Unsupervised High-Fidelity Facial Texture Generation and Reconstruction

Ron Slossberg*
Technion
ronslos@gmail.com

Ibrahim Jubran*
University of Haifa
ibrahim.jub@gmail.com

Ron Kimmel
Technion
ron@cs.technion.ac.il

Abstract

Many methods have been proposed over the years to tackle the task of facial 3D geometry and texture recovery from a single image. Such methods often fail to provide high-fidelity texture without relying on 3D facial scans during training. In contrast, the complementary task of 3D facial generation has not received as much attention. As opposed to the 2D texture domain, where GANs have proven to produce highly realistic facial images, the more challenging 3D geometry domain has not yet caught up to the same levels of realism and diversity.

In this paper, we propose a novel unified pipeline for both tasks, generation of both geometry and texture, and recovery of high-fidelity texture. Our texture model is learned, in an unsupervised fashion, from natural images as opposed to scanned texture maps. To the best of our knowledge, this is the first such unified framework independent of scanned textures.

Our novel training pipeline incorporates a pre-trained 2D facial generator coupled with a deep feature manipulation methodology. By applying precise 3DMM fitting, we can seamlessly integrate our modeled textures into synthetically generated background images forming a realistic composition of our textured model with background, hair, teeth, and body. This enables us to apply transfer learning from the domain of 2D image generation, thus, benefiting greatly from the impressive results obtained in this domain.

We provide a comprehensive study on several recent methods comparing our model in generation and reconstruction tasks. As the extensive qualitative, as well as quantitative analysis, demonstrate, we achieve state-of-the-art results for both tasks.

1. Introduction

Generation of 3D facial geometry and full texture, as well as their reconstruction from a single 2D image, are highly challenging and important tasks at the intersection of computer vision, graphics, and machine learning. These tasks arise within endless applications ranging from virtual reality and computer gaming to facial editing.

Figure 1. **Left:** Generation. **Right:** Reconstruction.

At the heart of such generation and reconstruction methods lies a hidden common assumption that natural facial geometries and textures reside on a low-dimensional manifold. Following this assumption, the above tasks can be carried out within this simpler representation space, instead of the original high-dimensional space. The recovery of this manifold is termed *facial modeling* and the mathematical bridge between the high and low dimensional representations is termed a *facial model*.

Many different types of facial models have been in use, including linear models, non-linear deep learning-based models, hybrid models, implicit modeling, and direct dense landmark regression methods; see more details in Section 2. However, regardless of the model type used, it is of great importance to have two intertwined, rather than decoupled, models for the geometry and texture, for more realistic results. This is more crucial for synthesis tasks, where the newly generated faces must adhere to the combined geometry and texture manifold.

In previous efforts, training a synthesis model for fa-

*Authors contributed equally.

cial geometry and texture was either: (i) dependent on 3D facial scans (*i.e*. via supervised learning) and produces high-quality results, or (ii) dependent on 2D facial images only (*i.e*. unsupervised or semi-supervised), but produced low-quality, inaccurate, or incomplete models; see detailed overview in Section 2. Our work combines the best of both worlds, and provides an unsupervised training pipeline, independent of 3D facial scans, producing state-of-the-art modeling results on par even with fully supervised methods. Our high-resolution model is achieved by incorporating a linear as well as a direct regression facial model, a pre-trained 2D generative model, a deep feature manipulation component, and a differentiable rendering layer, as the building blocks for our unsupervised training pipeline.

2. Background and Related Efforts

Next, we review related efforts. Techniques that have been incorporated in the proposed pipeline are described in detail.

The 3D Morhpable Model (3DMM) [1] is arguably the most commonly used model both when generating or reconstructing facial geometries and textures; see survey [9] and Section 2. The 3DMM model is obtained by semantically aligning facial scans to a template model comprised of n vertices and performing PCA analysis [15] on the geometry, texture and expression vectors. The obtained k principal components for shape and expression $\mathbf{U} \in \mathbb{R}^{3n\times k}$ and mean shape $\mathbf{M} \in \mathbb{R}^{3n}$ comprise the 3DMM model. Given a set of shape and expression parameters $(\mathbf{p}_s \in \mathbb{R}^k, \mathbf{p}_e \in \mathbb{R}^k)$ the facial geometry is constructed as $\mathbf{S} = \mathbf{M} + \mathbf{U}_s \cdot \mathbf{p}_s + \mathbf{U}_e \cdot \mathbf{p}_e$. Texture modeling and formation are produced per-vertex in a similar manner. Many improvements were suggested, for example, [3, 2], who improve the data acquisition and registration processes. However, due to their linear nature, such models usually produce unrealistic over-smoothed samples [31]. As a remedy, many non-linear models have shown successful results in recent years; see more details in what follows.

3DMM fitting. Given a 2D face image and the 3DMM geometry and expression basis, the goal of *3DMM fitting (or, regression)* is to recover the 3DMM geometry and expression coefficients as well as a 3D rigid transformation $(\mathbf{R}, \mathbf{t})$. This is achieved by minimizing some rendering loss between the given 2D image and a face rendered using the above coefficients and parameters. Numerous approaches have been suggested for tackling this problem, ranging from optimization-based methods, like [1, 11], to one-shot deep learning pipelines such as the pioneering papers by [27, 41], more recently followed by [35, 13, 7, 14] to name a few. In our proposed texture generation pipeline we utilize the work of [7], due to their state-of-the-art precision in estimating the geometry, expression, texture, and illumination parameters, as well as their available code and pre-trained model; see Section 3.

Non-linear and hybrid model fitting. Recent efforts have built upon classical 3DMMs, proposing both hybrid [28, 34, 29, 31, 32, 11, 4, 30] and completely non-linear models [38, 40]. These deep network-based methods may incorporate linear parts supplemented by DNN-based parts and in other cases are entirely network-based. Some models are presented only in the context of monocular geometry and texture recovery while others are also utilized in the context of synthesis. We find that models dedicated to one of those tasks often do not extend well to the other.

