3D Face Style Transfer with a Hybrid Solution of NeRF and Mesh Rasterization

Jianwei Feng
Amazon
jianwef@amazon.com

Prateek Singhal
Amazon
prtksngh@amazon.com

Figure 1. Given a set of multi-view input images of a human face (a), our approach reconstructs a 3D human face, transfers the style of a style image (b) to it and generates 3D consistent stylized novel views of the face (c).

Abstract

Style transfer for human face has been widely researched in recent years. Majority of the existing approaches work in 2D image domain and have 3D inconsistency issue when applied on different viewpoints of the same face. In this paper, we tackle the problem of 3D face style transfer which aims at generating stylized novel views of a 3D human face with multi-view consistency. We propose to use a neural radiance field (NeRF) to represent 3D human face and combine it with 2D style transfer to stylize the 3D face. We find that directly training a NeRF on stylized images from 2D style transfer brings in 3D inconsistency issue and causes blurriness. On the other hand, training a NeRF jointly with 2D style transfer objectives shows poor convergence due to the identity and head pose gap between style image and content image. It also poses challenge in training time and memory due to the need of volume rendering for full image to apply style transfer loss functions. We therefore propose a hybrid framework of NeRF and mesh rasterization to combine the benefits of high fidelity geometry reconstruction of NeRF and fast rendering speed of mesh. Our framework consists of three stages: 1. Training a NeRF model on input face images to learn the 3D geometry; 2. Extracting a

mesh from the trained NeRF model and optimizing it with style transfer objectives via differentiable rasterization; 3. Training a new color network in NeRF conditioned on a style embedding to enable arbitrary style transfer to the 3D face. Experiment results show that our approach generates high quality face style transfer with great 3D consistency, while also enabling a flexible style control.

1. Introduction

Style transfer for human face has been a popular research area in recent years. It has various applications in animations, advertising and gaming industry. Existing style transfer approaches for human face mainly focus on 2D image domain, where the input of the system is generally a style image and a content image, and the output is a stylized image which preserves the identity of the content image while having the style of the style image. The approaches for 2D face style transfer are usually achieved by 2D convolutional neural networks and pose 3D inconsistency issue when applied on a video or multi view images of the same face, which constraints usage of these 2D style transfer approaches in movies, animations or gaming for a consistent visual experience.

Several recent studies on 3D style transfer leverage NeRF to stylize a 3D scene. They generally supervise NeRF training with style transfer objectives applied on images rendered from NeRF, which introduces training time and memory challenge due to volume rendering on large number of pixels to form the full image needed to compute style transfer losses. Stylizing-3D-Scene [5] proposed a hyper network which was conditioned on style embedding of a style image and transferred style information to the color network of NeRF. They applied style transfer losses on small image patches (32x32) to avoid issues in training time and memory. UPST-NeRF [4] also utilized a hyper network and trained on small image patches. Training with small image patches has difficulty in capturing global semantic information and leads to a loss in style transfer quality. ARF [39] proposed a nearest neighbor-based Gram matrix loss for style transfer and deferred gradient descend to optimize on full image instead of image patch. However, deferred gradient descend significantly slows down the training process as it doesn't reduce the computation needed for volume rendering full resolution image.

To reduce training time and memory of NeRF, recent work [7,43] proposed to only sample points near object surface for volume rendering. In this paper, we take it one step forward and propose to use just one single surface intersection point to render, in which case the volume rendering falls back to its simplest form and becomes equivalent to rendering a mesh extracted from NeRF. Compared to volume rendering, mesh rasterization is faster and consumes less GPU memory. We then propose a three stage approach for 3D face style transfer, where we apply different 3D representation and rendering techniques in different stages to optimize for different loss objectives in consideration of their computation needs. In the first stage, we train a NeRF model to reconstruct 3D geometry from input face images, optimized by an RGB loss applied on a batch of randomly sampled pixels through volume rendering. In the second stage, we extract a mesh from the trained NeRF model, and stylize the mesh color from a style image. The mesh color is optimized by style transfer objectives applied on full image rendered from differentiable mesh rasterization [15]. We generate 200 stylized meshes from 200 style images in a training dataset. In the third stage, we fix the geometry network weight of NeRF, and train a hyper network to predict the color network weight from a style image, to generalize for arbitrary style transfer. During each training iteration, we randomly sample a style image and its corresponding stylized mesh, and renders a full image through mesh rasterization. The hyper network is then optimized by an RGB loss between a random batch of predicted pixels from NeRF's volume rendering, and corresponding pixels from mesh rendered image. With the combination of NeRF and mesh rasterization, we are able to do 3D face style transfer at original resolution of up to 2K.

During mesh optimization, we observe that using raw style image for style transfer objectives usually leads to poor convergence due to the large difference in identity and head pose with the content images rendered at different view points. We therefore propose to generate pair data of stylized images with similar head pose and identity by applying a 2D style transfer model [38] on content images randomly rendered at different head pose angle. Mesh optimization with pair data shows better style transfer quality on the mesh.

To summarize, our contributions are:

- We propose a novel three stage approach which achieves arbitrary 3D face style transfer with good style transfer quality and 3D consistency.
- We combine NeRF and mesh rasterization to optimize for different loss objectives which enables 3D face style transfer on original image resolution of up to 2K at a reasonable training cost.
- We propose to generate pair data of stylized images to fill the gap of head pose and identity. Optimizing mesh colors with pair data shows better style transfer quality.

