ReenactArtFace: Artistic Face Image Reenactment

Linzi Qu, Jiaxiang Shang, Xiaoguang Han, and Hongbo Fu

Abstract—Large-scale datasets and deep generative models have enabled impressive progress in human face reenactment. Existing solutions for face reenactment have focused on processing real face images through facial landmarks by generative models. Different from real human faces, artistic human faces (e.g., those in paintings, cartoons, etc.) often involve exaggerated shapes and various textures. Therefore, directly applying existing solutions to artistic faces often fails to preserve the characteristics of the original artistic faces (e.g., face identity and decorative lines along face contours) due to the domain gap between real and artistic faces. To address these issues, we present *ReenactArtFace*, the first effective solution for transferring the poses and expressions from human videos to various artistic face images. We achieve artistic face reenactment in a coarse-to-fine manner. First, we perform *3D artistic face reconstruction*, which reconstructs a textured 3D artistic face through a 3D morphable model (3DMM) and a 2D parsing map from an input artistic image. The 3DMM can not only rig the expressions better than facial landmarks but also render images under different poses/expressions as coarse reenactment results robustly. However, these coarse results suffer from self-occlusions and lack contour lines. Second, we thus perform *artistic face refinement* by using a personalized conditional adversarial generative model (cGAN) fine-tuned on the input artistic image and the coarse reenactment results. For high-quality refinement, we propose a contour loss to supervise the cGAN to faithfully synthesize contour lines. Quantitative and qualitative experiments demonstrate that our method achieves better results than the existing solutions.

Index Terms—face reenactment, artistic faces, 3DMM, generative models

1 Introduction

Human face reenactment aims to generate a sequence of photo-realistic face images by transferring the pose and expression from a driving face video to a source face. Benefiting from large-scale datasets and deep generative models (e.g., GANs [1]), various face reenactment techniques, such as one-shot reenactment [2], [3], [4] and few-shot reenactment [5], [6], have been explored. Such one-shot face reenactment techniques can be potentially used to animate single artistic human face images in paintings, cartoons, mangas, sketches, etc. This would greatly simplify the animation process of such artistic images, thus benefiting various applications, e.g., making artistic talking heads for entertainment or privacy protection during online chatting.

However, directly applying existing face reenactment techniques to artistic face images often fails due to the domain gap between real and artistic faces. Artistic faces have the following characteristics, which increase the difficulties of artistic face reenactment. First, artistic faces often have a large variance in geometry and contain exaggerated edges (Figure 2 (a)). Recent human face reenactment methods [7], [8], [9], [10] rely on facial landmarks, which cannot represent such exaggerated edges well. Second, artistic face textures have different styles (Figure 2 (b)), making it difficult to construct a unified dataset to train a generation model. Additionally, it is time-consuming to provide a series of multi-view images/videos for a specific artistic face. Without such an artistic face dataset, the previous neural methods [7], [8], [9], [10] cannot be retrained. One-shot human reenactment methods [2], [4] with a fine-tuning stage might mitigate the problem of training all diverse styles in one network. However, fine-tuning with one view is not robust to pose and expression variations. Finally, artists often draw lines along face contours to delimit color regions and thus emphasize the shape information (Figure 2 (c) and (d)). However, such contour lines do not exist in real face images and are thus not considered in real face reenactment works.

In this paper, we propose *ReenactArtFace*, the first effective pipeline for achieving artistic portrait reenactment. Our pipeline has two stages and achieves the goal in a coarse-to-fine manner, as illustrated in Figure 3. First, it reconstructs a 3D textured and rigged face model from an input artistic face image. This reconstructed model provides coarse reenactment results. We regard these results excluding the regions with artifacts as the ground truth to help the generation of finer results in the second step. Such a two-stage design successfully enables the training of the second step and addresses the limitations of one-shot face reenactment, making our method robustly synthesize various poses and expressions from a single artistic face image (Figure 1).

Specifically, ReenactArtFace consists of two main modules. In the *3D Artistic Face Reconstruction* module, we first fit a 3D morphable model (3DMM) [11] to an input artistic face image and then use an associated input parsing map to guide the deformation of the fitted 3DMM. Such a parsing map provides richer semantic and boundary information than facial landmarks often used in human face reenactment. We leverage image meshing [12] to map the whole source image and the parsing map respectively to the UV textures of the fitted 3DMM model. The resulting textured

- *Corresponding author: Hongbo Fu*
  L. Qu and H. Fu are with the School of Creative Media, City University of Hong Kong. E-mail: linziqu2-c@my.cityu.edu.hk, hongbofu@cityu.edu.hk
- *J. Shang is with the Department of Computer Science & Engineering, HKUST. E-mail: jshang@cse.ust.hk*
- *X. Han is with Shenzhen Research Institute of Big Data, Chinese University of Hong Kong, Shenzhen. E-mail: hanxiaoguang@cuhk.edu.cn*

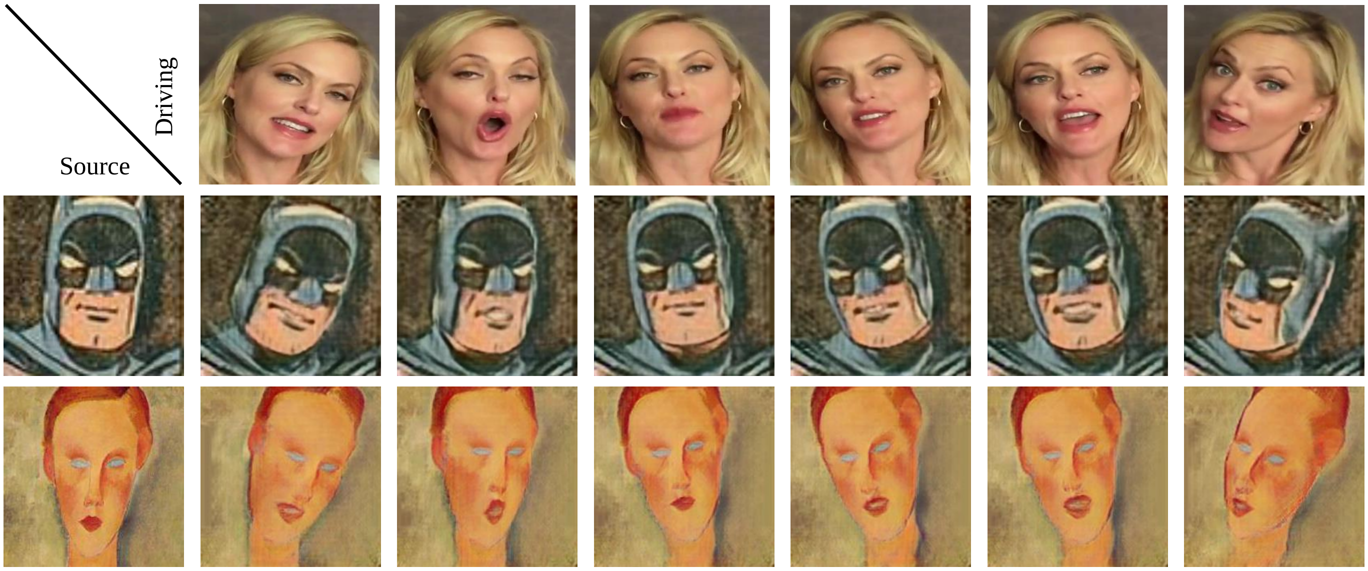


Fig. 1. Our ReenactArtFace, a novel artistic face reenactment technique, is able to transfer the pose and expression from a driving video to in-the-wild artistic face images to generate artistic talking heads, where the identity and texture of the original artistic images are preserved. Please refer to the accompanying video for animation effects.

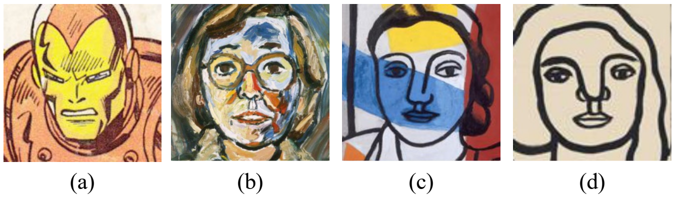


Fig. 2. The examples of artistic faces in different styles.

3DMM can produce faces/parsing maps with the same identity in various expressions and poses and thus provide good guidance for generation models. However, since images from direct rendering of the textured 3DMM suffer from self-occlusions and lack contour lines, they can serve as coarse reenactment results only. Meanwhile, benefiting from the semantics of 3DMM, the reenacted parsing maps avoid such self-occlusions. The second module, namely *Artistic Face Refinement*, takes as input a source artistic face image and the coarse reenactment parsing maps under different views and expressions. First, the model is trained on a large dataset of real human faces to learn the correspondence between semantics and textures, forming our pre-trained model. Then, we fine-tune this pre-trained model supervised by the rendered coarse reenactment results with the artifacts removed via inpainting and masking. To synthesize contour lines, we design a contour loss by leveraging cycle consistency between the reenacted results and the source face in the contour region.

Experiments show that our method is able to robustly reenact artistic images of various styles. Quantitative and qualitative comparisons prove that our ReenactArtFace outperforms the state-of-the-art techniques for real human face reenactment. Besides face reenactment, we also demonstrate the possibilities of our method for two additional applications, namely artistic face editing and facial video stylization.

- We are the first to transfer the poses and expressions from human videos to various artistic faces and fully consider the artistic facial characteristics.
- We achieve artistic face reenactment via a novel two-stage approach. In Stage 1, the 3D Artistic Face Reconstruction module ensures the geometry consistency via 3DMM fitting and deformation, resulting in coarse reenactment results. In Stage 2, the Artistic Face Refinement module achieves the texture consistency with the input face (including its contour lines) via utilizing the rendered coarse reenactment results and a contour loss to train a personalized cGAN for each artistic face.
- We demonstrate the effectiveness and practicality of our method via extensive experiments and applications.

