

GANHead: Towards Generative Animatable Neural Head Avatars

Sijing Wu¹ Yichao Yan^{2*} Yunhao Li¹ Yuhao Cheng²
 Wenhan Zhu² Ke Gao³ Xiaobo Li³ Guangtao Zhai^{1,2*}

¹Institute of Image Communication and Network Engineering, Shanghai Jiao Tong University

²MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University

³Alibaba Group

{wusijing, yanyichao, lyhsjtu, chengyuhao, zhuwenhan823, zhaiguangtao}@sjtu.edu.cn

{gaoke.gao, xiaobo.lixb}@alibaba-inc.com



Figure 1. We propose GANHead to generate diverse head avatars with complete geometry and realistic texture. The generated avatars can be deformed to target poses and expressions through FLAME [22] parameters, which generalize well to unseen poses and expressions.

Abstract

To bring digital avatars into people’s lives, it is highly demanded to efficiently generate complete, realistic, and animatable head avatars. This task is challenging, and it is difficult for existing methods to satisfy all the requirements at once. To achieve these goals, we propose GANHead (**Generative Animatable Neural Head Avatar**), a novel generative head model that takes advantages of both the fine-grained control over the explicit expression parameters and the realistic rendering results of implicit representations. Specifically, GANHead represents coarse geometry, fine-gained details and texture via three networks in canonical space to obtain the ability to generate complete and realistic

head avatars. To achieve flexible animation, we define the deformation filed by standard linear blend skinning (LBS), with the learned continuous pose and expression bases and LBS weights. This allows the avatars to be directly animated by FLAME [22] parameters and generalize well to unseen poses and expressions. Compared to state-of-the-art (SOTA) methods, GANHead achieves superior performance on head avatar generation and raw scan fitting.

1. Introduction

How to efficiently generate photorealistic and animatable head avatars without manual effort is an open problem in computer vision and computer graphics, which has numerous applications in VR/AR, games, movies, and the metaverse. In these applications, it is desirable that the head

*Corresponding author

Project page: <https://wsj-sjtu.github.io/GANHead/>

avatar models fulfill the following requirements: (1) **Complete**, *i.e.*, the 3D model can cover the entire head including the frontal face, the back of head, and the hair region; (2) **Realistic**, where the avatar is expected to display vivid texture and detailed geometry; (3) **Animatable**, *i.e.*, the avatar is supposed to be fully riggable over poses and expressions, and can be controlled with low-dimensional parameters; (4) **Generative** model can be more flexibly applied to various downstream tasks, therefore, the head avatar model is preferred to be generative rather than discriminative for large-scale content generation tasks.

We investigate the research on neural head avatars and summarize previous works in Tab. 1. 3D morphable models (3DMMs) built from registered meshes have been widely employed to model head avatars. Principal component analysis (PCA) is applied to shape and texture, and novel subjects can be generated by sampling the coefficients of PCA bases. However, registering real-world raw scans to a template mesh with fixed topology is non-trivial, and it is difficult to define a fixed topology for complex regions like hair. As a result, most of these methods only model the facial region [3–6, 13, 15, 23, 39], while a few cover the full head without hair [2, 11, 22, 33, 36]. Moreover, the oversimplification of PCA makes the models lack of realism.

In parallel with explicit meshes, implicit representations have been utilized to approximate complex surfaces. Some discriminative models [14, 19, 24, 31, 49] successfully model the complete head geometry with realistic texture. However, these methods can only be applied to the reconstruction task, incapable of generating new samples. Meanwhile, 3D-aware GANs based on implicit representations [7, 8, 29, 48] can generate multi-view-consistent frontal face images. Nevertheless, the heads are still incomplete. In addition, it is difficult to animate the neural head avatars generated by 3D-aware GANs. Recently, several implicit generative models [18, 41, 47, 50] achieve realistic and animatable head avatars. However, these models either cannot generate complete head with satisfactory geometry [18, 50], or can only be animated implicitly via the learned latent codes [47], which is inconvenient and limits the generalization ability to unseen poses and expressions.

It is natural to ask a question: can we build a model that can generate diverse realistic head avatars, and meanwhile be compatible with the animation parameters of the common parametric face model (such as FLAME [22])? In this work, we propose a generative animatable neural head avatar model, namely, GANHead, that simultaneously fulfills these requirements. Specifically, GANHead represents the 3D head implicitly with neural occupancy function learned by MLPs, where coarse geometry, fine-gained details and texture are respectively modeled via three networks. Supervised with unregistered ground truth 3D head scans, all these networks are defined in canonical space via

Scheme	Methods	Complete	Realistic	Animatable	Generative
Explicit 3DMMs	[2, 3, 15, 40]	✗	✗	✓	✓
3D-aware GANs	[7, 8, 29, 48]	✗	✓	✗	✓
Personalized Avatars	[16, 49]	✓	✓	✓	✗
	[14, 31, 44]	✗	✓	△	✗
Implicit Head Models	[50]	✗	✓	△	✓
	[18]	✗	✓	✓	✓
	[47]	✓	✓	△	✓
	Ours	✓	✓	✓	✓

Table 1. A summary of current head avatar methods. △ denotes that the head avatar can only be animated implicitly via the learned latent codes, and cannot generalize well to unseen expressions.

auto-decoder structures that are conditioned by shape, detail and color latent codes, respectively. This framework allows GANHead to achieve **complete** and **realistic** generation results, while yielding desirable **generative** capacity.

The only remaining question is how to control the implicit representation with animation parameters? To answer this question, we extend the multi-subject forward skinning method designed for human bodies [9] to human faces, enabling our framework to achieve flexible **animation** explicitly controlled by FLAME [22] pose and expression parameters. Inspired by IMAvatar [49], the deformation field in GANHead is defined by standard vertex based linear blend skinning (LBS) with the learned pose-dependent corrective bases, the linear blend skinning weights, and the learned expression bases to capture non-rigid deformations. In this way, GANHead can be learned from textured scans, and no registration or canonical shapes are needed.

Once GANHead is trained, we can sample shape, detail and color latent codes to generate diverse textured head avatars, which can then be animated flexibly by FLAME parameters with nice geometry consistency and pose/expression generalization capability. We compare our method with the state-of-the-art (SOTA) complete head generative models, and demonstrate the superiority of our method.

