# Exp-GAN: 3D-Aware Facial Image Generation with Expression Control*

Yeonkyeong Lee[1], Taeho Choi[1], Hyunsung Go[2], Hyunjoon Lee[1], Sunghyun Cho[3], and Junho Kim[2]

[1]Kakao Brain [2]Kookmin University [3]POSTECH

**Abstract.** This paper introduces Exp-GAN, a 3D-aware facial image generator with explicit control of facial expressions. Unlike previous 3D-aware GANs, Exp-GAN supports fine-grained control over facial shapes and expressions disentangled from poses. To this ends, we propose a novel hybrid approach that adopts a 3D morphable model (3DMM) with neural textures for the facial region and a neural radiance field (NeRF) for non-facial regions with multi-view consistency. The 3DMM allows fine-grained control over facial expressions, whereas the NeRF contains volumetric features for the non-facial regions. The two features, generated separately, are combined seamlessly with our depth-based integration method that integrates the two complementary features through volume rendering. We also propose a training scheme that encourages generated images to reflect control over shapes and expressions faithfully. Experimental results show that the proposed approach successfully synthesizes realistic view-consistent face images with fine-grained controls. Code is available at https://github.com/kakaobrain/expgan.


**Fig. 1.** Exp-GAN generates realistic images of human faces, with explicit control of camera pose, facial expression, and facial shape. It is also capable of generating faces with different appearances keeping given facial expression and camera pose unchanged.

* This work was done when the first author was with Kookmin University. Corresponding author: Junho Kim (junho@kookmin.ac.kr)

## 1 Introduction

Recent years have seen a significant increase in photo-realism of synthetic images built on generative models such as generative adversarial networks (GANs) [11], variational autoencoders (VAEs) [17] and diffusion models [30]. Among them, state-of-the-art GAN models such as StyleGAN [15,16] have realized generation of extremely realistic face images of massive identities with scale-wise style control. To control over more sophisticated semantic attributes, much research has been done to explore semantically meaningful directions in the latent space [26,13,27,34] or to learn mappings for disentangled representations [31,7,18,10,1,28,2]. However, face shapes and expressions can be controlled in a limited way because they are manipulated through attribute editing in a latent space.

For more intuitive control over semantic attributes including facial shapes and expressions, several methods that adopt 3D morphable models (3DMM) of faces to the 2D GAN framework have been proposed [31,10,2]. In [10,2], a 3D face mesh is rendered to inject various information (RGB, normal, neural features) of face shapes and expressions to the generator. However, despite using a 3D face mesh model, their results show entanglement between facial expressions and other attributes such as camera poses and identities due to the lack of multi-view consistency of the 2D GAN framework.

3D-aware GANs have been proposed to synthesize high-fidelity face images with multi-view consistency [4,5,12,35,3,8]. In general, 3D-aware GANs learn to generate an implicit volume feature field that can be realized as images with volume rendering. Since implicit volume features already contain 3D information, 3D-aware GANs can be successfully trained to generate face images with multi-view consistency. However, to our best knowledge, control over facial shapes and expressions in 3D-aware GANs has not yet been considered.

This paper proposes Exp-GAN, a 3D-aware facial image generator that gives us explicit controls over facial shapes and expressions with multi-view consistency. Specifically, Exp-GAN learns to synthesize a variety of facial expressions disentangled from identities and camera poses, as shown in Fig. 1. To accomplish this, Exp-GAN adopts a hybrid approach that combines the 3D morphable model (3DMM) of faces and the 3D-aware GAN into a single framework of conditional GAN that can be trained with a collection of 2D facial images. The 3DMM allows us fine-grained and intuitive control over the facial shape and expressions, while the 3D-aware GAN enables multi-view consistent photo-realistic image synthesis.

Specifically, Exp-GAN synthesizes the facial and non-facial parts separately using a neural face generator and a neural volume generator, respectively. The neural face generator adopts a 3DMM with the neural texture to synthesize features of a realistic and multi-view consistent face that fully reflect user controls over facial expression and shape given by blendshape coefficients. The neural volume generator adopts the 3D-aware GAN approach to generate volumetric features, supporting diverse and realistic image synthesis with multi-view consistency. For the seamless integration of the two separately generated features,

we also introduce a feature integration method based on the volume rendering process of NeRF [22]. Finally, we propose a training scheme based on the regression of blendshape coefficients with discriminators for faithful image synthesis with respect to user control parameters.

We empirically show that Exp-GAN can generate various expressions, poses, and shapes of human faces. We also show that the proposed method improves the result quantitatively compared to previous works that provide expression controls with 2D StyleGANs. Our contributions can be summarized as follows:

- We propose Exp-GAN, the first 3D-aware facial image generator to achieve both multi-view consistency and fine-grained control over facial expressions.
- We propose geometrically explicit conditioning of a 3D-aware GAN with facial expression controls based on 3DMMs.
- Our hybrid approach combines the 3DMM and volumetric features for the synthesis of the facial and non-facial regions, and adopts a novel depth integration method for seamless integration of separately synthesized features.
- We also propose a novel training scheme leveraging discriminators with regression branches to train our network to faithfully reflect user controls.

## 2 Related Work

**Expression Controls in Generative Models**. Semantic editing in the latent space of GANs has been studied in [34,27,28,1,26,13], in which facial expression controls are handled through semantic attribute editing. SeFa [27] and GANSpace [13] discover semantically interpretable directions through latent space factorization. InterFaceGAN [26] finds linear directions in the latent space using binary-classified samples with respect to semantic attributes. StyleFlow [1] finds non-linear paths in the StyleGAN's latent space for manipulating semantic attributes using attribute-conditioned normalizing flow models. However, these approaches treat facial expression controls by means of semantic attribute editing over the pretrained StyleGAN's latent space, the diversity of expression controls is limited to simple expressions, such as smiles. To support fine-grained control over facial expressions, StyleRig [31] presents a facial attribute editing approach based on rig-like controls via 3DMMs. While StyleRig adopts the 3DMM, it still relies on the predefined StyleGAN's latent space, thus it suffers from a similar limitation to the aforementioned approaches, i.e., limited to simple expressions.