2 Related Work

In this section, we review the existing techniques that are related to our method, including face reenactment, 3DMM-based facial texture completion, and cycle consistency in face image generation.

2.1 Face Reenactment

Existing face reenactment solutions can be roughly divided into graphics-based [13], [14], [15], [16], [17] and neural-based [7], [8], [9], [18], [19], [20] methods in terms of the utilization of Deep Neural Networks (DNNs).

Graphics-based methods in 2D space achieve reenactment via 2D sparse warps [15], [17] from control points, and they use a simple piece-wise linear interpolation defined by Delaunay triangulation to expand such warps. Nevertheless, sparse warps fail to generate the hidden regions in the source images, especially when the pose changes. With converting the input to 3D, [16] recombine the fitted

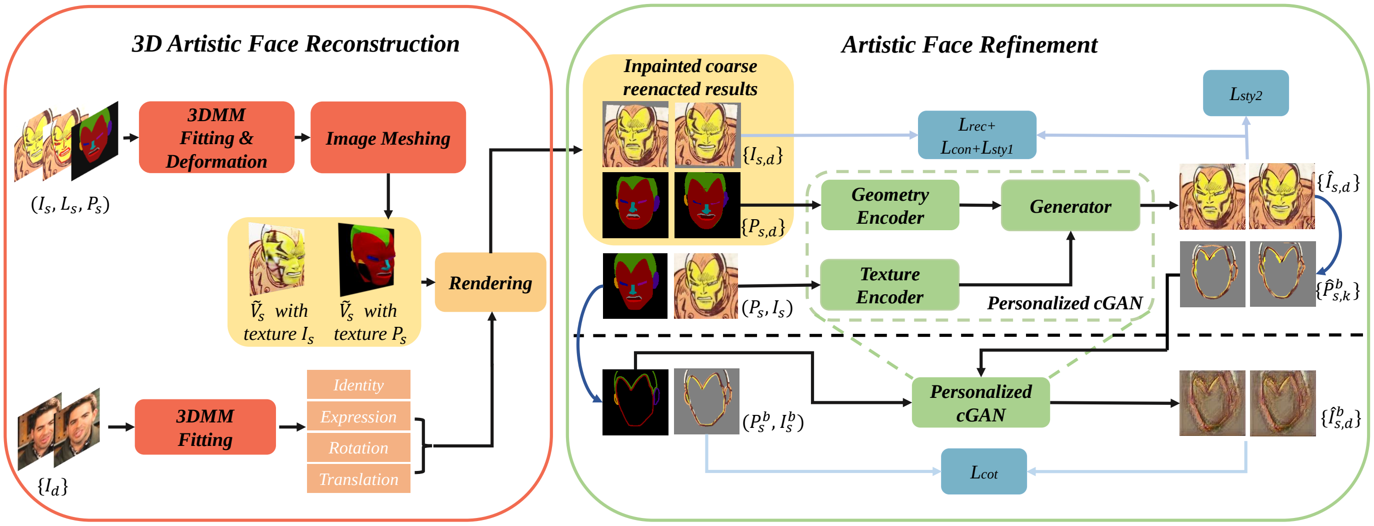


Fig. 3. An illustration of our two-stage pipeline for artistic face reenactment. It transfers the pose and expression of a real human face in a driving video $\{I_d\}$ to an input artistic face I_s (with the help of manually labeled face landmarks L_s and parsing map P_s) to get an artistic talking head $\{\hat{I}_{s,d}\}$.

3DMM parameters and render the reenacted 3D face model. However, 3DMMs are from human faces, which cannot be applied directly to artistic faces.

Benefiting from large human-face datasets, neural-based methods train powerful DNNs. For example, FOMM [18] uses the motion module to produce a dense warp field guided by keypoints. Then, they warp the source face to target poses and expressions before applying the generation module to synthesize occlusion areas. Similarly, the work of PIRenderer [19] has two networks for warping and synthesizing guided by fitted 3DMMs from 2D landmarks. [7], [8], [9], [21], [22] directly leverage generators to produce faces with the input reenacted landmarks for source identities from various geometry transfer modules. However, the facial landmark detector might fail when the input is an artistic portrait with an exaggerated shape. Besides, the commonly used 68 facial landmarks are too sparse to represent artistic faces. These methods fail to preserve the geometry, and such DNNs trained on the real human face data provide real-human-like results in at least one aspect of geometry or texture (Figure 7). To keep identities of artistic faces, on the one hand, we explicitly represent geometric inputs via deformed 3DMMs using the associated parsing maps. On the other hand, we train a generator not only with real human faces but also with more closely artistic faces. [5], [23] fine-tune their generators for a specific out-of-domain person by combining constant latent identity features and multi-pose features constrained by the source artistic face with a discriminator after training on various human identities. Although fine-tuning is a feasible way to mitigate the domain gap, the generator would easily overfit the single input and thus ignore the reenacted guidance. To address the overfitting problem, we first render coarse reenactment results from the fitting and deformation 3DMM, and then fine-tune our personalized cGAN by such coarse results rather than the single source input.

2.2 3DMM-based Facial Texture Completion

Our coarse reenactment results contain artifacts caused by the invisible regions of the single-view input, and the inpainting of such artifacts is related to the face completion task. Since 3DMM contains expressive geometry and texture prior, it provides a good starting point for face completion works [12], [14], [24], [25], [26], [27], [28]. After the 3DMM fitting step, these works are designed in a coarse-to-fine manner. For example, Gecer *et al.* [28] project an input face in 3D and fill the unseen regions by reconstructing the uncompleted projected images of different views through a pre-trained StyleGAN [29]. DVP [14] and AudioDVP [26] take the synthetic renderings of the fitted 3DMM as coarse inputs and learn generators to add details. They train the generators with video clips in a self-supervised manner. However, it is hard to build a dataset [28] covering all kinds of styles to train a StyleGAN due to the large variance among artistic faces, and it is time-consuming to prepare a video clip [14], [26] for each artistic face. Different from them, considering the limited data for each artistic portrait, we maximally use the information of multi-view coarse renderings as a guide rather than an input to train a cGAN network.

2.3 Cycle Consistency in Face Image Generation

Our Artistic Face Refinement module aims to synthesize faithful face results. Previous cGAN methods have performed impressive results on real human faces. Most generation models [1], [30] are trained with large paired datasets, but such a requirement is not always satisfied in real applications. Several works [31], [32], [33], [34] demonstrate that unpaired image training is effective with cycle consistency. For example, to create a cycle for unpaired supervision, GANimation [32] firstly generates an expression-changed image according to an arbitrary action unit (AU) and then uses this image with the original AU to produce the input to learn for expression translation. Zhou *et al.* [34] rotate

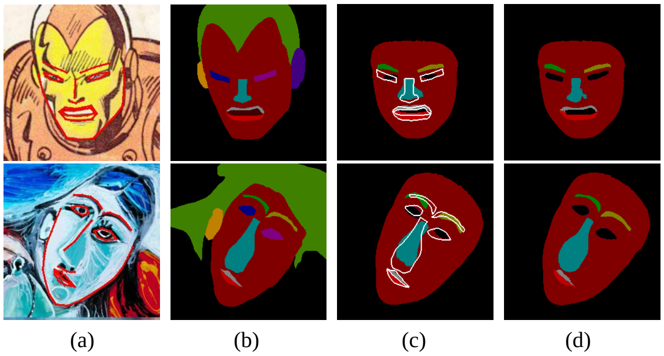


Fig. 4. Two examples of 3DMM fitting and deformation. In each row, (a) is an input image with manually specified facial landmarks (in red), and (b) is its corresponding parsing map. (c) The fitted mesh with the pre-annotated semantic regions overlaid with the boundaries of the parsing map in (b). (d) The deformed mesh with the pre-annotated semantic regions.

and render the roughly predicted 3D representation to a random pose and back to its original place to achieve face rotation. However, in our artistic face reenactment task, under cycle-consistency constraints, the optimal mapping between various reenacted faces and the single input source is hard to learn, which causes detail loss. Thus, we leverage the 3D-consistent coarse reenactment faces to supervise the generator. On the other hand, the contour regions are an important artistic style without 3D consistency. Inspired by cycle consistency, we propose a contour loss to deal with the generation of contour lines.

3 Method

For each input artistic portrait image I_s, with the corresponding parsing map P_s and facial landmarks L_s, our ReenactArtFace aims to animate this face according to the pose and expression of a real human face in a driving face video $\{I_d\}$. Due to the large shape variations in artistic faces, it is difficult to automatically detect facial landmarks in I_s, and we thus rely on manual intervention to get L_s and P_s.

To achieve artistic face reenactment, we use a two-stage pipeline, which consists of two main modules, namely, *3D Artistic Face Reconstruction* (Section 3.1) and *Artistic Face Refinement* (Section 3.2). The first module aims to reconstruct a 3D rigged and textured artistic face model, represented as $\tilde{V}_s$ from (I_s, L_s, P_s), and then renders it back to the image domain to get paired face images $\{I_{s,d}\}$ and parsing maps $\{P_{s,d}\}$ under various poses and expressions. $\{I_{s,d}\}$ essentially serve as a set of coarse reenactment results, which, however, suffer from artifacts due to self-occlusions and lack artistic contour lines. In the second module, a personalized cGAN is trained to synthesize the result $\hat{I}_{s,d}$ derived from the source artistic face I_s and the reenacted parsing map $P_{s,d}$. The cGAN is firstly pre-trained on a dataset of real human faces and then fine-tuned with coarse reenactment pairs $\{I_{s,d}, P_{s,d}\}$ after initially inpainting and masking. Thus, our pipeline explicitly disentangles the geometry and texture of an artistic face, which are defined by the inpainted parsing maps and the personalized cGAN, respectively.

