investigation, Validation. **Jiaojiao Jiang:** Project administration, Funding acquisition, Writing – review & editing.

References

[1] Allen-Zhu, Z., Li, Y., 2023. Backward feature correction: How deep learning performs deep (hierarchical) learning, in: The Thirty Sixth Annual Conference on Learning Theory, PMLR. pp. 4598–4598.

[2] An, J., Huang, S., Song, Y., et al., 2021. Artflow: Unbiased image style transfer via reversible neural flows, in: CVPR.

[3] Bouvrie, J.V., 2009. Hierarchical learning: Theory with applications in speech and vision. Ph.D. thesis. Massachusetts Institute of Technology.

[4] Carandini, M., Heeger, D.J., 2012. Normalization as a canonical neural computation. Nature Reviews Neuroscience 13, 51–62.

[5] Chen, D., Yuan, L., Liao, J., Yu, N., Hua, G., 2017. Stylebank: An explicit representation for neural image style transfer, in: CVPR.

[6] Chen, H., Wang, Z., Zhang, H., et al., 2021. Artistic style transfer with internal-external learning and contrastive learning, in: NeurIPS.

[7] Chen, T.Q., Schmidt, M., 2016. Fast patch-based style transfer of arbitrary style. arXiv preprint arXiv:1612.04337 .

[8] Chollet, F., 2017. Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1251–1258.

[9] Deng, Y., Tang, F., Dong, W., et al., 2021. Arbitrary video style transfer via multi-channel correlation, in: Proceedings of the AAAI Conference on Artificial Intelligence.

[10] Deng, Y., Tang, F., Dong, W., et al., 2022. Stytr2: Image style transfer with transformers, in: CVPR.

[11] Efros, A.A., Freeman, W.T., 2001a. Image quilting for texture synthesis and transfer, in: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, pp. 341–346.

[12] Efros, A.A., Freeman, W.T., 2001b. Image quilting for texture synthesis and transfer, in: Proceedings of the 28th annual conference on Computer graphics and interactive techniques (SIGGRAPH'01), ACM. pp. 341–346.

[13] Efros, A.A., Leung, T.K., 1999. Texture synthesis by non-parametric sampling, in: Proceedings of the seventh IEEE international conference on computer vision, IEEE. pp. 1033–1038.

[14] Gatys, L., Ecker, A.S., Bethge, M., 2015. Texture synthesis using convolutional neural networks, in: Advances in Neural Information Processing Systems.

[15] Gatys, L.A., Ecker, A.S., Bethge, M., 2016a. Image style transfer using convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2414–2423.

[16] Gatys, L.A., Ecker, A.S., Bethge, M., 2016b. A neural algorithm of artistic style, in: Vision Sciences Society.

[17] Gretton, A., Borgwardt, K.M., Rasch, M.J., Schölkopf, B., Smola, A., 2012. A kernel two-sample test. Journal of Machine Learning Research 13, 723–773.

[18] Heeger, D.J., Bergen, J.R., 1995. Pyramid-based texture analysis/synthesis, in: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques, pp. 229–238.

[19] Hou, Q., Zhou, D., Feng, J., 2021. Coordinate attention for efficient mobile network design, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13713–13722.

[20] Huang, X., Belongie, S., 2017. Arbitrary style transfer in real-time with adaptive instance normalization, in: Proceedings of the IEEE International Conference on Computer Vision, pp. 1501–1510.

[21] Johnson, J., Alahi, A., Fei-Fei, L., 2016. Perceptual losses for real-time style transfer and super-resolution, in: European Conference on Computer Vision (ECCV).

[22] Kalischek, N., Wegner, J.D., Schindler, K., 2021. In the light of feature distributions: moment matching for neural style transfer, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9382–9391.

[23] Kolkin, N., Salavon, J., Shakhnarovich, G., 2019. Style transfer by relaxed optimal transport and self-similarity, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10051–10060.

[24] Kong, C., Luo, A., Bao, P., et al., 2024. Moe-ffd: Mixture of experts for generalized and parameter-efficient face forgery detection. arXiv preprint arXiv:2404.08452 .

