StyleGene: Crossover and Mutation of Region-level Facial Genes for Kinship Face Synthesis

Hao Li[1], Xianxu Hou[2,5], Zepeng Huang[1], Linlin Shen[1,2,3,4]*
[1]Computer Vision Institute, College of Computer Science and Software Engineering, Shenzhen University
[2]National Engineering Laboratory for Big Data System Computing Technology, Shenzhen University
[3]Shenzhen Institute of Artificial Intelligence and Robotics for Society
[4]Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen University
[5]School of AI and Advanced Computing, Xi'an Jiaotong-Liverpool University
{haoli2021, huangzepeng2021}@email.szu.edu.cn, hxianxu@gmail.com, llshen@szu.edu.cn

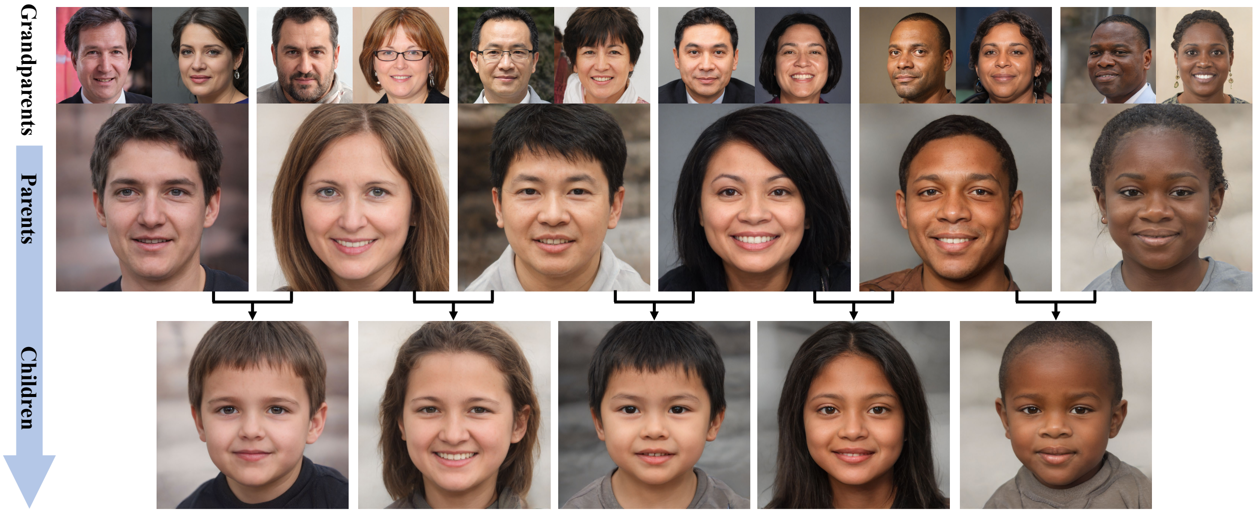


Figure 1. Our StyleGene method synthesizes kinship faces with resemblance to parents, exhibiting diversity and reasonable variations. The first row is the input grandparents, and the second and third rows are their descendants generated by our method.

Abstract

High-fidelity kinship face synthesis has many potential applications, such as kinship verification, missing child identification, and social media analysis. However, it is challenging to synthesize high-quality descendant faces with genetic relations due to the lack of large-scale, high-quality annotated kinship data. This paper proposes RFG (Region-level Facial Gene) extraction framework to address this issue. We propose to use IGE (Image-based Gene Encoder), LGE (Latent-based Gene Encoder) and Gene Decoder to learn the RFGs of a given face image, and the relationships between RFGs and the latent space of StyleGAN2. As cycle-like losses are designed to measure the $\mathcal{L}_2$ distances between the output of Gene Decoder and image encoder, and that between the output of LGE and IGE, only face images are required to train our framework, i.e. no paired kinship face data is required. Based upon the proposed RFGs, a crossover and mutation module is further designed to inherit the facial parts of parents. A Gene Pool has also been used to introduce the variations into the mutation of RFGs. The diversity of the faces of descendants can thus be significantly increased. Qualitative, quantitative, and subjective experiments on FIW, TSKinFace, and FF-Databases clearly show that the quality and diversity of kinship faces generated by our approach are much better than the existing state-of-the-art methods.

1. Introduction

Humans can identify kinship through photographs based on the resemblance between parents and children. Many works have investigated this intrinsic relation in the fields

*Corresponding Author

of kinship verification [9, 32, 42] and genetics [4, 5, 8, 19]. With the popularity of face synthesis and editing technology in recent years, high-fidelity kinship face synthesis has also attracted much attention. This task, aiming to synthesize the faces of descendants based on the appearance of the parents, has many potential applications, such as finding long-lost children, crime investigations, kinship verification, and multimedia social applications.

In recent years, many efforts have been made to make use of generative models [7, 12, 15, 16, 27, 29, 38, 43, 45, 48] for kinship face synthesis. These works can be categorized into two paradigms: one-stage and two-stage. The one-stage paradigm [12, 29, 38, 45] treats this problem as an image-to-image translation task and trains a one-to-one kinship face generator with paired data. However, these approaches can only produce low-resolution images and the resultant images can be blurry and lack diversity. Further, it would be quite difficult to obtain annotated kinship data. By contrast, the two-stage paradigm [7, 15, 16, 27, 43, 48] first extracts the genetic representation and assembles them into children's representation based on the parents' faces. To obtain genetic representation, existing methods try to learn the inheritance and variation of facial appearances by training deep neural networks [14, 27, 48] or via a knowledge rule [7]. However, the learned genetic representation is prone to overfitting due to the lack of high-quality kinship annotated training data, resulting in a lack of diversity in the generated children. In addition, these methods cannot provide fine-grained attributes representation, and thus the generated facial attributes lack interpretability.

In this paper, the facial genetic process is abstracted as the exchange and mutation of the parents' facial parts. We propose an Image-based Gene Encoder (IGE) to construct an independent representation for each facial part, called a Region-level Facial Gene (RFG), which is used to control the synthesis of facial regions. We further simulate the crossover and mutation process to assemble the RFGs of descendants by using those of the parents, and our proposed Gene Pool used in the mutation process can significantly increase the diversity of the generated descendants. We use the pre-trained StyleGAN2 [24] as the generator to synthesize high-fidelity faces. To achieve this, we use a Gene Decoder to map RFGs to the $\mathcal{W}^+$ space of StyleGAN2. Since IGE requires a facial parsing mask to generate the RFG, we additionally train a Latent-based Gene Encoder (LGE) to directly map the latent code of StyleGAN2 to RFGs. Thus, facial parsing mask is not required for the RFG extraction in the inference stage. The main contributions of this paper are summarized as follows:

- We propose StyleGene to synthesize high-fidelity kinship faces with controllable facial genetic regions, via modeling the facial genetic relations based on the proposed region-level facial genes.
- A novel genetic strategy is further introduced by simulating the crossover and mutation process to generate the RFGs of descendants. We introduce a Gene Pool into the mutation process to significantly increase the diversity of the kinship face.
- We validate the effectiveness of our approach on several benchmarks, demonstrating the superiority of our StyleGene framework over other state-of-the-art methods, in terms of the quality and diversity of the generated kinship faces.