Several generative networks have been proposed that employ 3DMMs to synthesize facial images with complicated expressions [7,10,18,2]. DiscoFaceGAN [7] trains a StyleGAN-like image generator via an imitative-contrastive paradigm. GIF [10] and VariTex [2] leverage generated 3DMM face images to learn controllable face image generation. GIF [10] generates face images with expressions, with FLAME [19] conditioning signals and rendered images. VariTex [2] learns to synthesize pose-independent textures for 3DMM faces with expressions and additive features for non-facial parts to generate facial images with camera pose and expression controls. Since previous approaches with 3DMMs [7,10,18,2] rely on 2D generators such that expression information from 3DMMs is injected as

projected facial features in 2D image spaces, entanglement between camera poses and expressions still exists due to the limitation of 2D approaches.

**3D-aware GANs**. Recently, 3D-aware GANs have been proposed to disentangle camera poses from other attributes to achieve multi-view consistency [23,25,4,5,12,35,3,8]. They learn to map a random noise vector to an implicit feature field that can be consistently rendered from multiple viewpoints. SofGAN [5] decouples the latent space of portraits into a geometry and a texture space and uses the geometry space to generate a 3D geometry with a canonical pose. $\pi$-GAN [4] proposes a SIREN-based network [29] to learn a neural radiance field (NeRF)-based generator that can synthesize 3D-aware images. StyleNeRF [12] combines the NeRF and a style-based generator to improve rendering efficiency and 3D consistency for high-resolution image generation. CIPS-3D [35] learns a style-based NeRF generator with a deep 2D implicit neural representation network that efficiently generates high-resolution rendering results with partial gradient backpropagation. EG3D [3] proposes a tri-plane-based hybrid 3D representation to learn high-quality 3D geometry with fine details. GRAM [8] learns generative neural radiance fields for 3D-aware GAN by constraining point sampling and radiance field learning on 2D manifolds to generate high-quality images with fine details. Recent 3D-aware GANs successfully disentangle pose and identity to provide high-quality multi-view-consistent images. However, disentanglement of facial expression has not yet been considered in 3D-aware face image generation.

## 3 Framework

Fig. 2 shows an overview of our framework. Our framework consists of four parts: a neural face generator, a neural volume generator, an image synthesis module, and a discrimination module. For the synthesis part of our framework, StyleGAN2-based generators [16] are used, namely $G_{\text{tex}}$ for the neural face texture, $G_{\text{vol}}$ for the volume feature, and $G_{\text{img}}$ for the final image, respectively. Two StyleGAN2-based discriminators are used for the discrimination module: $D_{\text{img}}$ for the final output and $D_{\text{aux}}$ for the low-resolution auxiliary output.

Similar to previous generative NeRF models, we provide a camera pose $(\mathbf{R}, \mathbf{t})$ as an input to our framework to generate images from various viewpoints, where $\mathbf{R}$ is a rotation matrix, and $\mathbf{t}$ is a translation vector. We assume a fixed intrinsic camera matrix as a hyperparameter. For the explicit control of shapes and expressions of faces, we use a blendshape-based 3DMM. Specifically, we adopt DECA [9] that allows us to control the facial shape and expression using coefficient vectors $\boldsymbol{\alpha} \in \mathbb{R}^{100}$ and $\boldsymbol{\beta} \in \mathbb{R}^{50}$, respectively. To model the jaw motion, which is not supported by DECA, we introduce additional three coefficients to the expression coefficients, i.e., we use $\boldsymbol{\beta} \in \mathbb{R}^{50+3}$ as an expression coefficient vector in our framework. With the 3DMM, a face mesh that reflects user-provided coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ can be created (Fig. 2, top-left).

Mathematically, our synthesis framework can be expressed as:

$$I = G(\mathbf{z}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{R}, \mathbf{t}), \tag{1}$$

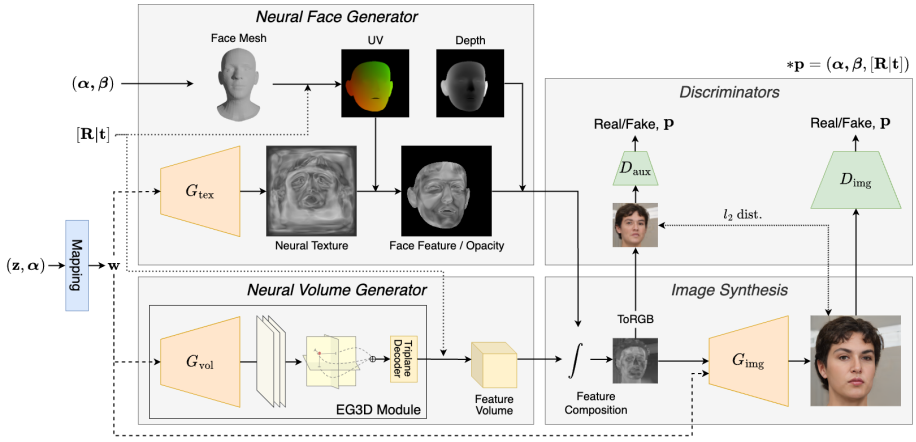


**Fig. 2.** Our framework. Facial feature map is generated in the *neural face generator* block, while the *neural volume generator* block synthesizes the feature volume representing non-facial regions. In the *image synthesis* block, the two features are composited by volume ray casting and upsampled to produce the output image. The synthesized output as well as a low-resolution auxiliary result are evaluated by the two discriminators in the *discriminators* block with adversarial and parameter regression losses.

where $I$ is a 2D output image, $G$ is our synthesis network, and $\mathbf{z} \in \mathbb{R}^{256}$ is a random latent vector. Our synthesis framework takes a form of a conditional GAN framework that produces a realistic 2D output image $I$ from a latent vector $\mathbf{z}$ conditioned by a camera pose $(\mathbf{R}, \mathbf{t})$, a facial shape vector $\boldsymbol{\alpha}$, and an expression vector $\boldsymbol{\beta}$. The latent code $\mathbf{z}$, sampled from a multivariate unit Gaussian distribution, enables the generation of diverse identities while conditioned on the other parameters. Note that, in our framework, both $\boldsymbol{\alpha}$ and $\mathbf{z}$ control the identities of generated face images; the blendshape coefficient vector $\boldsymbol{\alpha}$ provides fine control over the facial shape, while the latent code $\mathbf{z}$ controls the appearance like hair and skin.

