CAP-VSTNet: Content Affinity Preserved Versatile Style Transfer

Linfeng Wen[1]   Chengying Gao[1] *   Changqing Zou[2,3]

[1]Sun Yat-Sen University   [2] State Key Lab of CAD&CG, Zhejiang University   [3] Zhejiang Lab

wenlf5@mail2.sysu.edu.cn mcsgcy@mail.sysu.edu.cn aaronzou1125@gmail.com

Figure 1. Our model produces more consistent stylization results with content affinity preserved for photorealistic image style transfer (*left*), and photorealistic video and artistic video style transfer (*right*). Animations can be found in the supplementary material.

Abstract

Content affinity loss including feature and pixel affinity is a main problem which leads to artifacts in photorealistic and video style transfer. This paper proposes a new framework named CAP-VSTNet, which consists of a new reversible residual network and an unbiased linear transform module, for versatile style transfer. This reversible residual network can not only preserve content affinity but not introduce redundant information as traditional reversible networks, and hence facilitate better stylization. Empowered by Matting Laplacian training loss which can address the pixel affinity loss problem led by the linear transform, the proposed framework is applicable and effective on versatile style transfer. Extensive experiments show that CAP-VSTNet can produce better qualitative and quantitative results in comparison with the state-of-the-art methods.

1. Introduction

Photorealistic style transfer aims to reproduce content image with the style from a reference image in a photore-

*Corresponding Author

alistic way. To be photorealism, the stylized image should preserve clear content detail and consistent stylization of the same semantic regions. Content affinity preservation, including feature and pixel affinity preservation [23,25,28], is the key to achieve both clear content detail and consistent stylization in the transfer.

The framework of a deep learning based photorealistic style transfer generally uses such an architecture: an encoder module extracting content and style features, followed by a transformation module to adjust features statistics, and finally a decoder module to invert stylized feature back to stylized image. Existing photorealistic methods typically employ pre-trained VGG [30] as encoder. Since the encoder is specifically designed to capture object-level information for the classification task, it inevitably results in content affinity loss. To reduce the artifacts, existing methods either use skip connection modules [2, 14, 40] or build a shallower network [8, 23, 39]. However, these strategies, limited by the image recovery bias, cannot achieve a perfect content affinity preservation on unseen images.

In this work, rather than use the traditional encoder-transformation-decoder architecture, we resort to a reversible framework [1] based solution called CAP-VSTNet, which consists of a specifically designed reversible residual

network followed by an unbiased linear transform module based on Cholesky decomposition [19] that performs style transfer in the feature space. The reversible network takes the advantages of the bijective transformation and can avoid content affinity information loss during forward and backward inference. However, directly using the reversible network cannot work well on our problem. This is because redundant information will accumulate greatly when the network channel increases. It will further lead to content affinity loss and noticeable artifacts as the transform module is sensitive to the redundant information. Inspired by knowledge distillation methods [8, 35], we improve the reversible network and employ a channel refinement module to avoid the redundant information accumulation. We achieve this by spreading the channel information into a patch of the spatial dimension. In addition, we also introduce cycle consistency loss in CAP-VSTNet to make the reversible network robust to small perturbations caused by numerical error.

Although the unbiased linear transform based on Cholesky decomposition [19] can preserve feature affinity, it cannot guarantee pixel affinity. Inspired by [25, 28], we introduce Matting Laplacian [22] loss to train the network and preserve pixel affinity. Matting Laplacian [22] may result in blurry images when it is used with another network like one with an encoder-decoder architecture. But it does not have this issue in CAP-VSTNet, since the bijective transformation of reversible network theoretically requires all information to be preserved.

CAP-VSTNet can be flexibly applied to versatile style transfer, including photorealistic and artistic image/video style transfer. We conduct extensive experiments to evaluate its performance. The results show it can produce better qualitative and quantitative results in comparison with the state-of-the-art image style transfer methods. We show that with minor loss function modifications, CAP-VSTNet can perform stable video style transfer and outperforms existing methods.

2. Related Work

2.1. Style Transfer

Gatys et al. [11] expose the powerful representation ability of deep neural networks and propose neural style transfer by matching the correlations of deep features. Feed-forward frameworks [17, 33, 38] are proposed to address the issue of computational cost. To achieve universal style transfer, transformation modules are proposed to adjust statistics of deep features, such as the mean and variance [15] and the inter-channel correlation [24].

Photorealistic style transfer requires that stylized image should be undistorted and consistently stylized. DPST [28] optimizes stylized image with regularization term computed on Matting Laplacian [22] to suppress distortion. PhotoWCT [25] proposes post-processing algorithm by using Matting Laplacian as affinity matrix to reduce artifacts. However, both of these methods may blur the stylized images instead of preserving the pixel affinity. The following works [2, 8, 40] mainly focus on preserving clear details and speeding up processing by designing skip connection module or shallower network. Content affinity preservation including feature and pixel affinity preservation remains an unsolved challenge.