2. Related Work

2.1. Manipulation in Latent Space of StyleGAN

Generative Adversarial Networks (GANs) [17] have been widely used in face generation [21, 33, 47] and editing [3, 13, 50]. In particular, the StyleGAN [22–24] has attracted much attention due to its ability to synthesize high-fidelity images. The StyleGAN generator first maps a latent code $\boldsymbol{z} \in \mathcal{Z}$ drawn from a normal distribution to an intermediate latent code $\boldsymbol{w} \in \mathcal{W}$ by a mapping function to control image generation. Previous studies [20, 23, 36] have demonstrated that $\mathcal{W}$ space has learned facial attribute semantics and different layers of latent code control different levels of image attributes. Based on this finding, many works [6, 49] tried to invert the real image to $\mathcal{W}$ space for semantic face editing. Some recent works [1, 2, 34] show that the $\mathcal{W}^+$ space extended from the $\mathcal{W}$ space has lower reconstruction errors. In this paper, we build a link between our proposed RFGs and the $\mathcal{W}^+$ space of StyleGAN2 [24], based on which we can achieve fine-grained control over the synthesis of facial regions.

2.2. Kinship Face Synthesis

Kinship face synthesis aims to synthesize the face images of descendants given the images of parents. The challenge is to learn facial genetic relations with limited kinship data. Table 1 summarizes the key differences between ours and existing methods. Early works [14, 29, 48] design the image level mapping from parents to children via supervised learning. However, limited by the quality and scale of training data, the quality of images produced by these methods is usually low and prone to overfitting. Recent works

Table 1. Comparison between StyleGene and existing kinship face synthesis methods. N_r denotes the number of controllable regions.

Methods	Stages	Kinship annotation	Image resolution	Diversity Controlling	N_r
DNA-Net [14]	two	✓	128	Noise	-
KinshipGAN [29]	one	✓	128	Noise	-
ChildPredictor [48]	two	✓	128	Noise	-
CDFS [45]	one	✗	256	Noise	5
StyleDNA [27]	two	✓	1024	Noise	-
ChildGAN [7]	two	✗	1024	Interpolation	5
StyleGene (Ours)	two	✗	1024	Gene Pool	34

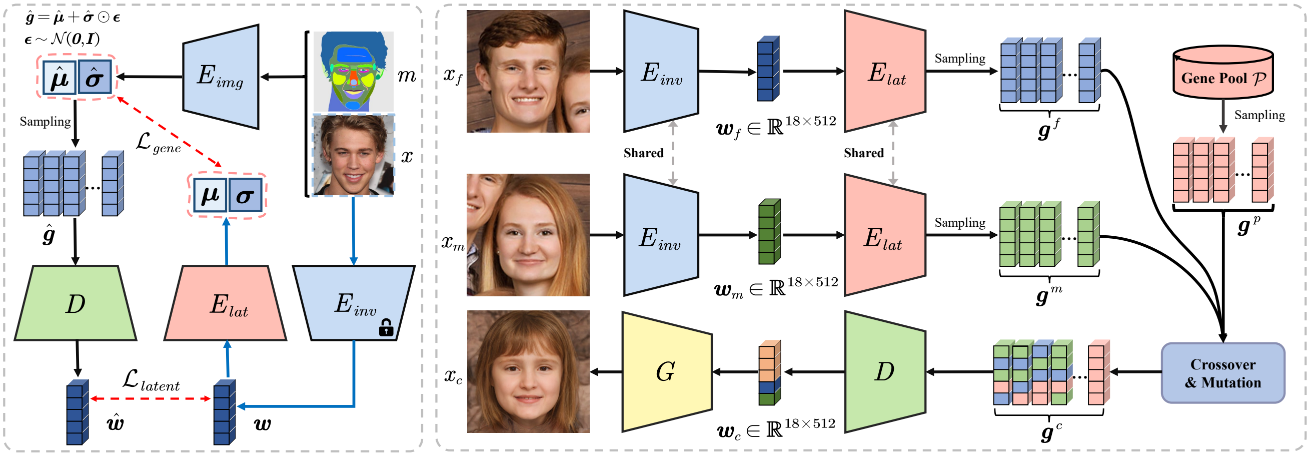


Figure 2. The overall framework of our method. The image-based gene encoder E_{img} learns to encode an independent RFG representation for each facial region in the training stage. The gene decoder D maps RFGs $\boldsymbol{g}$ to the $\mathcal{W}^+$ space of StyleGAN2. For ease of use, our latent-based gene encoder E_{lat} maps latent codes $\boldsymbol{w}$ (obtained by an image inversion encoder E_{inv}) to RFGs. In the inference stage, the RFGs of both parents are first extracted using latent-based gene encoder, and then the RFGs of the descendants are assembled by crossover and mutation module. The RFGs will be mapped back to the $\mathcal{W}^+$ space using gene decoder, and finally the high-fidelity face is generated by a pre-trained StyleGAN2 generator G.

[7,27] based on StyleGAN can generate high-resolution images by interpolating in the latent space to control the generation of descendants. However, StyleDNA [27] learns the mapping from parents to children in $\mathcal{W}$ space by supervised learning, resulting in poor diversity due to overfitting. In addition, the inherited regions of children are uncontrollable. ChildGAN [7] calculates the direction vectors of different facial regions through annotated landmarks, and controls the genetic regions by interpolation. However, due to attribute entanglement in $\mathcal{W}$ space, finding the disentangled direction vectors for small facial regions is difficult, which limits the diversity of generated descendants.

To address this problem, our proposed StyleGene first learns region-level facial genes (RFGs) to control the synthesis of face regions. Then we further model kinship relations based on RFGs, and leverage the pre-trained StyleGAN2 generator to generate kinship face.

3. Method

In this section, we present our approach for high-fidelity kinship face synthesis. Given a pair of parental face images, x_f and x_m, our goal is to synthesize face images of their descendants. We consider the facial genetic process as the inheritance of facial parts from the father and mother.

The overview of the proposed framework is shown in Fig. 2. In training, we build models to extract **Region-level Facial Genes (RFGs)** from StyleGAN2 latent space and then learn to map the disentangled genes representations back to the corresponding latent code. In particular, we use an **Image-based Gene Encoder (IGE)** to directly extract RFGs from the input face with region annotations. Then a Gene Decoder is trained to map the obtained RFGs to StyleGAN2 latent space. In addition, we use a GAN inversion method to embed the same face to the $\mathcal{W}^+$ latent space of StyleGAN2, and a **Latent-based Gene Encoder (LGE)** is used to decompose the obtained latent code into the corresponding RFGs.