Following the StyleGAN architecture [15,16], our framework adopts a mapping network that transforms $\mathbf{z}$ to an intermediate latent vector $\mathbf{w} \in \mathbb{R}^{512}$ instead of directly using $\mathbf{z}$. In addition, to constrain our synthesis process on the facial shape coefficient vector $\boldsymbol{\alpha}$, we feed $\boldsymbol{\alpha}$ to the mapping network by concatenating it with $\mathbf{z}$ as shown in Fig. 2.

### 3.1 Neural Face Generator

The goal of the neural face generator is to generate a 2D feature map representing the facial region that fully reflects the user control parameters and the latent vector $\mathbf{z}$. Inspired by neural texture approaches [32,2,20], we generate a neural texture of size $256 \times 256$ for the facial region from the intermediate latent vector $\mathbf{w}$. Each texel within the texture has a 32-dimensional feature vector representing the appearance of the facial region and an opacity value $a \in [0, 1]$. At the same

time, a face mesh is created and rendered from $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, $\mathbf{R}$ and $\mathbf{t}$, resulting in a texture UV map and a depth map of size $256 \times 256$. By sampling the neural texture with the UV map, we get a facial feature map and an opacity map of size $256 \times 256$. We then downsample the depth map, facial feature map, and opacity map to $64 \times 64$ using average pooling to suppress aliasing artifacts caused by the sampling process. We denote the downsampled depth map, facial feature map, and opacity map by $D$, $T$, and $A$, respectively.

**Disentanglement of Pose and Facial Attributes**. As done in [3], our texture decoder takes camera pose parameters $\mathbf{R}$ and $\mathbf{t}$ as inputs, although it synthesizes a pose-independent texture map. As noted in [3], most real-world datasets have biases that correlate camera poses and other attributes such as facial expressions, and improper handling of such biases leads to unwanted artifacts in generation results. We adopt the idea of generator pose conditioning proposed in [3] to address this problem. During training, the generator is conditioned by the camera pose so that it can learn the joint distribution of pose-independent and -dependent attributes inherent in the dataset. We refer the readers to [3] for more details about the generator pose conditioning.

### 3.2 Neural Volume Generator

The neural volume generator generates a 3D feature volume representing non-facial regions including the hair, clothes, and background. For this purpose, we employ EG3D [3] as a backbone 3D-aware generative model because of its generation performance and architectural simplicity. Nonetheless, our method can be incorporated with other 3D-aware generative models as long as they use volumetric feature fields [4,35,12,8].

Our neural volume generator takes an intermediate latent code $\mathbf{w}$ as input and feeds it to a StyleGAN2-based generator to produce a tri-plane representation [3], a light-weight representation for volumetric features, for the non-facial region. Then, we obtain a feature volume from the generated tri-plane representation. Specifically, given a camera pose $(\mathbf{R}, \mathbf{t})$, we shoot a bundle of rays from the camera to sample features from the tri-plane representation. In our implementation, we shoot $64 \times 64$ rays and sample 48 points per ray from the feature field. We aggregate features from the tri-planes and decode them using a small multi-layer perceptron for each point. Finally, we obtain a 3D feature volume $V$ of size $64 \times 64 \times 48$ where each voxel has a 32-dimensional feature vector and a scalar density value $\sigma$. We refer to [3] for more details about the tri-plane representation.

### 3.3 Image Synthesis with Feature Integration

The image synthesis module first integrates the facial feature map $T$ and the feature volume $V$ based on the depth map $D$ and performs volume rendering to obtain a 2D feature map $F \in \mathbb{R}^{64 \times 64 \times 32}$. To this end, we adopt the volume ray casting algorithm with per-ray feature composition [22]. Specifically, for each spatial location in $V$, we have 48 features $\{\mathbf{f}_0, \ldots, \mathbf{f}_{47}\}$, density values $\{\sigma_0, \ldots, \sigma_{47}\}$,


(a) (b) (c)

**Fig. 3.** Depth-based feature composition process and feature visualizations. (a) Depth-based feature composition process. Green dots represent features from $V$, as $\{\mathbf{f}_0, \mathbf{f}_1, \ldots\}$, and yellow dot represents a facial feature $\mathbf{f}^f$. (b) Visualization of a facial feature map. (c) Feature volume rendered without facial features. Non-facial regions including teeth and inner mouth are synthesized, complementing the facial feature map.

and their corresponding depths $\{d_0, d_1, \ldots, d_{47}\}$ where $d_0 \leq d_1 \leq \ldots \leq d_{47}$. For volume ray casting, we compute a set of opacity values $\{a_0, \ldots, a_{46}\}$ where $a_i = 1 - \exp(-\sigma_i(d_{i+1} - d_i))$. Then, we insert the feature $\mathbf{f}^f$ and opacity $a^f$ from $T$ and $A$ into the sets of features and opacity values according to the depth $d^f$ from $D$, and obtain:

$$\mathcal{F} = \left\{\mathbf{f}'_0 = \mathbf{f}_0, \ldots, \mathbf{f}'_i = \mathbf{f}_i, \mathbf{f}'_{i+1} = \mathbf{f}^f, \mathbf{f}'_{i+2} = \mathbf{f}_{i+1}, \ldots \mathbf{f}'_N = \mathbf{f}_{N-1}\right\}, \text{ and} \quad (2)$$

$$\mathcal{A} = \left\{a'_0 = a_0, \ldots, a'_i = a_i, a'_{i+1} = a_t, a'_{i+2} = a_{i+1}, \ldots a'_N = a_{N-1}\right\}, \quad (3)$$

where $d_i \leq d^f \leq d_{i+1}$. We then perform volume ray casting as:

$$\mathbf{f} = \sum_{i=0}^{N} T_i(1 - a'_i)\mathbf{f}'_i, \text{ where } T_i = \prod_{j=0}^{i-1} a'_j, \quad (4)$$

where $N = 48$ and $\mathbf{f}$ is an integrated feature vector. Collecting $\mathbf{f}$, we construct a 2D feature map $F$. Fig. 3 illustrates the composition process.

