Virtual Reality & Intelligent Hardware

•Article•

August 2024 Vol. 6 No. 4: 292–307
10.1016/j.vrih.2023.06.010

# Generating animatable 3D cartoon faces from single portraits

Chuanyu PAN[1*], Guowei YANG[2], Taijiang MU[2], Yu-Kun LAI[3]

1. *Department of Electrical Engineering and Computer Sciences, University of California Berkeley, Berkeley CA* 94704*, USA;*
2. *Department of Computer Science and Technology, Tsinghua University, Beijing* 100084*, China;*
3. *School of Computer Science and Informatics, Cardiff University, Cardiff CF*24 4*AG, UK*

**Received** 16 February 2023; **Revised** 20 May 2023; **Accepted** 20 June 2023

**Abstract: Background** With the development of virtual reality (VR) technology, there is a growing need for customized 3D avatars. However, traditional methods for 3D avatar modeling are either time-consuming or fail to retain the similarity to the person being modeled. This study presents a novel framework for generating animatable 3D cartoon faces from a single portrait image. **Methods** First, we transferred an input real-world portrait to a stylized cartoon image using StyleGAN. We then proposed a two-stage reconstruction method to recover a 3D cartoon face with detailed texture. Our two-stage strategy initially performs coarse estimation based on template models and subsequently refines the model by nonrigid deformation under landmark supervision. Finally, we proposed a semantic-preserving face-rigging method based on manually created templates and deformation transfer. **Conclusions** Compared with prior arts, the qualitative and quantitative results show that our method achieves better accuracy, aesthetics, and similarity criteria. Furthermore, we demonstrated the capability of the proposed 3D model for real-time facial animation.

**Keywords:** 3D reconstruction; Cartoon face reconstruction; Face rigging; Stylized reconstruction; Virtual reality

**Citation:** Chuanyu PAN, Guowei YANG, Taijiang MU, Yu-Kun LAI. Generating animatable 3D cartoon faces from single portraits. Virtual Reality & Intelligent Hardware, 2024, 6(4): 292–307.

## 1 Introduction

Virtual and augmented reality (VR/AR) has rapidly developed in recent years, in which creating virtual 3D faces and avatars for users is an essential and challenging task. These faces should achieve high performance in terms of aesthetics and recognizability, resembling the person being modeled. They should also be animatable for numerous downstream applications. However, traditional methods either require time-consuming heavy manual modeling or rely on existing general templates, which can result in losing recognizability. With the development of deep learning techniques, a few methods[1−3] that automatically reconstruct realistic 3D faces from images have been proposed. However, owing to numerous facial details,

*Corresponding author, chuanyu_pan@berkeley.edu

Peer review under the responsibility of Beijing Zhongke Journal Publishing Co.Ltd.

high similarity to the reference face is considerably hard to achieve in realistic 3D reconstruction. In comparison, cartoon faces can easily achieve high visual performance and can be represented with less memory. Therefore, many VR/AR applications use 3D cartoon faces as avatars for user images.

Our study focused on automatically creating 3D animatable cartoon faces based on a single real-world portrait. As shown in Figure 1, we split our pipeline into the following steps. First, we generated a stylized cartoon image from the input image using StyleGAN[4]. Subsequently, we reconstructed a static 3D cartoon face from the stylized image. Finally, we generated semantic-preserving facial rigs to animate the 3D face.

Existing face reconstruction methods[1,2] perform poorly at reconstructing cartoon faces because they introduce strong real-world priors that are difficult to generalize to the cartoon domain. Some studies[5] that reconstructed 3D caricatures also failed to perform well on real-world portrait images owing to domain gaps. However, to obtain accurate texture mapping and natural facial animation, precise correspondences between the reconstructed 3D face and the semantic labels on the 2D image are required. These correspondences are typically acquired by projecting the model back to the image. Therefore, incorrect shapes would cause incorrect correspondences, highlighting the necessity for accurate reconstruction in this task.

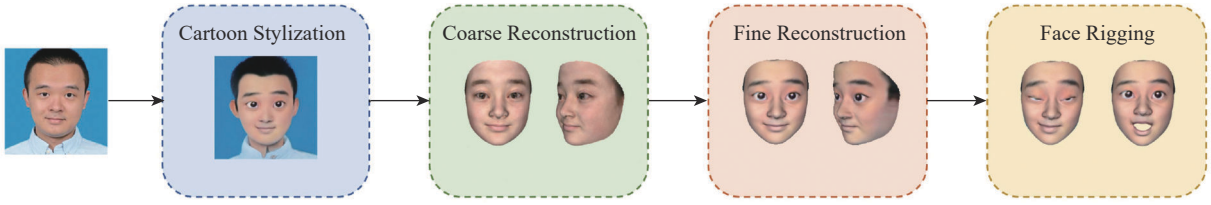



**Figure 1 Pipeline of the proposed animatable 3D cartoon face generation method. We first transform an input portrait to a 2D cartoon image and subsequently conduct template-based coarse reconstruction and deformation-based fine reconstruction to build an elaborate 3D cartoon face. Finally, we generate semantic face rigs for facial animation, making the static 3D model animatable.**

To address this problem, we proposed a two-stage reconstruction method. In the first stage, we utilized face templates and a reconstruction network to perform coarse estimation. In the second stage, our nonrigid deformation refinement adjusted the 3D model under the supervision of accurate 2D annotations. This refinement was not restricted to a specific domain. Some studies[3,6] introduced a similar idea of adding a refinement network to adjust the 3D model. However, these studies constrained the refinement on depth or normal directions. As a result, they are only effective in reconstructing facial details, such as wrinkles and moles. However, these refinements are insufficient for handling cartoon faces, which usually contain larger eyes and exaggerated expressions. Our method conducts an all-direction refinement, creating an accurate alignment without unnatural distortions. We demonstrate that our method performs well on both cartoon and real-world data.

Face rigging, the basis of facial animation, is the final part of our pipeline. Facial animation methods[7] that use 3D morphable models (3DMM)[8] usually lack semantics, making their application to industrial applications challenging. Some face-rigging methods[9,10] can generate semantic rigs but require user-specific training samples. Our semantic-preserving rigging method conducts deformation transfer from a set of handmade expression models to the target. The expression models are predefined and built by professional modelers, and the rigging process is free from any reference sample.

Our work is industry-oriented, aiming to achieve high-quality customized cartoon face reconstruction with real-time animation capabilities. Experiments show that our method outperforms prior methods in terms of both reconstruction accuracy and user subjective evaluation. In this study, we showed visualization results and an application of real-time "face-to-face" animation. In summary, our main contributions are as follows:

(1) We developed a complete system that generates a user-specific 3D cartoon face from a single portrait that is animatable in real time. It can be directly applied to VR/AR applications, such as virtual meetings and social networking for avatar customization.

(2) To achieve this, we proposed a two-stage 3D face reconstruction scheme that produces high-quality results for both real-world portraits and cartoon images. Our deformation-based refinement in the second stage improves the performance of texture mapping and facial animation.

(3) A solution for semantic-preserving face rigging without reference samples was also provided.

## 2 Related work

### 2.1 Model-based single image 3D face reconstruction

Three-dimensional face reconstruction has been studied extensively in 3D computer vision, which is widely applied in face recognition, character generation, facial data collection, etc. Reconstructed 3D faces are typically represented by 3D meshes with numerous vertices. To reduce the complexity of facial representations, 3D morphable models (3DMM)[8] have been proposed for face modeling. 3DMM is a set of bases that constructs a low-dimensional subspace of 3D faces. The geometry and texture of faces in the manifold can be expressed using linear combinations of the bases. Some works[11−14] aligned the reconstructed face model with facial landmarks on the input image to regress 3DMM coefficients. However, these methods have difficulties capturing the detailed geometry of faces owing to landmark sparsity. Other studies used features such as image intensity and edges[15] to preserve facial fidelity. With the development of deep learning and differentiable rendering, some recent studies[16−18] have used convolutional neural networks (CNNs) to learn the 3DMM coefficients and pose parameters. To address the lack of training data, Deng et al. utilized photometric information to train CNNs in a weakly supervised manner[1]. All of these 3DMM-based methods face the same problem of hardly preserving exaggerated shapes and geometry details owing to the lack of expressivity of the low-dimensional models. To address this, Guo et al. proposed a fine-tuning network to recover geometry details, such as wrinkles and moles, after 3DMM coarse reconstruction[3]. However, this method restricts the fine-tuning displacement to the depth direction and is incapable of reconstructing exaggerated expressions and shapes, such as large eyes and mouths, which are fairly common in cartoon images. There are also model-free single-image reconstruction methods[2,19−21]; however, it is difficult to align or animate the results of these methods because of topological inconsistencies in the output meshes.