[25] Kong, X., Deng, Y., Tang, F., et al., 2023. Exploring the temporal consistency of arbitrary style transfer: A channelwise perspective. IEEE Transactions on Neural Networks and Learning Systems .

[26] Kwatra, V., Schödl, A., Essa, I., Turk, G., Bobick, A., 2003. Graphcut textures: Image and video synthesis using graph cuts. ACM Transactions on Graphics 22, 277–286.

[27] Li, Y., Wang, N., Liu, J., Hou, X., 2017. Demystifying neural style transfer, in: Proceedings of the 26th International Joint Conference on Artificial Intelligence, pp. 2230–2236.

[28] Lin, T.Y., Maire, M., Belongie, S., et al., 2014. Microsoft coco: Common objects in context, in: European Conference on Computer Vision (ECCV).
[29] Liu, Y., Dong, Z., Zhu, P., Liu, S., 2022. Unsupervised underwater image enhancement based on feature disentanglement. Journal of Electronics & Information Technology 44, 3389–3398.
[30] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B., 2021. Swin transformer: Hierarchical vision transformer using shifted windows, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10012–10022.
[31] Luan, F., Paris, S., Shechtman, E., Bala, K., 2017. Deep photo style transfer, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4990–4998.
[32] Ma, Y., Zhao, C., Li, X., Basu, A., 2023. Rast: Restorable arbitrary style transfer via multi-restoration, in: WACV.
[33] Mechrez, R., Shechtman, E., Zelnik-Manor, L., 2017. Photorealistic style transfer with screened poisson equation, in: Proceedings of the British Machine Vision Conference, British Machine Vision Association.
[34] Mroueh, Y., 2020. Wasserstein style transfer, in: International Conference on Artificial Intelligence and Statistics, PMLR. pp. 842–852.
[35] Park, D.Y., Lee, K.H., 2019. Arbitrary style transfer with style-attentional networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5880–5888.
[36] Phillips, F., Mackintosh, B., 2011. Wiki art gallery, inc.: A case for critical thinking. Issues in Accounting Education 26, 379–390.
[37] Pitié, F., Kokaram, A.C., Dahyot, R., 2007. Automated colour grading using colour distribution transfer. Computer Vision and Image Understanding 107, 123–137.
[38] Portilla, J., Simoncelli, E.P., 2000. A parametric texture model based on joint statistics of complex wavelet coefficients. International Journal of Computer Vision 40, 49–70.
[39] Rabin, J., Peyré, G., Delon, J., Bernot, M., 2011. Wasserstein barycenter and its application to texture mixing, in: International conference on scale space and variational methods in computer vision, Springer. pp. 435–446.
[40] Reinhard, E., Adhikhmin, M., Gooch, B., Shirley, P., 2002. Color transfer between images. IEEE Computer graphics and applications 21, 34–41.
[41] Shuttleworth, R., Andreas, J., Torralba, A., et al., 2024. Lora vs full fine-tuning: An illusion of equivalence. arXiv preprint arXiv:2410.21228 .
[42] Simonyan, K., Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition, in: 3rd International Conference on Learning Representations (ICLR 2015).
[43] Solomon, J., De Goes, F., Peyré, G., Cuturi, M., Butscher, A., Nguyen, A., Du, T., Guibas, L., 2015. Convolutional wasserstein distances: Efficient optimal transportation on geometric domains. ACM Transactions on Graphics 34, 1–11.
[44] Zhang, C., Xu, X., Wang, L., Dai, Z., Yang, J., 2024. S2wat: Image style transfer via hierarchical vision transformer using strips window attention, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 7024–7032.
[45] Zhang, Y., Huang, N., Tang, F., Huang, H., Ma, C., Dong, W., Xu, C., 2023. Inversion-based style transfer with diffusion models, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 10146–10156.
[46] Zhang, Y., Tang, F., Dong, W., et al., 2022. Domain enhanced arbitrary image style transfer via contrastive learning, in: SIGGRAPH.
[47] Zhu, P., Liu, Y., Wen, Y., Xu, M., Fu, X., Liu, S., 2023. Unsupervised underwater image enhancement via content-style representation disentanglement. Engineering Applications of Artificial Intelligence 126, 106866. URL: https://www.sciencedirect.com/science/article/pii/S0952197623010503, doi:https://doi.org/10.1016/j.engappai.2023.106866.
[48] Zhu, P., Liu, Y., Xu, M., Fu, X., Wang, N., Liu, S., 2024. Unsupervised multiple representation disentanglement framework for improved underwater visual perception. IEEE Journal of Oceanic Engineering 49, 48–65. doi:10.1109/JOE.2023.3317903.

