Unified Implicit Neural Stylization
(Supplementary Material)

Zhiwen Fan[1,⋆], Yifan Jiang[1,⋆], Peihao Wang[1,⋆],
Xinyu Gong[1], Dejia Xu[1], Zhangyang Wang[1]

[1]The University of Texas at Austin
{zhiwenfan,yifanjiang97,peihaowang,xinyu.gong,dejia,atlaswang}@utexas.edu

1 Introduction

We provide the implementation details of our Style Implicit Module(SIM), Content Implicit Module (CIM), and Amalgamation Module (AM). More detailed comparisons by applying our INS on NeRF [5] with Style3D [1], single-image-based style transfer methods [3] and video-based methods [8, 2] are provided due to the space limitation in the main draft. A video demonstrates the conditional style interpolation and detailed comparisons are also included in our project page.

2 Implementation Details

2.1 Additional Details in Implementation INS on NeRF

For INS on SIREN [7] and SDF [9], we implement INS on the top of their published codes. For INS on NeRF [5], we re-implement the original method based on Pytorch library [6]. In the training stage of INS+NeRF, we first randomly select 4,096 rays in each GPU to train the Content Implicit Module (CIM), then a patch of size 72×72 is selected to generate 5,184 rays when joint training all modules with Sampling Stride set as 4. We use 100 training views on the NeRF-Synthetic dataset [5] and test all the methods with 200 testing views. While training on LLFF dataset [4], we adopt 35 training viewpoints and generate 120 new viewpoints for rendering. To accelerate the training and inference, we down-sample the NeRF-Synthetic dataset by 1/2 in all methods. We conduct comparisons with Style3D [1] on NeRF-synthetic and LLFF datasets with their original code and re-train Style3D using the same style image number with our method for fair comparisons.

2.2 Module Details

As presented in Figure 2 of the main draft, the Style Implicit Module (SIM) inputs with one-hot style embeddings in training stage, generates the 256-dimensional style latent code using the 4 layers of MLPs. The dimension of style embeddings

is determined by the style number, the embedding will be the 5-dimensional one-hot vector to specify each style image if we input with 5 different style images. Obtaining the 256 hidden dimensions latent style code, we will concatenate the style code with the output of Content Implicit Module(CIM), forming the input of Amalgamation Module (AM). Therefore, the input of the first layer of AM is 256+256 dimension vector and will be compressed to 256 dimensions. The framework of CIM is based on NeRF [5] which consists of 8-layer MLPs with 256 hidden dimensions. There are four layers of MLPs in both SIM and AM. The illustration of the details of *Sampling-Stride Ray Sampling* can be found in Figure 1.

3 Qualitative Evaluations

We provide more visualized results of our INS on SDF in Figure 5 and INS on NeRF with Style3D [1], single-image-based style transfer methods [3] and video-based methods [8, 2].

Fig. 1. Illustration of the Sampling Stride (SS) strategy in the ray sampling stage. With a given sampling stride larger than 1, we can approach a larger receptive field without sacrificing additional computational cost.

Qualitative Results with Other Methods. We present the qualitative evaluations in Figure 2, Figure 3 and Figure 4, where several baseline methods are included for comparisons. We can see Style3D [1] and video-based method MCCNet [2] ReReVST [8] generate consistent textures in different test viewpoints, but the results do not capture the target styles very well. Although image-based method [3] produces desired styles on image, it failed to produce consistent results in different views and not able to attach target textures to object surfaces(see the last row of Figure 4). A more detailed comparisons using all 200 testing viewpoints can be found in our project page. Different from our pure implicit-based representation, Style3D [1] requires CNNs and a hypernetwork to generate the model weights, which requires a much large storage space than ours (125MB vs. 14MB).