Recently, versatile style transfer has received a lot of attention. Many approaches focus on exploring a general framework capable of performing artistic, photorealistic and video style transfer. Li et al. [23] propose a linear style transfer network and a spatial propagation network [27] for artistic and photorealistic style transfer, respectively. DSTN [14] introduces a unified architecture with domain-aware indicator to adaptively balance between artistic and photorealistic stylization. Chiu et al. [7] propose an optimization-based method to achieve fast artistic or photorealistic style transfer by simply adjusting the number of iterations. Chen et al. [6] extend contrastive learning to artistic image and video style transfer by considering internal-external statistics. Wu et al. [39] apply contrastive learning by incorporating neighbor-regulating scheme to preserve the coherence of the content source for artistic and photorealistic video style transfer. While achieving versatile style transfer, VGG-based networks suffer from inconsistent stylization due to content affinity loss. We show that preserving content affinity can improve image consistent stylization and video temporal consistency.

2.2. Reversible Network

Dinh et al. [9] first propose an estimator that learns a bijective transform between data and latent space, which can be seen as a perfect auto-encoder pair as it naturally satisfies reconstruction term of auto-encoder [4, 34]. Follow-up work by Dinh et al. [10] introduces new transformation that breaks the unit determinant of Jacobian to address volume-preserving mapping. Glow [20] proposes a simple type of generative flow building on the works by Dinh [9, 10]. Since each layer's activation of reversible network can be exactly reconstructed from the next layer's, RevNet [12] and Reformer [21] present reversible residual layers to address memory consumption during deep network training. i-RevNet [16] builds an invertible type of RevNet with invertible down-sampling module. i-ResNet [3] inverts residual mapping by using Banach fixed point theorem to address the restriction of reversible network architecture.

Recently, An et al. [1] apply flow-based model [20] to address the content leak problem for artistic style transfer. However, content affinity may not be preserved due to transformation module and redundant information, which leads
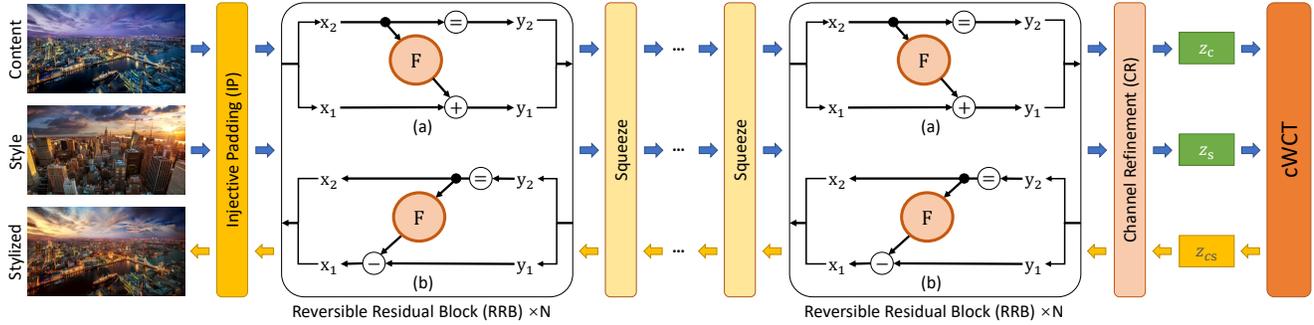
Figure 2. Architecture illustration of the proposed CAP-VSTNet. See Section 3 for details.
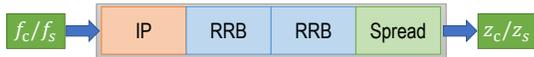
Figure 3. Structure of the adopted reversible residual blocks based channel refinement module (called CR-RRB for short). IP and RRB denotes injective padding module and reversible residual block, respectively.

to noticeable artifacts. The proposed method addresses this issue via a new reversible residual network enhanced by a channel refinement module and a Matting Laplacian loss based training.

3. Method

The architecture of CAP-VSTNet is shown in Figure 2. Given a content image and a style image, our framework first maps the input content/style images to latent space through the forward inference of network after an injective padding module which increases the input dimension by zero-padding along the channel dimension. The forward inference is performed through cascaded reversible residual blocks and spatial squeeze modules. After that a channel refinement module is then used to remove the channel redundant information in content/style image features for a more effective style transformation. Then a linear transform module cWCT is used to transfer the content representation to match the statistics of the style representation. Lastly the stylized representation is inversely mapped back to the stylized image through the backward inference.

3.1. Reversible Residual Network

In our network design, each reversible residual block performs a function of a pair of inputs $x_1, x_2 \rightarrow y_1, y_2$, which can be expressed in the form:

$$
\begin{aligned}
x_1, x_2 &= split(x), \\
y_1 &= x_1 + F(x_2), \quad y_2 = x_2.
\end{aligned}
\tag{1}
$$

Following Gome et al. [12], we use the channel-wise partitioning scheme that divides the input into two equal-sized

parts along the channel dimension. Since the reversible residual block processes only half of the channel dimension at one time, it is necessary to perturb the channel dimension of the feature maps. We find that channel shuffling is effective and efficient: $y = (y_2, y_1)$. Each block can be reversed by subtracting the residuals:

$$
\begin{aligned}
y_2, y_1 &= split(y), \\
x_2 = y_2, \quad x_1 &= y_1 - F(x_2).
\end{aligned}
\tag{2}
$$

Figure 2 (a) and (b) show the illustration of the forward and backward inference of reversible residual block, respectively. The residual function F is implemented by consecutive conv layers with kernel size 3. And each conv layer is followed by a relu layer, except for the last. We attain large receptive field by stacking multiple layers and blocks, in order to capture dense pairwise relations. We abandon the normalization layer as it poses a challenge to learn style representation. To capture large scale style information, the squeeze module is used to reduce the spatial information by a factor of 2 and increase the channel dimension by a factor of 4. We combine reversible residual blocks and squeeze modules to implement a multi-scale architecture.