2. Related Works

2.1. Novel View Synthesis

Novel View Synthesis aims at synthesizing image at arbitrary view point from a set of source images. Tra-

ditional approaches apply explicit 3D representations to model 3D scenes, such as 3D meshes [2, 6, 33, 36], 3D voxels [11, 13, 28, 37], point clouds [1, 22, 25, 35], depth maps [9, 17]. They further combine the 3D geometry defined with explicit representations with appearance representations such as colors, texutre maps, light fields or neural texture. The use of explicit 3D representations of geometry either requires supervision from ground truth 3D representation or poses strong assumption on the underlying 3D geometry.

In recent years, there has been advances in neural rendering approaches with neural radiance field (NeRF) [23, 40], where a 3D scene is represented implicitly by a multi-layer perceptron (MLP). The MLP maps the 3D coordinate and camera view direction to RGB value and density, and synthesizes a novel view via volume rendering which aggregates the colors of sampled 3D points along a ray. NeRF produces high quality novel view synthesis without the need of 3D supervision or assumption on the 3D geometry. Following works extend NeRF for faster training and inference, such as representing 3D scene with hashmap [24], or octree [19], followed by a reduced number of MLP layers to speed up. Other works extend NeRF to improve surface capture quality, such as NeuS [34].

2.2. Human Face Style Transfer

Given a content image of human face and a reference style image, human face style transfer aims to synthesize a stylized image with the style of the style image and the structure of the content image. Traditional approaches for human face style transfer mainly focus on 2D image domain. Some works realize human face style transfer with an image-to-image translation framework, where the main idea is to learn a bi-directional mapping between the real face domain and artistic face domain [26, 32, 42]. The other line of work falls on modifying and finetuning styleGAN [12]. Pinkney and Adler [27] first finetuned StyleGAN on cartoon data and achieved cartoon style transfer by simply applying the latent code in original StyleGAN to finetuned cartoon StyleGAN. Kwong et al. [8] further swapped the convolutional layer features between original styleGAN and a finetuned cartoon styleGAN to achieve style transfer. DualStyleGAN [38] modified the architecture of StyleGAN by introducing explicit extrinsic style path to have a deeper control on the style transfer. As these approaches focus on 2D image domain, they usually show 3D inconsistency issue when applied on multi view images of the same face. In contrast to 2D approaches, our approach achieves style transfer in 3D domain, with visually pleasing quality while preserving 3D consistency.

2.3. 3D Scene Style Transfer

There have been recent works [4, 5, 18, 39] on 3D scene style transfer which combines style transfer and novel view synthesis and aims to synthesize novel views with style from a style image while preserving the underlying 3D structure. They mainly leverage NeRF [23] as the 3D representation for the scene. These works mainly apply on in-the-wild 3D scenes and transfer the color tone of style images. However, they cannot capture the detail style patterns and semantics as required in human face style transfer. Further, to handle the training time and memory issue from NeRF, they propose solutions that may reduce style transfer quality, or fail to generalize to unseen styles. For example, [4, 5] applies style transfer losses on small image patches during training, which degrades the style transfer quality as it cannot capture global semantic information. ARF [39] proposed deferred gradient descend to train on full resolution image, which significantly slows down training and makes learning multiple styles impossible in practice. In contrast to these works, our approach focuses on 3D human face style transfer and captures local details and semantics in style transfer. We propose a novel NeRF-mesh hybrid framework which enables fast training speed at original image resolution and achieves good style transfer quality and 3D consistency.

3. Proposed Approach

3.1. Overview

As illustrated in Fig. 2, our approach consists of three stages: 1. geometry training stage, where we train a NeRF model to capture the 3D geometry of the real face; 2. mesh optimzation stage, where we derive a mesh from the trained NeRF model, refine its color through inverse projection, and stylize it by optimizing for style transfer objectives with pair data setting; 3. style training stage, where we train a hyper network to predict NeRF's color network weight from a style embedding extracted from a style image. Details of each stage are presented in the following sections.

3.2. Geometry Training

Neural Radience Field (NeRF) [23] uses multilayer perceptron (MLP) networks to model a 3D scene as fields of volume density and colors. Given a pixel of an image for a 3D scene at view direction, a ray from the pixel is emitted and several 3D points are sampled along the ray. For each 3D point, NeRF predicts its volume density and color by a geometry network and color network. The geometry network of NeRF maps a 3D point to volume density and features. The color network of NeRF then maps features from geometry network and view direction to RGB color. The predicted color of the pixel is derived by volume rendering which aggregates the color and volume density of the