3.1 3D Artistic Face Reconstruction

As shown in Figure 3, our 3D artistic face reconstruction contains two parts. We first introduce the 3DMM fitting for the source image in Section 3.1.1. Specifically, after the fitting, we further utilize the face boundary of the 2D parsing map to achieve the exaggerated 3D geometry of the artistic face. Then we propose to use the image meshing procedure to add non-face region meshes and textures in Section 3.1.2.

3.1.1 Fitting 3DMM and Deformation with Face Boundary

Although artistic faces might not perfectly lie in the shape space of real human faces, artistic and real human faces still look similar in terms of shape and structure. We thus first estimate the rough 3D geometry (denoted as V_c) via fitting the coefficients of a 3DMM [11] to the input artistic face I_s by minimizing the distance between the landmarks in L_s and the projected corresponding vertices of V_c, which are calculated from the coefficients and 3DMM bases.

As discussed earlier, compared with real human faces, which often have smooth curved edges, artistic faces might contain exaggerated edges (e.g., squared edges in Figure 4 (a)). Because of this characteristic, 68 landmarks commonly used on real human faces are too sparse to accurately represent the geometry for artistic faces. To solve this issue, guided by the boundary of each facial part in the parsing map P_s, we continue to deform all the vertices in V_c (already fitted to the facial landmarks) to produce a geometrically more accurate face model V_s. Specifically, to build the correspondence between V_c and the 2D parsing map P_s, we leverage the semantics of the 3DMM, which is manually specified for each point on the 3DMM in a pre-processing step. Since V_c shares the same topology as the 3DMM, the labeling procedure needs to be done once. Considering that there is no one-to-one mapping between 2D pixels and 3D points, we use the Chamfer distance to force the projected vertices located in the boundary of facial parts V_c^b (in V_c) to be aligned with the 2D pixels P_s^b in the semantic boundary of P_s. We denote the perspective projection from 3D space to 2D space as $Pro(,)$. Mathematically, the Chamfer distance is formulated as:

$$L_{ch} = \frac{Cham(P_s^b, Pro(V_s^b, \theta))}{|P_s^b|}, \tag{1}$$

where $Cham(,)$ denotes the Chamfer distance between two groups of points, $|\cdot|$ is the number of points, θ is the fitted camera pose for the source input.

The Chamfer loss only forces a part of 3D boundary vertices which are the closest to each 2D boundary pixel after the projection to move. To smoothly propagate the changes to the rest of the vertices, we apply Laplacian deformation [35] for regularization. Figure 4 (c) and (d) show the rendered images of the fitted mesh and the deformed mesh with labeled semantic regions, respectively. The Laplacian term is formulated as:

$$L_p = \|Lap(V_s, F) - Lap(V_c, F)\|_2 \,, \tag{2}$$

where $Lap(,)$ is a graph Laplacian matrix and F is the triangular face set of the 3D model. Finally, the refined mesh

Symbols	Explanation
I_s	Source image
$\{I_d\}$	Driving video
$I_{s,d}$	Coarse reenactment image
$\hat{I}_{s,d}$	Generated image
P_s	Source parsing map
$P_{s,d}$	Coarse reenactment parsing map
P_s^b	Semantic boundary of parsing map
V_s^b	Vertices on the semantic boundary of 3D face model
$\tilde{V}_s$	Full 3D face model with non-face region
$M_{c,d}$	Covisible mask
$M_{l,d}$	Boundary mask

TABLE 1
Important symbols used in this paper.

V_s is generated by optimizing V_c via the following energy function:

$$L_{total} = \lambda_p L_p + \lambda_c L_{ch}. \tag{3}$$

In our experiments, we set $\lambda_p = 10^6$ and $\lambda_c = 1$.

3.1.2 Image Meshing and Rendering

Inspired by [12], we use image meshing to estimate the depth of the non-face region (i.e., hair, background) based on the input refined mesh V_s only for the face region. Then, the output full mesh $\tilde{V}_s$ with texture map $I_s(P_s)$ is used to render coarse reenactment faces, including hair and background under various views.

To better restore the face in every pose, we leverage the points located in the boundary of V_s as boundary anchors rather than those corresponding to the 2D face contours. Since predicting the 3D-consistent parsing maps with the same identity is difficult, we extract the source artistic face and the parsing map as textures at the same time. In the subsequent discussions, $\tilde{V}_s$ denotes the mesh after image meshing. Then, according to any pose and expression coefficients, we render the corresponding 3D face model $\tilde{V}_s$ with textured I_s or P_s to get the faces $\{I_{s,d}\}$ or parsing maps $\{P_{s,d}\}$ in multiple poses and expressions.

3.2 Artistic Face Refinement

There are serious artifacts in the rough results rendered in Section 3.1.2 caused by the single-view source image. Considering that the expression changes introduce more artifacts, especially in the mouth and eyes areas, the coarse reenactment data for fine-tuning contains pose changes only. Thus, we first explain how to initially inpaint and mask these regions after rendering in Section 3.2.1. Then, we introduce the training pipeline for our personalized cGAN with these inpainting data and masks in Section 3.2.2.

3.2.1 Coarse Reenactment Result Inpainting

As shown in Figure 5 (a), the unseen regions in I_s bring obvious artifacts in new viewpoints. After the image meshing, the rendering procedure with pose changes causes artifacts. We separate such causes into three types: self-occlusions of the face, occlusions on the background, and replication of the contour lines in the source input. To remove these artifacts, we first inpaint the self-occluded regions of the face and then introduce a covisible (Cov) mask $M_{c,d}$, and a boundary (Bd) mask $M_{l,d}$. These two masks are used to erase the incorrect textures on the background and the unreasonable boundary lines on the face, respectively.

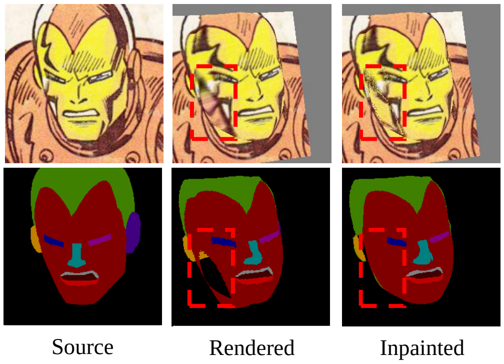


Fig. 5. An example including an artistic face and the corresponding parsing map after rendering and inpainting. The red dotted boxes highlight the inpainted area.

Inspired by [12], we use the symmetry property of a face to inpaint the missing regions due to the self-occlusions of the face region in $I_{s,d}$. The comparison of the results before and after inpainting is shown in Figure 6 (a) and (c).

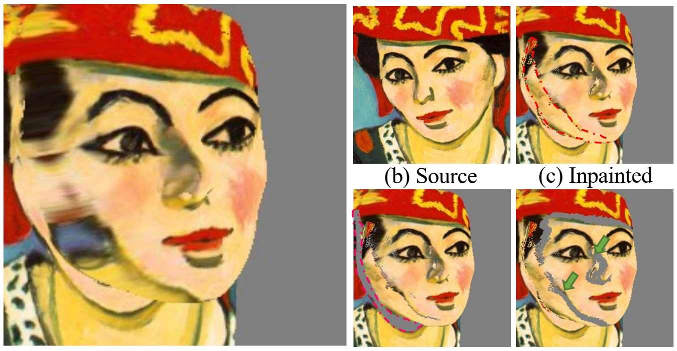
Then, by comparing the area circled by a pink dotted line in Figure 6 (d) with its corresponding region in Figure 6 (a), we can see that this region belongs to the background in the target view but the skin in the source view (Figure 6 (a)).

In the reconstructed 3D model $\tilde{V}_s$ with the texture map I_s, the triangular subset belonging to the background region will be occluded by the other triangular subset belonging to the skin region located in the same x-y coordinates at the source view. Thus, these two subsets have the same UV coordinates. When the model $\tilde{V}_s$ is observed from the target view, the unreasonable texture of this background area is exposed.

Considering that the texture of this part is invisible in the source view, we introduce a covisible (Cov) mask $M_{c,d}$ following [36] to erase such incorrect textures, as shown in Figure 6 (d) highlighted by a pink dotted line. The Cov mask is obtained by rasterizing faces in $\tilde{V}_s$ that are visible in both the source and target views, and we also regard the inpainted area as the visible part.

Finally, the phenomenon of replicated lines occurs when the pose changes. As shown in Figure 6 (a), the original contour line in I_s shifts to the inner region of the face, especially in cases where the source I_s is a profile. Thus, we design a boundary (Bd) mask $M_{l,d}$ to remove those unreasonable lines from the coarse reenactment results and thus avoid being learned by the model. First, we construct the one-to-one relationship between visible triangles of the 3D face mesh and image pixels via the rasterization of the $\tilde{V}_s$ from the estimated camera pose θ. Then, according to this relationship, we select all the triangles associated with the 2D boundary pixels P_s^b in the semantic boundary of P_s. The mask $M_{l,d}$ is produced through the projection of these triangles after they are rotated to a certain pose. As shown in Figure 6 (e)) by the green arrow, the boundary regions are erased by $M_{l,d}$. We use $M_{s,d} = M_{c,d} \cdot M_{l,d}$ to denote the combination of the two masks.