3.2. Channel Refinement

The cascaded reversible residual block and squeeze module design in CAP-VSTNet leads to redundant information accumulation during forward inference as the squeeze module exponentially increases the channels. The redundant information will negatively affect the stylization. In [8, 35], channel compression is used to address the redundant information problem and facilitate better stylization. In our network design, we instead use a channel refinement module (CR) which is more suitable for the connected cascaded reversible residual blocks.

As illustrated in Figure 3, the CR module first uses an injective padding module increasing latent dimension to ensure that the input content/style image feature channel can be divisible by the target channel. Then, it uses patch-wise reversible residual blocks to integrate large-field informa-

| IP | MLP | Spread | MLP | Spread |

| IP | IR | IR | Spread |

| IP | RRB | RRB | Spread |

| (a) Input | (b) w/o CR | (c) CR-MLP | (d) CR-IR | (e) CR-RRB |
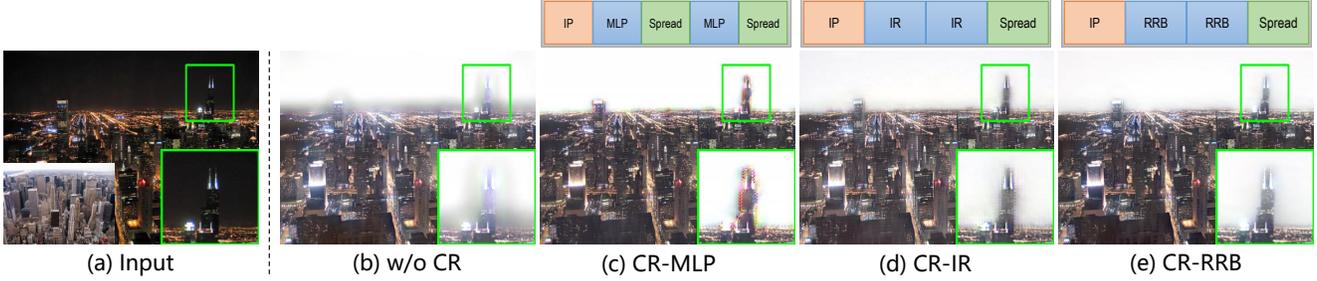
Figure 4. Alternative designs of the channel refinement module. (b) Artifacts are produced without any explicit redundant information removing. (c-d) Pointwise layers like fully connected layers or inverted residual layers cannot preserve good content affinity and result in aliasing artifacts. Please zoom in to see the details. The structures of the alternative designs are shown on top of the resulting images.

Method	AdaIN	WCT	LinearWCT	cWCT
Reversible	✓	✓		✓
Stability	✓		✓	✓
Learning-free		✓		✓
Time C=32	0.066	1.186	0.288	0.097
C=256	0.424	3.205	2.419	0.808

Table 1. Design choices of the linear transformation module. The adopted cWCT is reversible, stable, and learning-free. The execution time is evaluated on $C \times 512 \times 512$ feature maps for 100 times.

tion, after that it spreads the channel information into a patch of the spatial dimension. There are several alternative design choices for this CR module. MLP based pointwise layers can also be used for information distillation. Our preliminary experiments have found aliasing artifacts may appear when pointwise layers (e.g., fully connected layers or inverted residuals [29]) are employed (see the results produced by CR-MLP and CR-IR in Figure 4), but the adopted CR-RRB design does not have this issue.

3.3. Transformation Module

Existing photorealistic methods typically employ WCT [24] as transformation module, which contains whitening and coloring steps. Both of the above steps require the calculation of singular value decomposition (SVD). However, the gradient depends on the singular values σ by calculating $\frac{1}{min_{\sigma(i \neq j)}\sigma_i^2 - \sigma_j^2}$. If the covariance matrix of content (style) feature map $\Sigma_c = f_c f_c^T (\Sigma_s = f_s f_s^T)$ has the same singular values, or the distance between any two singular values is close to 0, the gradient becomes infinite. It will further cause the WCT module to fail and the model training to crash.

We use an unbiased linear transform based on Cholesky decomposition [19] to address this problem. The Cholesky decomposition is derivable with gradient depending on $\frac{1}{\sigma}$. It does not require that the two singular values are not equal

| Input | w/o L_{cyc} | w/ L_{cyc} |

Figure 5. Ablation results of cycle consistency loss. Numerical error may results in significant changes.

as SVD, thus is more stable. To avoid overflow, we can regularize it with an identity matrix: $\hat{\Sigma} = \Sigma + \epsilon I$. Another advantage of Cholesky decomposition is that its computational cost is much lower than that of SVD. Therefore, the adopted Cholesky decomposition based WCT (cWCT for short) is more stable and faster. We show the comparison of various linear transformation modules [15, 23, 24] in Table 1.