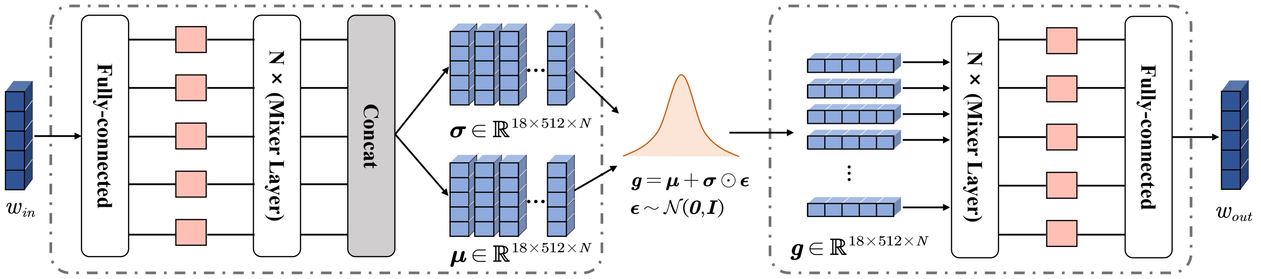
In inference, we first use LGE to extract the RFGs of parental faces, which are then used to obtain children's RFGs via crossover and mutation process. In addition, we build a Gene Pool to simulate the genetic variations.

3.1. Region-level Facial Gene

In order to build facial genetic relations, a key step of the proposed framework is to extract disentangled representations of different face regions. We propose to train IGE to directly extract RFGs by using fine-grained annotated facial parts. We then use a Gene Decoder to transform the obtained gene representations into the StyleGAN2 latent space, from which we can reconstruct the original face image.

Fine-grained Facial Parts Segmentation. We follow DatasetGAN [46] pipeline to generate face images and corresponding segmentation masks. DatasetGAN trains a shallow decoder to achieve semantic segmentation based on the features of StyleGAN by only using a few annotated data. In this work, we adapt their approach to StyleGAN2 to produce pixel-level labels for N ($N = 34$) facial parts. These fine-grained facial parts are then used to extract the corresponding facial genes.

Image-based Gene Encoder. Fig. 4 shows the overview of our IGE $E_{img}(\cdot, \cdot)$. The goal of IGE is to learn region-


Fully-connected
N × (Mixer Layer)
Concat
$\boldsymbol{\sigma} \in \mathbb{R}^{18\times512\times N}$
$\boldsymbol{\mu} \in \mathbb{R}^{18\times512\times N}$
$\boldsymbol{g} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}$
$\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0},\boldsymbol{I})$
$\boldsymbol{g} \in \mathbb{R}^{18\times512\times N}$
N × (Mixer Layer)
Fully-connected
w_{in}
w_{out}

(a) Latent-based Gene Encoder

(b) Gene Decoder

Figure 3. The proposed Latent-based Gene Encoder (a) and Gene Decoder (b) architecture. We mainly use the Mixer layer [39] to deal with the nonlinear transformation between high-dimensional vectors.

level facial gene representation $\boldsymbol{g}_i$ for each facial part.

Given a face image $x \in \mathbb{R}^{3\times512\times512}$ and the corresponding mask $m = \{m_1, \cdots, m_N\}$, we first transform x to a feature map $f(x)$ with 64×64 resolution by a series of convolutional layers f. Then we extract the region feature $\boldsymbol{r}_i$ for the i-th face part by masking the obtained $f(x)$ with the corresponding binary mask m_i, i.e., $\boldsymbol{r}_i = f(x) \odot m_i$, where $\odot$ is an element-wise matrix multiplication operator. Inspired by variational autoencoder [25], the posterior distribution for each region $\boldsymbol{r}_i$, mapped by a mapping function h_i, is modeled as a multivariate Gaussian distribution. It is defined by

$$\hat{\boldsymbol{g}}_i \sim q(\hat{\boldsymbol{g}}_i|\boldsymbol{r}_i) = \mathcal{N}(\hat{\boldsymbol{g}}_i; \hat{\boldsymbol{\mu}}_i, \hat{\boldsymbol{\sigma}}_i^2 \boldsymbol{I}), \tag{1}$$

where $\hat{\boldsymbol{\mu}}_{\boldsymbol{i}}, \hat{\boldsymbol{\sigma}}_{\boldsymbol{i}} \in \mathbb{R}^{18\times512}$ are the multi-dimensional output of $h_i(\boldsymbol{r}_i)$, representing the mean and standard deviation in a diagonal matrix form, respectively. According to [25], backpropagation is made differentiable via the reparameterization trick, thus $\hat{\boldsymbol{g}}_i \in \mathbb{R}^{18\times512}$ can be sampled by

$$\hat{\boldsymbol{g}}_i = \hat{\boldsymbol{\mu}}_i + \hat{\boldsymbol{\sigma}}_i \odot \boldsymbol{\epsilon}, \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \tag{2}$$

Finally, since we use N facial parts, all the RFG can be denoted by $\hat{\boldsymbol{g}} = [\hat{\boldsymbol{g}}_1, \cdots, \hat{\boldsymbol{g}}_N] \in \mathbb{R}^{18\times512\times N}$. We simplify this process denoted as $\hat{\boldsymbol{g}} = E_{img}(x, m)$.

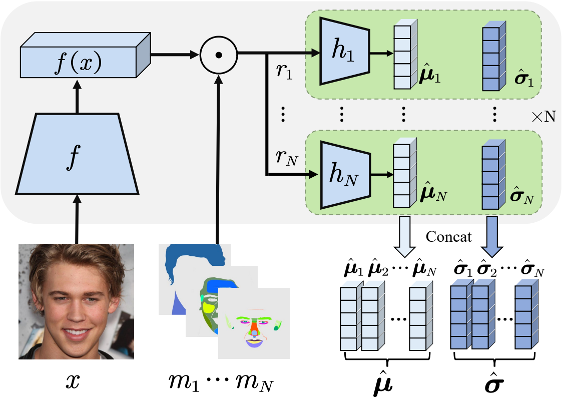

$f(x)$
f
r_1
h_1
$\hat{\boldsymbol{\mu}}_1$
$\hat{\boldsymbol{\sigma}}_1$
×N
r_N
h_N
$\hat{\boldsymbol{\mu}}_N$
$\hat{\boldsymbol{\sigma}}_N$
Concat
$\hat{\boldsymbol{\mu}}_1 \hat{\boldsymbol{\mu}}_2 \cdots \hat{\boldsymbol{\mu}}_N$
$\hat{\boldsymbol{\sigma}}_1 \hat{\boldsymbol{\sigma}}_2 \cdots \hat{\boldsymbol{\sigma}}_N$
x
$m_1 \cdots m_N$
$\hat{\boldsymbol{\mu}}$
$\hat{\boldsymbol{\sigma}}$

Figure 4. The proposed Image-based Gene Encoder.