Dense landmark regression. In [19], a regression network is trained to predict a dense collection of landmarks directly on a given facial 2D image. These landmarks represent the projected vertex locations of a 3D canonical facial model. This method achieves better fitting to the facial image and is not constrained to the limitations of the linear model. On the other hand, the extracted facial geometry is less accurate and detailed. Hence, this method is better suited for AR and other image-related tasks, *e.g*. mouth-region mask extraction from 2D images; see Section 3.4.

Realistic 2D face generation. In a long line of work culminating in [16], various models have been proposed for the task of 2D face image generation. Such models are capable of generating highly realistic 2D facial images as well as project real 2D facial images onto the model's latent manifold. As our model aims to mitigate the need for 3D scans of facial textures, we heavily rely on well-established 2D facial generative models as the basis for our proposed pipeline, successfully harnessing their high level of realism. Throughout the proposed pipeline, we utilize the architecture, training methodologies as well as pre-trained model weights produced by the seminal papers of Karras *et al*. [17, 16], which have been shown to produce high-resolution realistic images; see Section 3.

Manipulating facial properties via deep feature mapping. Most synthesis methods described above, specifically [17], learn to map an input random noise vector, through some latent representation, into a realistic 2D facial image. Following this popular approach, a variety of papers have emerged which learn to manipulate this intermediate latent vector to change some desired facial properties in the output 2D image. Such manipulation can be either statistics-based [5] or, more often, learning based [33]; see survey in [36] and references therein.

As will be discussed in detail in Section 3, our pipeline makes use of a 2D facial image generator to compensate for the lack of 3D facial scans. However, it is very challenging to compensate for 3D geometry and full facial texture using only *randomly generated* 2D facial images. To this end, we utilize a component that can control the pose of those generated images and show that the *controlled* 2D images indeed suffice for full-texture learning. To this end, we utilize the work of [33] for deep feature manipulation; see Section 3.

Reconstruction. In a long line of research, many methods have been suggested for 3D face reconstruction from a given 2D image. In [27, 28, 29, 8], a mapping from 2D images to a 3D geometric representation is learned based on synthetic data pairs. Real facial textures were used, for example, in [11, 12], to obtain, in a supervised manner, a realistic reconstruction. However, acquiring such textures requires 3D scanning of faces, which is very costly and laborious, hence, impractical to scale to large numbers. In this paper, we provide an unsupervised alternative that simultaneously requires no 3D scans, and achieves either comparable or higher quality reconstructions. A pipeline for completion of a facial texture containing large holes was suggested in [6]. A different approach that learns to ignore incomplete regions via masking was suggested in [30]. However, those also rely on textures as training data. A one-shot learning approach was proposed in [10] which applies an iterative and very slow optimization process to complete a facial texture. The only other unsupervised reconstruction model we know of was proposed by [20]. This work utilizes Graph Convolutional Networks to generate impressive high-quality reconstruction results; However, by not basing their pipeline on a 2D image generator which can produce controlled 2D images (*e.g*. *StyleGAN* combined with a model as [33]), their method is, by design, not intended for the task of generation of expressive 3D models, might lack important facial details, and does not account for the coupling of geometry and texture. See detailed comparison with [20] in Section 4.

Generation. While most prior efforts have focused on reconstruction, some methods have been proposed for the generation of random but realistic facial models. In [22], who focus on improving facial recognition models via synthetic augmentation a GAN-based approach is proposed as well; however, their pipeline focuses more on controlling model parameters intending to supplement the training data for facial recognition models. This focus, as opposed to the realistic generation of completely random faces, leads to a less desirable outcome in terms of realism and resolution. Hence, the results are not visually pleasing as depicted in Fig. 5. In [31, 30, 11], 3DMMs combined with generative models were used for either generation or reconstruction of realistic textures. However, these methods use high-quality facial scans during training, which were obtained by specialized facial scanners and are not publicly available. This makes these methods very difficult to utilize in practice. Another downside is that limited scanned data is far less diverse than abundant facial 2D images commonly found in many datasets.

The following question now naturally arises: **can we generate or reconstruct high quality, realistic, and diverse 3D facial texture and corresponding geometry, by learning from 2D image data only?**

2.1. Our contribution

The main contributions of our method are the following:

(i) We affirmatively answer the question above and provide the first unsupervised high-fidelity generation as well as reconstruction pipeline capable of producing realistic textures coupled with corresponding geometries.

(ii) Our pipeline decouples intrinsic texture features related to the person's identity, from extrinsic properties such as pose and lighting. This allows us to tackle the challenging problems of reconstructing a facial texture including full high-quality side views, as well as performing model re-illumination. See Section 3.2.

(iii) We present state-of-the-art results in both model generation as well as full texture recovery. We support this claim via both qualitative as well as quantitative results and comparisons. See Section 4.

(iv) Our results are fully reproducible as only freely available datasets and models are required during training and inference. In addition, we provide all our trained model weights for both generation and reconstruction tasks [23].

3. Unsupervised Learning of Facial Textures and Geometries

In this section, we dive into the details of our unsupervised 3D facial geometry and full texture generation pipeline and its components. While our pipeline learns to recover both geometry and texture, our novelty lies mainly in high-quality texture generation and retrieval, as we believe that the main effect on the perception of model realism stems from high-resolution texture rather than highly detailed geometry. This was also noted by [31] who demonstrated this point by varying texture and geometry quality while observing the overall effect on model realism. Nevertheless, recovery of highly detailed geometry is still an important research topic with many successful efforts such as [28, 29, 37, 4] to name a few.