A. More Related Work

A.1. Classical Image Style Transfer

Early style transfer methods predate deep learning and treat style as statistics or exemplars to be matched. Histogram matching and color transfer techniques align low-order moments between source and target, often in decorrelated color spaces to avoid channel coupling [40, 37]. Patch-based texture synthesis views style as an arrangement of local patterns: nonparametric sampling and graph-cut recomposition assemble an output by copying and stitching patches from an exemplar while enforcing spatial coherence [13, 12, 26]. Multi-scale parametric models further capture statistics of filter responses such as wavelets and steerable pyramids enabling compelling stationary textures but limited structural control [18, 38]. While these approaches are efficient and interpretable, their reliance on hand-crafted features and stationarity assumptions makes them struggle with complex, nonstationary artistic styles or strong semantic structure.

A.2. Theory of Image Style Transfer

Theoretical views of NST typically formalize style as statistics of a feature distribution and content as geometric/semantic structure. Gram-based losses match second-order moments of deep activations, which can be interpreted as matching uncentered covariances or feature correlations; this connects to classical texture models (e.g., Portilla–Simoncelli) and to moment-matching in statistical learning [38, 16]. From a distributional perspective, many stylization losses instantiate integral probability metrics: AdaIN approximates alignment of per-channel means/variances, WCT aligns full Gaussian approximations, and MMD/Wasserstein objectives match feature distributions more faithfully

[17, 23, 34]. [27] showed that matching the Gram matrices is equivalent to minimizing the MMD. Optimal transport provides a geometric account of transferring "mass" between content and style feature distributions; sliced/regularized OT yields tractable alignments that better preserve structure [39, 23, 34, 43]. Patch-based neural methods approximate a nearest-neighbor projection onto the style manifold in feature space, offering a nonparametric consistency prior for high-frequency details [7]. Recent analyses also study the content–style trade-off as a multi-objective optimization, where regularizers (total variation, Laplacian/affinity constraints) control the Pareto frontier between texture realism and structural distortion [31, 33].

B. Image Style Transfer as Optimal Transport Problem

We introduce the mathematical formulation of the universal transfer style frame work developed in [34]. Formally, UST framework operates as follows.

Encoder/Decoder. Let F be a fixed deep network. For an image I, let

$$F(I) = \{F_j(I) \in \mathbb{R}^C \,:\, j \in [N]\}$$

denote the collection of C-dimensional feature vectors at all spatial locations where $N := HW$ is the spatial dimension. Following the UST formalism, we view these features as an empirical distribution in feature space, and there is an encoder E that maps any image I to

$$\nu_I := \frac{1}{n}\sum_{j=1}^{N} \delta_{F_j(I)}$$

where $\delta_{F_j(I)}$ denotes the Dirac measure on $F_j(I)$. We assume the existence of a decoder D that approximately inverts E, i.e.,

$$D(E(I)) \approx I.$$

An encoder and a decoder trained between the pixel domain and a chosen VGG layer satisfies this in practice. For example, E and D can be image encoder and decoder trained from the pixel domain to a spatial convolutional layer output in VGG and vice-versa.

Transfer Module. Given a content image I_c and a style image I_s, UST applies a feature-space transform $T_{cs} : \mathbb{R}^{HW} \to \mathbb{R}^{HW}$ such that its pushforward matches the spatial feature distributions, i.e.,

$$(T_{cs})_{\#}\nu_{I_c} = \nu_{I_s}.$$

Here, $(T_{cs})_{\#}$ denotes the pushforward map, which sends the empirical distribution from the content space to the style space. The stylized output is then obtained by decoding the pushforward of the encoded content image, i.e.,

$$I_o = D((T_{cs})_{\#}E(I_c)).$$

In practice, $T_{c\to s}$ is instantiated by statistical alignment of features, e.g., channel-wise mean and variance matching we defined before.