Latent-based Gene Encoder. As shown above, the proposed IGE needs a face mask to extract the representations for each facial region. To eliminate the dependency, we train a LGE $E_{lat}(\cdot)$ to directly extract RFGs from StyleGAN2 latent space. As shown in Fig. 3(a), our LGE mainly consists of a fully connected layer and several Mixer Layers [39], and maps the latent code $\boldsymbol{w}$ of a face image x to the mean $\boldsymbol{\mu}_i$ and standard deviation $\boldsymbol{\sigma}_i$ for each pre-defined face region. This process can be formulated by

$$\{\boldsymbol{\mu}, \boldsymbol{\sigma}\} = E_{lat}(\boldsymbol{w}), \tag{3}$$

where $\boldsymbol{\mu} = [\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_N]$ and $\boldsymbol{\sigma} = [\boldsymbol{\sigma}_1, \cdots, \boldsymbol{\sigma}_N]$, and we use a pre-trained GAN inversion model $E_{inv}(\cdot)$ [40] as the image encoder to get the latent code of a face image, i.e. $\boldsymbol{w} = E_{inv}(x)$. Then the region-level gene $\boldsymbol{g}_i$ for the i-th facial region can be sampled by

$$\boldsymbol{g}_i = \boldsymbol{\mu}_i + \boldsymbol{\sigma}_i \odot \boldsymbol{\epsilon}, \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}). \tag{4}$$

Finally, the RFGs of N facial parts can be denoted by $\boldsymbol{g} = [\boldsymbol{g}_1, \cdots, \boldsymbol{g}_N]$.

Gene Decoder. We then use Gene Decoder $D(\cdot)$ to map the region-level facial genes into the StyleGAN2 latent space. Thus, we can reconstruct the face images from the obtained gene representation. Specifically, the Gene Decoder first processes the obtained RFGs with several Mixer Layers [39]. Then all the regional features are concatenated and further processed by a fully connected layer to obtain the latent code $\hat{\boldsymbol{w}} \in \mathbb{R}^{18\times512}$. This process can be formulated by

$$\hat{\boldsymbol{w}} = D(\hat{\boldsymbol{g}}), \tag{5}$$

where $\hat{\boldsymbol{g}} = [\hat{\boldsymbol{g}}_1, \cdots, \hat{\boldsymbol{g}}_N]$ is the RFGs of N facial parts.

Loss Functions. As shown in Fig. 2(a), given an input face image x, the corresponding region mask m and the latent code $\boldsymbol{w} = E_{inv}(x)$, we train our models by using following loss functions.

To build a bridge between RFGs and latent space, we first constrain the IGE E_{img} and Gene Decoder D to reconstruct the latent code of the input image. The latent reconstruct loss is defined by

$$\mathcal{L}_{latent} = \parallel \boldsymbol{w} - \hat{\boldsymbol{w}} \parallel_2, \tag{6}$$

where $\hat{\boldsymbol{w}} = D(E_{img}(x, m))$.

In addition, we use LGE E_{lat} to remove the reliance on face annotations. To decouple the latent code into the RFGs obtained from IGE, the LGE is trained by

$$\mathcal{L}_{gene} = \sum_{i=1}^{N} \left[\parallel \boldsymbol{\mu}_i - \hat{\boldsymbol{\mu}}_i \parallel_2 + \parallel \boldsymbol{\sigma}_i - \hat{\boldsymbol{\sigma}}_i \parallel_2 \right], \tag{7}$$

where $\hat{\boldsymbol{\mu}}_i$ and $\hat{\boldsymbol{\sigma}}_i$ are the outputs of $E_{img}(x, m)$ for the i-th facial region, and $\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i$ are the outputs of $E_{lat}(\boldsymbol{w})$ for the i-th facial region.

All modules are trained jointly, and the full loss is defined by

$$\mathcal{L} = \mathcal{L}_{latent} + \lambda \mathcal{L}_{gene}, \tag{8}$$

where λ is a hyperparameter, and we set it to be 1 in our experiments.

Note that this training process does not rely on kinship data, alleviating the problem of insufficient high-quality annotated kinship datasets.

3.2. Crossover & Mutation

As parent-offspring resemblance is often reflected in the local area of the face appearance [5, 31]. Thus, after obtaining the region-level facial genes (RFGs), we simulate a crossover process to generate the RFGs of descendants by combining the RFGs of the parents. Additionally, we further build a gene pool to simulate the gene mutation process, which can significantly increase the diversity of synthesized faces.

Gene Crossover. The gene crossover process aims to generate the gene representation of the descendants with a linear combination of the RFGs of the parents. In particular, given a pair of parental face images, x_f and x_m, we first apply Image Encoder and LGE to obtain a set of RFGs $\boldsymbol{g}^f = [\boldsymbol{g}_1^f, \cdots, \boldsymbol{g}_N^f]$ and $\boldsymbol{g}^m = [\boldsymbol{g}_1^m, \cdots, \boldsymbol{g}_N^m]$ from N facial parts, respectively. Then, the RFGs of descendants $\boldsymbol{g}^c = [\boldsymbol{g}_1^c, \cdots, \boldsymbol{g}_N^c]$ can be calculated by

$$\boldsymbol{g}_i^c = \alpha_i \boldsymbol{g}_i^f + \beta_i \boldsymbol{g}_i^m, \tag{9}$$

where α_i and β_i are randomly generated weights for each region and $\alpha_i + \beta_i = 1$ when mutation is not applied.

Fig. 5 shows four example faces generated using RFGs inherited from their parents based on Eq. 9, when different combinations of α_i and β_i are applied. The bigger value of α_i, the more similar to father for the i-th facial region. The figure shows that our approach can precisely and independently control the similarity of each facial region with their parents, by setting different values of α_i and β_i.

Gene Mutation. In genetics, the gene pool is the set of all genes of all individuals in a population [30]. For humans, all races share the same gene pool. We introduce the gene pool concept to better simulate the genetic variation in facial appearances. We define the gene pool $\mathcal{P}$ as the sets formed by grouping a large number of RFGs, i.e. $\mathcal{P} = \{\boldsymbol{g}^1, \cdots, \boldsymbol{g}^p, \cdots, \boldsymbol{g}^P\}$, where $\boldsymbol{g}^p = [\boldsymbol{g}_1^p, \cdots, \boldsymbol{g}_N^p]$. As shown in the supplementary materials, we divide the gene pool into nine groups by age, two by gender, and seven by race. Note that the RFG in the gene pool can independently control different facial traits. And its phenotype is related to the grouping condition.