An overview of our training and inference pipelines is depicted in Figs. 2 and 3 respectively. Our approach to unsupervised learning of facial textures utilizes an adversarial loss to train a texture generator, G_T, while harnessing a

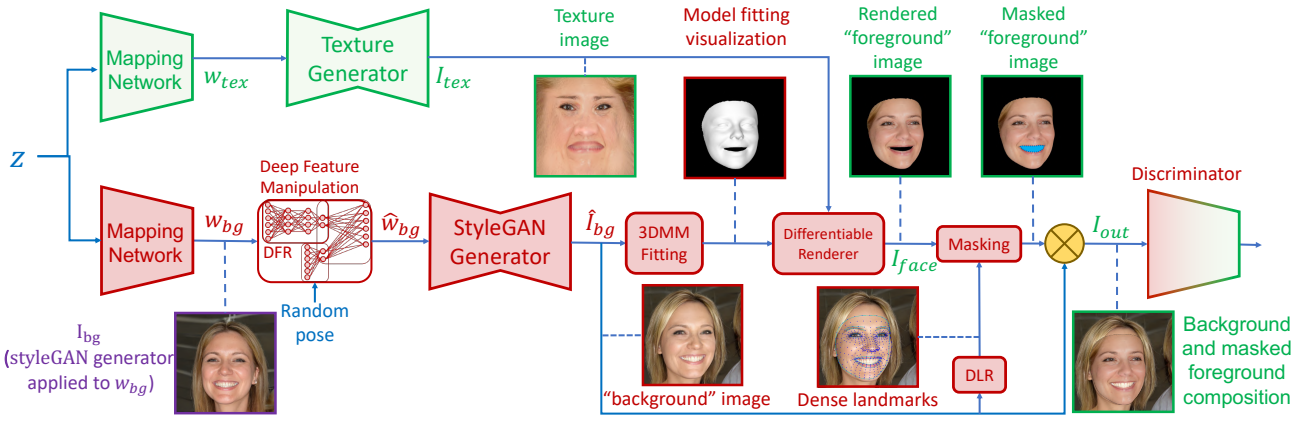

Figure 2. **Our training pipeline.** A vector $\mathbf{z} \in \mathbb{R}^{512}$ of Gaussian random noise is plugged into two mapping networks [18] with the same architecture, producing two latent vectors $\mathbf{w}_{tex}, \mathbf{w}_{bg} \in \mathbb{R}^{18\times512}$, respectively. The facial image encoded within $\mathbf{w}_{bg}$ is illustrated in purple. The vector $\mathbf{w}_{bg}$ is plugged into a deep feature manipulation network [33] called *StyleRig* to obtain the (manipulated) latent vector $\hat{\mathbf{w}}_{bg} \in \mathbb{R}^{18\times512}$ which encodes the same facial information as $\mathbf{w}_{bg}$ (*e.g.* facial expression, identity, lighting, etc.) but with a modified facial orientation or expression. We then feed $\mathbf{w}_{tex}$ and $\hat{\mathbf{w}}_{bg}$ into our texture and pre-trained *StyleGAN* generators G_T and G_b, outputting a texture image I_{tex} and a 2D facial image $\hat{I}_{bg}$ respectively. We then recover 3DMM parameters $\hat{\mathbf{p}}_{bg}$ that best fit $\hat{I}_{bg}$ [7] (illustrated by the textureless facial geometry), and use them to render the texture I_{tex} into a 2D face image I_{face} superimposed on $\hat{I}_{bg}$. We then mask out the mouth area of I_{face} using a mouth mask (in bright blue) recovered from I_{bg} using a dense landmark regression (DLR) model [19]. The masked facial (foreground) image and the (background) image $\hat{I}_{bg}$ are then composed together to form the output image I_{out}. Finally, I_{out} is fed into a pre-trained discriminator which is further trained. Trainable and pre-trained models are depicted in green and red respectively.

pre-trained 2D facial image generator, G_{bg}, in the following fashion. We start by generating, via G_{bg}, a 2D facial image which we term a *background image*. We then fit a corresponding geometry to the background image using a pre-trained 3DMM fitting model [7]. We then proceed to generate a facial texture I_{tex} via our trainable texture generator G_T. Our generated synthetic texture is stored as a 2D image coupled with a canonical UV parametrization relating between image locations to the vertices of the 3D facial model. The model fitted to the background image enables the seamless mapping of our synthetic texture image I_{tex} into the background image as depicted in Fig. 2.

Our texture generator is trained within a GAN framework for which a discriminator model, serving as a trainable loss function, is trained to differentiate between blended and real images and thus continuously improves the generator quality. In order to generate high-resolution facial textures from all viewing angles, it is crucial to control various properties within the images generated by G_{bg}. For example, we require that each generated identity appears under a range of poses. We, therefore, utilize a deep feature manipulation component, as proposed by [33], that encodes the desired properties within the input of G_{bg}. In addition, in order to disentangle between the albedo and shading components of the texture, we estimate the lighting conditions in G_{bg} and apply them to our texture within the rendering process. Section 3.2 further elaborates on these critical components.

Learning from 3D facial scans. Prior efforts approached the task of training facial texture models by relying on difficult-to-obtain 3D scans. For example, in [31, 11], high resolution scans obtained by a 3DMD scanner are geometrically aligned and mapped to a canonical 2D domain. The mapped textures are used as training data for a GAN which is tasked to generate new and realistic ones. This methodology suffers from several drawbacks, *e.g.*, (i) The 3D scans are not easily obtained or freely distributed, thus posing a significant barrier in reproducing such models. (ii) High-quality 3D scanners are expensive and cumbersome, limiting the ability to collect data. Hence, even when available, such datasets are comprised of at most a few thousand subjects, which can not encompass the huge variety of human faces. We mitigate the above issues by eliminating the dependency on scans and replacing them with widely available 2D facial images, thus providing a more accessible method and producing a more diverse texture model.