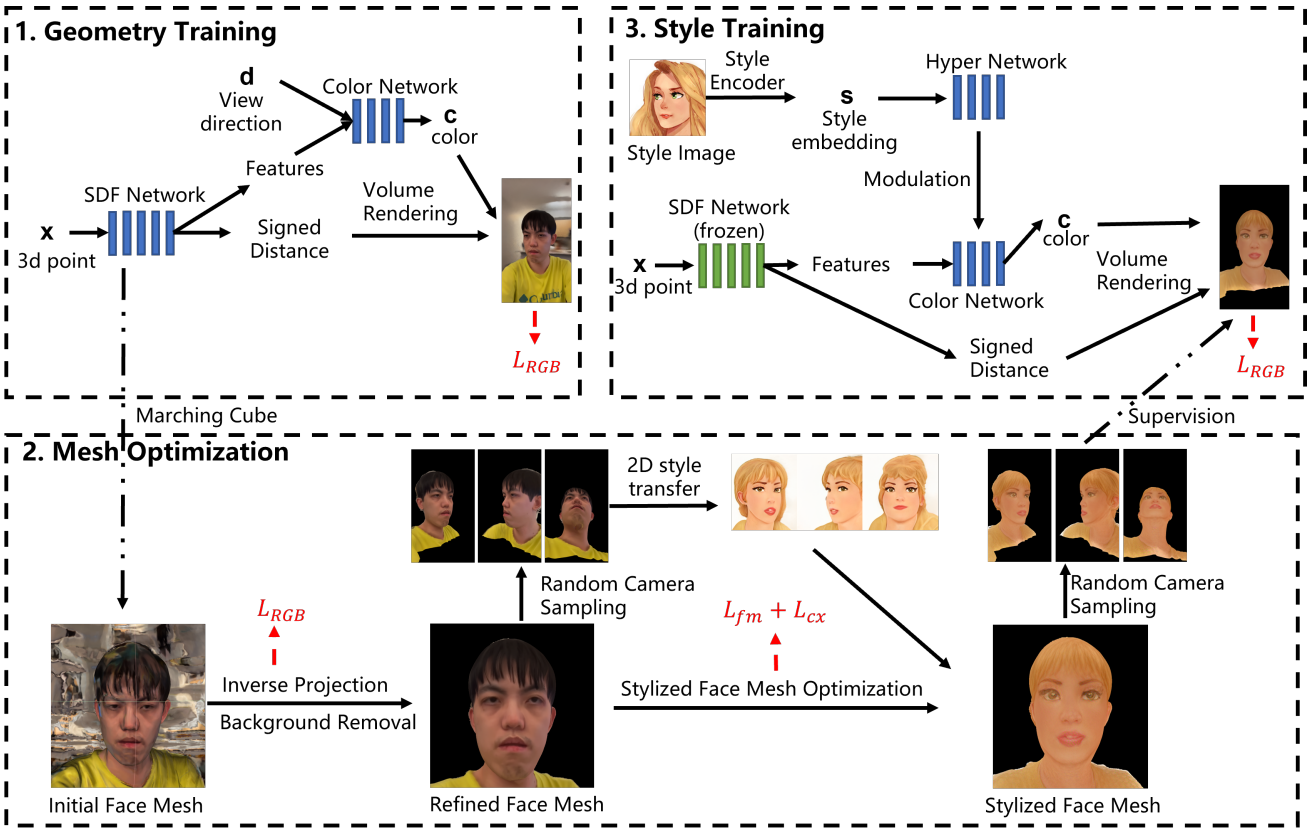


Figure 2. Overview of our approach. Our approach is in 3 stages: 1. Geometry training to learn the 3D geometry of a human face; 2. Mesh optimization to refine mesh colors and transfer style from a style image to the mesh; 3. Style training to train a hyper network conditioned on style image to generalize to arbitrary style.

3D points along the ray.

Original NeRF has issues with extracting high quality surface due to insufficient surface constraint during training. To derive higher quality mesh from a trained radiance field, we use NeuS [34] which proposes improvements in surface capture. NeuS represents surface as a signed distance function (SDF) and replaces geometry network in NeRF with an SDF network to predict signed distance from a 3D point. It also modifies volume rendering formulation based on SDF and introduces an extra loss terms for surface regularization.

In the geometry training stage, we train a NeuS model on input face images. The trained SDF network of NeuS represents the 3D geometry of the human face.

3.3. Mesh Optimization

After training a NeuS model on input images, we use marching cube [20] to export a face mesh from trained SDF network. To optimize face mesh, we apply differentiable rasterization [15] to render image from mesh, and apply losses on image level, where the gradients of the losses can be back propagated to the mesh. Optimizing topology of 3D mesh from image supervision usually leads to suboptimal convergence, as analyzed in [16, 29]. We therefore fix the vertex locations of the mesh and only optimize for vertex colors.

Mesh Refinement: The initial mesh from marching cube generally contains some artifacts in the colors. This is because the color network of NeuS model was trained with volume rendering which aggregates colors along the ray to form final color at pixel, and thus the color at surface point has some gap with the color seen in the image. We then refine the mesh color by optimizing an inverse projection problem.

$$argmin_{\mathbf{c}} L_{rgb}(\mathbf{M} \odot \phi_{\mathbf{c}}(\theta), \mathbf{M} \odot \mathbf{I_{gt}}) \quad (1)$$

where $\mathbf{c}$ is vertex colors and $\phi_{\mathbf{c}}(\cdot)$ represents an image generator by mesh rasterization, parameterized by vertex color. I_{gt} is a random ground truth image from the set of input images and θ is the corresponding view angle. $\mathbf{M}$ is the mesh segmentation mask. We optimized the mesh color on input images with masked RGB loss with an iterative process. After inverse projection, the mesh color is refined to be similar as presented in source images. We further remove background by applying a foreground segmentation