3.4. Training Loss

We train our network in an end-to-end manner with the integration of three types of losses:

$$L_{total} = L_s + \lambda_m L_m + \lambda_{cyc} L_{cyc}, \quad (3)$$

where L_m, L_s, and L_{cyc} denote Matting Laplacian loss, style loss, and cycle consistency loss, respectively. λ_m and λ_{cyc} are the weights corresponding to the losses.

The Matting Laplacian loss in our design can be formulated as:

$$L_m = \frac{1}{N} \sum_{c=1}^{3} V_c[I_{cs}]^T M V_c[I_{cs}], \quad (4)$$

where N denotes the number of image pixels, $V_c[I_{cs}]$ denotes the vectorization of the stylized image I_{cs} in channel c, and M denotes the Matting Laplacian matrix of the content image I_c.

Directly introducing Matting Laplacian loss in a network training could result in blurry images because Mat-
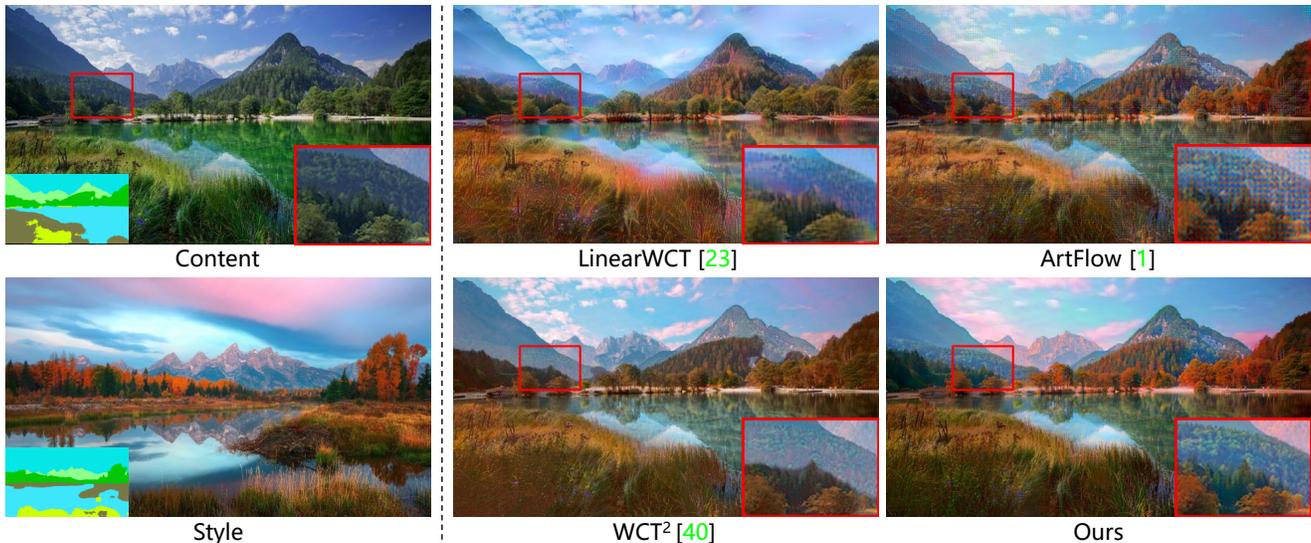
Figure 6. Visual comparison of content affinity preservation across various methods.

	Full	w/o RRB	w/o IP	w/o CR-RRB	w/ CR-MLP	w/o L_m&L_{cyc}	w/o L_{cyc}
SSIM↑	**0.650**	0.643	0.648	0.630	0.640	0.421	0.638
Gram loss↓	**0.750**	0.811	0.874	0.831	0.803	0.873	0.782

Table 2. Quantitative comparison of different design choices in terms of structure preservation (SSIM) and stylization effect (Gram loss).

ting Laplacian loss will force the network to smooth the image rather than preserve pixel affinity. Fortunately, introducing Matting Laplacian loss in our reversible network does not have the issue. It is because the bijective transformation in our reversible network requires all information to be preserved during forward and backward inference. The reversible network does not trick the loss by smoothing the image as it results in information loss. When performing linear transform, it depends on covariance matrix Σ_s. As the transformation of reversible network is deterministic, only a few style images with smooth texture may smooth the content structure. In this situation, it is reasonable to output a stylized image with the same smooth texture as we aim to transfer vivid style.

The style loss is formulated as:

$$L_s = \sum_{i=1}^{l} ||\mu(\phi_i(I_{cs})) - \mu(\phi_i(I_s))||$$
$$+ \sum_{i=1}^{l} ||\sigma(\phi_i(I_{cs})) - \sigma(\phi_i(I_s))||, \qquad (5)$$

where I_s denotes style image, ϕ_i denotes the i_{th} layer of the VGG-19 network (from $ReLu1\_1$ to $ReLu4\_1$), and μ and σ denote the mean and variance of the feature maps, respectively.