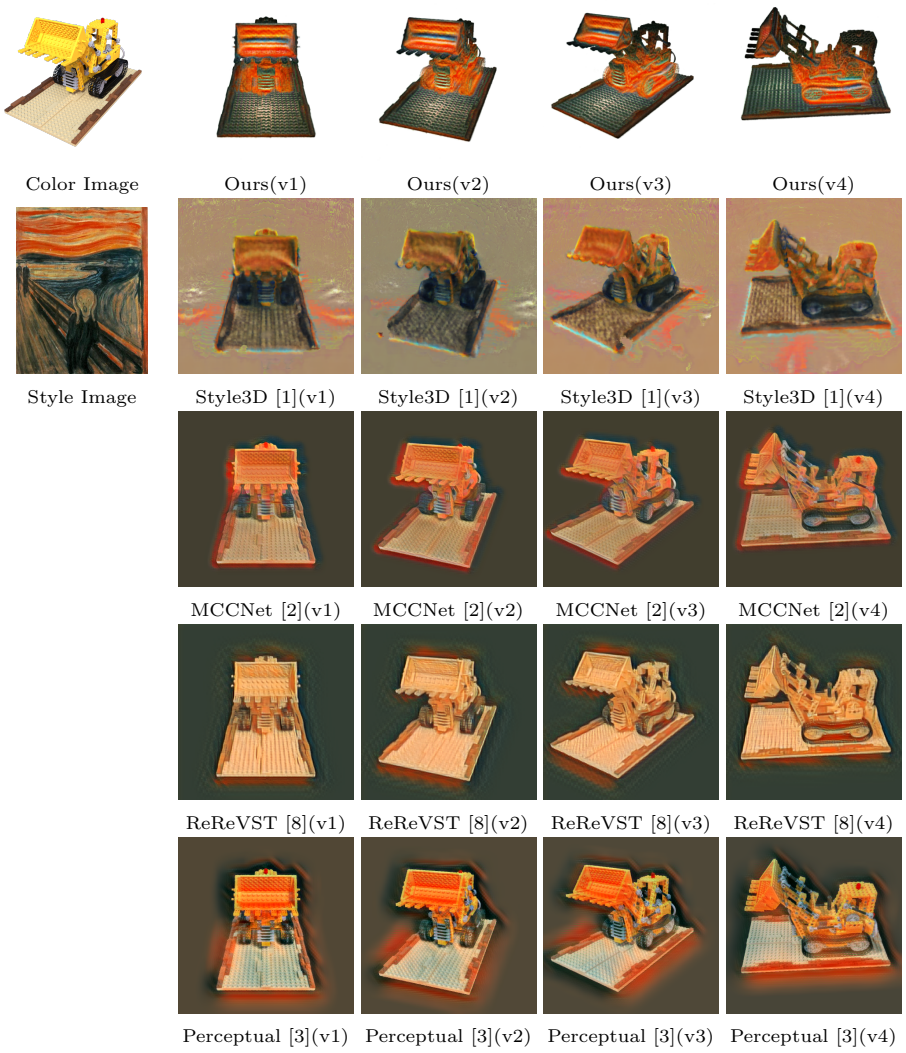

Fig. 2. Due to the limited space in the main draft, we demonstrate the novel view synthesis visualizations with more viewpoints (Scene: Lego, view number: 4). Our method produces stylizations with better view consistency than all other methods and can capture the target textures.