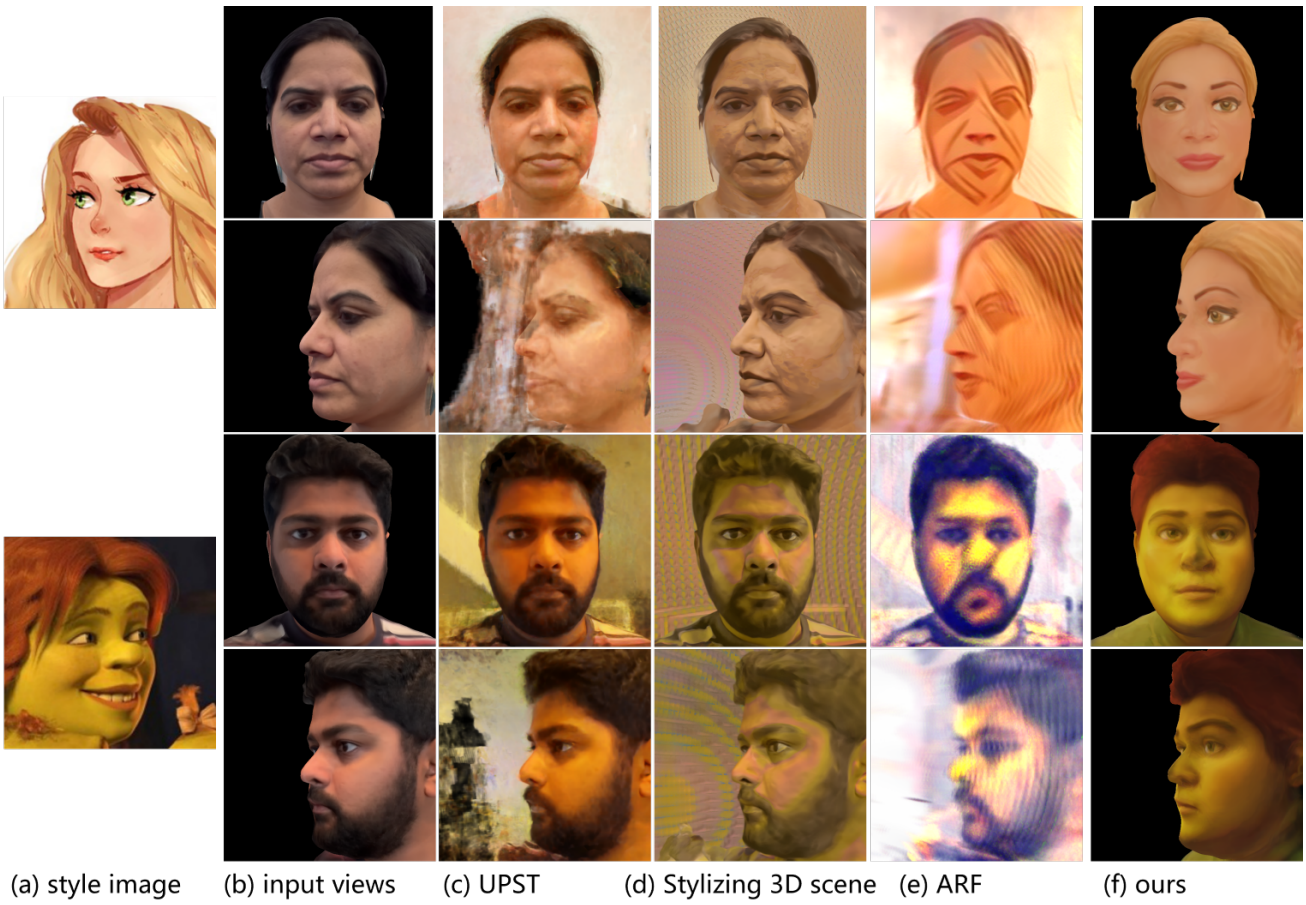


Figure 3. Qualitative Comparisons of transferring style in a style image (a) to input views (b). Our approach (f) shows better style transfer quality and 3D consistency compared to other 3D scene style transfer approaches (UPST [4] (c), Stylizing 3D Scene [5] (d), ARF [39] (e))

model [3] on input images and trim down mesh vertices that are visible in the input images as background pixels. After mesh refinement and background removal, the resulting mesh mainly contains human head and part of the upper body and has a photorealistic texture, which enables us to synthesize photorealistic images at different view points to use for content images for 2D style transfer.

Face Mesh Style Transfer: Given a refined face mesh and a style image, we aimed at transferring the style from the style image to the face mesh through optimization. Naturally, we can view the face mesh as an image generator $\phi_{\mathbf{c}}$ parameterized by vertex colors $\mathbf{c}$ that can generate content images of the face at arbitrary angle. And we apply style transfer objectives between content images and the input style image to optimize the vertex colors $\mathbf{c}$. For the style transfer objectives, we use a feature matching loss [10] and contextual loss [21]. This brings in our initial optimization objective below.

$$argmin_{\mathbf{c}} L_{fm}(\phi_{\mathbf{c}}(\theta), \mathbf{I_{style}}) + L_{cx}(\phi_{\mathbf{c}}(\theta), \mathbf{I_{style}}) \quad (2)$$

where $\mathbf{I_{style}}$ is the style image, and θ is the view angle of the mesh randomly sampled from a semi sphere in each iteration of optimization.

However, the initial objectives could not optimize the mesh color to have good style transfer quality. We find that it is because of a large gap in identity and head pose between the mesh rendered images and the style image. The mesh rendered images always resemble the identity of the input images that is different with the style image. And the mesh rendered images have diverse head pose that could be largely different with style image. Therefore, we propose to optimize with pair data that has similar identity and head pose.

More specifically, instead of using a fixed style image $\mathbf{I_{style}}$ for arbitrary content image $\phi_{\mathbf{c}}(\theta)$, we use a 2D style transfer model DualStyleGAN [38] $\psi(\cdot)$ to generate stylized image from a content image and a style image to have similar head pose and identity with the content image.

$$argmin_{\mathbf{c}} L_{fm}(\phi_{\mathbf{c}}(\theta), \psi(\phi_{\mathbf{c}}(\theta), \mathbf{I_{style}})) + L_{cx}(\phi_{\mathbf{c}}(\theta), \psi(\phi_{\mathbf{c}}(\theta), \mathbf{I_{style}}))) \quad (3)$$

During optimization, for each iteration, we randomly sample a view angle θ from a semi sphere, render an image $\phi_{\mathbf{c}}(\theta)$ from mesh, and generate a stylized image $\psi(\phi_{\mathbf{c}}(\theta), \mathbf{I_{style}})$ from 2D style transfer. The vertex color is optimized by the feature matching loss and contextual loss between these two.