Since all modules are reversible, we should be able to cyclically reconstruct content image $\tilde{I}_C$ by transferring the style information of content image I_c to stylized image I_{cs}. However, the reversible network suffers from numerical error and may result in noticeable artifacts (Figure 5). Thus, we introduce the cycle consistency loss to improve the network robustness.

The cycle consistency loss is calculated with $L1$ distance:

$$L_{cyc} = ||\tilde{I}_C - I_C||_1. \qquad (6)$$

3.5. Video Style Transfer

Single-frame methods [23,39,40] show that applying image algorithms that operate on each video frame individually is possible. Since our framework preserves the affinity of input videos, which is naturally consistent and stable, the content of stylized video is also visually stable. To constrain the style of stylized video [13, 23], we have two strategies: adjust the style loss (Eq.5) with lower layers of VGG-19 network (from $ReLu1\_1$ to $ReLu3\_1$) or add the regularization [36] to Eq.3 and fine-tune the model. Both strategies can achieve good temporal consistency. We choose the latter one as it can produce slightly better stylization effect.
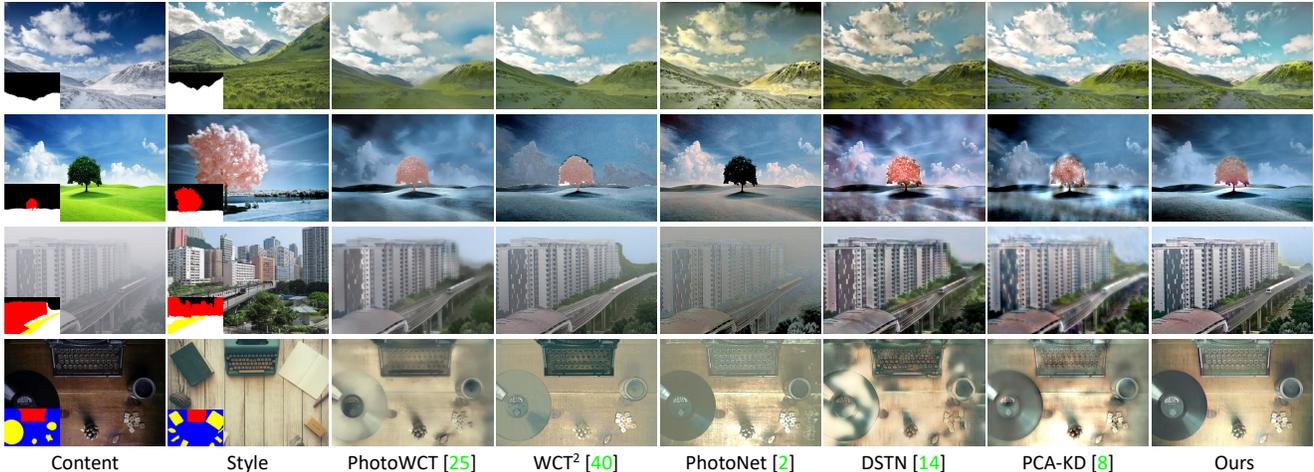
Figure 7. Visual comparisons of photorealistic image style transfer. All methods conduct style transfer with the assistance of masks, except PhotoNet which does not support masks.

Method	PhotoWCT [25]	WCT² [40]	PhotoNet [2]	DSTN [14]	PCA-KD [8]	Ours
SSIM↑	0.582	<u>0.644</u>	0.608	0.566	0.634	**0.650**
Gram loss↓	1.539	<u>0.796</u>	1.970	0.996	1.162	**0.750**
Time↓	16.88	0.32	0.19	0.92	**0.05**	<u>0.12</u>
Parameters	8.35M	10.12M	40.24M	103.45M	334K	4.09M

Table 3. Quantitative comparison of photorealistic style transfer methods. The execution time is evaluated on 1024×512 resolution.

4. Analysis

4.1. Content Affinity Preservation

To show the advantages of preserving feature and pixel affinity, we compare the stylization results with three types of methods. As shown in Figure 6, LinearWCT [23] applies linear transform to preserve feature affinity. However, the image details is unclear and the stylization is inconsistent as feature and pixel affinity could be damaged by VGG-base network. WCT² [40] aims to preserve spatial information rather than content affinity. While preserving clear details, it particularly relies on the precise masks, which otherwise produce noticeable seams. ArtFlow [1] uses the flow-based model to address content leak problem. However, it typically generates noticeable artifacts as linear transform and redundant information damage content affinity. Compared with other methods, ours model not only preserves clear details, but also achieves seamless style transfer.

4.2. Ablation Study

We conduct an ablation study to quantitatively evaluate how much each component (i.e., channel refinement components and training losses) affects the visual effects. Table 2 shows the ablation study results. When all the design components are used, the network can obtain the best re-

sults. Replacing residual block (RRB) with inverted residuals [29] degrades performance as the pointwise layer has smaller receptive field and damages content affinity. Removing injective padding (IP), the model fails to capture high-level content and style information from pixel image. Adding the channel refinement module (CR-RRB) helps remove redundant information for better content preservation and stylization effect. Implementing the channel refinement module with CR-MLP results in aliasing artifact, which degrades content affinity. Using VGG content loss (w/o $L_m \& L_{cyc}$) cannot guarantee pixel affinity due to the linear transform. With cycle consistency loss (L_{cyc}), the network achieves robustness to small perturbations.