### 2.2 Stylized face reconstruction

Stylized faces usually have larger variations in shape and expression, making it difficult to directly transfer realistic reconstruction methods to the cartoon domain. Liu et al. presented 3D caricatures using 3DMM. Because 3DMM is low-dimensional, the reconstructed geometry varies slightly[22]. Wu et al. reconstructed 3D stylized faces from 2D caricature images, in which they deformed a 3D standard face to address the limited deformation space of 3DMM for 3D caricatures by optimizing deformation gradients under the constraints of facial landmarks[23]. A follow-up study[24] utilized a CNN to learn the deformation gradients. These methods suffer from poor reconstruction accuracy owing to the sparsity of supervision and the large gap between the standard face and target. Based on a previous study[25], Qiu et al. predicted the surface of 3D caricatures using an implicit function, which was then aligned with 3DMM[5]. However, this method requires a large amount of 3D training data, which is difficult to collect. Overall, research on reconstructing 3D stylized faces is still fairly limited, and cartoon reconstruction remains a challenging task.

### 2.3 Face rigging

Face rigging is a crucial step in 3D facial animation. By introducing 3DMM, facial expressions can be represented by linear combinations of principal component analysis (PCA) bases[7,8]. Vlasic et al. proposed a multilinear model to encode facial identity, expression, and viseme[26]. Synthesizing from a large amount of real-world data, PCA models are generally built without semantics, making facial animation challenging to achieve. To generate user-specific blend shapes for each neutral face, hand-crafted or 3D-scanned blend-shape models are required[27,28]. Li et al. generated facial blend-shape rigs from sparse exemplars[9]. However, it still relies on existing well-crafted face models, and preparing exemplars for each subject is impractical. Pawaskar et al. transferred a set of facial blend shapes from one identity to another, but the topological difference between the two models could have a negative impact on its performance[29]. Some other work[30−32] automatically generated personalized blend shapes from video sequences or RGBD frames. Although these studies have achieved impressive performance, they require temporally continuous data; therefore, they are not applicable to single-image reconstruction.

## 3 Method

As shown in Figure 1, our pipeline is divided into three parts: stylization, reconstruction, and rigging. For stylization, existing methods such as StyleGAN[4] have achieved impressive performances. Therefore, we directly applied a StyleGAN-based transfer method[33] to generate cartoon images from real-world portraits. In this section, we focus on reconstruction and rigging methods.

To recover accurate geometry and detailed texture from a single cartoon image, we split the reconstruction into two stages. The first stage performs coarse estimation of the face geometry using a CNN-based 3DMM coefficient regression. The second stage aligns the face geometry to the input image via fine-grained Laplacian deformation. The two-stage reconstruction was designed for cartoon faces with exaggerated shapes by extending the representation space of the low-dimensional 3DMM. Finally, to animate the reconstructed model, we transferred the predefined expression basis from the standard face to the user-specific face for semantic-preserving facial rig generation.

### 3.1 Model-based coarse reconstruction

#### 3.1.1 Template models: 3DMM

When expressed using 3D meshes, human faces generally consist of numerous vertices and faces to show facial details. Directly predicting the position of each vertex during reconstruction is a daunting and time-consuming task. However, human faces share some common geometrical features, such as eyes and nose, making it possible to reduce the representation complexity. Thus, 3DMM[8] was proposed to encode 3D faces into a low-dimensional subspace through linear combinations of shape and texture bases:

$$\mathcal{S}=\bar{\mathcal{S}}+\alpha_{id}A_{id}+\alpha_{exp}A_{exp} \tag{1}$$

$$\mathcal{T}=\bar{\mathcal{T}}+\alpha_{tex}A_{tex} \tag{2}$$

where $\bar{\mathcal{S}}$ and $\bar{\mathcal{T}}$ represent the shape and texture of a standard face, respectively, whereas $S$ and $T$ represent those of the 3D face. $A_{id}$, $A_{exp}$, and $A_{tex}$ are the 3DMM bases for identity, expression, and texture, respectively. These bases were extracted and synthesized from numerous real facial scans. $\alpha_{id}$, $\alpha_{exp}$, and $\alpha_{tex}$ are combination coefficients of the bases. The proposed model-based reconstruction utilizes 3DMM to make a coarse estimation of the face geometry owing to its expressiveness and simplicity.

#### 3.1.2 Coarse 3D cartoon face reconstruction

Based on previous CNN-based methods[1,3], we utilized a CNN to predict 3DMM coefficients. As shown in

Figure 2a, the network takes a 2D cartoon image as input and predicts a vector of coefficients $x = (\alpha_{id}, \alpha_{exp}, \alpha_{tex}, \delta, p)$. The 3D face pose $p$ in the world coordinate system is defined as an affine transformation with rotation $R \in \mathrm{SO}(3)$ and translation $t \in \mathbb{R}^3$. $\delta$ is the sphere harmonics (SH) coefficient that estimates the global illumination of a Lambertian surface on each vertex as $\Phi(n_i, b_i|\delta) = b_i \cdot \sum_{k=1}^{B^2} \delta_k \phi_k(n_i)$, where human faces are assumed to be Lambertian surfaces[3,34]. $\phi_k$: $\mathbb{R}^3 \to \mathbb{R}$ represents SH basis functions, and $\Phi(n_i, b_i|\delta)$ computes the irradiation of a vertex with normal $n_i$ and scalar albedo $b_i$. Applying these coefficients to 3DMM provides the reconstructed 3D face.

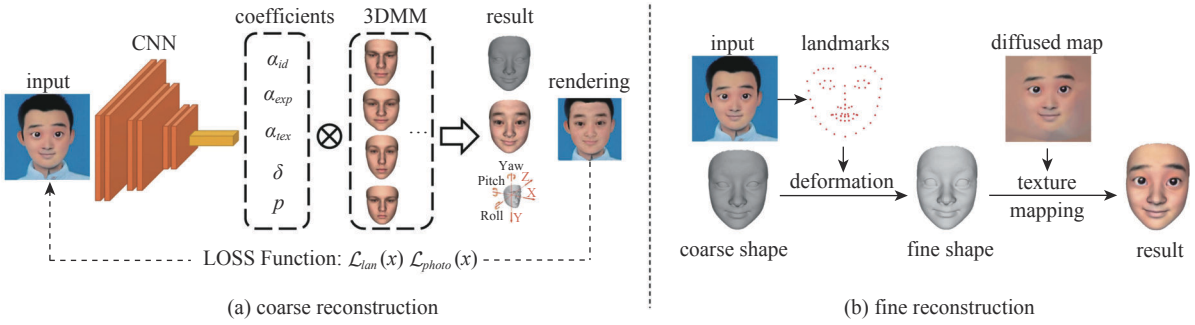



**Figure 2 Overview of the proposed two-stage 3D cartoon face reconstruction. (a) The coarse reconstruction method utilizes a CNN to predict 3DMM coefficients from an input image. The output coefficients contain a combination of parameters for identity $\alpha_{id}$, expression $\alpha_{id}$, texture $\alpha_{exp}$, lighting $\delta$, and pose $p$. (b) The fine reconstruction method refines the coarse shape using landmark supervision with Laplacian deformation. The refined model is then colored by diffused texture.**

To train the network, we first rendered the face image from the predicted 3D face model at pose $p$ and lighting approximation $\delta$ using differential rendering[35] techniques. The rendered image $I_{render}$ was then compared with the input image $I_{in}$ to calculate the loss.

