StylizedNeRF: Consistent 3D Scene Stylization as Stylized NeRF via 2D-3D Mutual Learning – *Supplemental Material* –

Yi-Hua Huang[1,2] **Yue He**[1,2] **Yu-Jie Yuan**[1,2] **Yu-Kun Lai**[3] **Lin Gao**[1,2*]
[1]Beijing Key Laboratory of Mobile Computing and Pervasive Device,
Institute of Computing Technology, Chinese Academy of Sciences
[2]School of Computer and Control Engineering, University of Chinese Academy of Sciences
[3]School of Computer Science & Informatics, Cardiff University
{huangyihua20g, heyue19s, yuanyujie, gaolin}@ict.ac.cn LaiY4@cardiff.ac.uk

1. Overview

This supplementary document provides some implementation details and further results that accompany the paper.

- Section 2 introduces the implementation details of our approach, including the network architecture and the proposed mutual learning algorithm.
- Section 3 provides additional results, including more visualizations.
- Section 4 discusses the limitations of our method and the future research topics.

2. Implementation Details

2.1. Network Architecture

The encoder and decoder of the pre-trained VAE are both consist of 4 fully connected layers and ReLU activation functions. The input of VAE is the style features of style images, defined as the concatenated vector of the mean and variance of feature maps extracted by VGG19 [4] along the spatial dimensions. The dimension of embedded latent space is 32, which is the length of learnable codes l. The input coordinate x to the stylized NeRF is transformed by the Fourier output $\gamma(x)$ with 10 bands and concatenated with the input latent code l. The style module of our stylized NeRF uses 8 fully connected layers and ReLU activation functions. The 4-th layer of the MLP is a residual layer, whose input is concatenated with the $\gamma(x)$ and l to avoid the gradient vanishing problem.

2.2. Mutual Learning Algorithm

Our mutual learning scheme aims to leverage the stylization capability of 2D method and the consistency of NeRF. To learn the consistency from NeRF, the applied 2D stylization method learns with the consistency loss determined by the warping error, which is based on the geometry prior from NeRF, as a pre-training process. Then the stylized NeRF and 2D stylization method are optimized simultaneously with a proposed mimic loss to align the outputs of each other. We demonstrate the details of our mutual learning process in Alg. 1.

Algorithm 1 Mutual Learning Process

Input $\{\mathcal{I}_i\}$: scene views $\{\mathcal{S}_j\}$: style images
$\mathcal{V}$: pretrained VAE $\mathcal{C}_c$: CNN-based decoder
σ: opacity function $\mathcal{G}$: pretrained VGG
Output c_s: style module
Procedure:
Extract style features $\{s_j\}$ from $\{\mathcal{S}_j\}$ with $\mathcal{G}$
$\{\mathcal{N}(\mu_j, \sigma_j)\} \leftarrow \{\mathcal{V}(s_j)\}$
$\{l_{i,j}\} \leftarrow$ sample $l_{i,j}$ from $\mathcal{N}(\mu_j, \sigma_j)$
Pretrain $\mathcal{D}$ with style, content and consistency loss
Initialize style module c_s
repeat
  $i, j \leftarrow$ random sampling
  512×512 patch $P_i \leftarrow$ sampling on $\mathcal{I}_i$
  Rays $r_{i,t} \leftarrow$ sampling on P_i
  Decoder forward: $\mathcal{C}_a(P_i, \mathcal{S}_j)$
  Points $r_{i,t,k} \leftarrow$ sampling on rays
  NeRF forward: $c_s(r_{i,t,k}, l_{i,j}), \sigma_{i,t,k}$
  $\mathcal{C}_n(p_{i,t}, l_{i,j}) \leftarrow$ composition of $c_s(r_{i,t,k}, l_{i,j})$ and $\sigma_{i,t,k}$
  Optimize C_s, $\mathcal{C}_n$ and $\{l_{i,j}\}$ with the objective functions: L_N, L_C
until Max Iteration
Return c_s

3. Additional Results

3.1. Conditional Stylization

The conditional probability modeling of learnable latent codes enables our stylized NeRF to stylize the scene with

*Corresponding Author is Lin Gao (gaolin@ict.ac.cn).

Figure 1. **Conditional stylization.** Conditional stylization with a stylized NeRF for the Playground in the T&T dataset [2].

Figure 2. **Conditional stylization.** Conditional stylization with a stylized NeRF for the Truck in the T&T dataset [2].