For the artifacts in the parsing maps from the rendered



(a) Rendered (b) Source (c) Inpainted (d) w/ Cov mask (e) w/ Bd mask

Fig. 6. The example for the inpainted results separately with Cov mask and Bd mask. (b) is in the source view. (a), (c), (d) and (e) are in the target view.

textured mesh, we leverage the semantics of the labeled 3DMM (Section 3.1.1). After the boundary alignment step in Section 3.1.1, the semantic map rendered in the original pose with the labeled semantic vertex textures is consistent with the P_s in the face region, as shown in Figure 4 (b) and (d). Thus, textures for invisible vertices are filled by the corresponding vertex semantic annotations. The inpainted parsing maps are shown in Figure 5 (Row 2).

3.2.2 Training Pipeline

Our personalized cGAN follows a standard generation model [37], consisting of a texture encoder, a geometry encoder, and a generator, as shown in Figure 3. The two conditions (i.e., the inputs) of the model are an input artistic image I_s to provide texture constraints, and a parsing map $P_{s,d}$ to provide geometry constraints. Then the model combines the texture and geometry information from the two conditions to synthesize a refined reenactment result. Here, we use a parsing map rather than facial landmarks as guidance, since the parsing map provides denser and richer geometry information.

The training of the model includes two stages. In the first stage, following [5], we pre-train the model on the VoxCeleb dataset [38] of various real human talking-head video sequences. We first utilize an off-the-shelf face parsing method [39] to extract a parsing map of each frame in the VoxCeleb dataset. Then, we randomly choose two frames in a video to achieve supervised training, i.e., one frame as a texture input, and the other frame and the corresponding parsing map as a ground-truth and a geometry input, respectively. The pre-training stage aims to learn the semantic alignment between the extracted texture and the parsing map via various examples.

With the pre-trained model, we observe that the output geometry is well controlled by the input parsing map, even though the geometry of the parsing map is different from that of a general human face. Additionally, the cGAN could initially extract the texture features of I_s and inject them into $P_{s,d}$ according to the associated semantics. However, the large domain gap between real human faces and artistic ones would lead to the failure of texture preservation after reenactment. Fine-tuning with the single ground-truth I_s using a style loss [40] or a cycle loss [31] might help the model learn more textures and further narrow this gap. Nevertheless, regressing multi-pose results from a single input is hard to optimize. Although the reenacted results are similar to the source artistic face, they lack the 3D consistency between each other. Details are shown in Section 4.2.2.

Hence, in the second stage, we use the inpainted reenactment data $(P_{s,d}, I_{s,d})$ and masks $M_{s,d}$ introduced in Section 3.2.1 to fine-tune the model. Since the geometry encoder is well-established, we only update the parameters of the texture encoder and the generator in this stage with a reconstruction loss L_{rec}, a content loss L_{con}, two style losses L_{sty}, and a contour loss L_{cot}, as shown in Figure 3. Both the content loss and style loss are based on the perceptual loss in [40].

The replicated lines show that the contour regions of artistic faces are not 3D-consistent, thus the rendered coarse results lack such lines. To avoid the textures in the contour regions of $I_{s,d}$ obscuring the lines that should be there, we first extract the binary boundary from $P_{s,d}$ and denote the extraction process as $EB(\cdot)$. Then, the model learns the non-contour regions (face, hair, background) and contour regions separately via $EB(P_{s,d})$, as described below.

For non-contour regions, all the reconstruction loss L_{rec}, content loss L_{con} and style loss L_{sty1} are calculated between the generated results $\hat{I}_{s,d}$ and the coarse ground-truth $I_{s,d}$, masked by $(1 - EB(P_{s,d})) \cdot M_{s,d}$, focusing on reducing the errors both in pixel and feature levels. Intuitively, although there is no ground-truth supervision for the invisible areas, our cGAN will provide an initial result for the masked regions guided by the semantic-consistency with such visible regions. Then, we utilize the other style loss L_{sty2} between the results $\hat{I}_{s,d}$ and the source face I_s, which provides a regression clue back to the texture from the original source. Following [19], L_{con} is calculated on the multi-resolutions of the paired $(\hat{I}_{s,d}, I_{s,d})$ via bilinear interpolation.

Specific to contour regions, inspired by the CycleGAN [31], with the unpaired data (synthesized contour lines extracted from the output $\hat{I}_{s,d}$ of cGAN and the source contour line), the contour loss L_{cot} forces the model to learn a mapping between the output contour lines and that of the source, as shown in the contour regions in Figure 3. The cycle-consistency for contour is, $P_{s,d}, I_s \rightarrow \hat{I}_{s,d} = G(P_{s,d}, I_s) \rightarrow G(EB(P_{s,d}) \cdot \hat{I}_{s,d}, EB(P_s) \cdot P_s) \approx EB(P_s) \cdot I_s$. Thus, the contour loss is defined as follows:

$$L_{cot} = \left\| G(EB(P_{s,d}) \cdot \hat{I}_{s,d}, EB(P_s) \cdot P_s) - EB(P_s) \cdot I_s \right\|_1 . \tag{4}$$

The total loss for the fine-tuning stage is:

$$\begin{aligned} L_{total} =& \lambda_r L_{rec} + \lambda_c L_{con} + \lambda_{s1} L_{sty1} \\ & + \lambda_{s2} L_{sty2} + \lambda_{ct} L_{cot}. \end{aligned} \tag{5}$$

In our experiments, we set $\lambda_r = 20$, $\lambda_c = 2.0$, $\lambda_{s1} = 300$, $\lambda_{s2} = 600$, and $\lambda_{ct} = 2000$.

All of the above procedures enable our personalized cGAN to inpaint the unreasonable textures masked by $M_{s,d}$ and the gray background regions lacking background textures (Figure 6 (a)). Additionally, to a certain extent, the cGAN makes the textures of self-occluded areas originally inpainted by symmetrical parts visually more natural and adds stylized lines of artistic faces in the contour regions.

4 Experiments

All the experiments are implemented with PyTorch on a PC with Intel i7-11700 CPU, 64 GB RAM, and a single 3090Ti GPU. To demonstrate the suitability of our pipeline for various artistic portraits, we choose the Artistic-Faces dataset [41] containing 160 artistic portraits, which represent a wide range of 16 artistic styles (e.g., cubism, comics, impressionism). In this paper, we first train cGAN on the VoxCeleb [38] dataset as our pre-trained model with 200 epochs, a learning rate of 10^{-4}, and a batch size of 8. The pre-processing method for VoxCeleb is described in [18]. For each artistic input, we then fine-tune such a model with 2K iterations and the same learning rate and batch size as before. The resolution of the input image and parsing map is 224×224. The number of iterations for fine-tuning has to be adjusted according to the similarity between the artistic real human faces. We suggest 1K iterations for human-like samples (Figure 7 (Rows 1 and 2)) and 2K iterations for imaginative samples (Figure 7 (Rows 3 and 4)). Our method takes 30 minutes for each 1K iterations to fine-tune on a single GPU.

4.1 Comparison with State-of-the-art Methods

We compare our ReenactArtFace with the state-of-the-art reenactment methods including FOMM [18], PIRenderer [19], Bi-layer [2], NeuralHead [23], MRAA [42], LIA [43] in terms of the visual quality including identity preservation, the accuracy of poses and expressions of the reenacted faces and contour lines. For quantitative comparison, we use $Fréchet\ Inception\ Distance$ [44] (FID) to measure the realism of the synthesized results via the similarity distribution between such results and the source artistic faces. Furthermore, cosine similarity (CSIM) of the embedding vectors from ArcFace [45] is used to evaluate the identity preservation between the reenacted faces and the provided faces. Note that with the same input of an artistic face, only NeuralHead [23] is fine-tuned with the single artistic portrait, and the other methods are only trained on the human face dataset, since they do not include a fine-tuning stage.

4.1.1 Qualitative Comparison

There are several representative results shown in Figure 7, in which the driving images contain different genders with various expressions and poses, and the source artistic faces are in diverse styles, ranging from human-like samples to imaginative samples. Our results show the best identity preservation and the most relevant expressions and poses to those of the driving images.

For human-like samples (Figure 7 (Rows 1 and 2)), all the compared methods generate reasonable results with roughly correct poses/expressions. However, the existing methods lack certain texture details such as the texture inside the eyes. In addition, compared with the source face, the geometry of the results by these methods changes a lot, such as the distance between two eyes (Row 1), the face shape especially for Bi-layer and NeuralHead. The expression change for LIA is not noticeable enough because the motion directions learned on the real human face are not suitable for the artistic faces. Thanks to the boundary-guided deformation for the 3D artistic face model and the personalized fine-tuning for cGAN, our method can preserve the identity the best.

For imaginative samples (Figure 7 (Rows 3 and 4)) with a large domain gap of human faces, all the alternative methods fail to preserve the identity and transfer the correct poses/expressions. FOMM, MRAA and PIRenderer can better retain color information. However, they completely distort the source geometry to fit the driving faces because of the large geometry between the artistic and real human faces in the training data. Bi-layer and LIA, respectively, produce unrelated results and obvious artifacts since their latent identity embeddings are far away from those learned in the human dataset. The results of NeuralHead fine-tuned with a single image are still not enough to restore the identities, and their expressions/poses are more like the source faces than the driving ones because their fine-tuning is easily overfitting (e.g., the third examples (Row 3)). Due to the explicit disentanglement of geometry and texture in our network, our method not only provides strong geometric guidance (reenacted parsing maps) in which expressions and poses are rigged by 3DMM, but also uses the corresponding multi-view coarse faces to learn a personalized neural renderer (fine-tuned cGAN). Thus, our results can not only better preserve the input identities, but also transfer the poses/expressions from the driving more faithfully.