Specifically, the loss function consists of three parts:

$$\mathcal{L}(x) = \omega_l \mathcal{L}_{lan}(x) + \omega_p \mathcal{L}_{photo}(x) + \omega_r \mathcal{L}_{reg}(x) \tag{3}$$

The first part is landmark loss, which is expressed as:

$$\mathcal{L}_{lan}(x) = \frac{1}{N} \sum_{n=1}^{N} \omega_n \left| q_n - \Pi\left(Rp_n + t\right) \right|^2 \tag{4}$$

where $q_n \in \mathbb{R}^2$ is the true position of the nth 2D facial landmark on the original image, and $p_n \in \mathbb{R}^3$ is the nth 3D facial landmark on the face mesh, which is predefined by 3DMM. Note that 3DMM base models share identical topologies, and the related vertices in each base model have the same semantics. Therefore, the 3D landmarks can be defined as certain vertices on the mesh. N is the number of landmarks, $\omega_n$ is the weight loss for each landmark, and $R$ and $t$ denote the rotation and transformation of pose $p$, respectively. $\Pi = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$ is the orthogonal projection matrix from 3D to 2D. The second part is photometric loss, which is expressed as

$$\mathcal{L}_{photo}(x) = \frac{1}{\left|\mathcal{A}_m\right|} \left| \mathcal{A}_m \cdot \left( I_{\text{render}} - I_{\text{in}} \right) \right|^2 \tag{5}$$

The above equation calculates the color difference between $I_{render}$ and $I_{in}$ per pixel. $\mathcal{A}_m$, acquired through face parsing[36], is a confidence map that evaluates whether an image pixel belongs to a human face. This strategy helps improve robustness in low-confidence areas, such as glasses or beards. Compared to landmark loss, photometric loss constrains the reconstructed texture and geometry at a fine-grained level. The final part is the regularization loss on 3DMM coefficients, whose purpose is to avoid getting far from the standard face.

$$\mathcal{L}_{reg}(x)=\omega_{id}|\alpha_{id}|^2+\omega_{exp}|\alpha_{exp}|^2+\omega_{tex}|\alpha_{tex}|^2 \quad (6)$$

### 3.1.3 Training with cartoon data

Most CNN-based methods train reconstruction networks by using normal face images. However, domain gaps exist between real and cartoon faces. To solve this problem, we propose a cartoon face dataset with landmark labels for network training.

Because cartoon face images are not as common as real-world images, we utilized a pretrained StyleGAN[4] to gather a large amount of cartoon data for cartoon face generation. Specifically, StyleGAN was trained on a set of cartoon face images collected from the Internet. We then randomly sampled latent codes from the input latent space $\mathcal{Z}$, forwarded them to StyleGAN, and obtained the cartoon face image. To ensure that a clear face appears on each image, we filtered out images where face detection confidence is lower than a threshold $\epsilon$ using a face detector[37].

Figure 3 shows some examples of our cartoon dataset, which contains 73852 images at a resolution of 1024×1024. Faces of different colors and ages were uniformly distributed in the dataset to minimize the bias caused by the data distribution. For each image, 68 landmarks were labeled using a landmark detector[38] to calculate the landmark loss in Eq. (4); this process is further explained in Section 3.2.1.

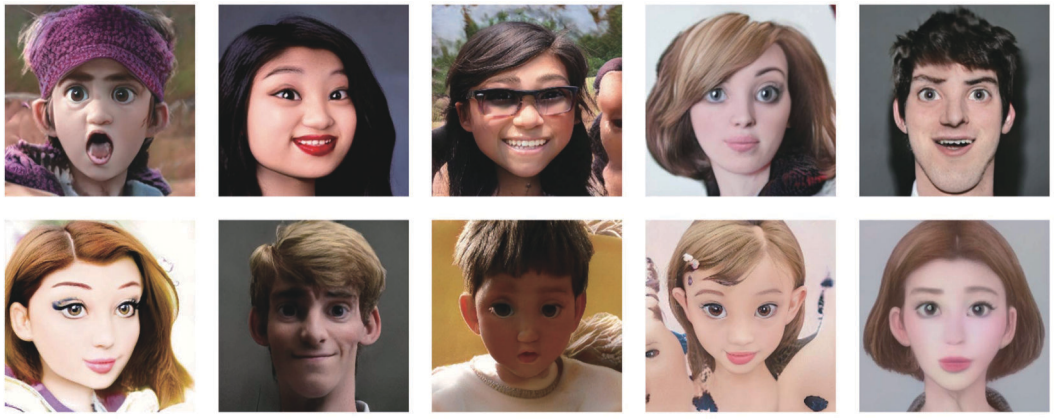


**Figure 3 Examples of the cartoon training dataset. For each sample, we ensure that a clear face exists using a face detector and apply annotations of 68 facial landmarks using a landmark detector.**

In addition, we used the same StyleGAN structure with a "layer swapping" interpolation scheme[33] to stylize users' real-world portraits. These images were then used as input for the coarse reconstruction process in the proposed application pipeline. The size of the stylized image was fixed for this study. However, studies on image enhancement[39,40] have shown the potential to increase the size and resolution of images. Thus, the image size will not be a limitation of this study.

### 3.2 Deformation-based fine reconstruction

Although using 3DMM for coarse reconstruction yields accurate results for the overall shape of the face, we found that it fails to recover some fine face structures, particularly the eyes. The low-dimensional parametric face model lacks expressivity for exaggerated facial parts, which is common in cartoon portraits. These reconstruction errors cannot be ignored because even a slight misalignment would significantly affect the model appearance and facial animation.

To address this issue, we introduced a deformation-based fine reconstruction process. As shown in Figure 2b, we aligned the 3D reconstructed face to the 2D landmarks on the input image via nonrigid deformation. We minimized the misalignment via accurate landmark supervision and a local deformation method. We

demonstrate that the proposed facial alignment strategy significantly improves texture mapping performance.

#### 3.2.1 Cartoon face 2D landmark annotation

Accurate 2D landmark annotation is crucial for the alignment. We observed a significant misalignment in the eye areas after projecting the predicted 3D face onto the image space. Some mainstream 68-landmark detectors[38] trained on ordinary face images can provide landmark annotations on the image. However, the annotation is not accurate for cartoon images, particularly in the eye areas, because of the domain gap. To solve this problem, we combined landmark detection with a state-of-the-art pixel-level face-parsing method[36]. We first obtained the prediction of 68 facial landmarks from the detectors and acquired the face-parsing result, which contained eye segmentation. Subsequently, we snap the position of each eye landmark to the nearest point on the boundary of the segmented eye area if the boundary exists. Using color clues, we set the eye landmarks at the border of the eye.

#### 3.2.2 Face alignment with Laplacian deformation

An intuitive way to align a 3D face with 2D landmark labels is to optimize the 3DMM coefficients by minimizing the distance between the projected 3D landmarks and 2D labels as follows:

$$\alpha_{id}^{*}, \alpha_{exp}^{*} = \underset{\alpha_{id}, \alpha_{exp}}{\operatorname{argmin}} \sum_{n=1}^{N} \omega_n |q_n - \Pi(Rp_n + t)| \tag{7}$$

$$p_n = K\left(\bar{\mathcal{S}} + \alpha_{id} A_{id} + \alpha_{exp} A_{exp}; n\right) \tag{8}$$

where $q_n$, $p_n$, $\Pi$, and $(R,\ t)$ have the same definitions as in Eq. (4). The parameter $K(\mathcal{S}; n) \in \mathbb{R}^3$ is used to get the nth 3D landmark position on shape $\mathcal{S}$. However, adjusting 3DMM coefficients in this manner will cause distortion and unnatural folds on the face due to the global nature and geometric restrictions of the template models, which will be demonstrated in Section 4.2.2.