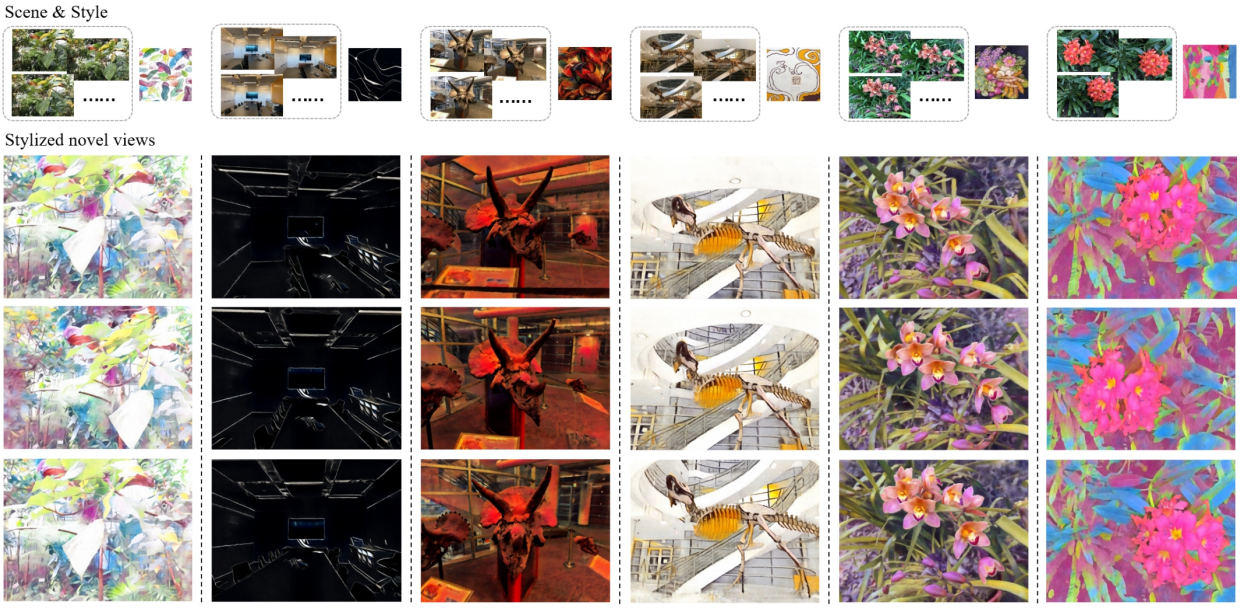

Figure 3. **Stylization results on the LLFF dataset.** We qualitatively test our method on the LLFF dataset to evaluate the stylization quality.

different styles by controlling the input latent codes. We evaluate the conditional stylization capability of our stylized NeRF on the Truck and Playground scenes in the T&T dataset [2] in Fig. 1 and Fig. 2. During the inference time, the input latent code is chosen as the mean vector μ of the distribution $\mathcal{N}(\mu, \sigma)$, which is encoded by the VAE from the style features. The conditional stylization capability enables our model to stylize 3D scenes with different styles.

3.2. Results on the LLFF dataset

We present more stylization results on the LLFF dataset [3]. We demonstrate that our approach can handle diversity of styles as shown in Fig. 3.

4. Limitations

Compared with 2D stylization methods, our method is inherently consistent in geometry. However, for style images whose texture has strong direction regularity, the mutual CNN decoder may generate stylized views with great ambiguity, which makes the style module of NeRF fail to learn the texture of the given style and could even cause blur. Fig. 4 shows the stylized results from ours and mutually trained CNN-based network. The styles with certain patterns full of directional lines will end up in inconsistency in 3D space, which can be captured by the CNN decoder but missed by our NeRF-based method. How to stylize 3D scenes with regular texture style will be our main topics for future research.

Our NeRF-based approach takes around 10 hours for training and around 2 minutes for rendering an image on an RTX 2080Ti GPU. Thus it is slower than point-cloud based methods like LSNV [1], which claims that it could stylize frames in near-real-time (17 fps). However, in combination with some recent work on NeRF acceleration, our inference speed may be boosted in the future.

Figure 4. Comparison of ours results and outputs of mutually trained decoder. Our method fails to capture 2D texture patterns inconsistent in 3D space.

References

[1] Hsin-Ping Huang, Hung-Yu Tseng, Saurabh Saini, Maneesh Singh, and Ming-Hsuan Yang. Learning to stylize novel views. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 13869–13878, 2021. 3

[2] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (TOG)*, 36(4):1–13, 2017. 2, 3

[3] Ben Mildenhall, Pratul P Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and

Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*, 38(4):1–14, 2019. 3

[4] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 1