Creating 3D Asset Variations Through 2D Style Transfer and Generated Texture Maps

I. Nikolov[1]

[1]Computer Graphics Group, Faculty of Architecture, Design and Media Technology, Aalborg University, Denmark

Figure 1: *Results from the proposed 3D asset variation creation pipeline, using a combination of an albedo transformed from a style transfer network and generated texture maps*

Abstract

Generating 3D object variations through style transfer models applied to their textures is an easy way for creating content for games and XR applications. Most workflows focus on either generating albedo textures only without changing the underlying surface and not touching the underlying surface or transforming the 3D object directly, which is computationally and resource-heavy. In this paper, we present an initial exploration of an in-between solution that aims to combine the style transfer of albedo textures with the generation of additional maps, such as normal, displacement, and roughness. The results show that the pipeline can generate variations based on different styles, which would enable the addition of smaller 3D-style surface features to objects without transforming their meshes. The project code and generated textures will be available at https://github.com/IvanNik17/3D-Assets-From-2D-Style-Transfer.

CCS Concepts

• ***Computing methodologies*** → ***Appearance and texture representations;*** *Reconstruction; Texturing;*

1. Introduction

With the development of more robust style transfer models [LLKY19, WLW*20, KKP*22], procedural 2D content generation has become increasingly open to a larger audience. The work by Mishra et al. [MG22] has shown that style transfer can be used for creating 3D asset variations from a set of images, while the research by Yin et al. [YGS*21] has demonstrated that the 3D objects themselves can be stylized and modified based on features from a 3D object input. Our proposed pipeline aims to investigate an in-between solution, where a style of images can be transferred to a texture and then that texture used to generate additional normal, displacement, and roughness maps, which in turn will add additional stylized details to the 3D object, without the need to modify its geometry. We test out a number of style transfer networks and choose Neural Neighbor Style Transfer (NNST) [KKP*22], as it preserves the finer texture features the best while introducing style variations. To transfer this style at least partially on the 3D object we use three U-Net networks [RFB15] to generate normal, displacement, and roughness maps from the style transferred output texture.

2. Method Overview

For implementing the pipeline we use the 3D objects provided by Nikolov et al. [NM20], as they display varied surface characteristics, shapes, and sizes (Figure 1). We generate textures of size 1024x1024 pixels for easier processing and decimate and smooth the meshes to $40k$ vertices, as the smaller details will be provided by the generated 2D maps.

For the style transfer, we test five widely used networks [HB17, GLK*17, LLKY19, WLW*20, KKP*22]. We set their respective content preservation hyperparameters to between $0.5-0.7$, depending on the authors' recommendations. We compare their outputs by calculating the Structural Similarity Index Measure (SSIM) between the style-transferred results and the input images. A higher

DOI: 10.2312/egp.20231021

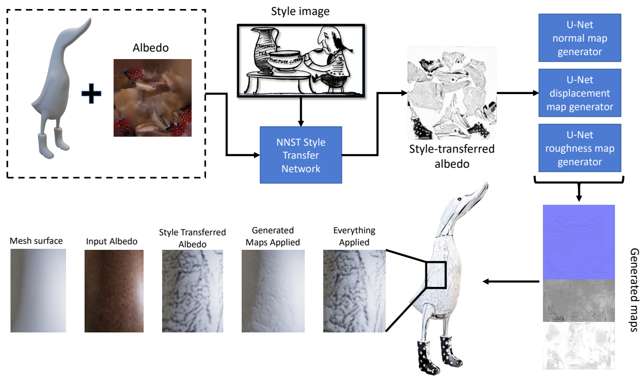

Figure 2: *Overview of the proposed pipeline. The albedo texture from a 3D model and a style image are input to the NNST model. The style-transferred texture is then input to the U-Net models, which generate the normal, displacement, and roughness maps. Examples of the mesh surface, base albedo, style-transferred albedo, and generated maps are also visualized.*

SSIM value would indicate a higher performance of content preservation. The averaged SSIM results are as follows: [HB17] - 0.353, [GLK*17] - 0.283, [LLKY19] - 0.402, [WLW*20] - 0.386 and [KKP*22] - **0.424**. This shows that the NNST model provides the best content preservation out of the tested set. For generating the stylized outputs for the NNST we use 200 iterations and a learning rate of $2e-3$, together with the content preservation value of 0.7. For creating the normal, displacement, and roughness maps we use a U-Net [RFB15] architecture based on the work presented by C. Chou [Cho21]. A U-Net model is trained for each of the three maps on $1.4k$ images. For training the U-Net models, 100 epochs are used, with a learning rate of $1e-4$ and batches of 8, together with an L2 squared norm loss. The proposed pipeline is shown in Figure 2.