Taking a closer look at contour lines, FOMM, MRAA and PIender keep a part of lines, but the inaccurate warp fields cause that to distort and break. It seems that the contour lines are washed away in Bi-layer, LIA and NeuralHead, mainly because there is no contour style in real human faces. Since our method explicitly learns the mapping of contour, the boundaries of our results are satisfactory.

Warped-based methods [18], [19] preserve more texture details of the source image. We attempt to fine-tune this kind of methods. However, it is unreasonable to fine-tune FOMM, because this method trains a keypoint detector neural network. There is a large gap between the keypoints of real human faces and artistic faces. After the simple fine-tuning, the keypoint detector will be suitable for artistic faces, but fail to extract the accurate keypoints for driving human faces. Ideally, the fine-tuning stage can be implemented for PIRenderer because it uses 3DMMs as motion descriptors for expressions and poses. Hence, we also fine-tune PIRenderer on artistic faces, and the results are shown in Supplementary Material Section B.

Although there is no special design for video stability in our pipeline, fine-tuning with 3D-consistent coarse images helps our model to achieve temporally coherent results. Without considering for the identity preservation, FOMM and Bi-layer generate smooth videos. The videos of PIRenderer, MRAA, and LIA are incoherent due to the varying geometry and unpredictable artifacts across frames. The overfitting of NeuralHead leads to a sudden change of frames in a video, as shown in the accompanying video.

4.1.2 Quantitative Comparison

We conducted the quantitative comparison on 160 images of the whole Artistic-Faces dataset. For each source input, we used 453 driving human videos from the VoxCeleb

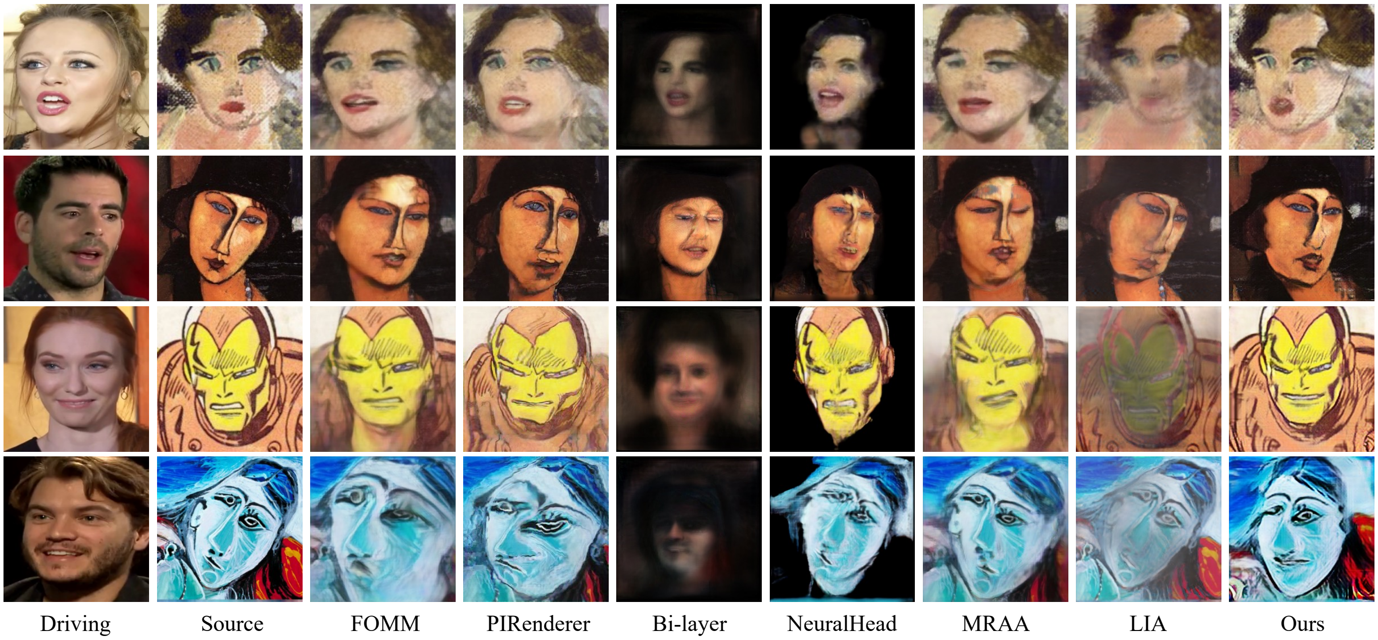


Fig. 7. Comparisons with the existing face reenactment methods. Our method shows the best identity preservation and pose transferring.

Method	CSIM ↑	FID ↓
FOMM [18]	0.49	112.29
PIRenderer [19]	0.55	101.06
Bi-layer [2]	0.42	322.75
NeuralHead [23]	0.44	165.99
MRAA [42]	0.54	103.06
LIA [43]	0.55	114.49
Ours_coarse	0.56	108.37
Ours	**0.58**	**89.52**

TABLE 2
Quantitative comparison between six existing methods and our ReenactArtFace on the Artistic-Faces dataset.

test set. Table 2 shows the quantitative comparison results, and our method significantly outperforms the compared methods in terms of FID and CSIM. That indicates that our results are the most similar to the ground truth on the deep features level. This is consistent with our findings based on the qualitative comparisons. Bi-layer get the worst results because it generates unfaithful faces for most artistic portraits (Figure 7 (Column 5)). CSIM is based on the face recognition model [45], so it focuses more on local areas of the face that are easier to identify rather than the whole face. Since our method is based on the coarse reenactment results, the increase of CSIM is little between our and our_coarse. The CSIM of LIA is relatively high, but it is meaningless since their results exhibit noticeable artifacts without obvious changes in poses and expressions (Figure 7 (Rows 3 and 4)). The results of PIRenderer and MRAA are close to ours, since they keep a lot of textures for faces and backgrounds via warping the source input.

4.2 Ablation Study

We report some ablation studies of our method in terms of the effectiveness of the 3D Artistic Face Reconstruction module in Section 4.2.1, the effectiveness of the Artistic Refinement module in Section 4.2.2, and the effectiveness of the contour loss in Section 4.2.3.

4.2.1 Effectiveness of 3D Artistic Face Reconstruction

We leverage parsing maps to preserve the geometry of input artistic faces. Since these parsing maps are rendered from the 3D mesh $\tilde{V}_s$ and inpainted based on its semantics (Section 3.2.1), the reconstructed mesh influences the geometry accuracy of results. To evaluate the effectiveness of the 3D Artistic Face Reconstruction module (Section 3.1), we compare results generated with and without the deformation step (Section 3.1.1).

Figure 8 shows representative results of the paired reenacted parsing maps and the corresponding generated artistic faces, with and without the deformation step. The shape of each facial semantic part (eyes, mouth, nose) without deformation (Figure 8 (b)) exhibits the characteristics of real human faces. For example, rectangular eyes ((a)-Bottom Row) are fitted as oval eyes ((b)-Bottom Row). Additionally, compared with the source faces (Figure 8 (a)), the sizes and proportions of these parts and their relative locations are not maintained (see the short nose and two symmetrically positioned eyes in (b)-Top Row). This issue is mainly because the identity bases in 3DMM are extracted from real human faces and the 68 landmarks commonly used on real human faces are too sparse for artistic faces. In contrast, the deformed 3D meshes guarantee a better geometry representation of the artistic faces (Figure 8 (c)).

4.2.2 Effectiveness of Artistic Refinement

Since a large artistic portrait dataset with a similar style both in texture and geometry is unachievable, one-shot reenactment training on the human face dataset is a feasible way to reenact several human-like artistic faces. Meanwhile, the fine-tuning procedure plays a key role in breaking the gap between the real human face domain and the artistic

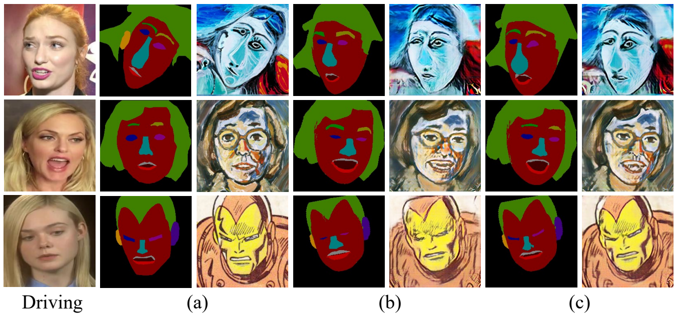


Fig. 8. The ablation study for the 3D Artistic Face Reconstruction module. Each example in (a) shows an input artistic face and its corresponding parsing map. The reenacted parsing maps and faces without and with the deformation step are given in (b) and (c), respectively.

face domain. To evaluate the effectiveness of the Artistic Refinement Module, we perform a set of ablation studies as shown in Figure 9, including the coarse reenactment data after inpainting, the cGAN without the fine-tuning stage, fine-tuned with the single source via the style loss, fine-tuned with the single source via the cycle loss, and fine-tuned with the coarse reenactment data respectively. Since there is no directly pixel-level relationship between the synthesized results and the source input (in different poses), the content loss and reconstruction loss cannot be used to fine-tune with one-shot setting.