In summary, our main contributions are:

- We propose a generative animatable head model that can generate complete head avatars with realistic texture and detailed geometry.
- The generated avatars can be directly animated by FLAME [22] parameters, robust to unseen poses and expressions.
- The proposed model achieves promising results in head avatar generation and raw scan fitting compared with SOTA methods.

2. Related Work

Explicit Face and Head Morphable Models. Explicit representation is widely used for 3D face modeling, which is built by performing Principal Component Analysis (PCA) on numerous registered 3D facial scans and represents a 3D face as the linear combination of a set of orthogonal bases. Blanz and Vetter [3] first proposed the concept of 3D Morphable Face Model (3DMM). Since then, many efforts [1, 4–6, 15] have been devoted to improve the performance of 3DMM by either improving the quality of captured face scans or the structure of 3D face model. Considering the limited representation power of traditional 3D Morphable Models and the difficulty of acquiring registered 3D data, deep learning based 3DMMs appeared [37–40], which learn 3D priors from 2D face images or videos with the help of differentiable rendering. However, these methods [3–6, 15, 32] can only model the facial region.

Recently, some 3D Morphable Models that can represent the entire head have been proposed [2, 11, 12, 22, 33, 36]. For example, Li *et al.* [22] propose the FLAME model which represents 3D head by rotatable joints and linear blend skinning. Although these methods can model the entire head, they still cannot model the hair region since it is hard to define a fixed topology for complex regions like hair and register the raw scan to it, while our model has the ability to generate complete head avatars with diverse hairstyles.

Implicit Face and Head Models. In parallel with explicit meshes, implicit representations [26–28, 30] can also be used to model 3D shapes [26, 30]. Park *et al.* [30] propose DeepSDF to represent shape using signed distance function predicted by an autoencoder. Since then, implicit representations have become popular in 3D modeling, as well as 3D face and head modeling [35, 46], since implicit representations are better at modeling complex surfaces and realistic textures. Many works [20, 35, 43] successfully reconstruct high fidelity static heads which cannot be animated. Recent works [14, 31, 34, 49] recover animatable realistic head avatars from monocular RGB videos, but needed to train a model for each person. In addition, 3D-aware GANs [7, 8, 17, 29, 48] are proposed to generate multi-view-consistent static frontal face images, but failed to extract complete head meshes (including the back of the head) due to the lack of 3D supervision.

Recently, several implicit generative models [18, 47, 50] are proposed to achieve realistic and animatable head avatars. Hong *et al.* [18] propose the first NeRF-based parametric human head model which controls the rendering pose, identity and expression by corresponding latent codes. Yenamandra *et al.* [47] propose i3DMM, a deep implicit 3D morphable model containing entire heads and can be animated by learned latent codes. However, these models either can not generate complete head meshes with satisfac-

tory geometry [18, 50], or can only be animated implicitly via the learned latent codes [47], which limits the generalization ability to unseen poses and expressions. In contrast, our method can generate animatable head avatars with complete geometry and realistic texture using implicit representation, which can also be generalized well to unseen poses and expressions.

3. Method

In this work, we propose GANHead, a generative model learned from unregistered textured scans. Once GANHead is trained, complete and realistic head avatars that are ready for animation can be obtained by sampling three latent codes. An overview of GANHead is illustrated in Fig. 2.

In this section, we first recap the deformation formulation of parametric head model FLAME [22], and illustrate its important role in helping GANHead build a deformation module with good generalization ability to unseen poses and expressions (Section 3.1). Second, we introduce the canonical generation module (Section 3.2) that generates diverse vivid head avatars in canonical space, followed by the deformation module (Section 3.3) which deforms the generated avatars to new poses and expressions controlled by FLAME parameters. Finally, to train GANHead model from raw scans, the data pre-processing procedures, training strategy and losses are introduced in Section 3.4.

3.1. Preliminary: GANHead vs FLAME

FLAME [22] is a widely used parametric model that covers the entire head (without hair), which is deformed by:

$$M(\beta, \theta, \psi) = LBS(T_P(\beta, \theta, \psi), J(\beta), \theta, \mathcal{W}), \quad (1)$$

where β , θ and ψ denote the shape, pose and expression parameters respectively. $LBS(\cdot)$ is the standard vertex based linear blend skinning (LBS) and \mathcal{W} denotes the skinning weights. $J(\cdot)$ calculates joints location from mesh vertices, and T_P is calculated by:

$$T_P(\beta, \theta, \psi) = \bar{T} + B_S(\beta; \mathcal{S}) + B_P(\theta; \mathcal{P}) + B_E(\psi; \mathcal{E}), \quad (2)$$

where, \bar{T} is the template head. $B_S(\beta; \mathcal{S})$, $B_P(\theta; \mathcal{P})$ and $B_E(\psi; \mathcal{E})$ are per-vertex offsets calculated by shape parameters β , pose parameters θ and expression parameters ψ with corresponding bases \mathcal{S} , \mathcal{P} and \mathcal{E} .

Different from FLAME, our framework aims to model the complete head geometry (including hair) and realistic texture. Therefore, we employ implicit representation due to its flexibility, rather than the polygon mesh used in FLAME. Specifically, the textured canonical shape (*i.e.* head with identity information in natural pose and expression) is represented by neural occupancy function learned by MLPs, which is controlled by the learned latent codes.