C. Theoretical Analysis

Definition 1 (Softmax)**.** *We define the function* $\mathrm{Softmax} : \mathbb{R}^d \to \mathbb{R}^d$ *as*

$$\mathrm{Softmax}(z) := \frac{\exp(z)}{\langle \exp(z), \mathbf{1}_n \rangle}.$$

Definition 2 (Self-Attention)**.** *Given* $X \in \mathbb{R}^{n\times d}$*, the full attention is defined as*

$$\mathrm{Attn}(X) := \mathrm{Softmax}\left(\frac{QK^\top}{\sqrt{d}}\right)V$$

where $Q = XW_Q, K = XW_K, V = XW_V$ *are the query, key, value projections, respectively, and* $W_Q, W_K, W_V \in \mathbb{R}^{d\times d}$ *are learnable weight matrices, and* Softmax *is applied row-wise.*

In vision tasks, the input image is typically a feature tensor $\mathsf{X} \in \mathbb{R}^{H\times W\times C}$. Flattening the spatial dimensions yields its matrix form $X = \mathrm{mat}(\mathsf{X}) \in \mathbb{R}^{HW\times C}$ where $X_{(i-1)W+j,c} = \mathsf{X}_{i,j,c}$ when feeding to the self-attention module, where HW and C correspond to n and d in Definition 2. We reserve the symbols X to denote the matrix representation of X.

Definition 3 (Window Attention [30])**.** *Given an input image feature $\mathsf{X} \in \mathbb{R}^{H\times W\times C}$, and a partition $\mathcal{P} = \{P_1, P_2, \dots, P_m\}$ of $[H]\times[W]$ (each P_j has size $H_j \times W_j$), we define the attention in the local window P_j as*

$$\mathrm{WindAttn}_{P_j}(\mathsf{X}) := \mathrm{Softmax}\left(\frac{Q_j K_j^\top}{\sqrt{d}} + B_j\right) V_j.$$

Here $Q_j = X[P_j]\cdot W_{Q_j}, K_j = X[P_j]\cdot W_{K_j}, V_j = X[P_j]\cdot W_{V_j}$ are the learnable query, key, value projections, respectively, where $W_{Q_j}, W_{K_j}, W_{V_j} \in \mathbb{R}^{C\times C}$, and $X[P_j] \in \mathbb{R}^{H_j W_j \times C}$ are the hidden states that restricted to the indices in P_j, and $B_j \in \mathbb{R}^{H_j W_j \times H_j W_j}$ is the positional bias weight.

The output of the window attention with an input $\mathsf{X} \in \mathbb{R}^{H\times W\times C}$ and a partition $\mathcal{P} = \{P_1, \dots, P_m\}$ is obtained by applying $\mathrm{WindAttn}_{P_j}$ on each window and writing the results back to their original positions.

Definition 3 allows arbitrary rectangular windows with heterogeneous sizes and it includes several types of windows discussed in [44]:

- Horizontal strip windows: choose a partition with $P_j = \{(h,w) : h \in I_j, w \in [W]\}$, $|I_j| = n$; then $H_j = n$, $W_j = W$.
- Vertical strip windows: choose $P_j = \{(h,w) : h \in [H], w \in J_j\}$, $|J_j| = n$; then $H_j = H$, $W_j = n$.
- Square windows: tile $[H]\times[W]$ by disjoint $M \times M$ blocks; then $H_j = W_j = M$.

Definition 4 (Modified Coordinate Attention)**.** *Let $\mathsf{X} \in \mathbb{R}^{H\times W\times C}$ be an input feature map of height H, width W, and C channels. The modified coordinate attention mechanism proceeds as follows:*

1. ***Height- and width-wise pooling.*** *We extract aggregated descriptors along the spatial dimensions using both average pooling and max pooling:*

$$Z^h_{i,c} = \frac{1}{W}\sum_{j=1}^{W} \mathsf{X}_{i,j,c}, \qquad Y^h_{i,c} = \max_{j\in[W]} \mathsf{X}_{i,j,c}, \qquad \forall i \in [H],\ c \in [C],$$

$$Z^w_{j,c} = \frac{1}{H}\sum_{i=1}^{H} \mathsf{X}_{i,j,c}, \qquad Y^w_{j,c} = \max_{i\in[H]} \mathsf{X}_{i,j,c}, \qquad \forall j \in [W],\ c \in [C].$$