As shown in Figure 9, only the results from our Artistic Refinement Module achieve identity consistency with the source. While the pre-trained model can only synthesize a real face, for the human-like sample (Row 1), it extracts the color of the face and hair, but for samples (Rows 2 and 3) with large deviations from real human faces, the pre-trained cGAN only transfers the color of the source background. Intuitively, the reason is that the texture variance of the background region is large, while that of the human face is very limited in the human dataset. Thus, the module regards such samples as the whole background. Although we provide the initially inpainting for the coarse reenactment faces (Column 3), there are still artifacts in the eyes, mouth, side face and stitching regions after completing with the symmetrical texture. In addition, it is obvious to see the textures losing in the background. Fine-tuned with the style loss (Column 5) provides more related texture but still loses some details. Meanwhile, when the pose changes greatly (Rows 2 and 3), the reenacted faces look like the copy of the source except for the nose, eyes and mouth, which is due to the lack of 3D supervision. Fine-tuned with the cycle loss (Column 6) fails to produce clear results because the many-to-one (multi-pose fake images to a single ground truth) optimization will easily fall into local optimum. The network does not learn the specific texture characteristics for each reenacted parsing map, but simply provides an easy way to restore the source artistic face. Thus, the effectiveness of our Artistic Refinement Module is demonstrated.

Our personalized cGAN is supervised with the source artistic face and a large number of coarse reenactment faces via different losses. We evaluate the effectiveness of each loss by testing the performance of our model trained in various ways, including trained only with the source artistic face (w/ L_{sty2}), trained with the coarse reenactment faces (w/ $L_{sty1} + L_{con}$), trained with both of them (w/ $L_{sty2} + L_{sty1} + L_{con}$), and trained with all the losses (w/ $L_{sty2} + L_{sty1} + L_{con} + L_{rec}$). Since the previous works [19], [46], [47] generally calculate the perceptual loss (i.e., the addition of the content loss and style loss) with the same ground truth, we regard the L_{sty1} and L_{con} as a whole in our ablation study.

As shown in Figure 10, training with the source face via L_{sty2} achieves semantic style transfer but fails to preserve the identity (Top and Middle Rows). That means that a single source artistic face is not enough for training. When we provide the coarse reenactment faces as the ground truth to calculate L_{sty1} and L_{con}, the identity consistency improves a lot, but the textures are rough, especially in the regions that are not present in the source view (Top and Bottom Rows). This is possibly because the incorrect textures of $\tilde{V}_s$ are masked by $M_{c,d}$ (Section 3.2.1). Combining the L_{sty2} with the L_{sty1} and L_{con} can ensure the quality of results. Then, the incorporation of L_{rec} can better supplement low-frequency information such as lines (Top Row) and color (Bottom Row), thus further improving visual quality. In short, L_{sty1} and L_{con} are the most effective in training the personalized cGAN while L_{sty2} and L_{rec} can help polish the results.

4.2.3 Effectiveness of Contour Loss

Since the contour lines are an important symptom of artistic face images and there is no existing work to discuss it, we propose the contour loss L_{cot} to supervise the training of contour regions extracted from $EB(P_{s,d})$.

To validate the efficiency of the contour loss, we first ablate the binary contour mask $EB(P_{s,d})$ and then with such a mask, we provide a dedicated supervision for contour regions from the source contour image $(EB(P_s) \cdot I_s)$ by the style loss and our contour loss. Figure 11 shows the results separately produced from the same structured cGANs but trained with different losses.

Without special consideration for the contour regions (Column 2), the results show weaker boundaries. This is because without the 3D consistency, the contour lines do not exist in coarse reenactment data and the strong guidance of the contour regions in such data makes the model learn wrong information. Thus, in the follow-up experiments, we use $EB(P_{s,d})$ to hide these regions in coarse data and learn the reenacted contour lines only from such regions in the source. Supervised by the style loss (Column 3), the model seems to generate a lot of line segments instead of smooth lines. Such lines disappear for parts, as seen in the one side of the face (Row 2). We speculate that the style loss helps the model obtain the overall style of color and texture, but ignores the whole structure of the contour in the source. Hence, we design the contour loss. From the results (Column 4), it can be found that our method can synthesize faithful lines that match the contour style of the original artistic face.

5 Applications

The personalized GAN model trained by our method can be applied to various applications, such as image editing,

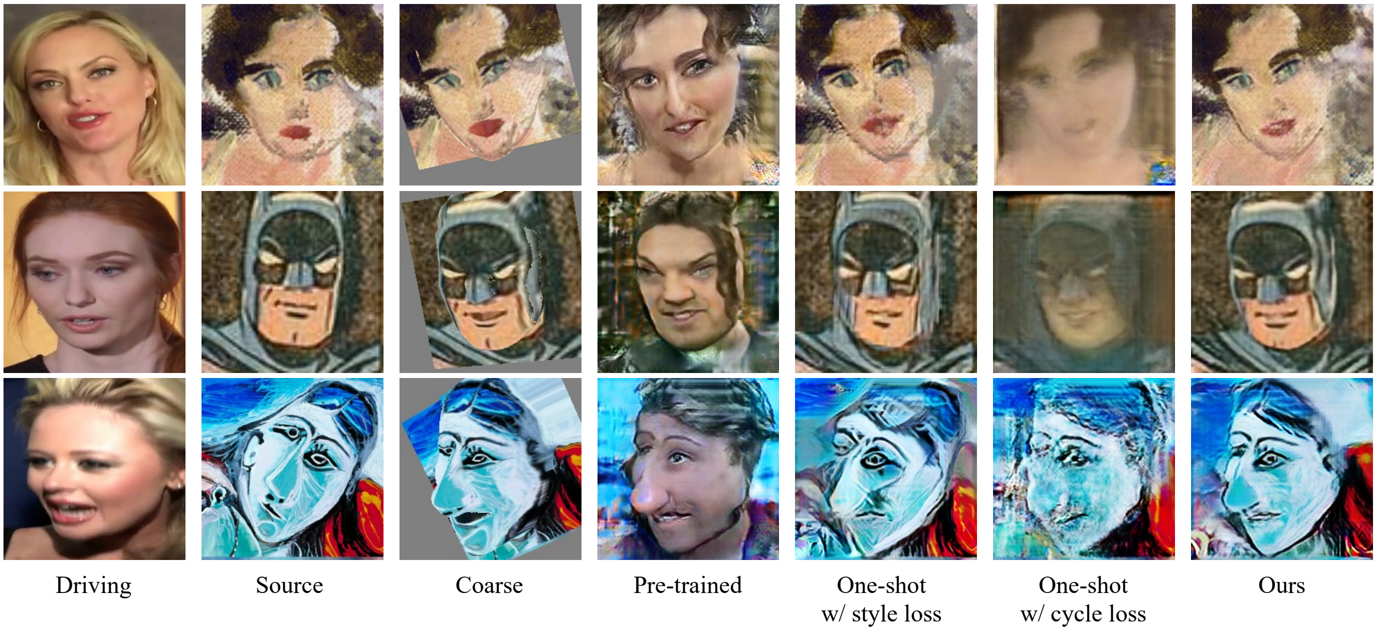


Fig. 9. Comparison of the results generated from our model variants with different settings given the same inputs. From column 3 to column 7: the textured face model (coarse), the cGAN without fine-tuning (pre-trained), fine-tuned with the single input (one-shot) with the style loss, fine-tuned with the single input (one-shot) with the cycle loss, and fine-tuned with both the single input and multiple coarse data (our final results). It is obvious that the coarse reenactment faces can greatly help cGAN to learn 3D-consistent texture distribution.

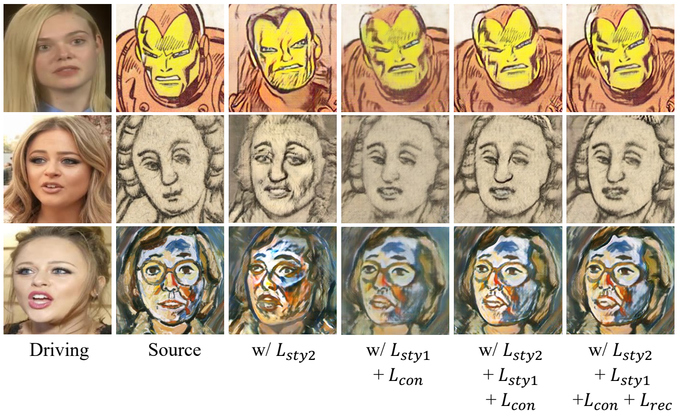

Driving
Source
w/ L_{sty2}
w/ L_{sty1} + L_{con}
w/ L_{sty2} + L_{sty1} + L_{con}
w/ L_{sty2} + L_{sty1} +L_{con} + L_{rec}

Fig. 10. The ablation study for the loss terms in the Artistic Refinement module.

image/video stylization, cartoon characters animation, etc. In this section, we show two typical applications in details.

5.1 Artistic Portrait Editing

The recent state-of-the-art editing methods [48], [49], [50] are based on StyleGAN [29]. They are trained from a large and diverse dataset which contains various identities, editing via the direction in the latent space. The results from the random latent codes are impressive. But given a specific portrait, the latent code generated from GAN inversion does not fully preserve facial features. Hence, using such a latent code for editing suffers from the identity preservation problems.

In this paper, we train a personalized cGAN with 3D prior for each artistic identity via coarse reenactment results. Then, the editing of geometry attributes for such an artistic face is directly determined by the parsing maps, including implicitly or explicitly control. The former is to control the expression/pose coefficients, and then our model produces the corresponding parsing map and result in turn. The latter is to subjectively edit the parsing map, considering everyone has some desired ideas for their beloved artistic characters, such as the exaggerated expressions and the shape of the face. With the input of a user-specified parsing map, our cGAN outputs the editing face. Several editing parsing maps and results are shown in Figure 12.