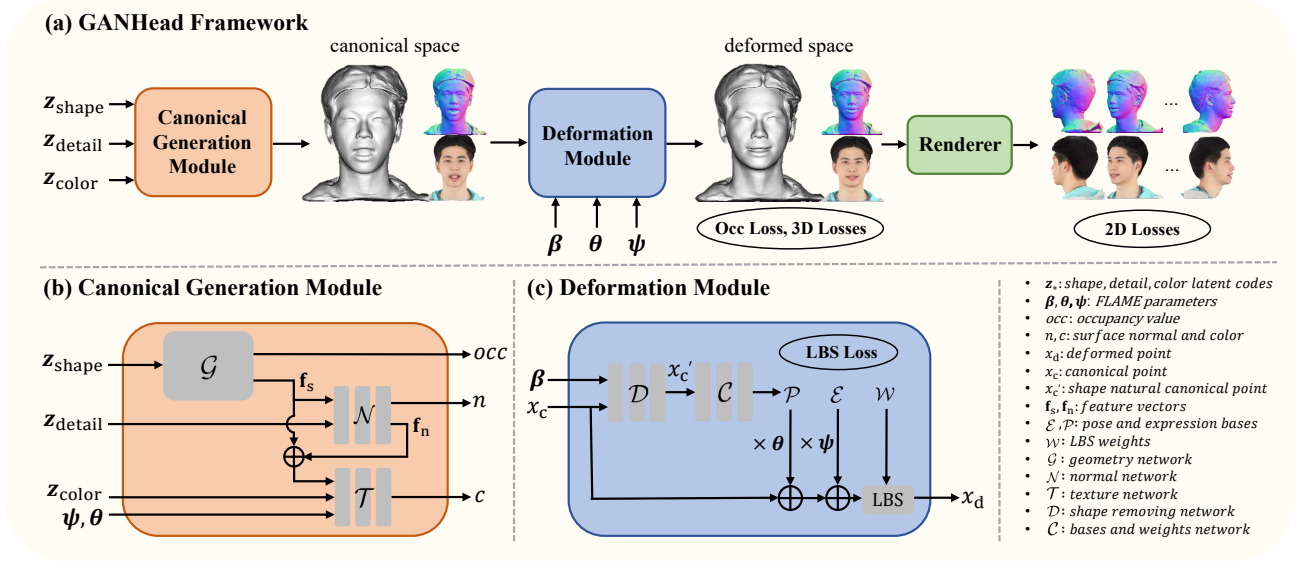


Figure 2. **Method overview.** Given shape, detail and color latent codes, the canonical generation model outputs coarse geometry and detailed normal and texture in canonical space. The generated canonical head avatar can then be deformed to target pose and expression via the deformation module. In the first training stage, occupancy values of the deformed shapes are used to calculate the occupancy loss, along with the LBS loss, to supervise the geometry network and the deformation module. In the second stage, the deformed textured avatars are rendered to 2D RGB images and normal maps, together with the 3D color and normal losses, to supervise the normal and texture networks.

Although implicit representations are more powerful, it is difficult for them to deform and generalize to unseen poses and expressions. To address this, we combine the implicit representations with the fine-grained control modeled in FLAME [22] to enjoy the merits of both sides. However, the number of vertices is not fixed in implicit representation, such that the original bases and LBS weights in the FLAME model cannot be directly used in our framework. To further tackle this issue, we utilize an MLP to learn continuous pose and expression bases, as well as the LBS weights. In order to control the avatars generated by GANHead using the same pose and expression parameters as the FLAME model, we calculate the ground truth by finding the nearest neighbors of the query points on the fitted FLAME surface, to supervise the learning of neural bases and weights.

3.2. Canonical Generation Module

GANHead models head shape and texture in canonical space via the canonical generation module, and we further design a deformation module in Section 3.3 to make it controllable by pose θ and expression ψ parameters which are consistent with FLAME [22]. It is remarkable that canonical heads are defined as: heads with identity information in natural pose and expression.

The canonical generation module consists of three neural networks that represent the shape (including coarse shape and fine-grained normal) and texture respectively.

Shape: We model the canonical head shape as the 0.5 level set of the occupancy function predicted by the geometry network \mathcal{G} :

$$\mathcal{G}(x_c, z_{\text{shape}}) : \mathbb{R}^3 \times \mathbb{R}^{n_s} \rightarrow \text{occ}, \mathbf{f}_s, \quad (3)$$

where x_c denotes the point in canonical space, and $z_{\text{shape}} \in \mathbb{R}^{n_s}$ is the shape latent code that conditions \mathcal{G} to generate diverse shapes. $\mathbf{f}_s \in \mathbb{R}^{n_f}$ is a feature vector carrying shape information which is then used to help predict fine-grained surface normal and texture. \mathcal{G} consists of a 3D style based feature generator followed by an MLP conditioned by the generated feature, similar to [9].

To model the details of the head, we use an MLP to predict the surface normal:

$$\mathcal{N}(x_c, z_{\text{detail}}, \mathbf{f}_s) : \mathbb{R}^3 \times \mathbb{R}^{n_d} \times \mathbb{R}^{n_f} \rightarrow n, \mathbf{f}_n, \quad (4)$$

where $z_{\text{detail}} \in \mathbb{R}^{n_d}$ is the detail latent code that controls the detail generation. n is the predicted normal of the query point x_c , together with a feature vector $\mathbf{f}_n \in \mathbb{R}^{n_f}$ used for texture prediction.

Texture: We model the head texture in canonical space via a texture MLP \mathcal{T} :

$$\mathcal{T}(x_c, z_{\text{color}}, \mathbf{f}, \theta, \psi) : \mathbb{R}^3 \times \mathbb{R}^{n_c} \times \mathbb{R}^{2n_f} \times \mathbb{R}^{15} \times \mathbb{R}^{50} \rightarrow c, \quad (5)$$

where $z_{\text{color}} \in \mathbb{R}^{n_c}$ is the color latent code that enables controllable texture generation. $\mathbf{f} = \mathbf{f}_s \oplus \mathbf{f}_n$ is the concatenation of the shape and normal feature vectors. θ and

ψ denote the pose and expression parameters, respectively, which are consistent with FLAME [22].

3.3. Deformation Module

To achieve flexible deformation with 3D geometry consistency and good generalization to unseen poses and expressions, we design our deformation module upon the FLAME [22] deformation field, as discussed in Section 3.1. The deformation module first predicts the continuous pose and expression bases, as well as LBS weights of the canonical points x_c , and then deforms them to x_d via added per-vertex offsets followed by linear blend skinning (LBS).