2. ***Channel reduction.*** *Let $r \geq 2$ be the reduction ratio. With a shared projection $W^{\mathrm{shared}} \in \mathbb{R}^{C\times(C/r)}$ and a nonlinearity activation ϕ, we obtain reduced features:*

$$U^h = \phi(Z^h W^{\mathrm{shared}}), \quad U^w = \phi(Z^w W^{\mathrm{shared}}),$$
$$V^h = \phi(Y^h W^{\mathrm{shared}}), \quad V^w = \phi(Y^w W^{\mathrm{shared}}),$$

where $U^h, V^h \in \mathbb{R}^{H\times(C/r)}$ and $U^w, V^w \in \mathbb{R}^{W\times(C/r)}$.

3. ***Channel fusion.*** *Concatenating average- and max-based descriptors, we apply learnable transformations $W^h, W^w \in \mathbb{R}^{(2C/r)\times C}$ with a nonlinearity activation σ:*

$$A^h = \sigma([U^h, V^h]W^h) \in \mathbb{R}^{H\times C}, \quad A^w = \sigma([U^w, V^w]W^w) \in \mathbb{R}^{W\times C}.$$

4. ***Attention generation.*** *The final coordinate attention is applied multiplicatively along both height and width:*

$$\mathrm{CoordAttn}(\mathsf{X})_{i,j,c} = \mathsf{X}_{i,j,c}\cdot A^h_{i,c}\cdot A^w_{j,c}, \quad \forall i \in [H],\ j \in [W],\ c \in [C].$$

Proposition 5 (Matrix Form of Coordinate Attention)**.** *Given an input image feature* $\mathsf{X} \in \mathbb{R}^{H\times W\times C}$*, the coordinate attention satisfies*

$$\mathrm{mat}(\mathrm{CoordAttn}(\mathsf{X})) = (A^h \otimes \mathbf{1}_W) \odot (\mathbf{1}_H \otimes A^w) \odot X,$$

where A^h, A^w *are defined in Definition 4.*

Proof. Let $X = \mathrm{mat}(\mathsf{X}) \in \mathbb{R}^{HW\times C}$ with $X_{(i-1)W+j,c} = \mathsf{X}_{i,j,c}$. Let $\mathbf{1}_W \in \mathbb{R}^{W\times 1}$ and $\mathbf{1}_H \in \mathbb{R}^{H\times 1}$ be the all-ones vectors, and define

$$\widehat{A} := A^h \otimes \mathbf{1}_W \in \mathbb{R}^{HW\times C}, \quad \widetilde{A} := \mathbf{1}_H \otimes A^w \in \mathbb{R}^{HW\times C}.$$

By definition of Kronecker product, we have, for any $i \in [H], j \in [W], c \in [C]$,

$$\widehat{A}_{(i-1)W+j,c} = A^h_{i,c}, \qquad \widetilde{A}_{(i-1)W+j,c} = A^w_{j,c}.$$

Hence

$$(\widetilde{A} \odot \widehat{A} \odot X)_{(i-1)W+j,c} = A^h_{i,c} A^w_{j,c} X_{(i-1)W+j,c} = A^h_{i,c}\, A^w_{j,c} \mathsf{X}_{i,j,c}.$$

By Definition 4, this is exactly $\mathrm{CoordAttn}(\mathsf{X})_{i,j,c}$. Thus we complete the proof. □

Coordinate attention is just a separable, axis-wise mask multiplied onto the feature map, it can be viewed as a diagonal operator in space (per channel), giving global row/column consistency without blending content from other locations.