Obtaining 3D facial models from 2D images. To replace the difficult-to-obtain 3D facial scans with prevalent 2D facial images, it is common to utilize a differentiable rendering layer. The rendering of 3D textured models into 2D images enables the utilization of 2D image-related architectures and losses. This process also requires a 3D mesh, usually represented by a pair (V, Tri) of vertex coordinates $V \in \mathbb{R}^{N_v\times3}$ and triangulation $Tri \in \mathbb{R}^{N_f\times3}$, as well as a uv parametrization $\phi : \{1, \cdots, N_v\} \rightarrow [0, 1] \times [0, 1]$

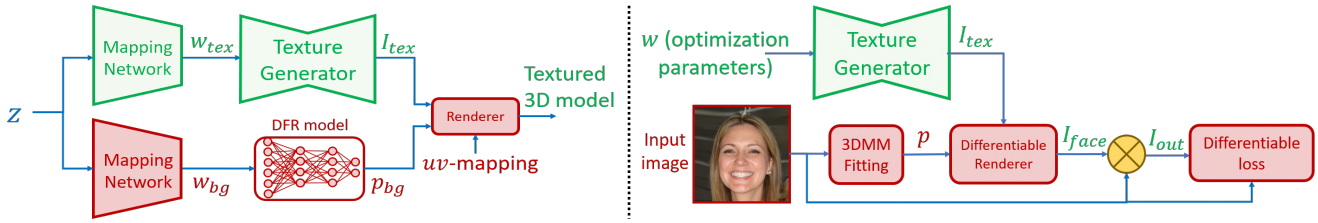

Figure 3. **Our inference pipelines.** During inference, we drop some components related to the training pipeline (see Fig. 2). **(Left) Generation:** As before, a single latent vector $\mathbf{z}$ is used to generate $\mathbf{w}_{tex}$ and $\mathbf{w}_{bg}$ via two mapping networks. The latent vector $\mathbf{w}_{bg}$ is used to generate 3DMM geometry parameters $\mathbf{p}_{bg}$ via the trained DFR model while $\mathbf{w}_{tex}$ is introduced to the trained texture generator yielding the corresponding texture image I_{tex}. The parameters $\mathbf{p}_{bg}$ are used, along with our canonical UV parametrization, to generate the 3DMM geometry which we render using I_{tex} as the mesh texture. **(Right) Reconstruction:** A given input image I is first plugged into a fitting model producing its 3DMM parameters $\mathbf{p}$. A latent vector $\mathbf{w}$ containing our optimization parameters is then inserted into our trained texture generator producing a texture image I_{tex}. Using a differentiable renderer, $\mathbf{p}$ and I_{tex} are rendered into a 2D face image I_{face}, which is superimposed on I to produce our output I_{out}. Finally, a VGG loss similar to [16] is evaluated between I and I_{out}.

that maps every vertex to coordinates on the canonical plane. The vertex coordinates are first projected onto the 2D camera plane and the final pixel colors are determined by a rasterization process mapping the facial texture onto the projected mesh according to the predetermined UV parametrization, to obtain the desired facial rendering I_{face}.

Using this methodology, we can transform our training losses from the 3D to the 2D facial domain. We can thus incorporate the vast corpus of prior art regarding 2D images, including pre-trained models as well as large, high resolution, and freely available datasets; see Section 3.1.

Having established the above, the question remains how to obtain synthetic facial renderings which are indistinguishable from real facial images, considering that the rendered images lack hair, ears, inside of the mouth, background, etc. Possible solutions include: **(i)** To segment the foreground (the face) in the real images, and ignore the rest of the image. Thus both real and rendered images contain a foreground (facial) region, imposed on an empty background. Since the segmentation of the real images is not correlated with the geometry which produces the fake renderings, the resulting foreground can be easily distinguished from the rendered examples. **(ii)** Alternatively, one can apply the same geometry fitting methodology used during the generation of the fake images, to the real facial images. This fitting provides the facial boundary, which can be used to mask out the background. While this yields better results, the geometry fitting is not perfectly aligned at the pixel level causing easily distinguishable artifacts at the face boundary.

To overcome the above limitations, we propose to generate an additional 2D facial image I_{bg}, *e.g.* using *StyleGAN*, and utilize I_{bg} as the background to our (foreground) rendered facial image I_{face}. This is achieved by first fitting a geometric model to I_{bg}, which serves as the 3D mesh required for rendering I_{tex} into a 2D image I_{face}, as previously detailed. This process embeds our synthetic facial texture image I_{tex} into I_{bg}, enforcing the facial texture to be generated in a way that realistically blends with the surrounding parts in I_{bg} (*e.g.*, hair and ears); see Fig. 2.

3.1. Transfer learning

The process described above results in a 2D facial image, enabling the use of standard 2D image losses. As common in generative models, we use an adversarial loss to discriminate between real and fake images. Fortunately, many such pre-trained GANs are available for the task of 2D facial image generation [16]. For the mapping network, texture generator, and discriminator, we use the architecture proposed in *StyleGAN*2 [18]. As facial textures are closely related to 2D facial images, we initialize the above models with the pre-trained *StyleGAN*2 weights. This transfer learning approach has dramatically reduced our pipeline training time and improves texture quality, as was also reported by [16].

3.2. Pose and illumination invariant textures

As detailed above, our unsupervised approach relies on rendering 2D images from the generated textures. However, this approach, without further improvements, has one inherent problem: when a vast majority of the background images I_{bg} contain, for example, frontal faces, our textures generator adequately learns to generate textures with high-resolution frontal face details but fails to produce high-resolution details on the periphery of the face. We propose to solve this issue by introducing random facial rotations during training via deep feature manipulation.

In addition, without properly addressing scene lighting, the generator will incorporate the lighting into the generated texture; see Fig. 6. However, it is desirable to decouple the albedo from the illumination effects, enabling post-relighting of the texture. Hence, we relight the models during training using the lighting parameters recovered by [7].

Deep feature pose manipulation. In order to overcome the first problem above, we manipulate the latent vector

w_{bg}, from which the background image I_{bg} is generated, to enforce the generation of faces in a variety of orientations. To this end, we perform deep feature manipulation by adopting the methodology proposed in [33].