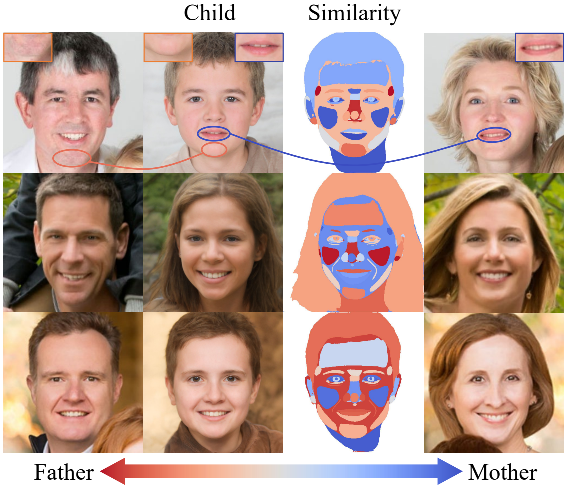


Figure 5. Example of different weights used in crossover of RFGs. While the redder regions look more similar to the father, and the bluer regions look more similar to the mother.

Given a pair of parental face images and information about generated descendants, such as age, gender, and race. We first query the gene pool to extract a subset $\mathcal{P}_c \subseteq \mathcal{P}$ that satisfies the descendants' information, which will be used to simulate gene mutations for better diversity in the descendants. Then we use Image Encoder and LGE to extract RFGs $\boldsymbol{g}^f = [\boldsymbol{g}_1^f, \cdots, \boldsymbol{g}_N^f]$ and $\boldsymbol{g}^m = [\boldsymbol{g}_1^m, \cdots, \boldsymbol{g}_N^m]$ of parents. For mutation, we randomly select η percent of facial regions whose RFGs are sampled from the gene pool. Given a $\boldsymbol{g}^p$ sampled from $\mathcal{P}_c$, a one-hot vector $t = \{t_i \in \{0, 1\}, i = 1, 2, \cdots, N\}$ is randomly generated to denote whether the $\boldsymbol{g}_i^c$ of descendant is fully copied from $\boldsymbol{g}_i^p$ ($t_i = 1$), or not ($t_i = 0$). When $\boldsymbol{g}_i^c$ is inherited from parents ($t_i = 0$), we also introduce mutation by introducing $\boldsymbol{g}^p$ into the crossover process defined in Eq. 9. Finally, we can extend Eq. 9 as follows to obtain the RFGs of the descendants $\boldsymbol{g}^c = [\boldsymbol{g}_c^1, \cdots, \boldsymbol{g}_c^N]$:

$$\boldsymbol{g}_i^c = \begin{cases} \boldsymbol{g}_i^p, & t_i = 1 \\ \alpha_i \boldsymbol{g}_i^f + \beta_i \boldsymbol{g}_i^m + \gamma \boldsymbol{g}_i^p, & t_i = 0 \end{cases}, \tag{10}$$

where γ is the intensity of mutation, α_i and β_i are randomly generated weights and $\alpha_i + \beta_i = 1 - \gamma$, $\boldsymbol{g}_i^p = S(\mathcal{P}_c)$ and $S(\cdot)$ is the random sampling operator. Once the RFG of descendant $\boldsymbol{g}^c$ is generated, the learned Gene Decoder is applied to map $\boldsymbol{g}^c$ to the latent space of StyleGAN2, i.e. $\boldsymbol{w}_c = D(\boldsymbol{g}^c)$, $\boldsymbol{w}_c \in \mathbb{R}^{18 \times 512}$, which is further processed for face synthesis.

While the latent code in $\mathcal{W}^+$ latent space of StyleGAN2 usually consists of 18 layers, Richardson et al. [34] demonstrate that the first 8 layers of the latent code mainly contribute to the ID information of the synthesized faces, and higher layers mainly control the skin color and microstructure of the synthesized faces. While the appearance of facial regions for descendants is mainly decided by $\boldsymbol{g}^c$, other attributes like skin, hair color, image lighting, and background, can thus be inherited by inclusion of the latent codes of parents. Therefore, we further fuse the latent codes of parents $\boldsymbol{w}_f$ and $\boldsymbol{w}_m$. We keep the first l layer of $\boldsymbol{w}_c$, and the higher layers are fused from $\boldsymbol{w}_f$ and $\boldsymbol{w}_m$. The final latent code $\boldsymbol{w}_c = [\boldsymbol{w}_c^1, \cdots, \boldsymbol{w}_c^{18}]$ of the descendant can be obtained by

$$\boldsymbol{w}_c^i = \frac{1}{2}(\boldsymbol{w}_f^i + \boldsymbol{w}_m^i), \quad i \in \{l+1, \cdots, 18\}, \tag{11}$$

4. Experiments

4.1. Experimental Settings

Datasets. Our training set consists of three parts, all images of CelebAHQ [21], 50,000 faces sampled from MS-Celeb-1M [18], and 10,000 faces generated by StyleGAN2, to make the model adaptive to different image qualities. CelebA-HQ is a high-quality face dataset consisting of 30,000 aligned face images with 1024 × 1024 resolution. The MS-Celeb-1M is a large-scale face recognition dataset containing about 10 million images, including many noisy face images. We evaluate our model on the FIW [35], TSKinFace [32], and FF-Database [48] datasets. FIW dataset contains the faces of 1,000 families with 1,997 sets of father-mother-child relations. The TSKinFace and FF-Database datasets provide 1,015 and 3,744 sets of father-mother-child kinship face images, respectively. We align and crop the images in these two dataset to 256 × 256. Our Gene Pool is built based on the FFHQ [23] dataset, which consists of 70,000 high-quality face images, and has a better diversity among human races and ages.

Baselines. We compare our method against the state-of-art kinship face synthesis baselines, i.e. DNA-Net [14], ChildGAN [7], ChildPredictor [48], and StyleDNA [27]. Since the source codes of StyleDNA and ChildPredictor are available, we compare our approach with them through all of the evaluations. However, due to the unavailability of the source codes, we can only visually compare the visual quality of the generated faces available in the papers of DNA-Net and ChildGAN.

Training Details. We use AdamW [28] optimizer with batch size of 32. The initialized learning rate is 0.001, which is divided by 2 every 10 epochs, and we stop the training at the 30th epoch.

4.2. Qualitative evaluation

Disentanglement of RFGs. Fig. 6 shows two examples of facial region editing using our proposed RFG, where the nose, eyes, jaw and lips of source faces shown in the first column, are replaced sequentially by the corresponding regions of reference faces shown in the first row. One can observe from the examples that, when a certain region is edited, all other facial regions are kept intact, which clearly shows the disentanglement capability of our RFG.