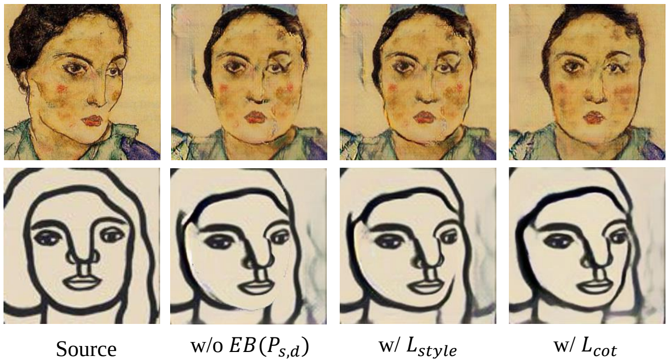


Fig. 11. The ablation study for the contour loss.

5.2 Face Video Stylization

Face Video Stylization task aims to smoothly transfer the style of an artistic image to each frame of a driving face video. Different from transferring the whole texture style using Adaptive Instance Normalization (AdaIN) [51], our model transfers the style in a semantically meaningful manner as [48], [49], [52] by using the parsing maps of the

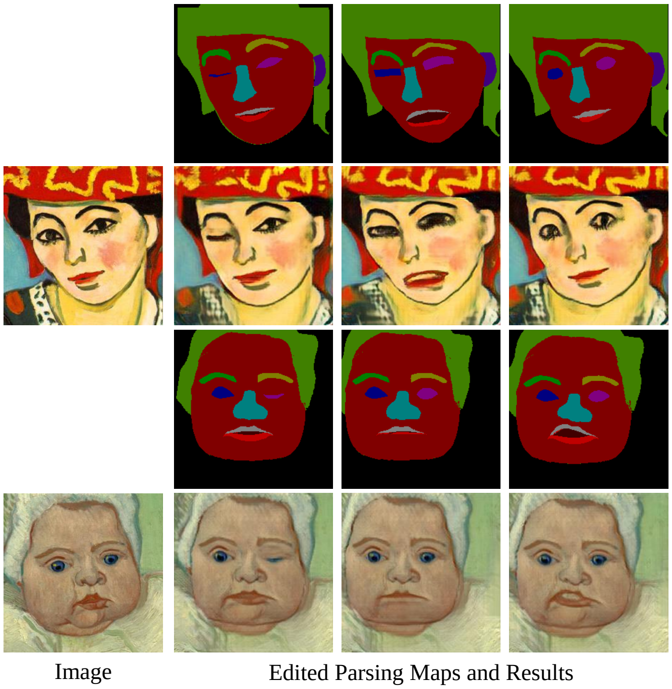


Fig. 12. Application of artistic portrait editing. The top example is edited by a user in the source view. The bottom example is controlled by the expression coefficients in the front view. In each example, the first row is the input edited parsing maps and the second row is the edited artistic faces.

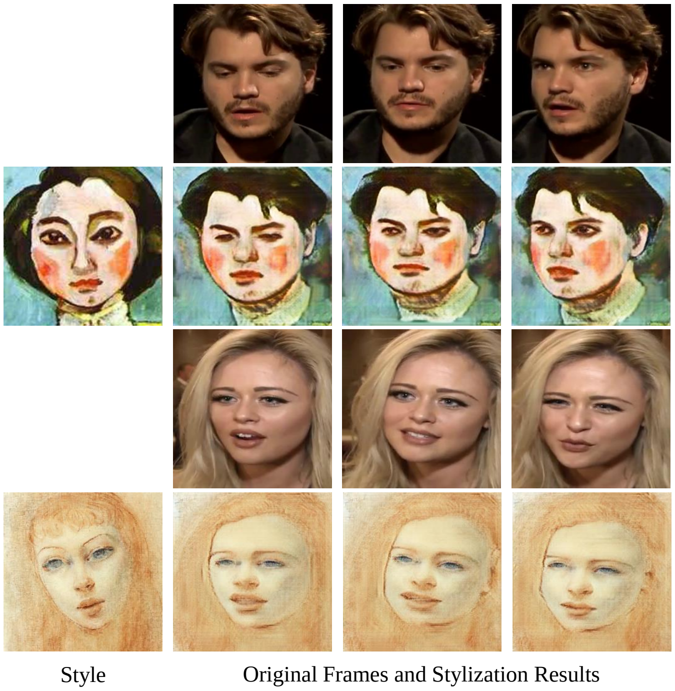


Fig. 13. Application of face video stylization. In each example, the first and second rows are three frames of a face video and the corresponding sylization results, respectively.

driving video to guide the personalized cGAN. We start to train a style-customized cGAN for the provided artistic texture image. We then use the method in [39] to predict a parsing map for each frame of the input video. Finally, such cGAN predicts stylized frames with the inputs of these corresponding parsing maps. Two stylization examples are shown in Figure 13.

6 Conclusions and Discussions

We presented ReenactArtFace, a novel coarse-to-fine pipeline for reenacting single artistic face images with various styles. To address varying styles of artistic images and the lack of a large artistic face image dataset, we proposed to use the reconstructed 3D model to render a large number of coarse results as the guide, enabling the personalized cGAN training. Our technique might promote the development of cartoon production in the future.

Additionally, we for the first time have tried to handle the specific contour style of 2D artistic faces and proposed a feasible solution for such a style. Extensive experiments have demonstrated that our ReenactArtFace generates the reenacted artistic faces preserving both exaggerated geometry and the imaginative texture consistency with the source input and outperforms the existing methods. The flexible control of inputs and attractive results from the applications show the practicality and significance of our method.

But there are still some limitations in our method. First, the blank background regions after pose transfer without corresponding inpainting textures are only supervised by the source input via the style loss. Although it can bring reasonable results, it can also produce ambiguity and artifacts, especially when the area is large, as shown in Figure 14 (Top). This issue might be addressed by modeling the background area and the other areas (face, hair, upper body) separately at the image meshing and then treating the background static rather than rotating with the head pose to reduce the area of missing textures. Second, we directly use the symmetry texture to inpaint without considering the relationship between light and shade in the initial pose. Especially for some realistic oil painting portraits, this simple copy-and-paste leads to discordant results, as shown in Figure 14 (Bottom). With such supervision, our cGAN alleviates this issue, but the results are still not satisfactory. A possible direction is to use Poission blending [53] in the inpainting step, and the choice of boundary constraints without the pixels located in the contour lines is important. Another limitation is that we deal with the contour style by extracting these regions from the boundaries of the parsing map. Thus, our method can only get boundaries of uniform thickness and assume that the contour line style is consistent across the extracted areas. For complex artworks, the style of contour lines with inconsistent thickness and various texture is common (e.g., the first example in Figure 9). Thus, how to model the contour and interior of an artistic face separately and then seamlessly merge them would be a worthy research direction. Lastly, our model requires manual annotation for the landmark and parsing map. Since our method requires rendering coarse reenactment results at each iteration during the fine-tuning stage, the whole process for reenacting an artistic face is time-consuming. In the future, we are interested in reducing or even eliminating the dependency on such auxiliary information, modeling exaggerated expressions of artistic portraits, and decreasing the fine-tuning time. Additionally, artistic faces by different artists exhibit very different styles. Since the numbers of images with similar or same styles are small in our current dataset, we treat all the artistic faces as one category in our current FID evaluation. Hence, we also are interested in constructing a larger dataset of artistic images to enable a more thorough evaluation of our proposed technique (e.g., calculating FID scores for each category of images with similar or same styles.

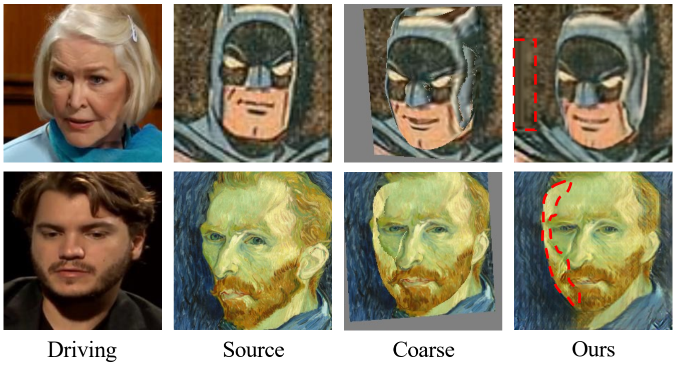


Fig. 14. Two less successful examples. The top row shows the artifacts appearing in the background regions highlighted by the red dotted box, while the bottom row shows the unnatural assembling result between the red dotted region and the original part.

Acknowledgments

We thank the anonymous reviewers for their constructive comments. This work was supported by unrestricted gifts from Adobe and the Centre for Applied Computing and Interactive Media (ACIM) of the School of Creative Media, City University of Hong Kong.

References

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[2] E. Zakharov, A. Ivakhnenko, A. Shysheya, and V. Lempitsky, "Fast bi-layer neural synthesis of one-shot realistic head avatars," in *European Conference on Computer Vision*, pp. 524–540, Springer, 2020.

[3] T.-C. Wang, A. Mallya, and M.-Y. Liu, "One-shot free-view neural talking-head synthesis for video conferencing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10039–10049, 2021.

[4] G. Yao, Y. Yuan, T. Shao, and K. Zhou, "Mesh guided one-shot face reenactment using graph convolutional networks," in *Proceedings of the 28th ACM International Conference on Multimedia*, pp. 1773–1781, 2020.

[5] E. Zakharov, A. Shysheya, E. Burkov, and V. Lempitsky, "Few-shot adversarial learning of realistic neural talking head models," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9459–9468, 2019.

[6] S. Ha, M. Kersner, B. Kim, S. Seo, and D. Kim, "Marionette: Few-shot face reenactment preserving identity of unseen targets," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 10893–10900, 2020.

[7] P. Sun, Y. Li, H. Qi, and S. Lyu, "Landmarkgan: Synthesizing faces from landmarks," *arXiv preprint arXiv:2011.00269*, 2020.