5. Experiments

5.1. Implementation Details

We implement a three-scale architecture with 30 blocks and 2 squeeze modules. For photorealistic style transfer, we sample content and style images from MS COCO dataset [26] and randomly crop them to 256×256. We set the weight factors of loss function as: $\lambda_m = 1200$ and $\lambda_{cyc} = 10$. We train the network for 160,000 iterations using Adam optimizer with batch size of 2. The initial learning rate is set to 1e-4 and decays at 5e-5. For artistic

Figure 8. Comparisons of short-term temporal consistency on photorealistic video style transfer. The odd rows show the previous frame. The even rows show the temporal error heatmap.
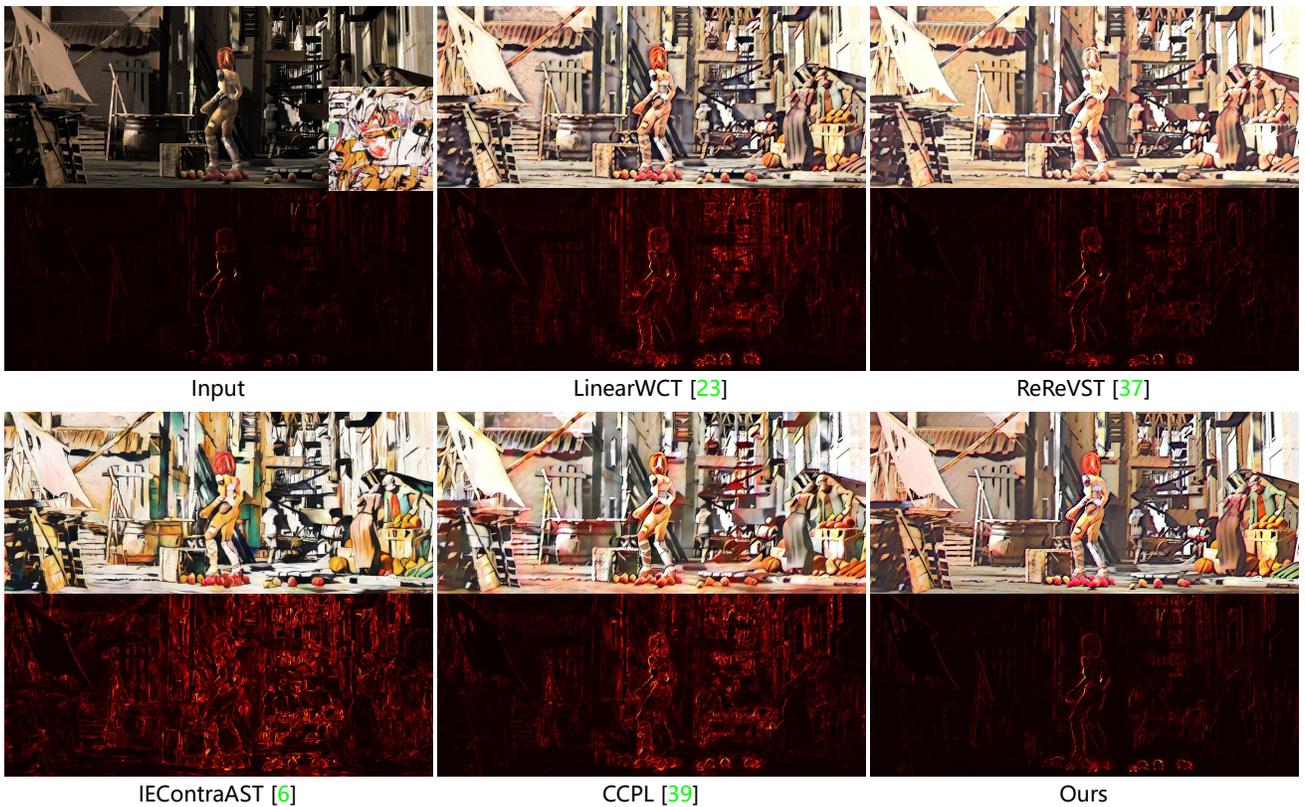
Figure 9. Comparisons of short-term temporal consistency on artistic video style transfer. The odd rows show the previous frame. The even rows show the temporal error heatmap.

style transfer, we set $\lambda_m = \lambda_{cyc} = 1$ to allow more variation of image pixel and sample style images from WikiArt dataset [18]. All the experiments are conducted on a single NVIDIA RTX 3090 GPU.