Thus, we exploited Laplacian deformation[41] to align the landmarks accurately and locally without affecting the overall shape. The deformation is driven by anchors, which are landmarks in this context. The goal is to preserve the local normal of each vertex on the mesh as much as possible while moving the anchors. Specifically, the Laplacian coordinates of vertex $v_i$ are defined as:

$$L(v_i) = \frac{1}{|\mathcal{N}(v_i)|} \sum_{v_j \in \mathcal{N}(v_i)} (v_i - v_j) \tag{9}$$

where $\mathcal{N}(v_i)$ is the set of vertices that share common edges with $v_i$ (i. e., 1-ring neighboring vertices). Preserving $L(v_i)$ during deformation imposes a constraint on local geometry, thereby preventing unnatural distortions. To be driven by the anchors, the corresponding vertices should follow the anchors and remain close. Therefore, the objective function to be minimized is:

$$\min_{v \in V} \left( \sum_{i=1}^{|V|} |L(v_i) - L_i'|^2 + \lambda \sum_{k \in M} |v_k - p_k|^2 \right) \tag{10}$$

where $L_i'$ is the initial value of $L(v_i)$, $M$ is the set of vertex indices for 3D landmarks on the mesh as deformation anchors, $v_k \in \mathbb{R}^3$ is the kth 3D landmark position, and $p_k \in \mathbb{R}^3$ is the corresponding ground truth 3D position. Transforming the 2D landmark supervision $q_n$ to 3D anchors $p_k$ requires depth information. We used the depth value of the initial 3D landmark vertex $v_k$ as an approximation of $p_k$'s.

$$d_{cam} - \left(Rp_k + t\right)\ |_z = d_{cam} - \left(Rv_k + t\right)\ |_z \tag{11}$$

where $d_{cam}$ is the depth of the camera center, and $(R, t)$ is a transformation to the camera coordinate system.

#### 3.2.3 Texture mapping

Texture plays a decisive role in improving the visual quality of the reconstructed model. The texture acquired from coarse reconstruction is a combination of the 3DMM texture basis, which is significantly

rough to create an elaborate cartoon face. Therefore, to maximize the similarity of the model with the input cartoon image, we projected each vertex onto the image with the transformation $(R,t)$ predicted in the coarse reconstruction stage. The normalized 2D projected position was then used as the texture coordinates of the vertex.

$$\text{tex\_coord}(v)=\text{Norm}(\Pi(Rv+t)) \tag{12}$$

**Diffused texture.** Because of the small reconstruction inaccuracy, some background pixels may be mistakenly mapped as a part of the face texture. This error is amplified in the 3D model, as shown in Figure 4 (Origin). To tackle this problem, we first segmented the cartoon face from the background with face parsing[36] and subsequently replaced the background with the diffusion of the face color, as shown in Figure 4 (Diffusion). Each background pixel was traversed using a breadth-first search algorithm, and its color was replaced with the average color of the surrounding visited pixels. The processed images were then used for texture mapping.


**Figure 4 Examples of the cartoon training dataset. For each sample, we ensure that a clear face exists using a face detector and apply annotations of 68 facial landmarks using a landmark detector.**

### 3.3 Semantic-preserving facial rig generation

Animating a static 3D cartoon face requires additional action guidance. Motivated by 3DMM, we utilized a template-based method for facial animation.

$$S^*=S_0+B_{exp}\beta \tag{13}$$

where $S_0$ is the neutral 3D face, and $B_{exp}$ is the expression basis. Controlled by coefficients $\beta$, the output face $S^*$ changes expression accordingly. Typically, the expression components of the 3DMM basis lack semantics and are mutually coupled, making it difficult to control each part of the face independently. Inspired by FACS[42], we manually constructed a set of standard face models $\{S_i\}$, $i=1, 2, 3, \cdots, m$, each of which represents a specific movement of a single face part, such as "left eye close" and "mouth open". Subsequently, we have $B_{exp}=(S_1-S_0,\ S_2-S_0,\ \cdots,\ S_m-S_0)$, $\beta=(\beta_1,\ \beta_2,\ \cdots,\ \beta_m)$, where $\beta_i$ ranges from 0 to 1.

However, directly applying standard expression models $\{S_i\}$ to an arbitrary neutral face results in unnatural expressions because of the shape variance between different identities. Therefore, we utilized deformation transfer [43] to robustly generate a user-specific face rig. As Figure 5 shows, the deformation from $S_0$ to $S_i$ is transferred to adapt to the newly reconstructed $S_0'$ and generate $S_i'$. The expression transfer is based on the geometric relations between the standard neutral face $S_0$, standard expression $S_i$, and target neutral face $S_0'$.

For the deformation from $S_0$ to $S_i$, vertices and faces between them correspond to each other because they are topologically consistent. For a triangular face $f_j$ in the mesh, suppose $v_k$ and $\widetilde{v_k}$ ($k$ = 1, 2, and 3) are the undeformed and deformed vertices of $f_j$, respectively. To include normal information, [43] introduced the fourth vertex $v_4$ in the direction perpendicular to $f_j$ with a unit distance as:

$$v_4 = v_1 + \frac{(v_2 - v_1) \times (v_3 - v_1)}{\sqrt{|(v_2 - v_1) \times (v_3 - v_1)|}} \tag{14}$$

The deformation of $f_j$ can then be described by a 3×3 matrix $Q_j$ and translation vector $t_j$ as

$$\widetilde{v_k} = Q_j v_k + t_j, k = 1, 2, 3, 4 \tag{15}$$

For the transformation from $S_i$ to $S_i'$, the goal is to preserve $Q_j$. Thus,

$$\min_{\widetilde{v_1'}, \ldots, \widetilde{v_n'}} \sum_{j=1}^{m} |Q_j - Q_j'| \tag{16}$$

where $Q_j$ is the transformation matrix of the jth triangular face on the mesh from $S_0$ to $S_i$, and $Q_j'$ is that from $S_0'$ to $S_i'$; $m$ is the number of faces, and $\{\widetilde{v_1'}, \ldots, \widetilde{v_n'}\}$ are the vertices of $S_i'$. $t_j$ remains unchanged when transferred to $S_0'$.

We can now obtain the expression models $\{S_i'\}$ for the newly reconstructed model by applying the above expression transfer to each $\{S_i\}$. Subsequently, the 3D face can be animated in real time driven by the input coefficients $\beta$.

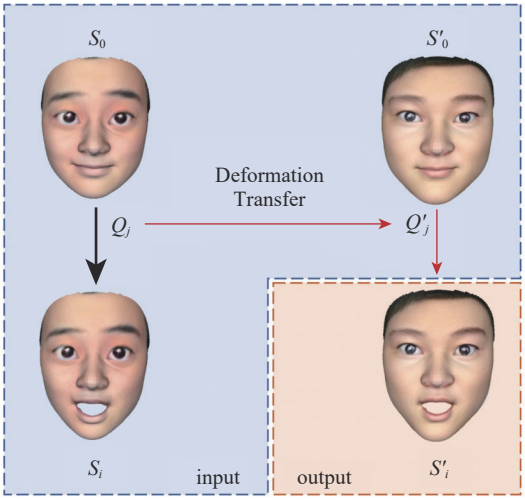



**Figure 5 Expression transfer. The deformation from $S_0$ to $S_i$ ($Q_j$ for face $f_j$) is transferred to the deformation from $S_0'$ to $S_i'$ ($Q_j'$ for face $f_j$) by generating new expression models $S_i'$. $S_0'$ can be animated using the subject-specific expression models.**

## 4 Experiments

### 4.1 Setup

**Implementation details.** We implemented the coarse reconstruction network using the PyTorch framework[44]. The network takes a stylized face image with size 224×224×3 as input and outputs a coefficient vector $x \in \mathbb{R}^{239}$, with $\alpha_{id} \in \mathbb{R}^{80}$, $\alpha_{exp} \in \mathbb{R}^{64}$, $\alpha_{tex} \in \mathbb{R}^{9}$, and $\delta \in \mathbb{R}^{6}$, respectively. In our experiment, we set the weights to $\omega_{id} = 1.2$, $\omega_{exp} = 1.0$, $\omega_{tex} = 1.2e-3$, $\omega_l = 2e-3$, $\omega_p = 2.0$, and $\omega_r = 3e-4$. Similar to [1], we used a ResNet-50 network as the backbone, followed by a fully connected layer to regress the coefficients. For the fine reconstruction stage, the optimization problem in Eq. (10) can be transformed into a linear equation using the least squares method. We solved the linear equation using sparse matrices and Cholesky decomposition. The same process was applied to the expression transfer optimization problem in Eq. (16) for facial rig generation. Our manually constructed standard expression models were built on a blender[45] by professional modelers and contained 46 different expressions defined by FACS[42].