[8] P.-H. Huang, F.-E. Yang, and Y.-C. F. Wang, "Learning identity-invariant motion representations for cross-id face reenactment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7084–7092, 2020.

[9] J. Zhang, X. Zeng, M. Wang, Y. Pan, L. Liu, Y. Liu, Y. Ding, and C. Fan, "Freenet: Multi-identity face reenactment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5326–5335, 2020.

[10] S. Tripathy, J. Kannala, and E. Rahtu, "Facegan: Facial attribute controllable reenactment gan," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 1329–1338, 2021.

[11] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter, "A 3d face model for pose and illumination invariant face recognition," in *2009 sixth IEEE international conference on advanced video and signal based surveillance*, pp. 296–301, Ieee, 2009.

[12] X. Zhu, Z. Lei, J. Yan, D. Yi, and S. Z. Li, "High-fidelity pose and expression normalization for face recognition in the wild," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 787–796, 2015.

[13] S. Rao, R. Ortiz-Cayon, M. Munaro, A. Liaudanskas, K. Chande, T. Bertel, C. Richardt, A. JB, S. Holzer, and A. Kar, "Free-viewpoint facial re-enactment from a casual capture," in *SIGGRAPH Asia 2020 Posters*, pp. 1–2, 2020.

[14] H. Kim, P. Garrido, A. Tewari, W. Xu, J. Thies, M. Niessner, P. Pérez, C. Richardt, M. Zollhöfer, and C. Theobalt, "Deep video portraits," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, pp. 1–14, 2018.

[15] H. Averbuch-Elor, D. Cohen-Or, J. Kopf, and M. F. Cohen, "Bringing portraits to life," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 6, pp. 1–13, 2017.

[16] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2face: Real-time face capture and reenactment of rgb videos," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2387–2395, 2016.

[17] Y. Zhou, X. Han, E. Shechtman, J. Echevarria, E. Kalogerakis, and D. Li, "Makelttalk: speaker-aware talking-head animation," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 6, pp. 1–15, 2020.

[18] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe, "First order motion model for image animation," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[19] Y. Ren, G. Li, Y. Chen, T. H. Li, and S. Liu, "Pirenderer: Controllable portrait image generation via semantic neural rendering," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13759–13768, 2021.

[20] Y. Zhang, S. Zhang, Y. He, C. Li, C. C. Loy, and Z. Liu, "One-shot face reenactment," *arXiv preprint arXiv:1908.03251*, 2019.

[21] O. Fried, A. Tewari, M. Zollhöfer, A. Finkelstein, E. Shechtman, D. B. Goldman, K. Genova, Z. Jin, C. Theobalt, and M. Agrawala, "Text-based editing of talking-head video," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 4, pp. 1–14, 2019.

[22] Y. Lu, J. Chai, and X. Cao, "Live speech portraits: real-time photorealistic talking-head animation," *ACM Transactions on Graphics (TOG)*, vol. 40, no. 6, pp. 1–17, 2021.

[23] E. Burkov, I. Pasechnik, A. Grigorev, and V. Lempitsky, "Neural head reenactment with latent pose descriptors," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13786–13795, 2020.

[24] J. Deng, S. Cheng, N. Xue, Y. Zhou, and S. Zafeiriou, "Uv-gan: Adversarial facial uv map completion for pose-invariant face recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7093–7102, 2018.

[25] K. Nagano, H. Luo, Z. Wang, J. Seo, J. Xing, L. Hu, L. Wei, and H. Li, "Deep face normalization," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 6, pp. 1–16, 2019.

[26] X. Wen, M. Wang, C. Richardt, Z.-Y. Chen, and S.-M. Hu, "Photorealistic audio-driven video portraits," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 12, pp. 3457–3466, 2020.

[27] X. Tu, Y. Zou, J. Zhao, W. Ai, J. Dong, Y. Yao, Z. Wang, G. Guo, Z. Li, W. Liu, *et al.*, "Image-to-video generation via 3d facial dynamics," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[28] B. Gecer, J. Deng, and S. Zafeiriou, "Ostec: one-shot texture completion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7628–7638, 2021.

[29] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4401–4410, 2019.

[30] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[31] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232, 2017.

[32] A. Pumarola, A. Agudo, A. M. Martinez, A. Sanfeliu, and F. Moreno-Noguer, "Ganimation: Anatomically-aware facial animation from a single image," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 818–833, 2018.

[33] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8789–8797, 2018.

[34] H. Zhou, J. Liu, Z. Liu, Y. Liu, and X. Wang, "Rotate-and-render: Unsupervised photo realistic face rotation from single-view images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5911–5920, 2020.

[35] O. Sorkine, D. Cohen-Or, Y. Lipman, M. Alexa, C. Rössl, and H.-P. Seidel, "Laplacian surface editing," in *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pp. 175–184, 2004.

[36] J. Shang, T. Shen, S. Li, L. Zhou, M. Zhen, T. Fang, and L. Quan, "Self-supervised monocular 3d face reconstruction by occlusion-aware multi-view geometry consistency," in *European Conference on Computer Vision*, pp. 53–70, Springer, 2020.

[37] M.-Y. Liu, X. Huang, A. Mallya, T. Karras, T. Aila, J. Lehtinen, and J. Kautz, "Few-shot unsupervised image-to-image translation," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10551–10560, 2019.

[38] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[39] Y. Lin, J. Shen, Y. Wang, and M. Pantic, "Roi tanh-polar transformer network for face parsing in the wild," *Image and Vision Computing*, vol. 112, p. 104190, 2021.

[40] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European conference on computer vision*, pp. 694–711, Springer, 2016.

[41] J. Yaniv, Y. Newman, and A. Shamir, "The face of art: landmark detection and geometric style in portraits," *ACM Transactions on graphics (TOG)*, vol. 38, no. 4, pp. 1–15, 2019.

[42] A. Siarohin, O. J. Woodford, J. Ren, M. Chai, and S. Tulyakov, "Motion representations for articulated animation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13653–13662, 2021.

[43] Y. Wang, D. Yang, F. Bremond, and A. Dantcheva, "Latent image animator: Learning to animate images via latent space navigation," *arXiv preprint arXiv:2203.09043*, 2022.

[44] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *Advances in neural information processing systems*, vol. 30, 2017.

[45] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4690–4699, 2019.

[46] Z. Chen, C. Wang, B. Yuan, and D. Tao, "Puppeteergan: Arbitrary portrait animation with semantic-aware appearance transformation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13518–13527, 2020.

[47] A. Chen, R. Liu, L. Xie, Z. Chen, H. Su, and J. Yu, "Sofgan: A portrait image generator with dynamic styling," *ACM Transactions on Graphics (TOG)*, vol. 41, no. 1, pp. 1–26, 2022.

[48] A. Selim, M. Elgharib, and L. Doyle, "Painting style transfer for head portraits using convolutional neural networks," *ACM Transactions on Graphics (ToG)*, vol. 35, no. 4, pp. 1–18, 2016.

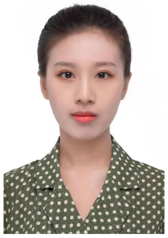
[49] J. Fišer, O. Jamriška, D. Simons, E. Shechtman, J. Lu, P. Asente, M. Lukáč, and D. Sỳkora, "Example-based synthesis of stylized facial animations," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, pp. 1–11, 2017.

[50] D. Futschik, M. Kučera, M. Lukáč, Z. Wang, E. Shechtman, and D. Sỳkora, "Stalp: Style transfer with auxiliary limited pairing," in *Computer Graphics Forum*, vol. 40, pp. 563–573, Wiley Online Library, 2021.

[51] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proceedings of the IEEE international conference on computer vision*, pp. 1501–1510, 2017.

[52] A. Texler, O. Texler, M. Kučera, M. Chai, and D. Sỳkora, "Faceblit: Instant real-time example-based style transfer to facial videos," *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, vol. 4, no. 1, pp. 1–17, 2021.

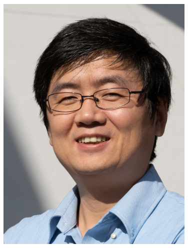
[53] P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," in *ACM SIGGRAPH 2003 Papers*, pp. 313–318, 2003.



Linzi Qu received her B.S. degree in Electronic Engineering from XiDian University in 2018. She is currently a PhD candidate in the School of Creative Media, City University of Hong Kong. Her research interests lie in computer graphics and computer vision.

Jiaxiang Shang is a Ph.D. candidate in the Department of Computer Science and Engineering at the Hong Kong University of Science and Technology. Before joining HKUST, he had his Bachelor degree in the Department of Computer Science and Engineering at Sun Yat-sen University. His primary research topic is about face reconstruction.

Xiaoguang Han is now an Assistant Professor and President Young Scholar of the Chinese University of Hong Kong (Shenzhen) and the Future Intelligence Network Research Institute. He received his PhD degree from the University of Hong Kong in 2017. His research interests include computer vision and computer graphics. He has published nearly 50 papers in well-known international journals and conferences, including top conferences and journals SIGGRAPH (Asia), IEEE TVCG, CVPR, ICCV, ECCV, NeurIPS, ACM TOG, etc. He is currently a guest editor of Frontiers of Virtual Reality, and also an associate editor of the journal of Computers & Graphics.



Hongbo Fu received a BS degree in information sciences from Peking University, China, in 2002 and a PhD degree in computer science from the Hong Kong University of Science and Technology in 2007. He is a Full Professor at the School of Creative Media, City University of Hong Kong. His primary research interests fall in the fields of computer graphics and human computer interaction. He has served as an Associate Editor of The Visual Computer, Computers & Graphics, and Computer Graphics Forum.