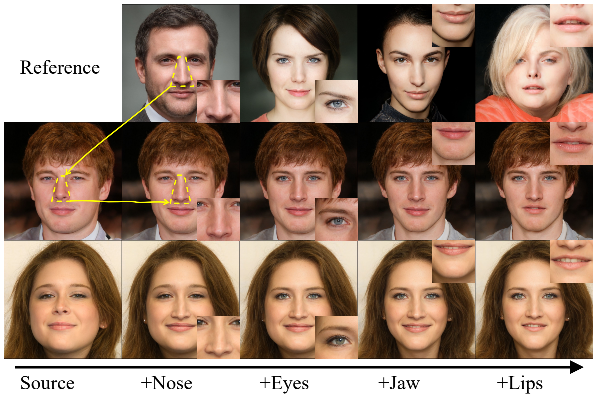


Figure 6. Disentangled editing of facial regions.

Effectiveness of Gene Pool. Based on the proposed idea of RFG, our approach further introduce a gene pool to increase the variations among the generated faces of descendants. Given a couple of parents shown in the first row, Fig. 7 shows the faces of five children generated by our approach with/without the involvement of gene pool, together with that synthesized by StyleDNA and ChildPredictor. While the faces generated by StyleDNA and ChildPredictor (4th and 5th rows) all look very similar, that synthesized by our approach are much more diverse and the introduction of gene pool (1st row) can further increase variations among children.

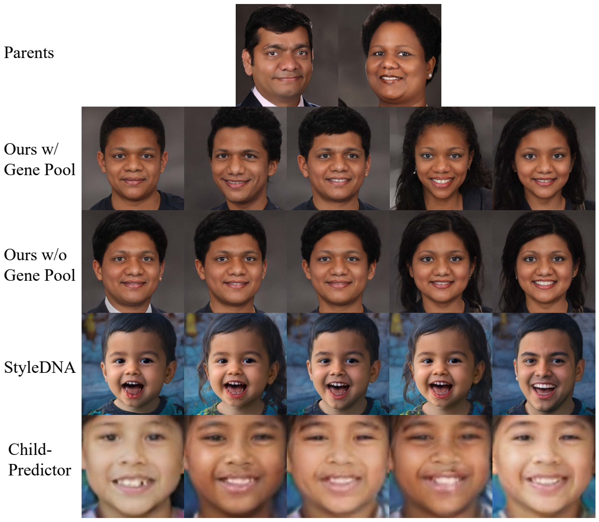
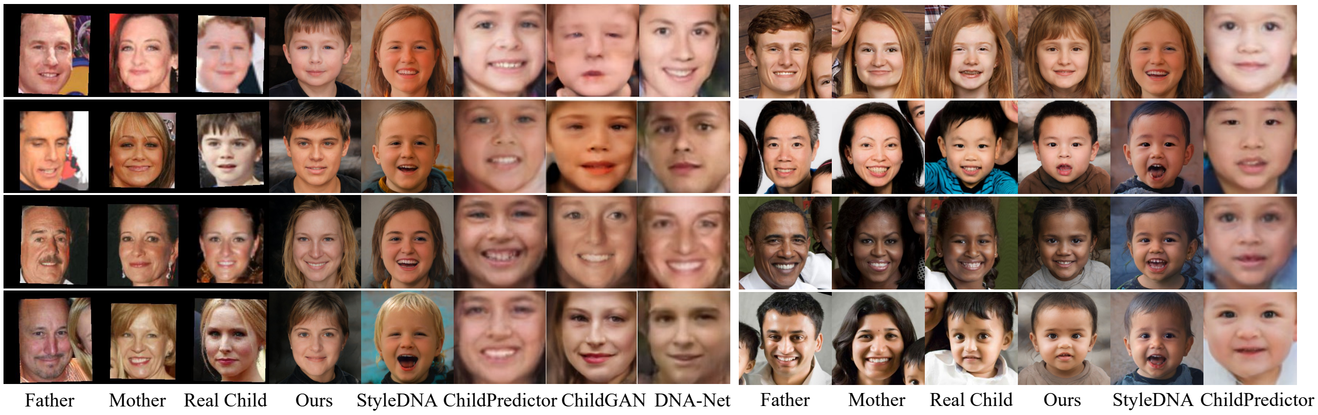


Figure 7. Visual comparison of diversity with StyleDNA and ChildPredictor. The first row shows the father and mother. The next two rows are generated by our method with and without Gene Pool. The last two rows are generated by StyleDNA [27] and ChildPredictor [48], respectively.



(a) FIW dataset

(b) FF-Database

Figure 8. Comparison of children faces synthesized by StyleGene and baselines. Left most three columns of (a) FIW and (b) FF-Database depict the father, mother, and real children, and right five or three columns depict the faces synthesized by ours, StyleDNA [27], ChildPredictor [48], ChildGAN [7] and DNA-Net [14].

Comparison with the state-of-the-art. Fig. 8(a) and (b) show qualitative results on FIW and FF-Database, respectively. We first compare StyleGene with StyleDNA, DNA-Net, ChildGAN, and ChildPredictor on FIW with low-resolution facial images. Note that the results of ChildGAN and DNA-Net are directly taken from their papers. As can be seen, DNA-Net, ChildGAN, and ChildPredictor struggle to yield high-quality facial images. Thanks to a powerful generator, our method can generate high-fidelity faces with hair. Though ChildGAN also uses the StyleGAN generator, its results are significantly affected by the quality of the input. Note that all comparisons have considered the facial information of parents. However, DNA-Net, ChildPredictor, and StyleDNA model facial genetics by learning the implicit mapping between parents and children, without considering regional similarity, which significantly compromised the similarity between the generated faces and their parents. Although ChildGAN considered parents' facial features, the results were too similar to their parents. Thanks to the region-level facial genes and the mutation introduced by gene pool, our results can maintain the local similarity to parents' faces, and at the same time, present reasonable global variations and diversity. When experimenting with faces of different races as shown in Fig. 8(b), our approach can still well preserve the race similarity of children with their parents.

4.3. Quantitative evaluation

Kinship Verification. We now use kinship verification accuracy to assess whether a genetic relationship exists between the synthetic descendant and parents. Higher accuracy means more realistic synthetic descendants. We use ArcFace pre-trained on the MS-Celeb-1M dataset as the backbone to create a one-versus-one kinship classifier [42], which is fine-tuned on the FF-Database training set. Then we performed a cross-database evaluation.

Specifically, we randomly sample 100 families with one kid from the FF-database test set, FIW, and TSKinFace datasets, respectively, and different methods randomly generate 40 children for each family. We randomly sample 8,000 positive and 8,000 negative pairs for each method from each dataset and calculate in Table 2 the kinship verification accuracy. One can observe that our method achieves substantially higher accuracy than StyleDNA and ChildPredictor, i.e. as high as 81.74%, 80.38%, and 62.29% accuracies are achieved on TSKinFace, FF-Database, and FIW datasets, respectively. The results suggest that the kinship relations of faces generated by our approach are very close to the real children.