The manipulation model, termed *StyleRig*, is comprised of two parts. A Differentiable Face Reconstruction Network, or DFR model, which takes as input the latent vector $\mathbf{w}$ and produces estimated 3DMM parameters $\mathbf{p} = DFR(\mathbf{w})$ which include $(\mathbf{p_s}, \mathbf{p_e}, \mathbf{p_t}, \gamma, \mathbf{R}, \mathbf{t})$, shape, expression, texture and lighting, rotation and translation parameters respectively. We train our model utilizing the highly versatile 3DMM model generated by [3]

A second network termed *StyleRig* takes as input a latent vector $\mathbf{w}$ and a set of parameters $\mathbf{p}$ and outputs a modified set of latent parameters $\hat{\mathbf{w}}$, where ideally the image produced by $I = G_{StyleGan}(\hat{\mathbf{w}})$ portrays the face $G_{StyleGan}(\mathbf{w})$ produced by $\mathbf{w}$ but modified to fit the parameters $\mathbf{p}$. In order to produce a rotated version of I_{bg} we first modify the rotation parameters of $\mathbf{p}_{bg} := DFR(\mathbf{w}_{bg})$ to derive $\hat{\mathbf{p}}_{bg}$ and then apply $\hat{\mathbf{w}}_{bg} = Stylerig(\hat{\mathbf{p}}_{bg}, \mathbf{w}_{bg})$. The image $\hat{I}_{bg}$ derived from $\hat{\mathbf{w}}_{bg}$ contains a rotated version of the same person as in I_{bg}. We then generate a texture image using the latent vector $\mathbf{w}_{tex}$, regardless of the rotation angles which were modified in $\mathbf{w}_{bg}$. This yields the desired pose-invariance within the texture generator; see Fig. 2

The same DFR model used above will be also utilized during inference in order to recover corresponding geometries for our generated textures; see Section 3.3 and Fig. 3. This allows us to efficiently generate corresponding geometries directly from latent vectors without using the trained *StyleGAN*generator during inference.

Training for re-illumination. To generate textures with no illumination effects, we first estimate the background scene lighting and relight the texture during training. Assuming a Lambertian reflectance model, we estimate the parameters $\gamma_b \in \mathbb{R}^{3x9}$ from I_{bg}, as coefficients of 9 Spherical Harmonics (SH) basis functions [25, 26] for R,G and B illumination bands, and relight the rendered image I_{face} under the recovered illumination. The coefficients γ_b with the computed vertex normals $\mathbf{n}_i$ and SH functions Φ_b produce the per-vertex lighting value as $\mathbf{C}(\mathbf{n}_i|\gamma) = \sum_{b=1}^{B^2} \gamma_b \Phi_b(\mathbf{n}_i)$. We perform two renderings, one for illumination and one for albedo, and perform pixel-wise multiplication to derive the illuminated rendering

$$I_{face} = \mathcal{R}(\mathbf{S}(p_s, p_e), G(w_{tex})) \cdot \mathcal{R}(\mathbf{S}(p_s, p_e), \mathbf{C}(\mathbf{n}_i|\gamma)),$$

where $\mathcal{R}(G, T)$ signifies the rendering operator applied to a geometry G and a texture T, $\mathbf{p}_s, \mathbf{p}_e$ are respectively shape and expression parameters recovered from I_{bg} and w_{tex} is the input latent vector for the texture generator.

This process results in the texture generator producing textures with no baked-in lighting effects, *i.e*., textures that incorporate the albedo only, so that the re-illuminated texture via γ would match the lighting present in the background image I_{bg} and seem realistic to the discriminator.

3.3. Recovering corresponding geometry

As detailed in Section 3.2, a trained DFR model can recover the geometry parameters (such as the parameters obtained via a geometry fitting pipeline, as in Sections 2). However, we note that the DFR extracts those parameters directly from a latent vector w and not from an image. As already part of our pipeline, the latent vector w_{bg} is plugged into the deep feature manipulation component, which contains the DFR model; see Fig. 2. Hence, as a byproduct of this pipeline, we obtain geometry parameters designated for the facial image to be later rendered. However, the simplicity of the DFR model and the fact that it does not directly observe the 2D image causes imprecise fitting results. In order to retain realism of the composed I_{out} during training, we must employ better fitting methodologies onto I_{bg} such as [7]. We will find a use for the DFR model when performing 3D synthesis as it relates between latent and 3D model parameters effectively, thus, encoding the desired coupling between texture and geometry; see Fig. 3.

3.4. Masking mouth area

As fitting a 3DMM to a 2D image is a challenging task, minor fitting errors are inevitable. The facial region mask rendered using the recovered 3DMM parameters thus might not perfectly align with the face in the 2D image, causing the (empty) mouth region in the mask to misalign with the mouth in the background image. Hence, for the rendered texture in the mask region to realistically blend with the background image, the generator is falsely forced to generate undesired details, *e.g*. teeth on top of the lips region; see example in Fig. 6. To prevent this from happening, we incorporate the work of [19], which performs a dense facial landmark regression on the 2D image, to find the mouth region with high accuracy. We then remove this predicted mouth region from the facial mask obtained by the 3DMM fitting procedure. This forces the rendered mask to contain an "empty region" in the mouth region of the background image. By further masking the inner-mouth area we can eliminate the appearance of teeth within our generated textures, as illustrated in Fig. 6.