5.2. Photorealistic Image Style Transfer

Qualitative evaluation. Figure 7 shows the comparison of the stylization results with advanced photorealistic style transfer methods, including PhotoWCT [25], WCT^2 [40], PhotoNet [2], DSTN [14] and PCA-KD [8]. We can see that PhotoWCT usually generates blurry images with loss of details. Although WCT^2 faithfully preserves image spatial information, it produces noticeable seams. PhotoNet generates poor stylization effect due to discarding masks. DSTN stylizes images with noticeable artifacts and distorts image structure. PCA-KD is not able to produce consis-

Method	Gram loss↓	Temporal loss↓	
		i=1	i=10
WCT^2 [40]	0.665	0.040	0.108
CCPL [39]	0.527	0.069	0.132
Ours	**0.435**	**0.039**	**0.107**

Table 4. Quantitative comparison of photorealistic video style transfer methods. 'i' denotes frame interval.

Method	Gram loss↓	Temporal loss↓	
		i=1	i=10
LinearWCT [23]	0.473	0.117	0.237
ReReVST [37]	0.815	0.108	0.235
IEContraAST [6]	1.062	0.141	0.262
CCPL [39]	**0.371**	0.128	0.251
Ours	0.436	**0.104**	**0.228**

Table 5. Quantitative comparison of artistic video style transfer methods. 'i' denotes frame interval.

tent stylization. Compared with the existing methods, our method faithfully preserves image details and achieves better stylization effect. Besides, image stylization is consistent without artifacts, which greatly enhances photorealism.

Quantitative evaluation. Following previous works [14, 40], we use structural similarity (SSIM) to evaluate photorealism and Gram loss [11] to evaluate stylization effect. We use all pairs of content and style images with semantic segmentation masks provided by DPST [28] for quantitative evaluation. Table 3 shows the comparison of quantitative results. Our method not only preserves structure better, but also achieves stronger stylization effect. Since the reversible residual network naturally satisfies the reconstruction condition, we reduce network parameters and make it more lightweight than most of standard VGG-based networks. PCA-KD [8] applies knowledge distillation method to crate the lightweight model for ultra-resolution style transfer. We note that our model is also applicable for ultra-resolution (i.e., 4K resolution) and achieves better performance as well.

5.3. Video Style Transfer

Photorealistic video style transfer. We compare our method with state-of-the-art methods [39, 40]. To visualize video stability, we show the heatmap of temporal error between the consecutive frames in Figure 8. To quantitatively evaluate, we collect 20 pairs of video clips of multiple scenes and semantically related style images from the Internet. Following [37, 39], we adopt the temporal loss to measure temporal consistency. We use RAFT [32] to estimate the optical flow for short-term consistency (two adjacent frames) and long-term consistency (9 frames in be-

Figure 10. Limitation. Both our artistic and photorealistic models fail to transfer complex textures like milky way.

tween) evaluation. Table 4 shows that our framework performs well against the other methods.

Artistic video style transfer. Figure 9 shows the comparison with four advanced methods [6, 23, 37, 39]. To quantitatively evaluate, we use all the sequences of MPI Sintel dataset [5] and collect 20 artworks of various types to stylize each video. For short-term consistency, MPI Sintel provides ground truth optical flows. For long-term consistency, we use PWC-Net [31] to estimate the optical flow following [37, 39]. Table 5 shows that our framework achieves the best temporal consistency, thanks to the content affinity preservation. Our model also produces vivid stylization effect comparable to CCPL [39].

5.4. Limitation

Preserving content affinity helps to achieve consistent stylization. However, both our artistic and photorealistic models fail to capture complex texture and may generate artifacts (Figure 10). Generating realistic textures remains a challenge for style transfer and image generation tasks. Existing stylization methods typically build on small models (e.g., VGG). Since realistic texture requires much high-frequency details, an interesting direction is to investigate whether large models can solve this problem.

6. Conclusion

In this paper, we propose a new framework named CAP-VSTNet for versatile style transfer, which consists of a new effective reversible residual network and an unbiased linear transform. It can preserve two major content affinity: pixel and feature affinity with the introduction of Matting Laplacian training loss. We show that CAP-VSTNet achieves consistent and vivid stylization with clear details. CAP-VSTNet is also flexible for photorealistic and artistic video style transfer. Extensive experiments demonstrate the effectiveness and superiority of CAP-VSTNet in comparisons with state-of-the-art approaches.

Acknowledgement

References

[1] Jie An, Siyu Huang, Yibing Song, Dejing Dou, Wei Liu, and Jiebo Luo. Artflow: Unbiased image style transfer via reversible neural flows. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 862–871, 2021. 1, 2, 6

[2] Jie An, Haoyi Xiong, Jun Huan, and Jiebo Luo. Ultrafast photorealistic style transfer via neural architecture search. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10443–10450, 2020. 1, 2, 6, 7

[3] Jens Behrmann, Will Grathwohl, Ricky TQ Chen, David Duvenaud, and Jörn-Henrik Jacobsen. Invertible residual networks. In *International Conference on Machine Learning*, pages 573–582. PMLR, 2019. 2

[4] Yoshua Bengio, Yann LeCun, et al. Scaling learning algorithms towards ai. *Large-scale kernel machines*, 34(5):1–41, 2007. 2

[5] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *European conference on computer vision*, pages 611–625. Springer, 2012. 8