**Data Collection.** As introduced in Section 3.1.3, we built a training dataset with 73852 cartoon face images for the coarse reconstruction training. For the testing data, we collected real-world portraits and stylized them using a pretrained StyleGAN[4]. We then annotated using the landmark detector 68 facial landmarks for each stylized cartoon image[38] and manually adjusted their positions. The test set contained 50 images with various lighting conditions and shapes.

### 4.2 Results of cartoon face reconstruction

#### 4.2.1 Comparison with prior art

We compared our method with PRN[2] and Deep3D[1], which is a template-free method that predicts face shapes using a CNN and a baseline method that predicts 3DMM coefficients in an unsupervised manner, respectively. Both methods have been proposed recently, showing impressive performances in 3D face

reconstruction. We also reported the results of our two stages, coarse and fine reconstructions, to validate the effectiveness of the two-stage design. We measured the reconstruction quality by computing the 2D landmarks and photometric errors of the test set. Specifically, for each test image, we projected the results onto the image plane after reconstruction. The landmark error measures the Euclidean distances between the projected landmarks and annotations and evaluates the correspondence and shape accuracy. We separately evaluated the errors for different facial parts. We also used the photometric error, which is the average Manhattan distance of the pixel colors between the rendered and input images, to evaluate the appearance similarity. The average results of the test data are presented.

As shown in Table 1, our method achieves a significantly lower landmark error than PRN and Deep3D. Although they have a similar network structure, our coarse reconstruction method slightly outperforms Deep3D owing to the cartoon data training. Compared with coarse reconstruction, our fine reconstruction method significantly improves the alignment accuracy of the eyes. The accuracy of other facial parts, such as the nose, eyebrows, and mouth, was also improved, thereby validating the effectiveness of the proposed deformation-based alignment strategy. To map the texture from the input image, alignment with the image should be accurate. Otherwise, it would produce unnatural facial colors. Moreover, our fine reconstruction method achieves the lowest photometric error owing to accurate reconstruction, alignment, and texture mapping. Although PRN utilizes the input image for texture mapping, similar to the proposed method, which is the reason PRN result looks similar to the input image, it has a larger photometric error because the inaccuracy of the shape and alignment causes background pixels to be mistakenly mapped to the texture. Visualization comparisons are shown in Figure 6.

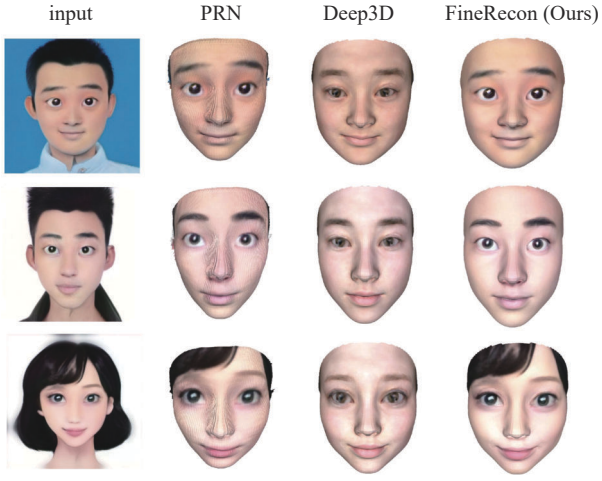



**Figure 6 Comparison of our results with PRN and Deep3D.**

**Table 1 Comparison with prior art**

| Method | Landmark Error ↓ | | | | | | Photometric Error ↓ |
|---|---|---|---|---|---|---|---|
| | eyes | nose | brow | mouth | contour | total | |
| PRN[2] | 269.45 | 200.41 | 156.34 | 397.67 | 435.86 | 1459.73 | 1.06 |
| Deep3D[1] | 100.23 | 0.51 | 11.60 | 1.57 | 16.17 | 130.08 | 3.31 |
| CoarseRecon (Ours) | 98.33 | 0.55 | 11.40 | 1.54 | 16.14 | 127.96 | 3.27 |
| FineRecon (Ours) | **8.27** | **0.23** | **11.33** | **1.50** | **16.11** | **37.44** | **0.83** |

### 4.2.2 Evaluation of face alignment

**Comparison with the template-based method.** Eq. (7) shows an intuitive way of adjusting 3DMM coefficients $\alpha_{id}$ and $\alpha_{exp}$ to align with the 2D landmark labels. There are two schemes for optimizing the coefficients based on templates, that is, adjusting $\alpha_{exp}$ only (Adjust Exp) and adjusting both $\alpha_{id}$ and $\alpha_{exp}$ simultaneously (Adjust Id+Exp).

Table 2 presents a comparison of the proposed deformation-based method with the two template-based methods. We used landmark errors as the criteria, with the same definition as in Table 1. Our method has lower landmark errors than the baselines. Interestingly, "Adjust Id+Exp" has a lower landmark error than "Adjust Exp" owing to its higher degree of freedom (DoF) and larger representation space. In this regard,

the proposed method exhibits the highest DoF and lowest error. The visualization results are shown in Figure 7. Although "Adjust Id+Exp" has a lower landmark error than "Adjust Exp", the visualization shows unnatural wrinkles and distortion owing to the restrictions of the templates. This suggests that the refinement exceeds the representation capability of the templates. Meanwhile, the proposed method can retain high-quality visual performance while simultaneously making accurate adjustments.

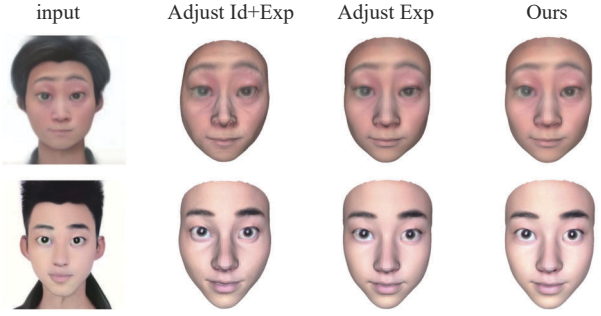
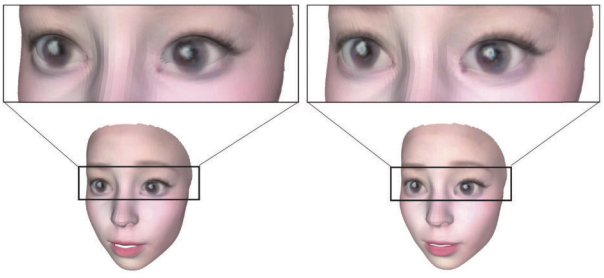



**Figure 7 Comparison of the results using different face alignment strategies: template-based method (Adjust Id+Exp/ Adjust Exp) and deformation-based method (ours).**

Figure 8 shows a comparison of the results for the eye areas with and without face alignment. Before alignment, a part of the eye texture was mistakenly mapped to the facial skin because the coarsely reconstructed eyes were extremely small. During animation, such as when the eyes are closing, the wrongly mapped texture was amplified.