The feature map $F$ is then fed to a StyleGAN2-based superresolution network $G_{\text{img}}$ to produce a high-resolution final RGB image. $G_{\text{img}}$ also takes the intermediate latent vector $\mathbf{w}$ to synthesize realistic-looking high-resolution details for the final output image.

### 3.4 Training

We train our entire network in an end-to-end fashion, as our framework is composed of differentiable modules except for the morphing and rendering steps for the face mesh, which do not have learnable parameters. To synthesize novel images, we use adversarial learning using only 2D real images. Specifically, we attach a discriminator $D_{\text{img}}$ to the output of the superresolution network to predict whether the final output looks real or fake.

To encourage our generator to synthesize images with correct camera poses and facial expressions, $D_{\text{img}}$ has an additional branch that estimates the pose

and expression coefficients of an input image. Using $D_{\text{img}}$, we train our generator by minimizing $\mathcal{L}_{\text{img}}^{\text{gen}}$, which is defined as:

$$\mathcal{L}_{\text{img}}^{\text{gen}} = \mathbb{E}_{\mathbf{z},\mathbf{p}}\left[f(-D_{\text{img},s}(G(\mathbf{z},\mathbf{p}))) + \lambda_{\mathbf{p}}\|D_{\text{img},\mathbf{p}}(G(\mathbf{z},\mathbf{p})) - \mathbf{p}\|^2\right] \tag{5}$$

where $f(\cdot)$ represents the softplus function [15,16] and $\mathbf{p} = (\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{R}, \mathbf{t})$. $\mathbb{E}_{\mathbf{z},\mathbf{p}}$ is the expectation under the joint distribution of $\mathbf{z}$ and $\mathbf{p}$. $D_{\text{img},s}$ and $D_{\text{img},\mathbf{p}}$ are the outputs of the prediction score branch and the pose and expression parameter branch, respectively. $\lambda_{\mathbf{p}}$ is a weight to balance the loss terms.

The discriminator $D_{\text{img}}$ is trained by $\mathcal{L}_{\text{img}}^{\text{disc}}$, which is defined as:

$$\begin{aligned}\mathcal{L}_{\text{img}}^{\text{disc}} = {} & \mathbb{E}_{\mathbf{z},\mathbf{p}}\left[f(D_{\text{img},s}(G(\mathbf{z},\mathbf{p})))\right] \\ & + \mathbb{E}_I\left[f(-D_{\text{img},s}(I)) + \lambda_{\mathbf{p}}\|D_{\text{img},\mathbf{p}}(I) - \mathbf{p}_{gt}\|^2 + \lambda_{r1}\|\nabla D_{\text{img},s}(I)\|^2\right]\end{aligned} \tag{6}$$

where $I$ is a real image sample, $\mathbb{E}_I$ is the expectation under the distribution of real images, and $\mathbf{p}_{gt}$ is the ground-truth (GT) label for the camera pose and expression parameters of $I$. We obtain $\mathbf{p}_{gt}$ by applying a pretrained DECA encoder [9] to the real image samples before training. The last term in $\mathbb{E}_I$ is an R1 regularization term [21] to stabilize GAN training.

Training our generator with only $D_{\text{img}}$ may converge to a low-quality local minimum as the superresolution network can be trained to synthesize images with an average pose and facial expression regardless of its input $F$. To resolve this, we let our network to produce an auxiliary low-resolution RGB image $I_{\text{aux}}$ directly from $F$, and introduce another discriminator $D_{\text{aux}}$, which predicts whether $F$ looks realistic. For this purpose, we assume the first three channels of $F$ as the low-resolution RGB output; a similar technique is also used in DNR [32]. We then train the generator minimizing a loss function $\mathcal{L}_{\text{aux}}^{\text{gen}}$, defined as:

$$\mathcal{L}_{\text{aux}}^{\text{gen}} = \mathbb{E}_{\mathbf{z},\mathbf{p}}\left[-D_{\text{aux},s}(F_{1,2,3}(\mathbf{z},\mathbf{p})) + \lambda_{\mathbf{p}}\|D_{\text{aux},\mathbf{p}}(F_{1,2,3}(\mathbf{z},\mathbf{p})) - \mathbf{p}\|^2\right] \tag{7}$$

where $F_{1,2,3}(\mathbf{z},\mathbf{p})$ represents the first three channels of $F$ as a function of $\mathbf{z}$ and $\mathbf{p}$. The loss $\mathcal{L}_{\text{aux}}^{\text{disc}}$ for the discriminator $D_{\text{aux}}$ is also defined similarly to $\mathcal{L}_{\text{img}}^{\text{disc}}$. Specifically, $\mathcal{L}_{\text{aux}}^{\text{disc}}$ is defined as:

$$\begin{aligned}\mathcal{L}_{\text{aux}}^{\text{disc}} = {} & \mathbb{E}_{\mathbf{z},\mathbf{p}}\left[D_{\text{aux},s}(F_{1,2,3}(\mathbf{z},\mathbf{p}))\right] \\ & + \mathbb{E}_I\left[-D_{\text{aux},s}(I\downarrow) + \lambda_{\mathbf{p}}\|D_{\text{aux},\mathbf{p}}(I\downarrow) - \mathbf{p}_{gt}\|^2 + \lambda_{r1}\|\nabla D_{\text{aux},s}(I\downarrow)\|^2\right]\end{aligned} \tag{8}$$

where $I\downarrow$ is a downsampled version of $I$. For both $D_{\text{img}}$ and $D_{\text{aux}}$, we employ the network architecture of the discriminator of StyleGAN2 [16].