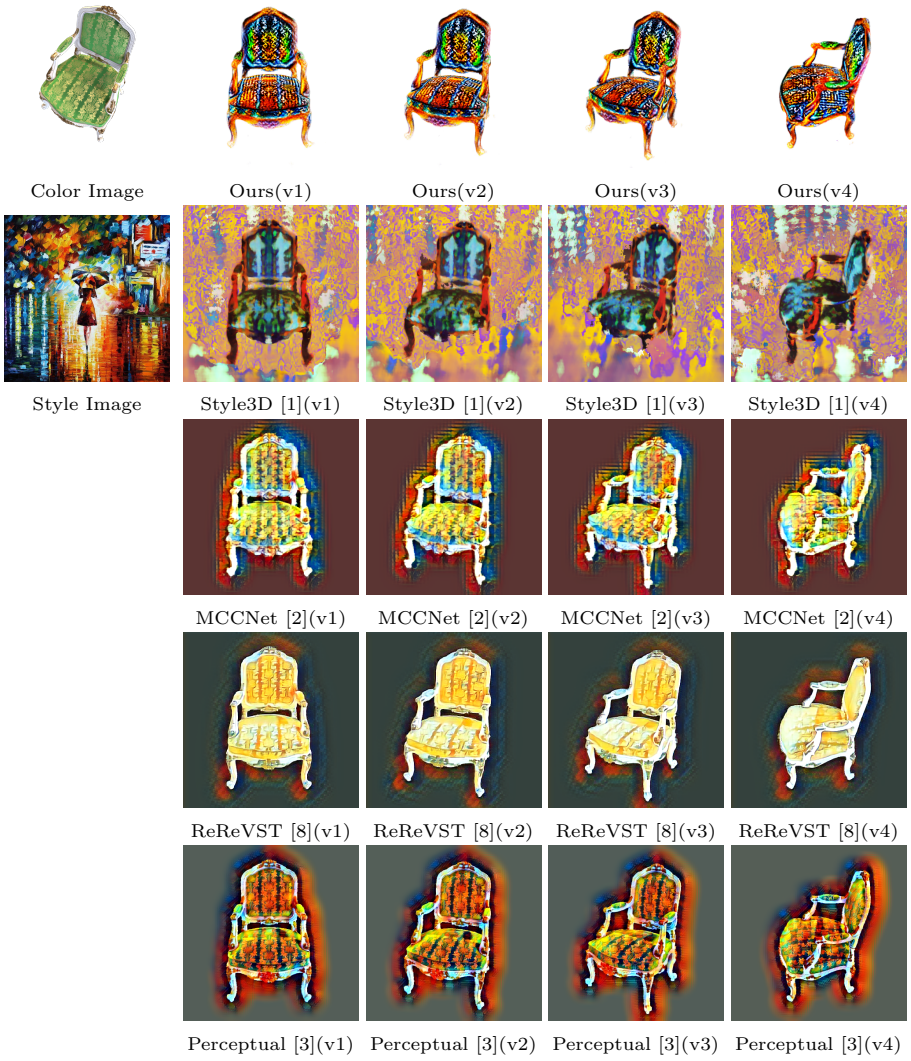

Fig. 3. We demonstrate the novel view synthesis visualizations with more viewpoints (Scene: Chair, view number: 4). Our method produces stylizations with better view consistency than all other methods and can capture the target textures.