Table 2. Kinship verification accuracy (%) on the TSKinFace [32], FF-Database [48], and FIW [35] dataset.

Methods	TSKinFace	FF-Database	FIW
StyleDNA [27]	53.15	55.11	49.47
ChildPredictor [48]	58.24	59.62	51.81
StyleGene (Ours)	**81.74**	**80.38**	**62.29**

Diversity Evaluation. We used the LPIPS [44] metric to measure the diversity of synthetic descendants, which calculates the L1 distance between pairs of image features extracted by AlexNet [26] pre-trained on the ImageNet [10] dataset. We use the same test data of kinship verification. First, we calculated the distance among the 40 synthesized descendants for each family. Then we take the average of 100 families as the LPIPS score and list them in Table 3. As shown in the table, our method achieves the highest LPIPS across all three datasets, i.e. 0.3270, 0.3418, and 0.3279 LPIPS on TSKinFace, FF-Database, and FIW datasets are achieved, which is significantly higher than that of StyleDNA and ChildPredictor.

Distribution of Synthesized Children. We are now try-

Table 3. Quantitative comparison of the diversity of the generated descendants. * means we cropped the face i.e. no hair.

Methods	TSKinFace	FF-Database	FIW
StyleDNA* [27]	0.0756	0.0763	0.0736
ChildPredictor [48]	0.1697	0.1723	0.1750
StyleGene (Ours)*	0.1748	0.1735	0.1740
StyleDNA [27]	0.1559	0.1542	0.1573
StyleGene (Ours)	**0.3270**	**0.3418**	**0.3279**

ing to model the distribution of the children synthesized by different approaches, and compare them with that of real children. Specifically, we randomly sample 50 families from the TSKinFace dataset, each consisting of a father, mother, son, and daughter. Based on the given parents, we also applied different synthesis methods to generate two children, i.e. a son and a daughter, for each family. After facial features are extracted using ArcFace [11] and reduced to one dimension using t-SNE [41], Kernel Density Estimation (KDE) [37] is further applied to estimate the features' probability density function. Fig. 9 shows the distribution of the 100 real children (blue) and that of children generated by different approaches. As can be seen, the distribution of faces generated by our approach (red) overlaps most with the real children (blue). In contrast, the distributions of faces synthesized by StyleDNA (black) and ChildPredictor (yellow) are located on both sides of the real ones. The distribution for StyleDNA even looks like a mixture of two Gaussian distributions, which is significantly different from that of real children.

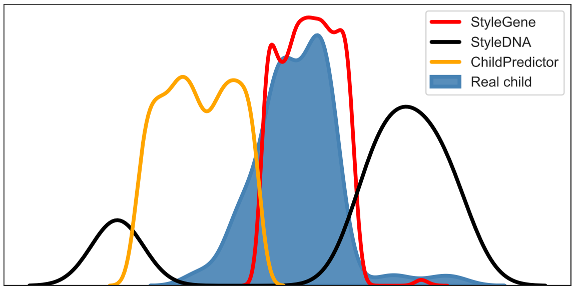


Figure 9. Distribution of real children (blue) and children generated using StyleDNA (black), ChildPredictor (yellow), and ours (red). Best viewed in color.

User Study. To further demonstrate the effectiveness of the proposed method, we conducted a user study involving 59 participants. The faces of children generated by our StyleGene, StyleDNA and ChildPredictor based on 20 parents, are present to each of the 59 participants, who are asked to rank the quality of the faces, in terms of realness and the similarity with given parents. In total 1,180 (59×20) votes are received for each of the three approaches and the average rankings are listed in Table 4. As shown in the table, our StyleGene receives the highest rank among the compared approaches, which suggests that the faces of children synthesized by our approach are the best, in terms of both quality and similarity with parents.

Table 4. The rank of different approaches in user study.

	ChildPredictor	StyleDNA	StyleGene (Ours)
Avg. rank	2.46	2.22	**1.32**

4.4. Ablation study

In this section, we perform ablation studies on the effectiveness of LGE, IGE and GP, in terms of kinship verification accuracy (ACC) and diversity (LPIPS) using FF-database with the same configuration as in Section 4.3. As shown in Table 5, the integration of IGE (2nd row) can improve both ACC and LPIPS of the synthesized descendant. As expected, the use of GP (3rd row) increases the diversity of synthesized faces. When both IGE and GP are integrated with LGE (4th row), the diversity can be significantly increased to 0.1735, with a comparable ACC (80.38%).

Table 5. Ablation study on FF-Database.

	LGE	IGE	GP	ACC (%)	LPIPS
1	✓	✗	✗	79.19	0.0625
2	✓	✓	✗	79.66	0.0646
3	✓	✗	✓	80.21	0.0839
4	✓	✓	✓	80.38	0.1735

4.5. Parameter sensitivity

We test different values of η, γ, and l to see how the performance of synthesis varies. Due to the page limit, details about the results of different parameters are presented in supplementary. In our experiments, we choose $\eta = 40\%$, $\gamma = 0.47$, and $l = 8$ to achieve a balance between the diversity and fidelity of the generated faces.

5. Conclusion

In this paper, we have proposed StyleGene, to extract RFGs (Region-level Facial Genes) for kinship face synthesis. While crossover and mutation of RFGs are proposed to model the facial genetic process, Gene Pool is further designed to increase the diversity among generated faces. Quantitative, qualitative, and subjective experimental results show that the realness, similarity with parents, and diversity of kinship faces generated by our approach are much better than existing state-of-the-art methods.

Acknowledgement. This work was jointly supported by National Natural Science Foundation of China (62206180 and 82261138629), Guangdong Basic and Applied Basic Research Foundation (2023A1515010688 and 2022A1515011018), and Shenzhen Municipal Science and Technology Innovation Council (JCYJ20220531101412030).