In addition, as similarly done in [12,3], we employ an MSE loss to make $I$ and $I_{\text{aux}}$ similar to each other, defined as:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{N_{I_{\text{aux}}}}\sum_{i}^{N_{I_{\text{aux}}}}\left(I_{\text{aux}}(j) - I\downarrow(j)\right)^2, \tag{9}$$

where $j$ is a pixel index and $N_{I_{\text{aux}}}$ is the number of pixels in $I_{\text{aux}}$. Lastly, based on the observation that the facial region is opaque, we introduce an opacity loss

$$\mathcal{L}_{\text{opacity}} = \frac{1}{N_A}\sum_{j}^{N_A} -\log\left(1 - A(j)\right), \tag{10}$$


**Fig. 4.** Qualitative results with varying expression coefficients $\boldsymbol{\beta}$ and camera poses. In each method, an identity is fixed with a random seed. (top row) Three reference images from which $\boldsymbol{\beta}$ are extracted and corresponding face meshes are also rendered. In (a-c), we generate samples with respect to $\boldsymbol{\beta}$, except for $\pi$-GAN which cannot control facial expressions. For (d) and (e), we use $\boldsymbol{\beta}$ from (b) and (c), respectively, with different camera poses.

where $N_A$ is the number of pixels in $A$. Our final loss for the generator is then

$$\mathcal{L}^{\text{gen}} = \mathcal{L}^{\text{gen}}_{\text{img}} + \mathcal{L}^{\text{gen}}_{\text{aux}} + \lambda_{\text{MSE}}\mathcal{L}_{\text{MSE}} + \lambda_{\text{opacity}}\mathcal{L}_{\text{opacity}} \tag{11}$$

where $\lambda_{\text{MSE}}$ and $\lambda_{\text{opacity}}$ are weights for $\mathcal{L}_{\text{MSE}}$ and $\mathcal{L}_{\text{opacity}}$, respectively.

## 4 Experiments

We train and evaluate our network with the FFHQ dataset [15]. In our training stage, we randomly sample parameters from the GT labels $\mathbf{g}_{gt}$ for each training sample to expand the sampling space, i.e., for each training sample, we randomly sample image indices $i$, $j$, and $k$ and use the GT labels $\boldsymbol{\alpha}_i$, $\boldsymbol{\beta}_j$, and ($\mathbf{R}_k$,$\mathbf{t}_k$) of the $i$-, $j$-, and $k$-th training images in the FFHQ dataset, respectively. The GT labels $\mathbf{g}_{gt}$ are obtained by applying a pretrained DECA [9] encoder to the FFHQ dataset. All of our qualitative results are generated with the truncation trick [15], with truncation $\psi = 0.5$. For quantitative comparisons truncation trick is not used except for the multi-view consistency score. Qualitative comparisons are conducted with images of different identities, due to the nature of GAN training. Further training details and more results, including additional comparisons and ablations, are presented in the supplementary material.

### 4.1 Comparison

We compare qualitative and quantitative results between our Exp-GAN and several baseline methods. We first compare our method against 3DMM-based 2D generative models, such as DiscoFaceGAN [7], GIF [10], and VariTex [2], to show how well our Exp-GAN reflects facial expression conditions in the final images with all the attributes disentangled. We also compare our method against $\pi$-GAN [4] as a baseline 3D-aware GAN to show how accurately Exp-GAN disentangles all the attributes. For DiscoFaceGAN [7] and GIF [10], we use pretrained models provided the authors. For VariTex [2], we change its 3DMM model parameters to match ours and train the model from scratch using the FFHQ dataset and the authors' implementation. For $\pi$-GAN [4] we train the model from scratch on the FFHQ dataset using the authors' implementation.

**Qualitative Results**. Fig. 4 shows a qualitative comparison where we fix the identity $(\mathbf{z}, \boldsymbol{\alpha})$ but vary the camera pose $(\mathbf{R}, \mathbf{t})$ and expression $\boldsymbol{\beta}$. As shown in the figure, $\pi$-GAN [4] generates view-consistent facial images but has no expression control. On the other hand, all the 2D generative models incorporating 3DMM provide control over the pose and expression, but they do not guarantee view consistency. Also, despite using the 3DMM, DiscoFaceGAN [7] shows only slight facial expression changes. Compared to all the other methods, our method generates high-fidelity 3D-aware face images that faithfully reflect input facial expressions while keeping all the other attributes unchanged.

**Quantitative Results**. Table 1 provides quantitative comparisons in Frechét Inception Distance (FID) [14], blendshape metric (BS), multi-view consistency score (MV), and identity consistency score (ID). As DiscoFaceGAN [7] uses the Basel Face Model (BFM) [24] for 3DMM differently from others, for a fair comparison, we estimate BFM blendshape coefficients from all the images in FFHQ and use them in place of DECA blendshape coefficients to generate images with DiscoFaceGAN [7]. We exclude VariTex [2] from our quantitative comparisons due to the domain difference caused by background masking.