3.5. Fully unsupervised training

The proposed pipeline above generates full facial textures along with corresponding geometries, and, using a differentiable renderer, synthesizes a 2D facial image that is plugged into a discriminator model. Besides the synthetic 2D images above, real 2D facial images are also fed into the discriminator during training. Such 2D real images are widespread and can be taken from any dataset of facial im-

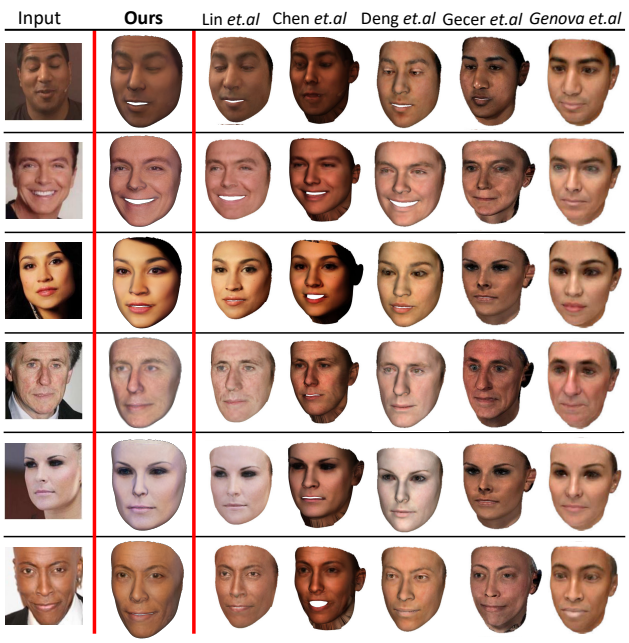

Figure 4. **Qualitative Reconstruction Comparison Results:** We present our texture reconstruction results on the MOFA test-set [35] and compare to previous results [20, 4, 7, 11, 13], respectively. This figure is best viewed when zoomed in.

ages, *e.g*. [17]. Moreover, we can utilize the (already in use) *StyleGAN* generator to generate such 2D images during training, rather than use an existing dataset. As the common 2D image datasets are huge, replacing them with a pre-trained model can be very useful when either the storage memory is limited or communication time is crucial.

4. Experimental results

We compare our proposed approach to several state-of-the-art texture reconstruction and 3D generation methods, most of which utilize facial scans for training. We provide quantitative as well as qualitative evidence demonstrating that our model outperforms previous methods, both supervised and unsupervised by scanned textures, in terms of texture reconstruction quality, realism, and details.

Implementation details. We implemented our pipeline in Python using Pytorch [24] library and trained it on 4 RTX 3090 GPUs on the FFHQ dataset [17]. As mentioned in Section 3.1, we initialized our models from the pre-trained weights of *StyleGAN* [18], using default parameters and their recommended losses.

4.1. Face generation

We randomly generated textures and corresponding geometries via our inference pipeline; see Section 3 and Fig. 3. We present the texture images with zoomed-in areas to highlight the high level of detail and realism. We compare our results to the supervised model proposed by of [30] as well as the unsupervised model from [22]; see Fig. 5. Additional randomly generated faces are included within the supplementary material.

Metric	[7]	[22]	[6]	**Ours**
L_1 distance ↓	0.052	0.034	/	**0.0244**
PSNR ↑	26.58	29.69	22.9∼26.5	**32.889**
SSIM ↑	0.826	0.894	0.887∼0.898	**0.972**
LightCNN [39] ↑	0.724	0.900	/	**0.96**

Table 1. **Quantitative Evaluation:** We evaluate our reprojected reconstruction similarity on the CelebA[21] test-set. Our analysis shows that our method achieves better similarity scores in all reported metrics when compared to previous methods [6, 7, 22].

4.2. Facial texture reconstruction

Fig. 4 presents a qualitative comparison between our texture reconstruction pipeline from Fig. 3 to several state-of-the-art prior [20, 4, 7, 11, 13]. The comparison demonstrates that our model can reproduce challenging textures *e.g*. difficult lighting conditions, makeup, and extreme expressions and compares favorably to previous approaches, including methods based on supervised training from 3D scans. Note that we utilize [7] for geometry recovery and thus focus our comparison on texture recovery only. Additional reconstruction results produced from high-resolution images are depicted in Fig. 1 and the supplementary material. The results demonstrate that our model is capable of high-resolution texture recovery when presented with high-quality input images.

4.3. Ablation study

In Fig. 6 we present an ablation study, where the full proposed model is shown to consistently produce more realistic results compared to its variants with missing components. This suggests that each of our pipeline components is crucial for producing satisfactory output results. We show that: (i) model rotations during training are crucial for generating high details on the peripheral areas of the texture; see Section 3.2, (ii) mouth masking eliminates the unwanted teeth artifacts; see Section 3.4, and (iii) model illumination during training successfully disentangles albedo from shading, producing models that can be realistically integrated into scenes with varying lighting conditions; see Section 3.2.

4.4. Quantitative results

Table 1 presents our quantitative study on the task of texture reconstruction conducted on the CelebA [21] test-set, containing nearly 20k images. We test our reconstruction quality using three classical image reconstruction metrics and one based on a face recognition model [39]. Our method achieves better scores in all tested metrics compared