Proposition 6 (Coordinate Attention as a diagonal operator in space)**.** *Given an input image feature* $\mathsf{X} \in \mathbb{R}^{H\times W\times C}$*, the coordinate attention is given by*

$$\mathrm{CoordAttn}(\mathsf{X})_{:,:,c} = \mathrm{Diag}(A^h_{:,c}) \mathsf{X}_{:,:,c} \,\mathrm{Diag}(A^w_{:,c}) \quad \forall c \in [C],$$

where A^h, A^w *are defined in Definition 4.*

Proof. Fix $c \in [C]$ and denote $X^{(c)} := \mathsf{X}_{:,:,c} \in \mathbb{R}^{H\times W}$. Let $D^{(c)}_h := \mathrm{Diag}(A^h_{:,c}) \in \mathbb{R}^{H\times H}$ and $D^{(c)}_w := \mathrm{Diag}(A^w_{:,c}) \in \mathbb{R}^{W\times W}$. For any $i \in [H]$, $j \in [W]$,

$$\begin{aligned}
(D^{(c)}_h X^{(c)} D^{(c)}_w)_{i,j} &= \sum_{i'=1}^{H} \sum_{j'=1}^{W} (D^{(c)}_h)_{i,i'} X^{(c)}_{i',j'} (D^{(c)}_w)_{j',j} \\
&= (D^{(c)}_h)_{i,i} X^{(c)}_{i,j} (D^{(c)}_w)_{j,j} \qquad \text{(since } D^{(c)}_h, D^{(c)}_w \text{ are diagonal)} \\
&= A^h_{i,c} \mathsf{X}_{i,j,c} A^w_{j,c}.
\end{aligned}$$

By Definition 4, $\mathrm{CoordAttn}(\mathsf{X})_{i,j,c} = \mathsf{X}_{i,j,c} \cdot A^h_{i,c} \cdot A^w_{j,c}$, which equals the right-hand side above. Therefore,

$$\mathrm{CoordAttn}(\mathsf{X})_{:,:,c} \;=\; \mathrm{Diag}(A^h_{:,c})\, \mathsf{X}_{:,:,c}\, \mathrm{Diag}(A^w_{:,c}) \quad \forall\, c \in [C].$$

Thus we complete the proof. □

Remark 7 (Comparison of Window Attention and Coordinate Attention)**.** *Within each window, standard self-attention applies a linear position–mixing operator A together with a channel–mixing value projection* W_V*. By contrast, coordinate attention first aggregates one-dimensional descriptors along rows and columns, producing per-row and per-column summaries (e.g., statistics that capture intensity or texture trends). These summaries are then combined to form a separable spatial gating vector that modulates each location independently. Because the modulation is scalar per location and does not mix features across positions, the operation preserves the underlying image geometry and maintains sharp boundaries. Practically, this tends to enhance large homogeneous regions (e.g., sky) and smooth facial areas while avoiding the texture bleeding often introduced by windowed attention, which forms dense linear combinations within each local window.*

D. Training Details

D.1. Training information

We conduct all experiments with a learning rate of 0.0001 and a batch size of 4, training for 40000 iterations on a NVIDIA RTX 3090 GPU.

D.2. Dataset

We adopt MS-COCO [28] as the content dataset and WikiArt [36] as the style dataset. During training, images are resized to 512 on the shorter side and randomly cropped to 224×224, while during testing, images of arbitrary size are accepted.

E. Network Architecture

E.1. Encoder

The encoder adopts default settings with stage dimensions of 192, 384, and 768; strip widths of 2, 4, and 7; and attention heads of 3, 6, and 12 across the three stages.

E.2. Transfer Module

The Transfer Module is configured with a default dimension $d = 768$, number of attention heads $H = 8$, and number of layers $L = 3$.

Algorithm 1 TransferModule(X_c, X_s)

Input: a content image feature X_c, a style image feature X_s
Output: a synthesized image feature X_{cs}