**Table 1.** Quantitative comparisons. Bold is the best result, and underscore is the second-best result. Refer the manuscript for details of the comparison protocols specific to algorithms and metrics.

| | FID ↓ | BS ↓ | MV ↑ | ID ↑ |
|---|---|---|---|---|
| $\pi$-GAN [4] ($128^2$) | 16.91 | — | **24.58** | — |
| DiscoFaceGAN [7] | <u>15.57</u> | 0.147 | 23.28 | **0.699** |
| GIF [10] | 28.0 | <u>0.1</u> | 16.56 | 0.435 |
| Ours | **7.44** | **0.05** | <u>23.84</u> | <u>0.622</u> |

We first evaluate image quality with FID between 50K real images from FFHQ and 50K generated images. For DiscoFaceGAN and GIF, we generate 50K facial images with random latent vectors and GT parameters, similarly to ours. For $\pi$-GAN, 50K images are generated with random latent vectors and sampled GT camera poses. As shown in the FID column in Table 1, our Exp-GAN generates higher-quality images than the other methods. The FID score of Exp-GAN is comparable to 4.8 of EG3D, which is reported in [3], while Exp-GAN also provides explicitly control over shape and expression unlike EG3D.

Next, we evaluate how well input conditional facial expressions are reflected in generated images with the BS metric. For this, we generate 50K images and re-estimate blendshape coefficients from them. The BS metric is measured by the mean squared distance between the input blendshape coefficients and re-estimated ones. As shown in the BS column in Table 1, our Exp-GAN achieves a better result than previous 2D-based generative models, validating that our method can faithfully reflect the input facial expression.

To evaluate multi-view consistency of our results, we measure MV scores as proposed in StyleNeRF [12]. From the given parameters $(\mathbf{z}, \boldsymbol{\alpha}, \boldsymbol{\beta})$, we generate 9 images by varying camera poses from left to right, changing the yaw value in $[-0.5, 0.5]$ radian. Among 9 generated images, we use 5 of them as reference images and reconstruct the remaining ones with IBRNet [33]. Then we compute the MV score in terms of PSNR for the reconstructed images with the generated images as references. For evaluation, we generate 1K test cases, each with 9 images, and measure the average MV score of the test cases. Our Exp-GAN shows comparable results with that of $\pi$-GAN. DiscoFaceGAN, interestingly, shows good multi-view consistency even though it is a 2D-based approach, but it achieves relatively low facial expression accuracy as discussed earlier.

We evaluate ID score that measures how well the facial identity is preserved in various camera poses and expressions. We evaluate ID score by computing the average ArcFace [6] cosine similarity from 50K pairs of generated images. For each pair, we generate images by fixing $\mathbf{z}$ and $\boldsymbol{\alpha}$ and changing $\boldsymbol{\beta}$, $\mathbf{R}$ and $\mathbf{t}$. ID score is evaluated only on the models that allow explicit control of expression. The 'ID' column of Table 1 shows that DiscoFaceGAN performs the best, followed by ours. However, DiscoFaceGAN generates images where facial expressions are not changed as expected.

**Fig. 5.** Limitation of a naïve baseline. All the results are generated using a single baseline model with the same code for $\mathbf{z}$ but different values for $\boldsymbol{\beta}$. Despite the fixed $\mathbf{z}$, the images show different identities due to the entanglement between $\mathbf{z}$ and $\boldsymbol{\beta}$.

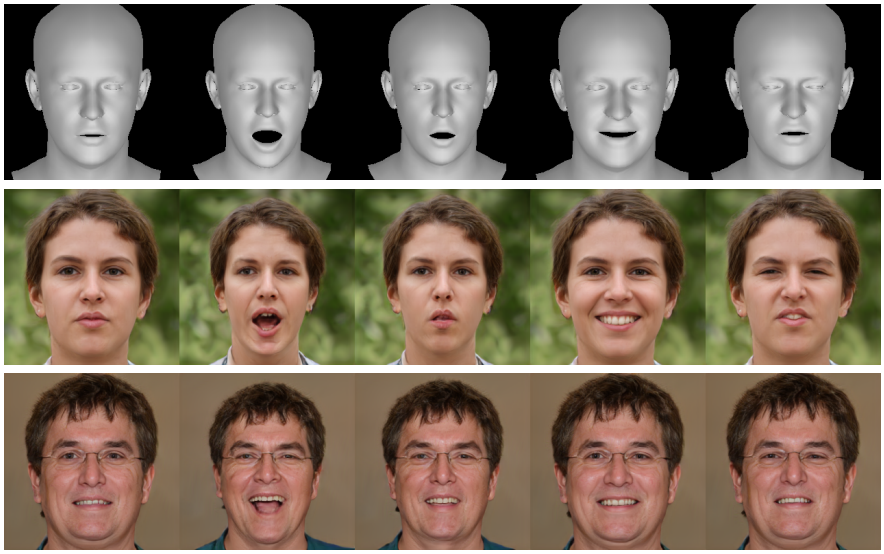

**Fig. 6.** Effect of the expression coefficient regression. (top) input expressions, as neutral, open mouth, half-open mouth, smile, and frown, respectively. (middle) our result, (bottom) without regression loss of expression coefficients.

### 4.2 Ablation Study

**Comparison with a Naïve Baseline**. As a naïve baseline, we train an EG3D network with a simple modification to control facial expressions. The mapping network is modified to get not only latent vectors but also expression coefficients, as the form of a concatenated vector $[\mathbf{z}, \boldsymbol{\beta}]$. Here, the 3DMM-related components are ablated, i.e., the neural face generator and the feature composition in Fig. 2. As shown in Fig. 5, changing facial expressions affects identities, evidencing entanglement between the blendshape coefficients and latent vectors.

**Feature Integration Scheme**. Similar to [10,2], facial and non-facial parts may be integrated with feature concatenation and then fed to the image synthesis module by ablating the depth composition in Sec. 3.3. We train with feature concatenation in place of our composition method for ablation and evaluate with the FID metric. The FID score is 13.19, which is worse than that of ours, 7.44. Additional qualitative comparisons are provided in the supplementary material.