The stylized images are generated from 2D style transfer and could contain 3D inconsistencies. As we are fixing vertex locations, the 3D consistency of the optimized mesh is guaranteed, and the optimization objectives only supervise the style of the mesh and avoid potential 3D inconsistencies from the generated stylized images. After optimization, we obtain a stylized face mesh from a style image. With mesh rasterization, the optimization is pretty fast and only takes 2 minutes per mesh.

3.4. Style Training

In this stage, we would like to generalize the color network of NeuS model for arbitrary style transfer. For this purpose, it should be trained with multiple styles seen so that it could generalize to unseen style. Therefore, we generate 200 stylized meshes corresponding to 200 different style images to use as our ground truth generators for training.

We modulate the weight of the color network in NeuS model by a hyper network $\Omega(\cdot)$ whose input is a style embedding extracted from a style image by a PSP style encoder [30]. Given different style images, the hyper network is capable of generating different color network weight to render for different stylized outputs.

We freeze the SDF network from stage 1 to reuse the learned 3D geometry, and only train the hyper network. We train with RGB loss supervised by the stylized mesh in stage 2. For each iteration, we randomly sample a style image $\mathbf{I_{style}}$, its corresponding stylized mesh $\phi(\cdot)$, a view angle θ at a semi sphere and a batch of pixels. We query the hyper network from a style embedding to generate weight of color network and render the color of sampled pixel through volume rendering. For RGB supervision, we use the stylized mesh $\phi(\cdot)$ to render an image from the same view angle θ. Formally,

$$argmin_{\Omega} L_{rgb}(\Omega(\mathbf{z_{style}}, \theta), \phi(\theta)) \quad (4)$$

where $\mathbf{z_{style}}$ represents a style embedding from a style image, $\Omega(\mathbf{z_{style}}, \theta$ represents a batch of pixels from hyper network rendering.

In test time, the trained hyper network can be used for arbitrary style transfer. With a style image, we extract its style embedding and predict the weight of color network. And the predicted color network and the pretrained SDF network are used to generate stylized novel views through volume rendering with the style in style image applied.

4. Experiment

Dataset We collect a video dataset of 8 subjects, where each of them records a video of 10-15 seconds of 300-500 frames at 30 FPS. The videos are further processed with COLMAP [31] to estimate camera intrinsics and poses for every video frame. For style transfer, We use a cartoon dataset [27] with 317 cartoon images. We use 200 images during training and hold off the remaining 117 images as unseen styles to evaluate for the generalizability of our approach. For each subject, we train a separate model for arbitrary style transfer on this subject. For a single model, the training can be finished in 23 hours, with 7 hours for stage 1, 6 hours for stage 2 and 10 hours for stage 3.

Methods for Comparison We compare our approach with state of the art 3D scene style transfer approaches (UPST [4], Stylizing 3D Scene [5], ARF [39]), and two baselines: 1. 2D style transfer $\rightarrow$ NeuS, where we first run 2D Image style transfer [38] on input images and then train a NeuS model directly on top of the stylized source images; 2. Neus $\rightarrow$ 2D style transfer, where we first train a NeuS model on top of the source images to synthesize novel views for real human face, and then apply 2D style transfer on top of the synthesized novel view images.

4.1. Qualitative Results

We compare our approach and 3D scene style transfer approaches (UPST [4], Stylizing 3D Scene [5], ARF [39]) qualitatively in Fig. 3. Among the 3D scene style transfer approaches, UPST [4] is significantly under-stylized and has a bad novel view synthesis on the side view. Stylizing 3D scene [5] generates 3D consistent frontal and side views, but can only transfer overall color tone and have artifacts in the background. ARF [39] applies stronger style transfer than other two approaches, but loses details in facial structure and contains blurriness. The compared 3D scene style transfer approaches only transfer the overall color tone of the style image and fail to capture the semantics of the face, whereas our approach transfers the color of hair, skin and lip well and also achieves good 3D consistency.

4.2. Quantitative Results

Consistency Measurement We use the short range consistency error and the long range consistency error from [14] to measure the 3D consistency between stylized images at different view points, aligned with the other 3D scene style transfer approaches. The consistency error is implemented by a warped LPIPS metric [41] where a view