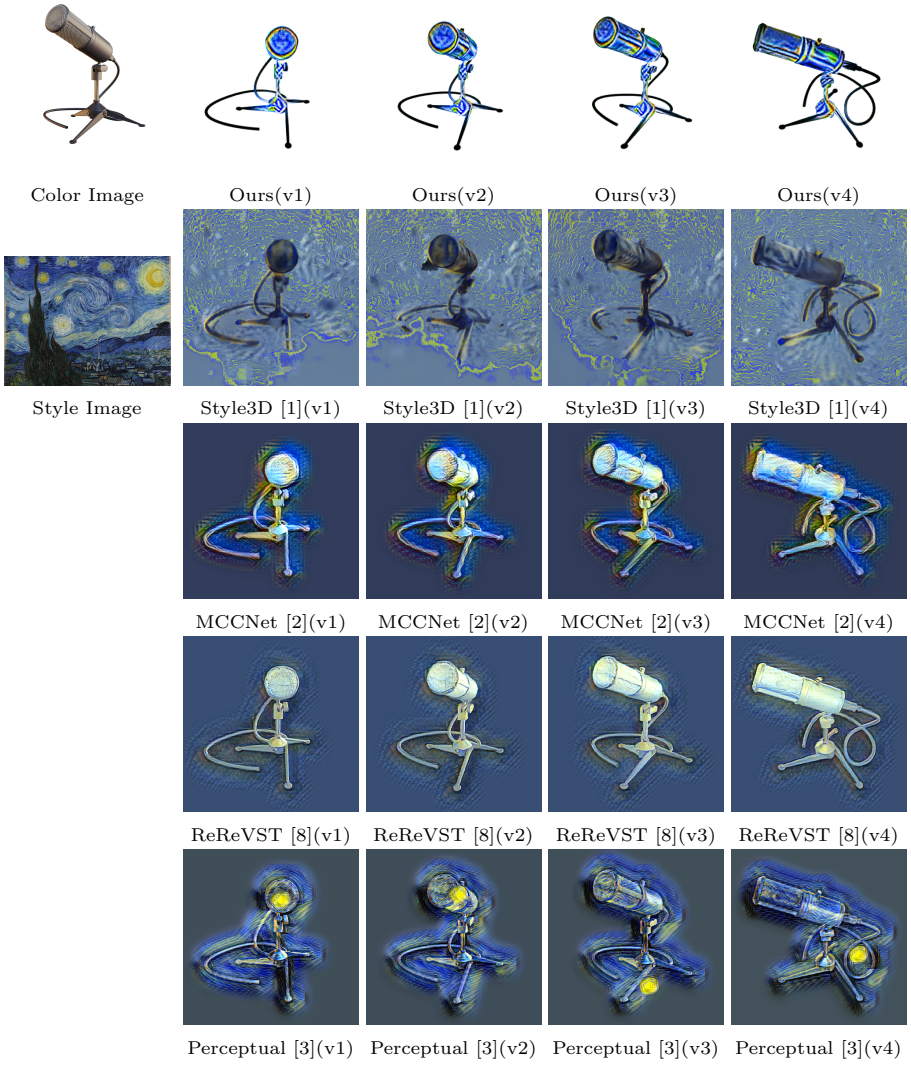

Fig. 4. We demonstrate the novel view synthesis visualizations with more viewpoints (Scene: Mic, view number: 4). Our method produces stylizations with better view consistency than all other methods and can capture the target textures.

Fig. 5. Additional visualization results of applying implicit neural stylization framework upon Signed Distance Function. Given multi-view color image and style image, IDR [9] can learn the style statistics for the disentangled geometry and appearance.

References

1. Chiang, P.Z., Tsai, M.S., Tseng, H.Y., Lai, W.S., Chiu, W.C.: Stylizing 3d scene via implicit representation and hypernetwork. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1475–1484 (2022)
2. Deng, Y., Tang, F., Dong, W., Huang, H., Ma, C., Xu, C.: Arbitrary video style transfer via multi-channel correlation. arXiv preprint arXiv:2009.08003 (2020)
3. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European conference on computer vision. pp. 694–711. Springer (2016)
4. Mildenhall, B., Srinivasan, P.P., Ortiz-Cayon, R., Kalantari, N.K., Ramamoorthi, R., Ng, R., Kar, A.: Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. ACM Transactions on Graphics (TOG) **38**(4), 1–14 (2019)
5. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In: European conference on computer vision. pp. 405–421. Springer (2020)
6. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch (2017)
7. Sitzmann, V., Martel, J., Bergman, A., Lindell, D., Wetzstein, G.: Implicit neural representations with periodic activation functions. Advances in Neural Information Processing Systems **33**, 7462–7473 (2020)
8. Wang, W., Yang, S., Xu, J., Liu, J.: Consistent video style transfer via relaxation and regularization. IEEE Transactions on Image Processing **29**, 9125–9139 (2020)
9. Yariv, L., Kasten, Y., Moran, D., Galun, M., Atzmon, M., Ronen, B., Lipman, Y.: Multiview neural surface reconstruction by disentangling geometry and appearance. Advances in Neural Information Processing Systems **33**, 2492–2502 (2020)