**Table 2.** Ablation study of the loss terms.

| | FID ↓ | BS ↓ | MV ↑ | ID ↑ |
|---|---|---|---|---|
| **Ours (full)** | 7.44 | 0.05 | 23.84 | 0.622 |
| w/o blendshape coeff. reg. | 8.90 | 0.10 | 25.93 | 0.723 |
| w/o $\mathcal{L}_{\text{MSE}}$ | 12.08 | 0.05 | 23.30 | 0.628 |
| w/o $\mathcal{L}_{\text{opacity}}$ | 10.53 | 0.05 | 23.96 | 0.617 |

**Impact of Loss Terms**. We conduct an ablation study to study the impact of each loss term. The quantitative results are reported in Table 2. Fig. 6 shows the impact of the expression coefficient regression loss in our discriminator. Here, we ablate the blendshape coefficient $\boldsymbol{\beta}$ in $\mathbf{p}$ estimated by the discriminators $D_{\text{aux}}$ and $D_{\text{img}}$ in Fig. 2. Thanks to the face mesh, which guides the generation process, it is still possible for the network to reflect expressions without the regression loss to some extent. Still, its expressiveness is limited compared to that of our full model. Next, We ablate each term from Eq. (11) to evaluate its impact on the generator. Without $\mathcal{L}_{\text{aux}}^{\text{gen}}$, we cannot generate plausible images. Without $\mathcal{L}_{\text{MSE}}$ or $\mathcal{L}_{\text{opacity}}$, FID scores are far inferior to that of our full model. As shown in Table 2, our final model can achieve the best performance, especially in both FID and BS scores.

### 4.3 Additional Results

Fig. 7 shows the effect of changing shape coefficients $\boldsymbol{\alpha}$. Although $\boldsymbol{\alpha}$ is entangled with the latent vector $\mathbf{z}$ in our framework, it is shown that changing $\boldsymbol{\alpha}$ results in natural-looking results with similar appearances. Fig. 8 shows various facial expressions while the camera pose changes. Lastly, as shown in Fig. 9, our Exp-GAN successfully synthesizes asymmetrical facial expressions that are rare in the FFHQ dataset [16]. See the supplementary material for more results, including examples of GAN inversion and facial reenactment.

## 5 Conclusion

We presented Exp-GAN, a novel 3D-aware GAN that can explicitly control camera poses, facial shapes and expressions. Leveraging the advantages of 3DMM and NeRF, our Exp-GAN generates features for facial and non-facial parts separately with appropriate neural approaches and seamlessly combines them to synthesize high-fidelity images via neural rendering. We showed that the depth-based feature integration in our generator and blendshape coefficient regressions in our discriminator play essential roles in the training of Exp-GAN for synthesizing images that faithfully reflect input shape and expression parameters.

Although Exp-GAN successfully disentangled several attributes as a 3D-aware GAN, it still lacks control over gaze and placement of accessories (e.g., glasses, earrings, etc.). Furthermore, Exp-GAN shows limited rendering qualities

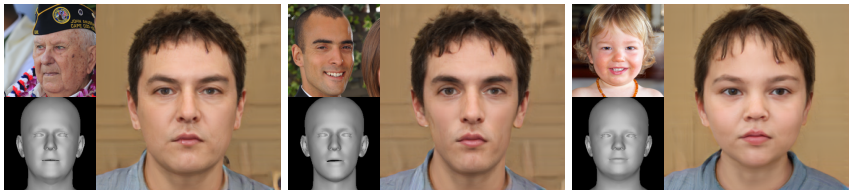

**Fig. 7.** Example of changing the face shape coefficients. (left) reference images from which $\boldsymbol{\alpha}$ is extracted; (right) our synthesis results. Fixing the latent code $\mathbf{z}$ and changing only $\boldsymbol{\alpha}$, we can obtain images with similar appearances but different face shapes.

**Fig. 8.** While the camera pose changes, facial expressions of each example are controlled as neutral, smile, open mouth, half-open mouth, and frown, respectively. See uncurated results in the supplementary material.

**Fig. 9.** Asymmetrical facial expressions generated by our method. Our method can generate asymmetrical facial expressions that are rare in the FFHQ dataset [16] thanks to the explicit modeling of facial expressions using a 3DMM model.

for the inside of the mouth mainly due to the lack of examples containing such a region in the dataset. We plan to address these issues for future work.

**Acknowledgements**. This was was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (2022R1F 1A1074628, 2022R1A5A7000765) and Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2020-0-01826, Problem-Based Learning Program for Researchers to Proactively Solve Practical AI Problems (Kookmin University) and No.2019-0-01906, Artificial Intelligence Graduate School Program (POSTECH)).