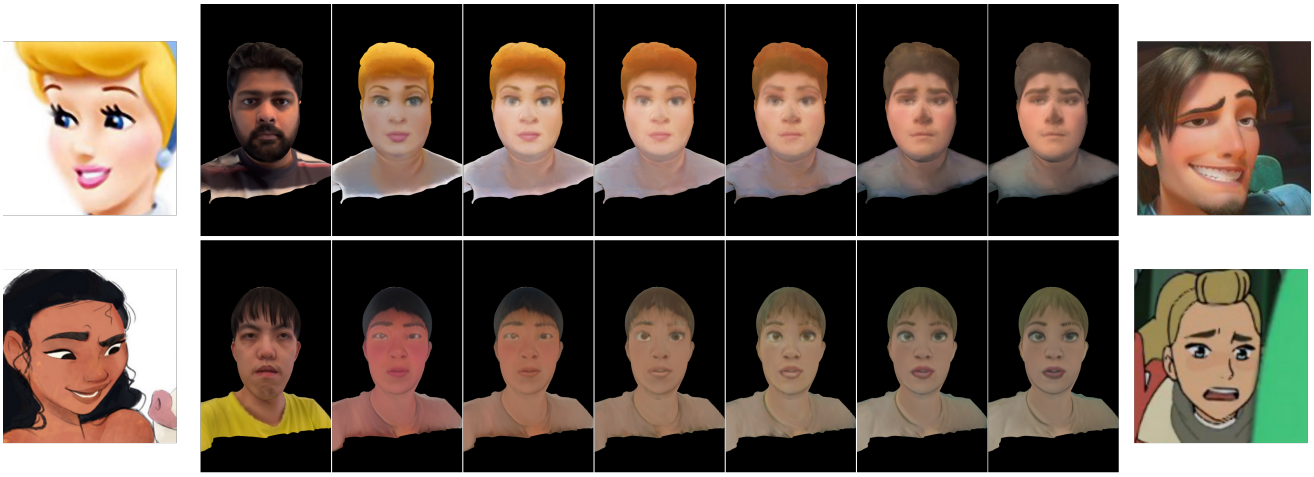


Figure 4. Style blending, our approach can interpolate between two styles and generate a mixed style of both. We show two rows of examples with style gradually changing from style 1 to style 2.

Table 1. Quantitative Comparison on short range and long range 3D consistency error. Our approach outperforms the compared approaches by a magnitude.

Method	Short Range Consistency Error (LPIPS $\times 10^{-2} \downarrow$)	Long Range Consistency Error (LPIPS$\times 10^{-2} \downarrow$)
2D style transfer $\rightarrow$ NeuS	1.21	3.47
NeuS $\rightarrow$ 2D style transfer	3.23	5.07
UPST [4]	1.71	4.14
Stylizing 3D Scene [5]	1.20	2.06
ARF [39]	1.88	5.12
Ours	**0.29**	**0.38**

is warped to another view with a depth estimation.

$$E(V_i, V_j) = LPIPS(M_{ij} \odot V_i, M_{ij} \odot f_{ij}^{w}(V_j)) \quad (5)$$

where $E(V_i, V_j)$ is the consistency error between view i and view j, f_{ij}^{w} is the warping function and M_{ij} is the warping mask. When computing LPIPS metric, only the pixels within the warping mask are taken. For short range consistency, the consistency error is computed with every adjacent frames in the testing video. For long range consistency, the consistency error is computed with all the view pairs with the gap of 7 frames.

Table. 1 shows that our approach outperforms the compared approaches by a magnitude in both short range consistency and long range consistency. The large improvement in the 3D consistency benefits from our multi stage training where explicit mesh guidance is applied. Among the other approaches, NeuS $\rightarrow$ 2D style transfer has the lowest 3D consistency as it absorbs most of the 3D consistency issues from 2D style transfer. Other NeRF based approaches show better 3D consistency but is significantly worse than our approach as they do not have explicit mesh guidance as ours which strengthen the 3D consistency.

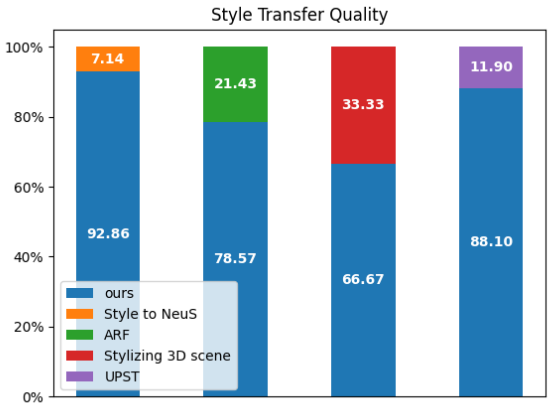
User Study We perform a user study to evaluate the style transfer quality and 3D consistency between different approaches. We compare our approach with four different approaches (Style to NeuS, ARF [39], Stylizing 3D Scene [5] and UPST [4]). For each comparison, we generate videos of two approaches for two identities (four videos in total). For each identity in a comparison, we ask users to make two selection: 1. select the video of better style transfer quality; 2. select the video of better 3D consistency. We collect votes from 20 participants per comparison, in total 320 votes (320=4 (comparisons) $\times$ 20 (participants) $\times$ 2 (identities) $\times$ 2 (questions). Results are shown in Fig. 5. Our approach outperforms other approaches in both style transfer quality and 3D consistency.

4.3. Ablation Studies

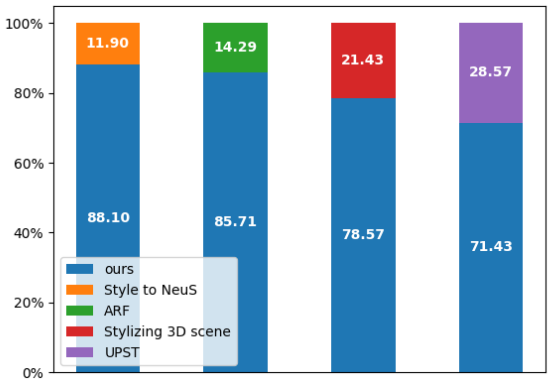
Mesh Optimization without pair data To show the effectiveness of our pair data setting during mesh optimization stage, we do an ablation study and show that without pair data setting, the mesh optimization could not converge well, due to the large identity and head pose gap between the style image and the content image from mesh rendering. Visualization can be seen at Fig 6.