The continuous bases and weights are predicted via an MLP:

$$\mathcal{C}(x_c) : \mathbb{R}^3 \rightarrow \mathcal{E}, \mathcal{P}, \mathcal{W}, \quad (6)$$

where $\mathcal{E} \in \mathbb{R}^{3 \times 50}$, $\mathcal{P} \in \mathbb{R}^{36 \times 3}$ and $\mathcal{W} \in \mathbb{R}^5$ are the predicted expression bases, pose-dependent corrective bases and LBS weights of each canonical point x_c . Different from [49], we expect the network \mathcal{C} can be used to multiple individuals like traditional parametric face models, rather than a single person. To this end, we define the network \mathcal{C} in shape natural space by adding a shape removing network \mathcal{D} in front of \mathcal{C} :

$$\mathcal{D}(x_c, \beta) : \mathbb{R}^3 \times \mathbb{R}^{100} \rightarrow x_c', \quad (7)$$

with x_c' denotes the shape natural canonical point, and β is the shape parameter consistent with FLAME [22]. Accordingly, Eq. 6 can be rewritten as:

$$\mathcal{C}(x_c') : \mathbb{R}^3 \rightarrow \mathcal{E}, \mathcal{P}, \mathcal{W}. \quad (8)$$

Once the continuous bases and weights are predicted, the canonical head avatar can be deformed to target pose and expression by adding offsets followed by performing standard linear blend skinning (LBS):

$$\begin{aligned} X_P(\theta, \psi) &= X_c + B_P(\theta; \mathcal{P}) + B_E(\psi; \mathcal{E}), \\ X_d(\beta, \theta, \psi) &= LBS(X_P(\theta, \psi), J(\beta), \theta, \mathcal{W}), \end{aligned} \quad (9)$$

where $X_c = \{x_{c1}, \dots, x_{cn}\}$ and $X_d = \{x_{d1}, \dots, x_{dn}\}$ denote the sets of the canonical and deformed points respectively. β , θ and ψ are the shape, pose and expression parameters consistent with FLAME [22], which makes the generated avatars easy to animate by FLAME parameters. To be clear that \mathcal{E} , \mathcal{P} and \mathcal{W} are the predicted continuous bases and weights rather than the corresponding FLAME components in Eq. 1.

As mentioned before, the shape, normal and texture networks are all defined in canonical space to learn more details and generalize well to unseen poses and expressions, which means that if we input canonical query points to the canonical generation module, the output is an head avatar

in canonical space, while if we input the canonical correspondence of the deformed query points, we will obtain the occupancy values, normals and colors of the deformed head avatar. The canonical correspondence of the deformed points X_d is obtained by iteratively finding the root of Eq. 9 given deformed points X_d [10].

3.4. Training

Data: We use the textured scans in FaceVerse-Dataset [42] to train our generative model GANHead. To obtain FLAME fitting results (shape parameters β , pose parameters θ and expression parameters ψ) of the dataset for training, 3D facial landmarks are required for rigid alignment (*i.e.* calculate the scale, translation and rotation factors to align the FLAME model with raw scans). To this end, we first calculate the 3D to 2D correspondence by rendering the scans to RGB images and depth images, then use Dlib [21] to detect the 2D landmarks and project them onto the 3D scans.

Training strategy: We train the GANHead model in two stages similar to gDNA [9]: the coarse geometry network and deformation module are trained in the first stage, while the detail normal and texture networks are trained in the second stage. The canonical space is defined as the avatar opening its mouth slightly to help the model learn more details of the inner mouth.

Losses: In the first stage, We define the loss function as:

$$\mathcal{L}_{stage1} = \mathcal{L}_{occ} + \lambda_d \mathcal{L}_{deshape} + \lambda_l \mathcal{L}_{lbs} + \lambda_r \mathcal{L}_{reg}. \quad (10)$$

Specifically, \mathcal{L}_{occ} measures the binary cross entropy between the predicted occupancy $\mathcal{G}(x_d, z_{shape}, \beta, \theta, \psi)$ and the ground truth occupancy $o_{gt}(x_d)$ of the sampled points x_d . $\mathcal{L}_{deshape}$ supervises the shape removing network by enforcing the shape removing ability on FLAME vertices similar to [9]:

$$\mathcal{L}_{deshape} = \|\mathcal{D}(M(\beta, \theta_0, \psi_0), \beta) - M(\beta_0, \theta_0, \psi_0)\|_2^2, \quad (11)$$

with $M(\cdot)$ and $\mathcal{D}(\cdot)$ denotes the FLAME model and shape removing network, respectively. β , θ and ψ are FLAME parameters, and the subscript 0 indicates that the parameter is all zeros. The LBS loss \mathcal{L}_{lbs} provides weak supervision for the learned LBS weights \mathcal{W} , pose bases \mathcal{P} and expression bases \mathcal{E} by constraining them to the correspondent FLAME components similar to [49]:

$$\mathcal{L}_{lbs} = \|\mathcal{W} - \mathcal{W}_{gt}\|_2^2 + \lambda_p \|\mathcal{P} - \mathcal{P}_{gt}\|_2^2 + \lambda_e \|\mathcal{E} - \mathcal{E}_{gt}\|_2^2, \quad (12)$$

where \mathcal{W}_{gt} , \mathcal{P}_{gt} and \mathcal{E}_{gt} are the LBS weights, pose-dependent corrective bases and expression bases of FLAME model. It is remarkable that all the FLAME components are sampled to the same dimension as the query points X_c by finding the nearest neighbours of the query points on the fitted FLAME surface. Following [9, 10], we add two auxiliary losses during the first training epoch (see Sup. Mat.