[6] Haibo Chen, Zhizhong Wang, Huiming Zhang, Zhiwen Zuo, Ailin Li, Wei Xing, Dongming Lu, et al. Artistic style transfer with internal-external learning and contrastive learning. *Advances in Neural Information Processing Systems*, 34:26561–26573, 2021. 2, 8

[7] Tai-Yin Chiu and Danna Gurari. Iterative feature transformation for fast and versatile universal style transfer. In *European Conference on Computer Vision*, pages 169–184. Springer, 2020. 2

[8] Tai-Yin Chiu and Danna Gurari. Pca-based knowledge distillation towards lightweight and content-style balanced photorealistic style transfer models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7844–7853, 2022. 1, 2, 3, 6, 7, 8

[9] Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014. 2

[10] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016. 2

[11] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016. 2, 8

[12] Aidan N Gomez, Mengye Ren, Raquel Urtasun, and Roger B Grosse. The reversible residual network: Backpropagation without storing activations. *Advances in neural information processing systems*, 30, 2017. 2, 3

[13] Agrim Gupta, Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Characterizing and improving stability in neural style transfer. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4067–4076, 2017. 5

[14] Kibeom Hong, Seogkyu Jeon, Huan Yang, Jianlong Fu, and Hyeran Byun. Domain-aware universal style transfer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14609–14617, 2021. 1, 2, 6, 7, 8

[15] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision*, pages 1501–1510, 2017. 2, 4

[16] Jörn-Henrik Jacobsen, Arnold Smeulders, and Edouard Oyallon. i-revnet: Deep invertible networks. *arXiv preprint arXiv:1802.07088*, 2018. 2

[17] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 2

[18] Sergey Karayev, Matthew Trentacoste, Helen Han, Aseem Agarwala, Trevor Darrell, Aaron Hertzmann, and Holger Winnemoeller. Recognizing image style. *arXiv preprint arXiv:1311.3715*, 2013. 7

[19] Agnan Kessy, Alex Lewin, and Korbinian Strimmer. Optimal whitening and decorrelation. *The American Statistician*, 72(4):309–314, 2018. 2, 4

[20] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018. 2

[21] Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*, 2020. 2

[22] Anat Levin, Dani Lischinski, and Yair Weiss. A closed-form solution to natural image matting. *IEEE transactions on pattern analysis and machine intelligence*, 30(2):228–242, 2007. 2

[23] Xueting Li, Sifei Liu, Jan Kautz, and Ming-Hsuan Yang. Learning linear transformations for fast image and video style transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3809–3817, 2019. 1, 2, 4, 5, 6, 8

[24] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Universal style transfer via feature transforms. *Advances in neural information processing systems*, 30, 2017. 2, 4

[25] Yijun Li, Ming-Yu Liu, Xueting Li, Ming-Hsuan Yang, and Jan Kautz. A closed-form solution to photorealistic image stylization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 453–468, 2018. 1, 2, 6, 7

[26] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 6

[27] Sifei Liu, Shalini De Mello, Jinwei Gu, Guangyu Zhong, Ming-Hsuan Yang, and Jan Kautz. Learning affinity via spatial propagation networks. *Advances in Neural Information Processing Systems*, 30, 2017. 2

[28] Fujun Luan, Sylvain Paris, Eli Shechtman, and Kavita Bala. Deep photo style transfer. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4990–4998, 2017. 1, 2, 8

[29] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the*

IEEE conference on computer vision and pattern recognition, pages 4510–4520, 2018. 4, 6

[30] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[31] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8934–8943, 2018. 8

[32] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pages 402–419. Springer, 2020. 8

[33] Dmitry Ulyanov, Vadim Lebedev, Andrea Vedaldi, and Victor S Lempitsky. Texture networks: Feed-forward synthesis of textures and stylized images. In *ICML*, volume 1, page 4, 2016. 2

[34] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, Pierre-Antoine Manzagol, and Léon Bottou. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(12), 2010. 2

[35] Huan Wang, Yijun Li, Yuehai Wang, Haoji Hu, and Ming-Hsuan Yang. Collaborative distillation for ultra-resolution universal style transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1860–1869, 2020. 2, 3

[36] Wenjing Wang, Jizheng Xu, Li Zhang, Yue Wang, and Jiaying Liu. Consistent video style transfer via compound regularization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12233–12240, 2020. 5

[37] Wenjing Wang, Shuai Yang, Jizheng Xu, and Jiaying Liu. Consistent video style transfer via relaxation and regularization. *IEEE Transactions on Image Processing*, 29:9125–9139, 2020. 8

[38] Xin Wang, Geoffrey Oxholm, Da Zhang, and Yuan-Fang Wang. Multimodal transfer: A hierarchical deep convolutional neural network for fast artistic style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5239–5247, 2017. 2

[39] Zijie Wu, Zhen Zhu, Junping Du, and Xiang Bai. Ccpl: Contrastive coherence preserving loss for versatile style transfer. 2022. 1, 2, 5, 8

[40] Jaejun Yoo, Youngjung Uh, Sanghyuk Chun, Byeongkyu Kang, and Jung-Woo Ha. Photorealistic style transfer via wavelet transforms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9036–9045, 2019. 1, 2, 5, 6, 7, 8