4.4. Application

Style Blending Our approach can perform smooth style blending between two styles by interpolating between the



(a) Style Transfer Quality



(b) 3D Consistency

Figure 5. User study in style transfer quality and 3D consistency. We ask the users to select the approach with better style quality or 3D consistency.

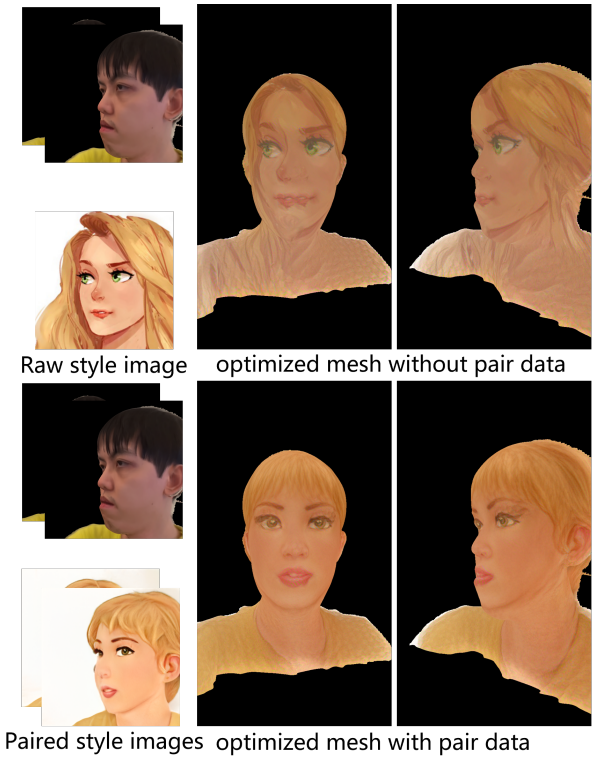


Figure 6. Comparison of mesh optimization with/without pair data setting

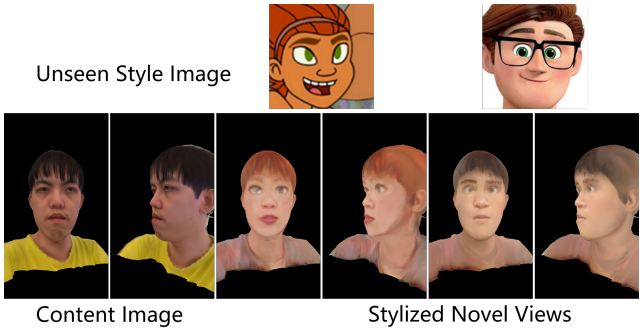


Figure 7. Our approach can generalize to unseen style images and generate style transfer with decent quality and 3D consistency

two embedding of the style images, generating smooth and harmonious style transfer of a mixed style blended from two style images, as shown in Fig. 4. This allows creation of non-existent styles through blending two styles.

Unseen Style Our approach trains a hyper network to generalize on multiple styles, hence is capable of generalizing to unseen style images in training, as illustrated in Fig. 7. This allows a broader use of our approach to apply on arbitrary cartoon images for 3D human face style transfer.

5. Conclusion

In this paper, we propose a novel three stage approach that achieves 3D face style transfer with good style quality and 3D consistency. We present a hybrid training strategy with volume rendering and mesh rasterization which enables style transfer at original image resolution. We design a novel mesh optimization stage where we propose a pair data setting to generate decent stylized meshes. We train a hyper network on stylized meshes to generalize for arbitrary style transfer. Our experiments demonstrate that our approach outperforms baselines approaches in terms of style quality and 3D consistency quantitatively and qualitatively, and is also capable to perform smooth and harmonious style blending as well as generalizing to unseen style.

References

[1] Kara-Ali Aliev, Artem Sevastopolsky, Maria Kolos, Dmitry Ulyanov, and Victor Lempitsky. Neural point-based graphics. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16*, pages 696–712. Springer, 2020. 3

[2] Chris Buehler, Michael Bosse, Leonard McMillan, Steven Gortler, and Michael Cohen. Unstructured lumigraph rendering. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 425–432, 2001. 3

[3] Xiangguang Chen, Ye Zhu, Yu Li, Bingtao Fu, Lei Sun, Ying Shan, and Shan Liu. Robust human matting via semantic guidance. In *Proceedings of the Asian Conference on Computer Vision*, pages 2984–2999, 2022. 5

[4] Yaosen Chen, Qi Yuan, Zhiqiang Li, Yuegen Liu Wei Wang Chaoping Xie, Xuming Wen, and Qien Yu. Upst-nerf: Universal photorealistic style transfer of neural radiance fields for 3d scene. *arXiv preprint arXiv:2208.07059*, 2022. 2, 3, 5, 6, 7

[5] Pei-Ze Chiang, Meng-Shiun Tsai, Hung-Yu Tseng, Wei-Sheng Lai, and Wei-Chen Chiu. Stylizing 3d scene via implicit representation and hypernetwork. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1475–1484, 2022. 2, 3, 5, 6, 7

[6] Paul E Debevec, Camillo J Taylor, and Jitendra Malik. Modeling and rendering architecture from photographs: A hybrid geometry-and image-based approach. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 11–20, 1996. 3

[7] Fangzhou Hong, Mingyuan Zhang, Liang Pan, Zhongang Cai, Lei Yang, and Ziwei Liu. Avatarclip: Zero-shot text-driven generation and animation of 3d avatars. *arXiv preprint arXiv:2205.08535*, 2022. 2