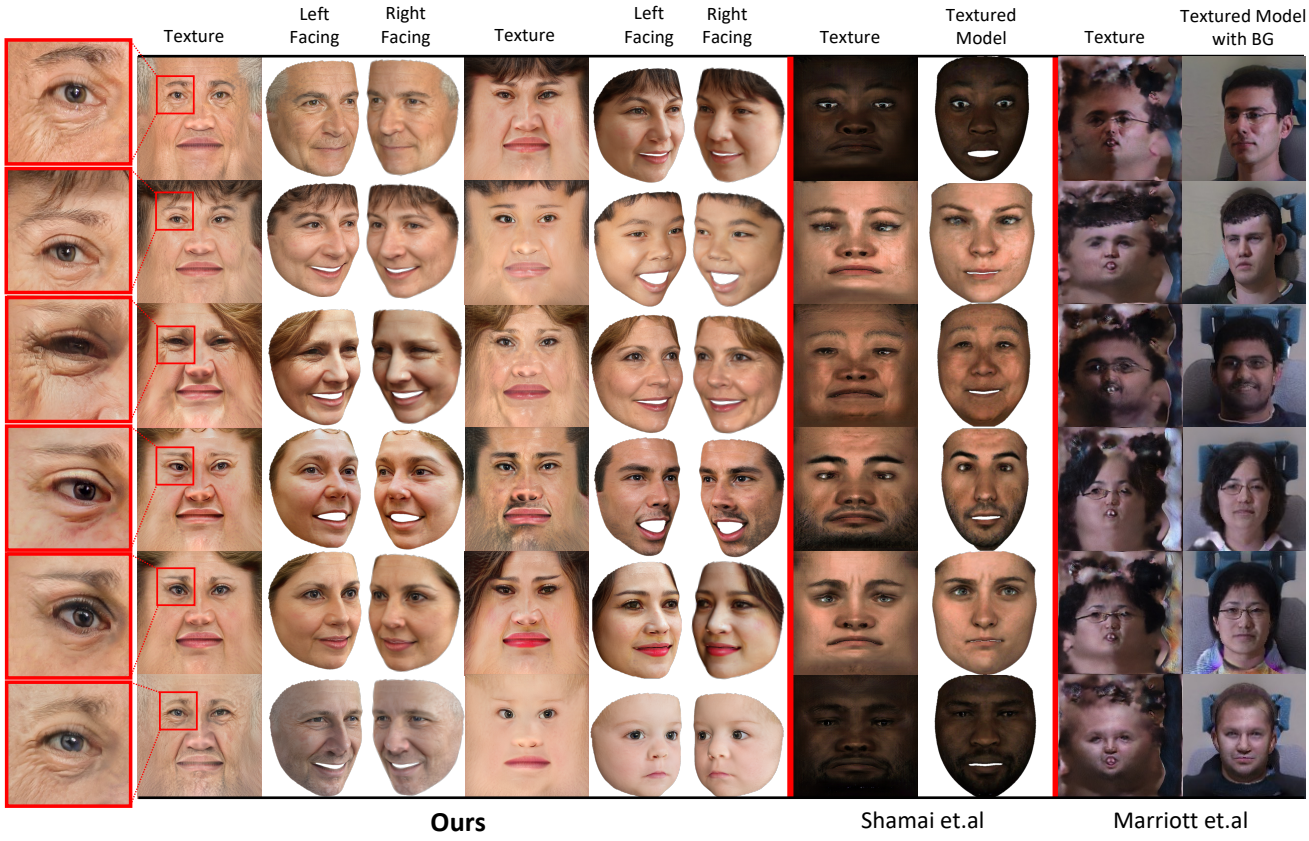

Figure 5. **Facial Synthesis Comparison:** We visually compare our results both in texture and rendered textured geometries (in multiple views) to: Shamai *et al*. [30] and Mariott *et al*. [22]. Our high resolution textures provide highly realistic faces spanning a wide variety of ages, ethnicity and appearance. The leftmost column provides a zoomed-in crop, highlighting the high resolution realistic details. Our method still presents higher level detail and realism as compared to both previous methods, although [30] is supervised by scanned textures.

Figure 6. **Ablation Study.** Left to right: full model, without rotations, without mouth masking, and without relighting. Removing those 3 components yields poor details in the texture periphery, unwanted teeth, and baked-in lighting, respectively.

to the state-of-the-art methods [7, 22, 6]. In contrast to [22], we do not omit problematic areas by masking.

5. Discussion, Limitations, and Future Work

We introduced a novel unsupervised pipeline for both generation and reconstruction of high resolution realistic facial textures utilizing the acclaimed *StyleGAN* framework. Our experiments demonstrate that we surpass prior art in both tasks, including models based on supervised training via scanned facial textures, in both reconstruction quality and generation realism. Our proposed model is capable of generating realistic geometries based on 3DMM modeling and importantly matches the geometry and texture via a single unified random input vector $\mathbf{z}$. We also plan to release our trained models including sample code for generation as well as reconstruction.

Due to the presence of subjects wearing glasses within the FFHQ dataset used for training, in some cases, our output texture might contain glasses. See Supplementary material for such examples. This can be mitigated in future work by using latent feature manipulation or simply removing subjects with glasses from the training set. In this paper, we did not utilize non-linear geometric representations as we note that high-resolution texture is the most crucial

component in the quest for realistic facial generation. Improving geometry quality by utilizing non-linear representations and leveraging normal maps is left for future work.

References

[1] 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] 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.

[3] 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.

[4] Anpei Chen, Zhang Chen, Guli Zhang, Kenny Mitchell, and Jingyi Yu. Photo-realistic facial details synthesis from single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9429–9439, 2019.

[5] Ying-Cong Chen, Huaijia Lin, Michelle Shu, Ruiyu Li, Xin Tao, Xiaoyong Shen, Yangang Ye, and Jiaya Jia. Faceletbank for fast portrait manipulation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3541–3549, 2018.

[6] Jiankang Deng, Shiyang Cheng, Niannan Xue, Yuxiang Zhou, and Stefanos Zafeiriou. Uv-gan: Adversarial facial uv map completion for pose-invariant face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7093–7102, 2018.

[7] Yu Deng, Jiaolong Yang, Sicheng Xu, Dong Chen, Yunde Jia, and Xin Tong. Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[8] Pengfei Dou, Shishir K Shah, and Ioannis A Kakadiaris. End-to-end 3d face reconstruction with deep neural networks. In *proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5908–5917, 2017.

[9] Bernhard Egger, William AP Smith, Ayush Tewari, Stefanie Wuhrer, Michael Zollhoefer, Thabo Beeler, Florian Bernard, Timo Bolkart, Adam Kortylewski, Sami Romdhani, Christian Theobalt, Volker Blanz, and Thomas Vetter. 3d morphable face models—past, present, and future. *ACM Transactions on Graphics (TOG)*, 39(5):1–38, 2020.

[10] Baris Gecer, Jiankang Deng, and Stefanos Zafeiriou. Ostec: One-shot texture completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7628–7638, 2021.

[11] Baris Gecer, Stylianos Ploumpis, Irene Kotsia, and Stefanos Zafeiriou. Ganfit: Generative adversarial network fitting for high fidelity 3d face reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1155–1164, 2019.