(a) without alignment (b) with alignment

**Figure 8 Comparison of the results on eye areas (a) without and (b) with face alignment.**

### 4.2.3 User subjective evaluation

For a more comprehensive evaluation of our reconstruction results, we conducted a user study to collect subjective evaluations of the reconstruction. Each participant was sent a questionnaire containing six independent questions. For each question, we randomly selected a cartoon face image from the testing set and reconstructed its 3D model using PRN, Deep3D, and the proposed method. We demonstrated the results of these methods in random order and asked participants to rate the aesthetics, accuracy, and similarity. Aesthetics determines whether the 3D model is aesthetically pleasing. Accuracy assesses the correctness of the overall shape and position of each facial part. Similarity evaluates whether the 3D model appears similar to the input image. Participants were asked to rate each aspect from 1 to 5, where 1 = very poor and 5 = very good.

**Table 2 Comparison with the template-based method**

| | Adjust Exp | Adjust Id+Exp | Ours |
|---|---|---|---|
| Landmark Error ↓ | 143.38 | 110.70 | **37.44** |

We invited 55 participants (30 males and 25 females) from diverse backgrounds. Table 3 shows the average score and standard deviation of all participants for each question and method. The proposed method outperforms the other two methods in terms of subjective criteria, including aesthetics, accuracy, and similarity. Based on our observation, the face shape and texture play important roles in improving the performance of these subjective criteria. The results suggest that our method does not overfit the landmark constraints but rather uses the appropriate constraints to achieve an overall high visual quality.

**Table 3 User subjective evaluations**

| | Aesthetics | Accuracy | Similarity |
|---|---|---|---|
| PRN[2] | 2.63/1.07 | 2.99/1.04 | 3.12/1.17 |
| Deep3D[1] | 2.66/1.22 | 2.62/1.01 | 2.46/1.02 |
| Ours | **3.75/0.92** | **3.88/0.81** | **3.95/0.87** |

Note: The table shows the mean and standard deviation (mean/std. dev.) of users' evaluation scores. Our method achieves the highest ratings on all three subjective criteria (aesthetics, accuracy, and similarity).

### 4.3 Results of cartoon face reconstruction

**Visualizations on template-based facial animation.** Figure 9 shows the linear combination of the neutral face $S_0$ and an expression template model $S_i$ with coefficient $\beta$, according to Eq. (13). The semantic of $S_i$ is "right eye close", which allows us to control the right eye independently. A total of 46 template models with different semantics, such as mouth opening, left brows up, and lips funnel, were developed in this study.

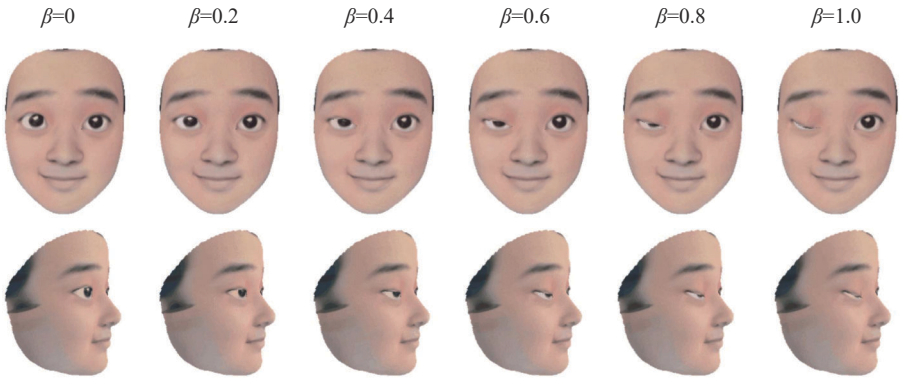



**Figure 9 Template-based facial animation.**

**Results on expression transfer.** The effectiveness of our expression transfer method is demonstrated in Figure 10. We hid the texture to clearly show the face geometry and demonstrate the transfer of two typical expressions: "right eye close" and "mouth open". The results show that the transferred expression can adapt well despite the varying shape of the target model $S_0'$. This is because we transferred the transformation of the triangular faces onto the mesh instead of simply applying the vertex shift to the target model.

**Eye-ball modeling.** To animate the eyes without eyeball distortion, we modeled the eyeballs independently during face rigging. A sphere was fitted in the eyeball area and then moved inside the head for a small distance $\Delta$ to avoid collision with the eyelids. The texture was correspondingly mapped to the sphere, and the invisible parts were set to white by default.

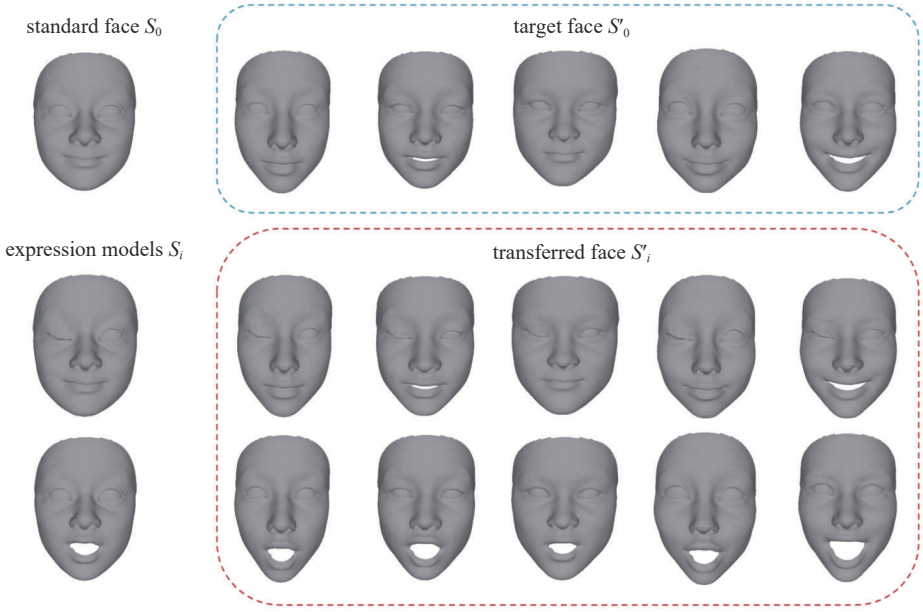



**Figure 10 Visualization results of expression transfer on part of the test data. Two expressions are demonstrated: "mouth open" and "close right eye".**

### 4.4 Application results

**Efficiency evaluation.** Generally, applications require high-efficiency reconstruction and animation. Our experiment was performed on a computer with an Intel(R) Xeon(R) E5-2678 v3 @ 2.50GHz CPU and a TITAN RTX GPU. We repeated the experiment on each test sample ten times and calculated the average time consumption. As shown in Table 4, the proposed method requires an average of 24s for reconstruction and face rigging, which is an acceptable waiting time for a user. Currently, the fine reconstruction algorithm is implemented on a CPU; however, we believe that its efficiency will be significantly improved if this step is accelerated by a GPU. For the runtime, the results show that our reconstructed model can change its expression with real-time performance of more than 280 FPS.

**Table 4 Efficiency evaluation**

| Reconstruction | | | Run-time |
|---|---|---|---|
| CoarseRecon | FineRecon | FaceRigging | Deformation |
| 1.46 s | 20.02 s | 1.84 s | 3.56 ms |

Note: We show that our pipeline can reconstruct an arbitrary face model within 30s and perform real-time facial animation over 280 FPS.

**Real-time face-to-face animation.** Using a fast expression animation driver[46], we demonstrated the potential of real-time face-to-face facial animation in Figure 11. The upstream driver predicted expression coefficients $\beta$ from a real human face. A reconstructed 3D cartoon face was then animated by $\beta$. The driving process can be implemented online with a separate frontend and backend, in which the driver and animatable 3D model serve as the backend and frontend, respectively. Intuitively, this functionality allows users to drive their own avatars to follow their facial actions in VR applications.