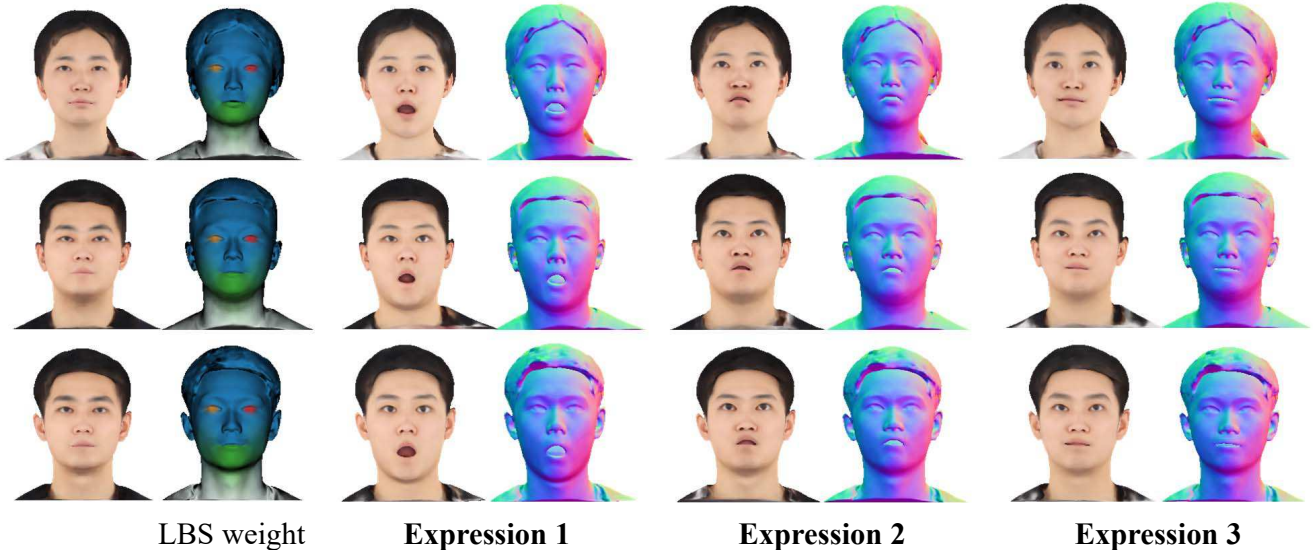


Figure 3. **Head avatars generation and animation.** We generate three head avatars by randomly sampling GANHead, and then deform them to the target expressions controlled by FLAME parameters. We show three samples, and they are all deformed to the same expressions. We also show the learned LBS weights of the canonical heads. Shapes are visualized as normal maps to highlight the geometry details.

for details). In addition, we employ a regularization term for the shape code via $\mathcal{L}_{reg} = \|z_{shape}\|_2^2$.

In the second stage, the training loss is defined as:

$$\mathcal{L}_{stage2} = \lambda_c \mathcal{L}_{color} + \lambda_n \mathcal{L}_{normal} + \lambda_r \mathcal{L}_{reg}. \quad (13)$$

The color loss \mathcal{L}_{color} includes the 2D and 3D supervision of the texture:

$$\mathcal{L}_{color} = \|I_r - I_{gt}\|_2^2 + \lambda \|c - c_{gt}\|_2^2, \quad (14)$$

where I_{gt} and I_r are the rendered RGB images of the ground truth scan and the output of GANHead respectively. c_{gt} and c denotes the ground truth color and the predicted color of the query points. The normal loss also includes 2D and 3D supervision:

$$\mathcal{L}_{normal} = \|N_r - N_{gt}\|_2^2 + \lambda (1 - n_{gt}^T \cdot n), \quad (15)$$

where N_{gt} and N_r are the rendered normal maps. n_{gt} and n are the normalized normal values of the query points, and we enforce them point to the same direction. Furthermore, we regularize the detail and color code via $\mathcal{L}_{reg} = \|z_{detail}\|_2^2 + \lambda \|z_{color}\|_2^2$.

4. Experiments

GANHead is proposed to generate diverse realistic head avatars that can be directly animated by FLAME [22] parameters. In this section, we evaluate the superiority of GANHead in terms of the head avatar generation quality and the animation flexibility of the generated avatars. Furthermore, we also fit GANHead to unseen scans and compare the performance to SOTA animatable head models to evaluate its expressiveness.

4.1. Implementation Details

Dataset: We train our model on 2289 textured scans out of 2310 (110 identities, each with 21 expressions) from the training set of the FaceVerse-Dataset [42]. Scans of a subject with hat are removed to avoid interfering with the learning of hair. And the test set of the FaceVerse-Dataset (375 scans from 18 subjects) are used to evaluate the raw scan fitting. We further conduct experiments on a subset of Multiface dataset [45] to verify the generalization ability of our model on different datasets (See Sup. Mat.).

Training details: We use PyTorch to implement our model, and Adam optimiser is used for training. We train 250 epochs with a batch size of 32 for the first stage, and 200 epochs with a batch size of 4 for the second stage. The 3D and 2D correspondence are precomputed before the second stage. The whole training takes about 3 days on 4 NVIDIA 3090 GPUs. Please refer to Sup. Mat. for more details.

4.2. Generation and Animation Capacities

Random Generate: We randomly sample the shape, detail and color latent codes to generate head avatars. The generated canonical head avatars and the visualization of their respective LBS weights are shown in Fig. 3 (the first column). We find that GANHead can generate diverse head avatars with detailed geometry and nice textures.

Deform to Target Poses and Expressions: The avatars generated by GANHead can be easily animated controlled by FLAME [22] parameters. Here we deform the generated avatars (The first column in Fig. 3) to the target expressions



Figure 4. **Extreme poses and expressions.** We generate a head avatar and then deform it to several extreme poses and expressions which have never showed up in the training set.

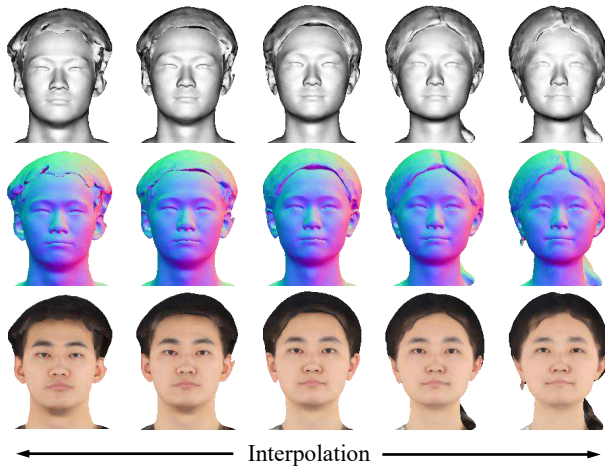


Figure 5. **Interpolation.** We interpolate the shape, detail and color latent codes between two samples, and show the geometry, detail and texture of the interpolation results.

represented by FLAME [22] parameters. The results show that the generated avatars can be well controlled by FLAME parameters, and the poses and expressions are well disentangled from the geometry.

Deform to Unseen Extreme Expressions and Poses:

In GANHead, the deformation module controlled by FLAME [22] parameters makes the generated avatars generalize well to unseen poses and expressions. Here we generate a head avatar, and deform it to six extreme poses and expressions by changing the FLAME parameters θ and ψ . As shown in Fig. 4, the generated avatar can be deformed to extreme expressions and poses that are not included in the training set, while displaying great geometry consistency. This is hard to achieve by previous implicit model.