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4432–4441, 2019. 2

[2] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan++: How to edit the embedded images? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8296–8305, 2020. 2

[3] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8789–8797, 2018. 2

[4] Peter Claes, Jasmien Roosenboom, Julie D White, Tomek Swigut, Dzemila Sero, Jiarui Li, Myoung Keun Lee, Arslan Zaidi, Brooke C Mattern, Corey Liebowitz, et al. Genome-wide mapping of global-to-local genetic effects on human facial shape. *Nature genetics*, 50(3):414–423, 2018. 2

[5] Joanne B Cole, Mange Manyama, Jacinda R Larson, Denise K Liberton, Tracey M Ferrara, Sheri L Riccardi, Mao Li, Washington Mio, Ophir D Klein, Stephanie A Santorico, et al. Human facial shape and size heritability and genetic correlations. *Genetics*, 205(2):967–978, 2017. 2, 5

[6] Edo Collins, Raja Bala, Bob Price, and Sabine Susstrunk. Editing in style: Uncovering the local semantics of gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5771–5780, 2020. 2

[7] Xiao Cui, Wengang Zhou, Yang Hu, Weilun Wang, and Houqiang Li. Heredity-aware child face image generation with latent space disentanglement. *arXiv preprint arXiv:2108.11080*, 2021. 2, 3, 6, 7

[8] Maria F Dal Martello and Laurence T Maloney. Lateralization of kin recognition signals in the human face. *Journal of vision*, 10(8):9–9, 2010. 2

[9] Afshin Dehghan, Enrique G Ortiz, Ruben Villegas, and Mubarak Shah. Who do i look like? determining parent-offspring resemblance via gated autoencoders. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1757–1764, 2014. 2

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 7

[11] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019. 8

[12] Itir Önal Ertugrul and Hamdi Dibeklioglu. What will your future child look like? modeling and synthesis of hereditary patterns of facial dynamics. In *2017 12th IEEE international conference on automatic face & gesture recognition (FG 2017)*, pages 33–40. IEEE, 2017. 2

[13] Rinon Gal, Or Patashnik, Haggai Maron, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (TOG)*, 41(4):1–13, 2022. 2

[14] Pengyu Gao, Joseph Robinson, Jiaxuan Zhu, Chao Xia, MIng Shao, and Siyu Xia. Dna-net: Age and gender aware kin face synthesizer. In *2021 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2021. 2, 6, 7

[15] Pengyu Gao, Siyu Xia, Joseph Robinson, Junkang Zhang, Chao Xia, Ming Shao, and Yun Fu. What will your child look like? dna-net: Age and gender aware kin face synthesizer. *arXiv preprint arXiv:1911.07014*, 2019. 2

[16] Fady S Ghatas and ElSayed E Hemayed. Gankin: generating kin faces using disentangled gan. *SN Applied Sciences*, 2(2):1–10, 2020. 2

[17] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 2

[18] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *European conference on computer vision*, pages 87–102. Springer, 2016. 6

[19] Hanne Hoskens, Dongjing Liu, Sahin Naqvi, Myoung Keun Lee, Ryan J Eller, Karlijne Indencleef, Julie D White, Jiarui Li, Maarten HD Larmuseau, Greet Hens, et al. 3d facial phenotyping by biometric sibling matching used in contemporary genomic methodologies. *PLoS genetics*, 17(5):e1009528, 2021. 2

[20] Ali Jahanian, Lucy Chai, and Phillip Isola. On the" steerability" of generative adversarial networks. *arXiv preprint arXiv:1907.07171*, 2019. 2

[21] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 2, 6

[22] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems*, 34:852–863, 2021. 2

[23] 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. 2, 6

[24] 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. 2

[25] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 4

[26] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012. 7

[27] Che-Hsien Lin, Hung-Chun Chen, Li-Chen Cheng, Shu-Chuan Hsu, Jun-Cheng Chen, and Chih-Yu Wang. Styledna: A high-fidelity age and gender aware kinship face synthesizer. In *2021 16th IEEE International Conference on Auto-*

matic Face and Gesture Recognition (FG 2021), pages 1–8. IEEE, 2021. 2, 3, 6, 7, 8

[28] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[29] Savas Ozkan and Akin Ozkan. Kinshipgan: Synthesizing of kinship faces from family photos by regularizing a deep face network. In *2018 25th IEEE international conference on image processing (ICIP)*, pages 2142–2146. IEEE, 2018. 2

[30] Svante Pääbo. The diverse origins of the human gene pool. *Nature Reviews Genetics*, 16(6):313–314, 2015. 5

[31] Shouneng Peng, Jingze Tan, Sile Hu, Hang Zhou, Jing Guo, Li Jin, and Kun Tang. Detecting genetic association of common human facial morphological variation using high density 3d image registration. *PLoS computational biology*, 9(12):e1003375, 2013. 5

[32] Xiaoqian Qin, Xiaoyang Tan, and Songcan Chen. Tri-subject kinship verification: Understanding the core of a family. *IEEE Transactions on Multimedia*, 17(10):1855–1867, 2015. 2, 6, 7

[33] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 2

[34] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2287–2296, 2021. 2, 5

[35] Joseph P Robinson, Ming Shao, Yue Wu, and Yun Fu. Families in the wild (fiw) large-scale kinship image database and benchmarks. In *Proceedings of the 24th ACM international conference on Multimedia*, pages 242–246, 2016. 6, 7

[36] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of gans for semantic face editing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9243–9252, 2020. 2

[37] Bernard W Silverman. *Density estimation for statistics and data analysis*. Routledge, 2018. 8

[38] Raunak Sinha, Mayank Vatsa, and Richa Singh. Familygan: Generating kin face images using generative adversarial networks. In *European conference on computer vision*, pages 297–311. Springer, 2020. 2

[39] Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *Advances in Neural Information Processing Systems*, 34:24261–24272, 2021. 4

[40] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for stylegan image manipulation. *ACM Transactions on Graphics (TOG)*, 40(4):1–14, 2021. 4

[41] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 8

[42] Jun Yu, Mengyan Li, Xinlong Hao, and Guochen Xie. Deep fusion siamese network for automatic kinship verification. In *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*, pages 892–899. IEEE, 2020. 2, 7

[43] Ishtiak Zaman and David Crandall. Genetic-gan: Synthesizing images between two domains by genetic crossover. In *European Conference on Computer Vision*, pages 312–326. Springer, 2020. 2

[44] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 7

[45] Yong Zhang, Le Li, Zhilei Liu, Baoyuan Wu, Yanbo Fan, and Zhifeng Li. Controllable descendant face synthesis. *arXiv preprint arXiv:2002.11376*, 2020. 2

[46] Yuxuan Zhang, Huan Ling, Jun Gao, Kangxue Yin, Jean-Francois Lafleche, Adela Barriuso, Antonio Torralba, and Sanja Fidler. Datasetgan: Efficient labeled data factory with minimal human effort. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10145–10155, 2021. 3

[47] Zhifei Zhang, Yang Song, and Hairong Qi. Age progression/regression by conditional adversarial autoencoder. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5810–5818, 2017. 2

[48] Yuzhi Zhao, Lai-Man Po, Xuehui Wang, Qiong Yan, Wei Shen, Yujia Zhang, Wei Liu, Chun-Kit Wong, Chiu-Sing Pang, Weifeng Ou, et al. Childpredictor: A child face prediction framework with disentangled learning. *IEEE Transactions on Multimedia*, 2022. 2, 6, 7, 8

[49] Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. In-domain gan inversion for real image editing. In *European conference on computer vision*, pages 592–608. Springer, 2020. 2

[50] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 2