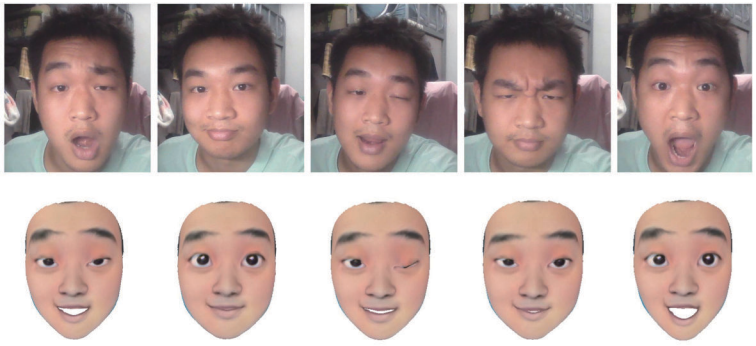


**Figure 11 Visualizations of real-time face-to-face animation. Our reconstructed model can be driven by a real-world reference face, utilizing an upstream expression driver.**

**Results on ordinary portrait images.** Although we focused on cartoon face reconstruction, our method can also be used to reconstruct high-quality realistic faces. Figure 12 shows examples of single-view 3D face reconstructions from ordinary portraits using the proposed method.

### 4.5 Additional results

Figure 13 shows additional visualization results on cartoon images with different styles. Our method is robust to exaggerated facial parts, such as large eyes and unnatural face shapes.

## 5 Conclusion

In this study, we introduced a novel pipeline for generating animatable 3D cartoon faces from a single real-world portrait. To achieve high-quality 3D cartoon faces, we proposed a two-stage face reconstruction scheme. We generated semantic-preserving face rigs using manually created models and expression transfer. Quantitative and qualitative results show that our reconstruction method achieves high performance in terms

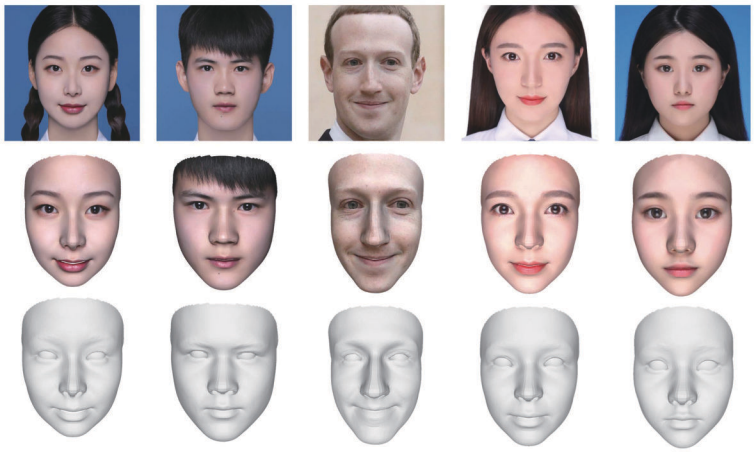


**Figure 12 Realistic 3D face reconstruction results from ordinary portraits. The upper row is the input image, and the middle and bottom rows are the reconstructed models with and without texture, respectively.**

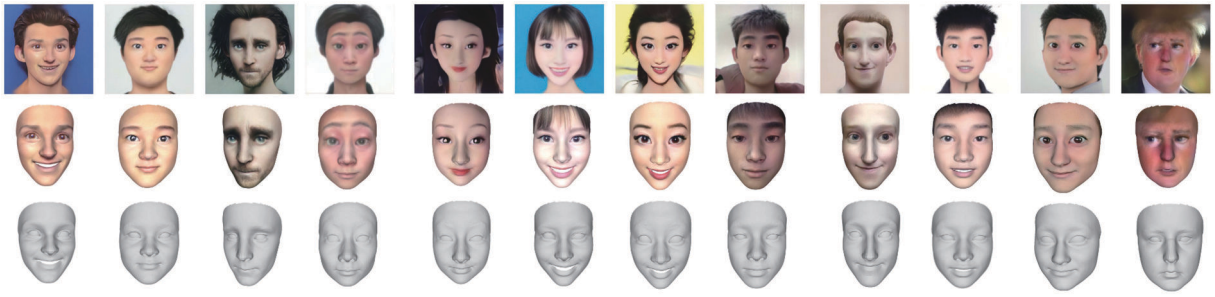


**Figure 13 Additional visualization results. We conduct reconstruction on images with different styles. For every three rows, the first row shows the input cartoon images, and the second and third rows show the 3D models with and without texture, respectively.**

of accuracy, aesthetics, and similarity criteria. Furthermore, we demonstrated the real-time animation capability of our model. The proposed pipeline can be used in creating user 3D avatars for VR/AR applications. Generating high-quality animatable 3D faces of various styles is a difficult task, and we aim to generalize our method to a larger range of styles in future studies.

## Declaration of competing interest

We declare that we have no conflict of interest.

## CRediT authorship contributions statement

**Chuanyu Pan:** Conceptualization, Data curation, Formal analysis, Methodology, Validation，Writing-original draft, Visualization; **Guowei Yang:** Writing-original draft, Supervision, Conceptualization, Project administration; **Taijiang Mu:** Writing-review & editing, Supervision, Project administration; **Yu-Kun Lai:** Writing-review & editing, Supervision.

## References

1 Deng Y, Yang J L, Xu S C, Chen D, Jia Y D, Tong X. Accurate 3D face reconstruction with weakly-supervised learning: from single image to image set. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Long Beach, CA, USA, IEEE, 2020, 285–295
DOI: 10.1109/cvprw.2019.00038

2 Feng Y, Wu F, Shao X H, Wang Y F, Zhou X. Joint 3D face reconstruction and dense alignment with position map regression network. In: Computer Vision-ECCV 2018. Springer International Publishing, 2018, 557–574
DOI: 10.1007/978-3-030-01264-9_33

3 Guo Y D , Zhang J Y, Jianfei C, Jiang BY, Zheng J M. CNN-based real-time dense face reconstruction with inverse-rendered photo-realistic face images. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019, 41(6): 1294–1307

DOI: 10.1109/tpami.2018.2837742

4 Karras T, Laine S, Aila T M. A style-based generator architecture for generative adversarial networks. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA, USA, IEEE, 2020, 4396–4405
DOI: 10.1109/cvpr.2019.00453

5 Qiu Y D, Xu X J, Qiu L T, Pan Y, Wu Y S, Chen W K, Han X G. 3DCaricShop: a dataset and a baseline method for single-view 3D caricature face reconstruction. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Nashville, TN, USA, IEEE, 2021, 10231–10240
DOI: 10.1109/cvpr46437.2021.01010

6 Cao C, Bradley D, Zhou K, Beeler T. Real-time high-fidelity facial performance capture. ACM Transactions on Graphics, 2015, 34(4): 1–9
DOI: 10.1145/2766943

7 Blanz V, Basso C, Poggio T, Vetter T. Reanimating faces in images and video. Computer Graphics Forum, 2003, 22(3): 641–650
DOI: 10.1111/1467-8659.t01-1-00712

8 Blanz V, Vetter T. A morphable model for the synthesis of 3D faces. Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques. New York, ACM, 1999, 187–194
DOI: 10.1145/311535.311556

9 Li H, Weise T, Pauly M. Example-based facial rigging. ACM Transactions on Graphics, 2010, 29(4): 1–6
DOI: 10.1145/1778765.1778769

10 Zhou J Y, Wu H T, Liu Z C, Tong X, Guo B N. 3D cartoon face rigging from sparse examples. The Visual Computer, 2018, 34(9): 1177–1187
DOI: 10.1007/s00371-018-1553-3

11 Blanz V, Mehl A, Vetter T, Seidel H P. A statistical method for robust 3D surface reconstruction from sparse data. In: Proceedings of 2nd International Symposium on 3D Data Processing, Visualization and Transmission, 2004.3DPVT. Thessaloniki, Greece, IEEE, 2004, 293–300
DOI: 10.1109/tdpvt.2004.1335212

12 Zhu X Y, Zhen L, Yan J J, Dong Y, Li S Z. High-fidelity pose and expression normalization for face recognition in the wild. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Boston, MA, USA, IEEE, 2015, 787–796
DOI: 10.1109/cvpr.2015.7298679