Latent Code Interpolation: We interpolate the shape, detail and color latent codes of two samples that look vary different, as shown in Fig. 5. We can see a smooth transition between the two samples.

4.3. Ablation Study

To validate the importance of each components of GANHead, we conduct ablation experiments on a subset (420 faces from 20 identities) of the training set.

How to deform the implicit head avatar is a significant

problem in implicit modeling. Deformation under the control of low dimensional meaningful variables is more difficult. Here we illustrate the superiority of the deformation module in GANHead by comparing our method to carefully designed baselines. These baselines are built by replacing our deformation module with the following deformation methods:

Forward skinning for human head (Head-FS). Since our deformation module is designed based on the multi-subject forward skinning for human body [9], we design a baseline that simply applies the forward skinning method to human head. The multi-subject forward skinning method is built upon the human body model SMPL [25], so we directly change the SMPL model to human head model FLAME [22] to model the deformation of human head. As can be observed in the top row of Fig. 6, the model can be well generated, but the generated avatar cannot be deformed to new expressions since the original forward skinning method does not model the non-rigid deformation controlled by expression blendshapes.

FLAME deformation field (F-Def). F-Def directly uses the pose-dependent corrective bases, expression bases and LBS weights, as well as the standard linear blend skinning of the FLAME head model [22] to deform the generated avatars. Since FLAME is based on explicit representation, the number of vertices is fixed, we sample the bases and weights to the same number of points as our query points. From Fig. 6 (the second row), we observe that the model can generate acceptable canonical shape, but jagged distortion will appear when deforming the avatar.

GANHead deformation module without LBS loss (w/o LBS loss). The LBS loss plays an important role in the learning of canonical geometry. Here we remove the LBS loss, and the results are shown in Fig. 6 (the third row). It can be observed that the canonical shape is poor, and the geometry is learned in the blendshapes.

4.4. Comparisons on Scan Fitting

Although the principal function of GANHead is to generate animatable head avatars with complete geometry and realistic texture, GANHead can also be fitted to raw scans like traditional 3DMMs. In this section, we demonstrate the fitting ability of GANHead through qualitative and quantitative results on the FaceVerse test set [42].

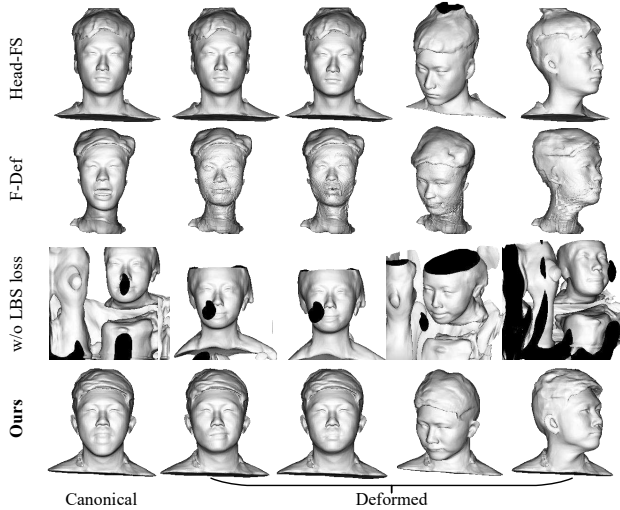


Figure 6. **Comparison with baseline methods.** The baseline methods either generate poor canonical geometry (w/o LBS Loss) or cannot deform well. In contrast, our method can generate realistic geometry which can also be well deformed to the target poses and expressions.

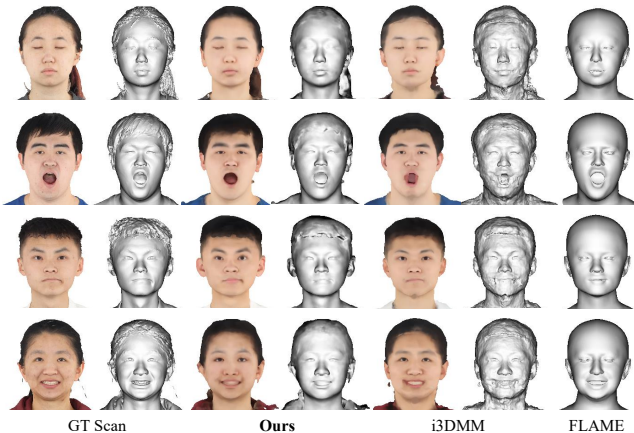


Figure 7. **Raw scan fitting.** We compare our fitting results on the test set of FaceVerse-Dataset [42] with two SOTA head models.

Fitting GANHead to raw scans can be achieved by optimizing the shape, detail and color latent codes using the following loss function:

$$\mathcal{L}_{fit} = \mathcal{L}_{occ} + \mathcal{L}_{3D} + \lambda_r \mathcal{L}_{reg}, \quad (16)$$

where \mathcal{L}_{occ} measures the binary cross entropy between the predicted occupancy and the ground truth occupancy. $\mathcal{L}_{3D} = \lambda_c \|c_{gt} - c\|_2^2 + \lambda_n (1 - n_{gt}^T \cdot n)$ supervises the reconstructed color and normal of the query points. $\mathcal{L}_{reg} = \lambda_1 \|z_{shape}\|_2^2 + \lambda_2 \|z_{detail}\|_2^2 + \lambda_3 \|z_{color}\|_2^2$ is the regularization term for the three latent codes.