[12] Baris Gecer, Stylianos Ploumpis, Irene Kotsia, and Stefanos Zafeiriou. Fast-ganfit: Generative adversarial network for high fidelity 3d face reconstruction. *arXiv preprint arXiv:2105.07474*, 2021.

[13] Kyle Genova, Forrester Cole, Aaron Maschinot, Aaron Sarna, Daniel Vlasic, and William T Freeman. Unsupervised training for 3d morphable model regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8377–8386, 2018.

[14] Jianzhu Guo, Xiangyu Zhu, Yang Yang, Fan Yang, Zhen Lei, and Stan Z Li. Towards fast, accurate and stable 3d dense face alignment. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[15] Ian T Jolliffe. Principal components in regression analysis. In *Principal component analysis*, pages 129–155. Springer, 1986.

[16] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In *Proc. NeurIPS*, 2020.

[17] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019.

[18] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020.

[19] Yury Kartynnik, Artsiom Ablavatski, Ivan Grishchenko, and Matthias Grundmann. Real-time facial surface geometry from monocular video on mobile gpus. In *Proceedings of CVPR Workshops*, 2019.

[20] Jiangke Lin, Yi Yuan, Tianjia Shao, and Kun Zhou. Towards high-fidelity 3d face reconstruction from in-the-wild images using graph convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5891–5900, 2020.

[21] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[22] Richard T Marriott, Sami Romdhani, and Liming Chen. A 3d gan for improved large-pose facial recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13445–13455, 2021.

[23] Pretrained Models. The weights for all our pretrained models., 2021. the authors commit to publish upon acceptance of this paper or reviewer request.

[24] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037, 2019.

[25] Ravi Ramamoorthi and Pat Hanrahan. An efficient representation for irradiance environment maps. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 497–500, 2001.

[26] Ravi Ramamoorthi and Pat Hanrahan. A signal-processing framework for inverse rendering. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 117–128, 2001.

[27] Elad Richardson, Matan Sela, and Ron Kimmel. 3d face reconstruction by learning from synthetic data. In *2016 fourth international conference on 3D vision (3DV)*, pages 460–469. IEEE, 2016.

[28] Elad Richardson, Matan Sela, Roy Or-El, and Ron Kimmel. Learning detailed face reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1259–1268, 2017.

[29] Matan Sela, Elad Richardson, and Ron Kimmel. Unrestricted facial geometry reconstruction using image-to-image translation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1576–1585, 2017.

[30] Gil Shamai, Ron Slossberg, and Ron Kimmel. Synthesizing facial photometries and corresponding geometries using generative adversarial networks. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 15(3s):1–24, 2019.

[31] Ron Slossberg, Gil Shamai, and Ron Kimmel. High quality facial surface and texture synthesis via generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018.

[32] Ayush Tewari, Florian Bernard, Pablo Garrido, Gaurav Bharaj, Mohamed Elgharib, Hans-Peter Seidel, Patrick Pérez, Michael Zöllhofer, and Christian Theobalt. Fml: Face model learning from videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10812–10822, 2019.

[33] Ayush Tewari, Mohamed Elgharib, Gaurav Bharaj, Florian Bernard, Hans-Peter Seidel, Patrick Pérez, Michael Zollhofer, and Christian Theobalt. Stylerig: Rigging stylegan for 3d control over portrait images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6142–6151, 2020.

[34] Ayush Tewari, Michael Zollhöfer, Pablo Garrido, Florian Bernard, Hyeongwoo Kim, Patrick Pérez, and Christian Theobalt. Self-supervised multi-level face model learning for monocular reconstruction at over 250 hz. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[35] Ayush Tewari, Michael Zollhofer, Hyeongwoo Kim, Pablo Garrido, Florian Bernard, Patrick Perez, and Christian Theobalt. Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 1274–1283, 2017.

[36] Ruben Tolosana, Ruben Vera-Rodriguez, Julian Fierrez, Aythami Morales, and Javier Ortega-Garcia. Deepfakes and beyond: A survey of face manipulation and fake detection. *Information Fusion*, 64:131–148, 2020.

[37] Anh Tuan Tran, Tal Hassner, Iacopo Masi, Eran Paz, Yuval Nirkin, and Gérard Medioni. Extreme 3d face reconstruction: Seeing through occlusions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3935–3944, 2018.

[38] Luan Tran and Xiaoming Liu. Nonlinear 3d face morphable model. In *In Proceeding of IEEE Computer Vision and Pattern Recognition*, Salt Lake City, UT, June 2018.

[39] Xiang Wu, Ran He, Zhenan Sun, and Tieniu Tan. A light cnn for deep face representation with noisy labels. *IEEE Transactions on Information Forensics and Security*, 13(11):2884–2896, 2018.

[40] 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.

[41] Xiangyu Zhu, Zhen Lei, Xiaoming Liu, Hailin Shi, and Stan Z Li. Face alignment across large poses: A 3d solution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 146–155, 2016.

A. Additional Results

We present additional results obtained by our method. In Figure 7 we demonstrated several generated results which include glasses. The glasses are generated within the facial texture causing a somewhat unrealistic result. This can be mitigated by latent feature manipulation or by simply eliminating samples with glasses from the training data. Figure 8 demonstrates numerous reconstruction results obtained from our reconstruction pipeline. We observe highly detailed fully textured faces reconstructed from any input pose. In Figure 9 we demonstrate more generation results, further demonstrating our high-fidelity texture generation capability coupled with matching realistic geometry. results are best viewed zoomed-in.

Figure 7. **Generation Results that Include Glasses.** Due to the presence of subjects wearing glasses within the FFHQ dataset used for training, in some cases, our output texture might contain glasses; see Discussion and Future Work Section.

Figure 8. **Additional Qualitative Reconstruction Results.** We applied our reconstruction pipeline to numerous facial images. The 2D input images and the output textured models are presented side by side. The figure is best viewed when zoomed in.

Figure 9. **Additional Qualitative Generation Results.** We used our generation pipeline to generate various random textures and geometries. The figure is best viewed when zoomed in.