[8] Jialu Huang, Jing Liao, and Sam Kwong. Unsupervised image-to-image translation via pre-trained stylegan2 network. *IEEE Transactions on Multimedia*, 24:1435–1448, 2021. 3

[9] Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang. Deepmvs: Learning multi-view stereopsis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2821–2830, 2018. 3

[10] 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. 5

[11] Mengqi Ji, Juergen Gall, Haitian Zheng, Yebin Liu, and Lu Fang. Surfacenet: An end-to-end 3d neural network for multiview stereopsis. In *Proceedings of the IEEE international conference on computer vision*, pages 2307–2315, 2017. 3

[12] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 3

[13] Kiriakos N Kutulakos and Steven M Seitz. A theory of shape by space carving. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 1, pages 307–314. IEEE, 1999. 3

[14] Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, and Ming-Hsuan Yang. Learning blind video temporal consistency. In *Proceedings of the European conference on computer vision (ECCV)*, pages 170–185, 2018. 6

[15] Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. Modular primitives for high-performance differentiable rendering. *ACM Transactions on Graphics (TOG)*, 39(6):1–14, 2020. 2, 4

[16] Yiyi Liao, Simon Donne, and Andreas Geiger. Deep marching cubes: Learning explicit surface representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2916–2925, 2018. 4

[17] Fayao Liu, Chunhua Shen, Guosheng Lin, and Ian Reid. Learning depth from single monocular images using deep convolutional neural fields. *IEEE transactions on pattern analysis and machine intelligence*, 38(10):2024–2039, 2015. 3

[18] Kunhao Liu, Fangneng Zhan, Yiwen Chen, Jiahui Zhang, Yingchen Yu, Abdulmotaleb El Saddik, Shijian Lu, and Eric Xing. Stylerf: Zero-shot 3d style transfer of neural radiance fields. *arXiv preprint arXiv:2303.10598*, 2023. 3

[19] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *Advances in Neural Information Processing Systems*, 33:15651–15663, 2020. 3

[20] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169, 1987. 4

[21] Roey Mechrez, Itamar Talmi, and Lihi Zelnik-Manor. The contextual loss for image transformation with non-aligned data. In *Proceedings of the European conference on computer vision (ECCV)*, pages 768–783, 2018. 5

[22] Moustafa Meshry, Dan B Goldman, Sameh Khamis, Hugues Hoppe, Rohit Pandey, Noah Snavely, and Ricardo Martin-Brualla. Neural rerendering in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6878–6887, 2019. 3

[23] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 3

[24] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)*, 41(4):1–15, 2022. 3

[25] Simon Niklaus, Long Mai, Jimei Yang, and Feng Liu. 3d ken burns effect from a single image. *ACM Transactions on Graphics (ToG)*, 38(6):1–15, 2019. 3

[26] Ori Nizan and Ayellet Tal. Breaking the cycle-colleagues are all you need. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7860–7869, 2020. 3

[27] Justin NM Pinkney and Doron Adler. Resolution dependent gan interpolation for controllable image synthesis between domains. *arXiv preprint arXiv:2010.05334*, 2020. 3, 6

[28] Charles R Qi, Hao Su, Matthias Nießner, Angela Dai, Mengyuan Yan, and Leonidas J Guibas. Volumetric and multi-view cnns for object classification on 3d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5648–5656, 2016. 3

[29] Edoardo Remelli, Artem Lukoianov, Stephan Richter, Benoit Guillard, Timur Bagautdinov, Pierre Baque, and Pascal Fua. Meshsdf: Differentiable iso-surface extraction. *Advances in Neural Information Processing Systems*, 33:22468–22478, 2020. 4

[30] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2287–2296, 2021. 6

[31] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 6

[32] Xuning Shao and Weidong Zhang. Spatchgan: A statistical feature based discriminator for unsupervised image-to-image translation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6546–6555, 2021. 3

[33] Michael Waechter, Nils Moehrle, and Michael Goesele. Let there be color! large-scale texturing of 3d reconstructions. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 836–850. Springer, 2014. 3

[34] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021. 3, 4

[35] Olivia Wiles, Georgia Gkioxari, Richard Szeliski, and Justin Johnson. Synsin: End-to-end view synthesis from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7467–7477, 2020. 3

[36] Daniel N Wood, Daniel I Azuma, Ken Aldinger, Brian Curless, Tom Duchamp, David H Salesin, and Werner Stuetzle. Surface light fields for 3d photography. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 287–296, 2000. 3

[37] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015. 3

[38] Shuai Yang, Liming Jiang, Ziwei Liu, and Chen Change Loy. Pastiche master: exemplar-based high-resolution portrait style transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7693–7702, 2022. 2, 3, 5, 6

[39] Kai Zhang, Nick Kolkin, Sai Bi, Fujun Luan, Zexiang Xu, Eli Shechtman, and Noah Snavely. Arf: Artistic radiance fields. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXI*, pages 717–733. Springer, 2022. 2, 3, 5, 6, 7

[40] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020. 3

[41] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 6

[42] Yihao Zhao, Ruihai Wu, and Hao Dong. Unpaired image-to-image translation using adversarial consistency loss. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16*, pages 800–815. Springer, 2020. 3

[43] Yufeng Zheng, Victoria Fernández Abrevaya, Marcel C Bühler, Xu Chen, Michael J Black, and Otmar Hilliges. Im avatar: Implicit morphable head avatars from videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13545–13555, 2022. 2