We compare our raw scan fitting results with two SOTA

Region	Method	Ch. (\downarrow)	F-Score (\uparrow)	Color (\downarrow)
Full Avatar	FLAME	4.883	72.78	–
	i3DMM	2.583	88.49	8.819
	Ours	2.186	90.37	8.324
Face	FLAME	1.755	89.59	–
	i3DMM	1.208	97.30	6.423
	Ours	0.695	99.23	6.529

Table 2. **Fitting comparison.** We report the symmetric Chamfer distance ($\times 10^{-2}$), F-Score computed with a threshold of 0.05, and color distance on the FaceVerse test set.

generative head models (i3DMM [47] and FLAME [22]) that can model complete head and can be animated, which are closest to the objective of this paper. For the fair comparison, we fit FLAME to raw scans by iteratively solving the optimization problem for each scan, and retrain the i3DMM model on the FaceVerse training set. The qualitative results are shown in Fig. 7. Apparently, our model achieves the best reconstruction quality on both shape (including expression) and texture. FLAME does not model hair, consequently it cannot fit the hair region of raw scans, while i3DMM and our method can model hair region. We further report symmetric Chamfer distance (Ch.) and F-Score for assessing the geometry reconstruction quality, and symmetric color distance for texture assessment, as shown in Tab. 2. Our method significantly superior to FLAME and i3DMM in shape and expression reconstruction, especially in the facial region. As for the texture, although i3DMM has slightly better symmetric color distance in the facial region, our method numerically outperforms i3DMM in the full head avatar (head and shoulder) by a margin and achieves a better overall visual effect.

5. Conclusion

We propose GANHead (Generative Animatable Neural Head Avatar model), a novel generative head model that combines the fine-grained control of explicit 3DMMs with the realism of implicit representations. Specifically, GANHead represents coarse geometry, detailed normal and texture via three networks in canonical space to generate complete and realistic head avatars. The generated head avatars can then be directly animated by FLAME parameters via the deformation module. Extensive experiments demonstrate the superiority of GANHead in head avatar generation and broader social impact in Sup. Mat.

Acknowledgments: This work was supported by NSFC (No. 62225112, 61831015, 62101325, 62201342), the Fundamental Research Funds for the Central Universities, National Key R&D Program of China (2021YFE0206700), Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102).

References

- [1] Victoria Fernández Abrevaya, Stefanie Wuhler, and Edmond Boyer. Multilinear autoencoder for 3d face model learning. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9. IEEE, 2018. 3
- [2] Linchao Bao, Xiangkai Lin, Yajing Chen, Haoxian Zhang, Sheng Wang, Xuefei Zhe, Di Kang, Haozhi Huang, Xinwei Jiang, Jue Wang, Dong Yu, and Zhengyou Zhang. High-fidelity 3d digital human head creation from rgb-d selfies. *ACM Transactions on Graphics*, 2021. 2, 3
- [3] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194, 1999. 2, 3
- [4] James Booth, Anastasios Roussos, Allan Ponniah, David Dunaway, and Stefanos Zafeiriou. Large scale 3d morphable models. *International Journal of Computer Vision*, 126(2):233–254, 2018. 2, 3
- [5] James Booth, Anastasios Roussos, Stefanos Zafeiriou, Allan Ponniah, and David Dunaway. A 3d morphable model learnt from 10,000 faces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5543–5552, 2016. 2, 3
- [6] Chen Cao, Yanlin Weng, Shun Zhou, Yiying Tong, and Kun Zhou. Facewarehouse: A 3d facial expression database for visual computing. *IEEE Transactions on Visualization and Computer Graphics*, 20(3):413–425, 2013. 2, 3
- [7] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16123–16133, 2022. 2, 3
- [8] Eric R Chan, Marco Monteiro, Petr Kellnhöfer, Jiajun Wu, and Gordon Wetzstein. pi-gan: Periodic implicit generative adversarial networks for 3d-aware image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5799–5809, 2021. 2, 3
- [9] Xu Chen, Tianjian Jiang, Jie Song, Jinlong Yang, Michael J Black, Andreas Geiger, and Otmar Hilliges. gdna: Towards generative detailed neural avatars. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20427–20437, 2022. 2, 4, 5, 7
- [10] Xu Chen, Yufeng Zheng, Michael J Black, Otmar Hilliges, and Andreas Geiger. Snarf: Differentiable forward skinning for animating non-rigid neural implicit shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11594–11604, 2021. 5
- [11] Hang Dai, Nick Pears, William Smith, and Christian Duncan. Statistical modeling of craniofacial shape and texture. *International Journal of Computer Vision*, 128(2):547–571, 2020. 2, 3
- [12] Hang Dai, Nick Pears, William AP Smith, and Christian Duncan. A 3d morphable model of craniofacial shape and texture variation. In *Proceedings of the IEEE international conference on computer vision*, pages 3085–3093, 2017. 3
- [13] Bernhard Egger, Sandro Schönborn, Andreas Schneider, Adam Kortylewski, Andreas Morel-Forster, Clemens Blumer, and Thomas Vetter. Occlusion-aware 3d morphable models and an illumination prior for face image analysis. *IJCV*, 126(12):1269–1287, 2018. 2
- [14] Guy Gafni, Justus Thies, Michael Zollhofer, and Matthias Nießner. Dynamic neural radiance fields for monocular 4d facial avatar reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8649–8658, 2021. 2, 3
- [15] Thomas Gerig, Andreas Morel-Forster, Clemens Blumer, Bernhard Egger, Marcel Luthi, Sandro Schönborn, and Thomas Vetter. Morphable face models—an open framework. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 75–82. IEEE, 2018. 2, 3
- [16] Philip-William Grassal, Malte Prinzler, Titus Leistner, Carsten Rother, Matthias Nießner, and Justus Thies. Neural head avatars from monocular rgb videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18653–18664, 2022. 2
- [17] Jiatao Gu, Lingjie Liu, Peng Wang, and Christian Theobalt. Stylenerf: A style-based 3d-aware generator for high-resolution image synthesis. *arXiv preprint arXiv:2110.08985*, 2021. 3
- [18] Yang Hong, Bo Peng, Haiyao Xiao, Ligang Liu, and Juyong Zhang. Headnerf: A real-time nerf-based parametric head model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20374–20384, 2022. 2, 3
- [19] Alexandru Eugen Ichim, Sofien Bouaziz, and Mark Pauly. Dynamic 3d avatar creation from hand-held video input. *ACM Transactions on Graphics (TOG)*, 34(4):1–14, 2015. 2
- [20] Petr Kellnhöfer, Lars C Jebe, Andrew Jones, Ryan Spicer, Kari Pulli, and Gordon Wetzstein. Neural lumigraph rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4287–4297, 2021. 3
- [21] Davis E King. Dlib-ml: A machine learning toolkit. *The Journal of Machine Learning Research*, 10:1755–1758, 2009. 5
- [22] Tianye Li, Timo Bolkart, Michael J Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4d scans. *ACM Trans. Graph.*, 36(6):194–1, 2017. 1, 2, 3, 4, 5, 6, 7, 8
- [23] Feng Liu, Luan Tran, and Xiaoming Liu. 3d face modeling from diverse raw scan data. In *ICCV*, pages 9408–9418, 2019. 2
- [24] Stephen Lombardi, Tomas Simon, Gabriel Schwartz, Michael Zollhofer, Yaser Sheikh, and Jason Saragih. Mixture of volumetric primitives for efficient neural rendering. *ACM Transactions on Graphics (TOG)*, 40(4):1–13, 2021. 2
- [25] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):1–16, 2015. 7
- [26] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks:

- Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4460–4470, 2019. 3
- [27] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 3
- [28] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5589–5599, 2021. 3
- [29] Roy Or-El, Xuan Luo, Mengyi Shan, Eli Shechtman, Jeong Joon Park, and Ira Kemelmacher-Shlizerman. Stylesdf: High-resolution 3d-consistent image and geometry generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13503–13513, 2022. 2, 3
- [30] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 165–174, 2019. 3
- [31] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5865–5874, 2021. 2, 3
- [32] Pascal Paysan, Reinhard Knothe, Brian Amberg, Sami Romdhani, and Thomas 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*, pages 296–301. Ieee, 2009. 3
- [33] Stylianos Ploumpis, Haoyang Wang, Nick Pears, William AP Smith, and Stefanos Zafeiriou. Combining 3d morphable models: A large scale face-and-head model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10934–10943, 2019. 2, 3
- [34] Amit Raj, Michael Zollhofer, Tomas Simon, Jason Saragih, Shunsuke Saito, James Hays, and Stephen Lombardi. Pixel-aligned volumetric avatars. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11733–11742, 2021. 3
- [35] Eduard Ramon, Gil Triginer, Janna Escur, Albert Pumarola, Jaime Garcia, Xavier Giro-i Nieto, and Francesc Moreno-Noguer. H3d-net: Few-shot high-fidelity 3d head reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5620–5629, 2021. 3
- [36] Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J Black. Generating 3d faces using convolutional mesh autoencoders. In *Proceedings of the European conference on computer vision (ECCV)*, pages 704–720, 2018. 2, 3
- [37] Ayush Tewari, Florian Bernard, Pablo Garrido, Gaurav Bharaj, Mohamed Elgharib, Hans-Peter Seidel, Patrick Pérez, Michael Zollhofer, and Christian Theobalt. Fml: Face model learning from videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10812–10822, 2019. 3
- [38] Ayush Tewari, Hans-Peter Seidel, Mohamed Elgharib, Christian Theobalt, et al. Learning complete 3d morphable face models from images and videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3361–3371, 2021. 3
- [39] Luan Tran, Feng Liu, and Xiaoming Liu. Towards high-fidelity nonlinear 3d face morphable model. In *CVPR*, pages 1126–1135, 2019. 2, 3
- [40] Luan Tran and Xiaoming Liu. On learning 3d face morphable model from in-the-wild images. *IEEE transactions on pattern analysis and machine intelligence*, 43(1):157–171, 2019. 2, 3
- [41] Daoye Wang, Prashanth Chandran, Gaspard Zoss, Derek Bradley, and Paulo Gotardo. Morf: Morphable radiance fields for multiview neural head modeling. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–9, 2022. 2
- [42] Lizhen Wang, Zhiyuan Chen, Tao Yu, Chenguang Ma, Liang Li, and Yebin Liu. Faceverse: a fine-grained and detail-controllable 3d face morphable model from a hybrid dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20333–20342, 2022. 5, 6, 7, 8
- [43] Xueying Wang, Yudong Guo, Zhongqi Yang, and Juyong Zhang. Prior-guided multi-view 3d head reconstruction. *IEEE Transactions on Multimedia*, 2021. 3
- [44] Ziyang Wang, Timur Bagautdinov, Stephen Lombardi, Tomas Simon, Jason Saragih, Jessica Hodgins, and Michael Zollhofer. Learning compositional radiance fields of dynamic human heads. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5704–5713, 2021. 2
- [45] Cheng-hsin Wu, Ningyuan Zheng, Scott Ardisson, Rohan Bali, Danielle Belko, Eric Brockmeyer, Lucas Evans, Timothy Godisart, Hyowon Ha, Alexander Hypes, Taylor Koska, Steven Krenn, Stephen Lombardi, Xiaomin Luo, Kevyn McPhail, Laura Millerschoen, Michal Perdoch, Mark Pitts, Alexander Richard, Jason Saragih, Junko Saragih, Takaaki Shiratori, Tomas Simon, Matt Stewart, Autumn Trimble, Xinshuo Weng, David Whitewolf, Chenglei Wu, Shou-I Yu, and Yaser Sheikh. Multiface: A dataset for neural face rendering. In *arXiv*, 2022. 6
- [46] Qiangeng Xu, Zexiang Xu, Julien Philip, Sai Bi, Zhixin Shu, Kalyan Sunkavalli, and Ulrich Neumann. Point-nerf: Point-based neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5438–5448, 2022. 3
- [47] Tarun Yenamandra, Ayush Tewari, Florian Bernard, Hans-Peter Seidel, Mohamed Elgharib, Daniel Cremers, and Christian Theobalt. i3dmm: Deep implicit 3d morphable model of human heads. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12803–12813, 2021. 2, 3, 8
- [48] Xiaoming Zhao, Fangchang Ma, David Güera, Zhile Ren, Alexander G Schwing, and Alex Colburn. Generative mul-

tiplane images: Making a 2d gan 3d-aware. *arXiv preprint arXiv:2207.10642*, 2022. [2](#), [3](#)

- [49] Yufeng Zheng, Victoria Fernández Abrevaya, Marcel C Bühler, Xu Chen, Michael J Black, and Otmar Hilliges. Im avatar: Implicit morphable head avatars from videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13545–13555, 2022. [2](#), [3](#), [5](#)
- [50] Yiyu Zhuang, Hao Zhu, Xusen Sun, and Xun Cao. Mofan-erf: Morphable facial neural radiance field. *arXiv preprint arXiv:2112.02308*, 2021. [2](#), [3](#)