13 Hassner T, Harel S, Paz E, Enbar R. Effective face frontalization in unconstrained images. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Boston, MA, USA, IEEE, 2015, 4295–4304
DOI: 10.1109/cvpr.2015.7299058

14 Bas A, Smith W A P, Bolkart T, Wuhrer S. Fitting a 3D morphable model to edges: a comparison between hard and soft correspondences. In: Computer Vision-ACCV 2016 Workshops. Springer International Publishing, 2017, 377–391
DOI: 10.1007/978-3-319-54427-4_28

15 Romdhani S, Vetter T. Estimating 3D shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR'05. DiegoSan, CA, USA, IEEE, 2005, 986–993
DOI: 10.1109/cvpr.2005.145

16 Kim H, Zollhöfer M, Tewari A, Thies J, Richardt C, Theobalt C. InverseFaceNet: deep monocular inverse face rendering. 2018

17 Jourabloo A, Liu X M. Large-pose face alignment via CNN-based dense 3D model fitting. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA, IEEE, 2016, 4188–4196
DOI: 10.1109/cvpr.2016.454

18 Zhu X Y, Lei Z, Liu X M, Shi H L, Li S Z. Face alignment across large poses: a 3D solution. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA, IEEE, 2016, 146–155
DOI: 10.1109/cvpr.2016.23

19 Hassner T, Basri R. Example based 3D reconstruction from single 2D images. In: 2006 Conference on Computer Vision and Pattern Recognition Workshop CVPRW'06. YorkNew, NY, USA, IEEE, 2006, 15
DOI: 10.1109/cvprw.2006.76

20 Kemelmacher-Shlizerman I, Seitz S M. Face reconstruction in the wild. In: 2011 International Conference on Computer Vision. Barcelona, Spain, IEEE, 2012, 1746–1753
DOI: 10.1109/iccv.2011.6126439

21 Hassner T. Viewing real-world faces in 3D. In: 2013 IEEE International Conference on Computer Vision. Sydney, NSW, Australia, IEEE, 2014, 3607–3614
DOI: 10.1109/iccv.2013.448

22 Liu J F, Chen Y Q, Miao C Y, Xie J J, Ling C X, Gao X Y, Gao W. Semi-supervised learning in reconstructed manifold space for 3D caricature generation. Computer Graphics Forum, 2009, 28(8): 2104–2116
DOI: 10.1111/j.1467-8659.2009.01418.x

23 Wu Q Y, Zhang J Y, Lai Y K, Zheng J M, Cai J F. Alive caricature from 2D to 3D. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA, IEEE, 2018, 7336–7345
DOI: 10.1109/cvpr.2018.00766

24 Cai H, Guo Y, Peng Z, Zhang J. Landmark detection and 3D face reconstruction for caricature using a nonlinear parametric model. Graphical Models, 2021, 115: 101103
DOI: 10.1016/j.gmod.2021.101103

25 Saito S, Huang Z, Natsume R, Morishima S, Li H, Kanazawa A. PIFu: pixel-aligned implicit function for high-resolution clothed human digitization. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV). Seoul, Korea (South), IEEE, 2020, 2304–2314
DOI: 10.1109/iccv.2019.00239

26 Vlasic D, Brand M, Pfister H, Popovic J. Face transfer with multilinear models. SIGGRAPH '06: ACM SIGGRAPH 2006 Courses. Boston, Massachusetts. New York, ACM, 2006
DOI: 10.1145/1185657.1185864

27 Alexander O, Rogers M, Lambeth W, Chiang J Y, Ma W C, Wang C C, Debevec P. The digital emily project: achieving a photorealistic digital actor. IEEE Computer Graphics and Applications, 2010, 30(4): 20–31
DOI: 10.1109/mcg.2010.65

28 Lewis J P, Anjyo K, Rhee T, Zhang M, Pighin F H, Deng Z. Practice and theory of blendshape facial models. Eurographics(State of the Art Reports), 2014, 1(8): 2

29 Pawaskar C, Ma W C, Carnegie K, Lewis J P, Rhee T. Expression transfer: a system to build 3D blend shapes for facial animation. In: 2013 28th International Conference on Image and Vision Computing New Zealand (IVCNZ 2013. Wellington, ZealandNew, IEEE, 2014, 154–159
DOI: 10.1109/ivcnz.2013.6727008

30 Garrido P, Zollhöfer M, Casas D, Valgaerts L, Varanasi K, Pérez P, Theobalt C. Reconstruction of personalized 3D face rigs from monocular video. ACM Transactions on Graphics, 2016, 35(3): 1–15
DOI: 10.1145/2890493

31 Ichim A E, Bouaziz S, Pauly M. Dynamic 3D avatar creation from hand-held video input. ACM Transactions on Graphics, 2015, 34(4): 1–14
DOI: 10.1145/2766974

32 Casas D, Feng A, Alexander O, Fyffe G, Debevec P, Ichikari R, Li H, Olszewski K, Suma E, Shapiro A. Rapid photorealistic blendshape modeling from RGB-D sensors. Proceedings of the 29th International Conference on Computer Animation and Social Agents. Geneva, Switzerland. New York, ACM, 2016, 121–129
DOI: 10.1145/2915926.2915936

33 Pinkney J N M, Adler D. Resolution dependent GAN interpolation for controllable image synthesis between domains. 2020: arXiv: 2010.05334

34 Ramamoorthi R, Hanrahan P. An efficient representation for irradiance environment maps. Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques. New York, ACM, 2001, 497–500
DOI: 10.1145/383259.383317

35 Laine S, Hellsten J, Karras T, Seol Y, Lehtinen J, Aila T M. Modular primitives for high-performance differentiable rendering. ACM Transactions on Graphics, 2020, 39(6): 1–14
DOI: 10.1145/3414685.3417861

36 Yu C Q, Gao C X, Wang J B, Yu G, Shen C H, Sang N. BiSeNet V2: bilateral network with guided aggregation for real-time semantic segmentation.International Journal of Computer Vision, 2021, 129(11): 3051–3068
DOI: 10.1007/s11263-021-01515-2

37 Zhang K P, Zhang Z P, Li Z F, Qiao Y. Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters, 2016, 23(10): 1499–1503
DOI: 10.1109/lsp.2016.2603342

38 King D E. Dlib-ml: a machine learning toolkit. Journal of Machine Learning Research, 2009, 10: 1755–1758

39 Zhang Y, Di X, Zhang B, Li Q, Yan S, Wang C. Self-supervised low light image enhancement and denoising. 2021: arXiv: 2103.00832

40 Muslim H S M, Ali Khan S, Hussain S, Jamal A, Qasim H S A. A knowledge-based image enhancement and denoising approach. Computational and Mathematical Organization Theory, 2019, 25(2): 108–121
DOI: 10.1007/s10588-018-9274-8

41 Zhou K, Huang J, Snyder J, Liu X G, Bao H J, Guo B N, Shum H Y. Large mesh deformation using the volumetric graph Laplacian. SIGGRAPH '05: ACM SIGGRAPH 2005 Papers. Los Angeles, California. New York, ACM, 2005, 496–503
DOI: 10.1145/1186822.1073219

42 Ekman P, Friesen W V. Facial action coding system. Environmental Psychology & Nonverbal Behavior,1978

43 Sumner R W, Popović J. Deformation transfer for triangle meshes. ACM Transactions on Graphics, 2004, 23(3): 399–405
DOI: 10.1145/1015706.1015736

44 Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Köpf A, Yang E, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai J, Chintala S. PyTorch: an imperative style, high-performance deep learning library. 2019: arXiv: 1912.01703

45 Community B O. Blender-a 3D modelling and rendering package. Blender Foundation, Stichting Blender Foundation, Amsterdam,2018

46 Lugaresi C, Tang J, Nash H, McClanahan C, Uboweja E, Hays M, Zhang F, Chang C, Yong M, Lee J, Chang W T, Hua W, Georg M, Grundmann M. MediaPipe: a framework for building perception pipelines. 2019: arXiv: 1906.08172