## References


1. Abdal, R., Zhu, P., Mitra, N.J., Wonka, P.: StyleFlow: Attribute-conditioned Exploration of StyleGAN-Generated Images using Conditional Continuous Normalizing Flows. ACM Trans. Graphics (Proc. SIGGRAPH 2021) **40**(3) (2021)
2. Bühler, M.C., Meka, A., Li, G., Beeler, T., Hilliges, O.: VariTex: Variational Neural Face Textures. In: Proc. ICCV. pp. 13890–13899 (2021)
3. Chan, E.R., Lin, C.Z., Chan, M.A., Nagano, K., Pan, B., Mello, S.D., Gallo, O., Guibas, L., Tremblay, J., Khamis, S., Karras, T., Wetzstein, G.: Efficient Geometry-aware 3D Generative Adversarial Networks. In: Proc. CVPR. pp. 16123–16133 (2022)
4. Chan, E.R., Monteiro, M., Kellnhofer, P., Wu, J., Wetzstein, G.: pi-GAN: Periodic Implicit Generative Adversarial Networks for 3D-Aware Image Synthesis. In: Proc. CVPR. pp. 5799–5809 (2021)
5. Chen, A., Liu, R., Xie, L., Chen, Z., Su, H., Yu, J.: SofGAN: A Portrait Image Generator with Dynamic Styling. ACM Trans. Graphics **42**(1) (2022)
6. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: ArcFace: Additive Angular Margin Loss for Deep Face Recognition. In: Proc. CVPR. pp. 4690–4699 (2019)
7. Deng, Y., Yang, J., Chen, D., Wen, F., Tong, X.: Disentangled and Controllable Face Image Generation via 3D Imitative-Contrastive Learning. In: Proc. CVPR. pp. 5154–5163 (2020)
8. Deng, Y., Yang, J., Xiang, J., Tong, X.: GRAM: Generative Radiance Manifolds for 3D-Aware Image Generation. In: Proc. CVPR. pp. 10673–10683 (2022)
9. Feng, Y., Feng, H., Black, M.J., Bolkart, T.: Learning an Animatable Detailed 3D Face Model from In-The-Wild Images. ACM Trans. Graphics (Proc. SIGGRAPH 2021) **40**(8), Article No. 88 (2021)
10. Ghosh, P., Gupta, P.S., Uziel, R., Ranjan, A., Black, M., Bolkart, T.: GIF: Generative Interpretable Faces. In: Proc. 3DV. pp. 868–878 (2020)
11. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative Adversarial Nets. In: Proc. NIPS. pp. 2672–2680 (2014)
12. Gu, J., Liu, L., Wang, P., Theobalt, C.: StyleNeRF: A Style-based 3D-Aware Generator for High-resolution Image Synthesis. In: Proc. ICLR (2022)
13. Härkönen, E., Hertzmann, A., Lehtinen, J., Paris, S.: GANSpace: Discovering Interpretable GAN Controls. In: Proc. NeurIPS (2020)
14. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In: Proc. NIPS. pp. 6629–6640 (2017)
15. Karras, T., Laine, S., Aila, T.: A Style-Based Generator Architecture for Generative Adversarial Networks. In: Proc. CVPR. pp. 4401–4410 (2019)
16. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and Improving the Image Quality of StyleGAN. In: Proc. CVPR. pp. 8110–8119 (2020)
17. Kingma, D.P., Welling, M.: Auto-Encoding Variational Bayes. In: Proc. ICLR (2014)
18. Kowalski, M., Garbin, S.J., Estellers, V., Baltrušaitis, T., Johnson, M., Shotton, J.: CONFIG: Controllable Neural Face Image Generation. In: Proc. ECCV. pp. 299–315 (2020)
19. Li, T., Bolkart, T., Black, M.J., Li, H., Romero, J.: Learning a model of facial shape and expression from 4D scans. ACM Trans. Graphics (Proc. SIGGRAPH Asia 2017) **36**(4), Article No. 194 (2017)

20. Ma, S., Simon, T., Saragih, J., Wang, D., Li, Y., la Torre, F.D., Sheikh, Y.: Pixel Codec Avatars. In: Proc. CVPR. pp. 64–73 (2021)
21. Mescheder, L., Geiger, A., Nowozin, S.: Which Training Methods for GANs do actually Converge? arXiv preprint arXiv:1801.04406 (2018)
22. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In: Proc. ECCV. pp. 405–421 (2020)
23. Nguyen-Phuoc, T., Li, C., Theis, L., Richardt, C., Yang, Y.L.: HoloGAN: Unsupervised Learning of 3D Representations From Natural Images. In: Proc. ICCV. pp. 7588–7597 (2019)
24. Paysan, P., Knothe, R., Amberg, B., Romdhani, S., Vetter, T.: A 3D Face Model for Pose and Illumination Invariant Face Recognition. In: IEEE International Conference on Advanced Video and Signal Based Surveillance. pp. 296–301 (2009)
25. Schwarz, K., Liao, Y., Niemeyer, M., Geiger, A.: GRAF: Generative Radiance Fields for 3D-Aware Image Synthesis. In: Proc. NeurIPS (2020)
26. Shen, Y., Gu, J., Tang, X., Zhou, B.: Interpreting the Latent Space of GANs for Semantic Face Editing. In: Proc. CVPR. pp. 9243–9252 (2020)
27. Shen, Y., Zhou, B.: Closed-Form Factorization of Latent Semantics in GANs. In: Proc. CVPR. pp. 1532–1540 (2021)
28. Shoshan, A., Bhonker, N., Kviatkovsky, I., Medioni, G.: GAN-Control: Explicitly Controllable GANs. In: Proc. ICCV. pp. 14083–14093 (2021)
29. Sitzmann, V., Martel, J.N.P., Bergman, A., Lindell, D.B., Wetzstein, G.: Implicit Neural Representations with Periodic Activation Functions. In: Proc. NeurIPS (2020)
30. Sohl-Dickstein, J., Weiss, E.A., Maheswaranathan, N., Ganguli, S.: Deep Unsupervised Learning using Nonequilibrium Thermodynamics. In: Proc. ICML (2015)
31. Tewari, A., Elgharib, M., Bharaj, G., Bernard, F., Seidel, H.P., Pérez, P., Zöllhofer, M., Theobalt, C.: StyleRig: Rigging StyleGAN for 3D Control over Portrait Images. In: Proc. CVPR. pp. 6142–6151 (2020)
32. Thies, J., Zollhöfer, M., Nießner, M.: Deferred Neural Rendering: Image Synthesis using Neural Textures. ACM Trans. Graphics (Proc. SIGGRAPH 2019) **38**(4) (2019)
33. Wang, Q., Wang, Z., Genova, K., Srinivasan, P., Zhou, H., Barron, J.T., Martin-Brualla, R., Snavely, N., Funkhouser, T.: IBRNet: Learning Multi-View Image-Based Rendering. In: Proc. CVPR. pp. 4690–4699 (2021)
34. Wu, Z., Lischinski, D., Shechtman, E.: StyleSpace Analysis: Disentangled Controls for StyleGAN Image Generation. In: Proc. CVPR. pp. 12863–12872 (2021)
35. Zhou, P., Xie, L., Ni, B., Tian, Q.: CIPS-3D: A 3D-Aware Generator of GANs Based on Conditionally-Independent Pixel Synthesis. arXiv preprint arXiv:2110.09788 (2021)