for $l = 1, \ldots, L$ **do**
  /* Layer Normalization */
  $X_c^{\mathrm{LN}} \leftarrow \mathrm{LayerNorm}(X_c)$
  /* Multihead Self Attention + Residual Connection */
  for $h = 1, \ldots, H$ **do**
    $Q^{(j,h)} \leftarrow X_c^{\mathrm{LN}} W_Q^{(j,h)}, \quad K^{(j,h)} \leftarrow X_c^{\mathrm{LN}} W_K^{(j,h)}, \quad V^{(j,h)} \leftarrow X_c^{\mathrm{LN}} W_V^{(j,h)}$
  end for
  $X_c \leftarrow \sum_{h=1}^{H} \mathrm{Softmax}(Q^{(j,h)}(K^{(j,h)})^{\top}/\sqrt{d})V^{(j,h)} + X_c$
  /* Layer Normalization */
  $X_c^{\mathrm{LN}} \leftarrow \mathrm{LayerNorm}(X_c)$
  /* Multihead Cross Attention + Residual Connection */
  for $h = 1, \ldots, H$ **do**
    $\widetilde{Q}^{(j,h)} \leftarrow X_c^{\mathrm{LN}} \widetilde{W}_Q^{(j,h)}, \quad K^{(j,h)} \leftarrow X_c^{\mathrm{LN}} \widetilde{W}_K^{(j,h)}, \quad V^{(j,h)} \leftarrow X_c^{\mathrm{LN}} \widetilde{W}_V^{(j,h)}$
  end for
  $X_c \leftarrow \sum_{h=1}^{H} \mathrm{Softmax}(\widetilde{Q}^{(j,h)}(\widetilde{K}^{(j,h)})^{\top}/\sqrt{d})\widetilde{V}^{(j,h)} + X_c$
end for
$X_{cs} \leftarrow X_c$
return X_{cs}

E.3. Decoder

The decoder progressively upsamples the stylized features while reducing the number of channels from 768 to 256, then to 128, 64, and finally to 3. The stylized features are initially shaped as

$$\frac{W}{8} \times \frac{H}{8} \times C,$$

and are reshaped by the patch reverse layer into a tensor of shape

$$C \times \frac{H}{8} \times \frac{W}{8}.$$

Table 3
Comparison of Model Complexity (#Params, GFLOPs, Latency@224, and peak memory).

Method	Total Params (M) ↓	Trainable Params (M) ↓	GFLOPs ↓	Latency@224 (ms) ↓	Memory (MB) ↓
S2WAT	64.96	64.96	299.43	38.41	751.36
StyTr2	48.34	48.34	54.38	23.75	250.89
GSCNet (Ours)	60.92	**34.19**	151.97	48.96	993.07

We define a basic processing unit, denoted as **BAP**, which consists of a padding operation, a 3×3 convolution, and a ReLU activation. The mirrored VGG decoder upsamples the features in three stages:

- **Stage 1**: BAP $\rightarrow$ Upsample $\times2$ $\rightarrow$ $3\times$ BAP
- **Stage 2 and 3**: BAP $\rightarrow$ Upsample $\times2$ $\rightarrow$ BAP

Each stage increases the spatial resolution while decreasing the number of channels. After the final stage, the output features of shape

$$C' \times H \times W$$

are processed by a padding layer and a 3×3 convolution to reduce the number of channels to 3, resulting in the final stylized image.

F. Model Efficiency

To verify the efficiency of the proposed DEA with LoRA, we compare the total parameters, trainable parameters, GFLOPs, inference latency, and peak inference memory of GSCNet against representative baselines. All GFLOPs, latency, and memory measurements are conducted at 224×224 input resolution on the same hardware. The results are summarized in Table 3.

As reported in Table 3, although GSCNet's total parameter count (60.92M) is relatively large because it retains a frozen pretrained backbone, its trainable parameter count (34.19M) is the lowest among all compared models, representing a 47% reduction relative to S2WAT and a 29% reduction relative to StyTr2. This demonstrates that the DEA with LoRA effectively reduces the number of parameters that need to be trained, reducing the number of trainable parameters and potentially lowering training-side optimization cost.

Although GSCNet's FLOPs (151.97) are lower than S2WAT's (299.43), its wall-clock latency is higher (48.96 ms vs. 38.41 ms) because the multi-scale depthwise convolutions in DEA are memory-bandwidth-bound and achieve lower GPU utilization than the large matrix multiplications that dominate attention-heavy architectures. We regard this as a deliberate quality–efficiency trade-off: the added inference cost brings the consistent artifact suppression shown in Sections 5.2 and 5.5, while the low trainable-parameter count keeps the training cost low. Relatedly, GSCNet's peak inference memory (993.07 MB) is higher than that of S2WAT (751.36 MB) and StyTr2 (250.89 MB), since the parallel multi-scale convolution branches in DEA retain additional intermediate feature maps.