3. Results

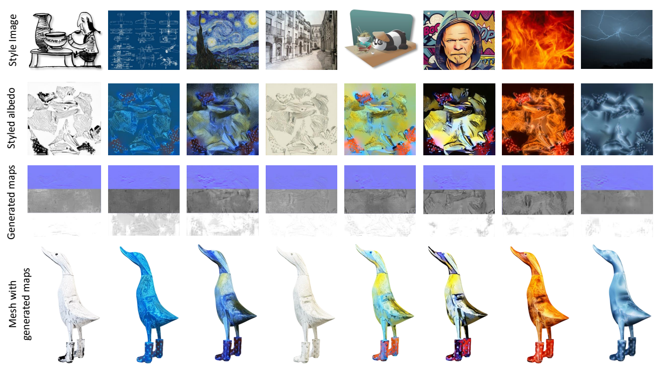

Figure 3: *Results for 8 different style images on the duck statue object. The style images were chosen to represent a mix of different features, to show how the resultant albedo textures and how generated maps change depending on the style.*

To test the proposed pipeline we choose 8 different style images, representing a mix of different styles. In Figure 3 we show the results from one of the 3D objects together with the different style images. We can see that in some cases the style transfer model makes colors bleed out and shift, which is a problem that is mentioned in the NNST paper. We can also see that some of the normal maps become over-smoothed and noisy, because of a lack of features in the input styled images, while some of the illumination spots from the real texture are propagated through the style transfer and become visual artifacts in the displacement maps. Even with these problems, the combination of the style-transferred textures and generated maps adds visual and surface details to the 3D model.

4. Conclusion

In this paper, we presented an initial exploration of the idea of introducing thematic variation in 3D models through the use of a style transfer network, together with U-Net-generated normal, displacement, and roughness maps. The initial results demonstrate some problems like visual artifacts and over-smoothing of the generated maps, but also show the possibilities of the proposed workflow, for quickly creating asset variations for games and XR applications. We plan to extend this pipeline by substituting the U-Net with a GAN or transformer network for better map generation, as well as look into generating ambient occlusion and metal maps.

References

[Cho21] Chou Y.-C.: Normal map generator. https://github.com/yuchen071/Normal-map-generator, 2021. Accessed: 2023-01-25. 2

[GLK*17] Ghiasi G., Lee H., Kudlur M., Dumoulin V., Shlens J.: Exploring the structure of a real-time, arbitrary neural artistic stylization network. *arXiv preprint arXiv:1705.06830* (2017). 1, 2

[HB17] Huang X., Belongie S.: Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision* (2017), pp. 1501–1510. 1, 2

[KKP*22] Kolkin N., Kucera M., Paris S., Sykora D., Shechtman E., Shakhnarovich G.: Neural neighbor style transfer. *arXiv e-prints* (2022), arXiv–2203. 1, 2

[LLKY19] Li X., Liu S., Kautz J., Yang M.-H.: Learning linear transformations for fast image and video style transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019), pp. 3809–3817. 1, 2

[MG22] Mishra S., Granskog J.: Clip-based neural neighbor style transfer for 3d assets. *arXiv preprint arXiv:2208.04370* (2022). 1

[NM20] Nikolov I., Madsen C.: Ggg-benchmarksfm- secondary: Secondary dataset for benchmarking close-range sfm software performance captured with different cameras. http://dx.doi.org/0.17632/t4d8mv3fxt.3, 2020. 1

[RFB15] Ronneberger O., Fischer P., Brox T.: U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (2015), Springer, pp. 234–241. 1, 2

[WLW*20] Wang H., Li Y., Wang Y., Hu H., Yang M.-H.: Collaborative distillation for ultra-resolution universal style transfer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2020), pp. 1860–1869. 1, 2

[YGS*21] Yin K., Gao J., Shugrina M., Khamis S., Fidler S.: 3dstylenet: Creating 3d shapes with geometric and texture style variations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2021), pp. 12456–12465. 1