Data Synthesis with Diverse Styles for Face Recognition via 3DMM-Guided Diffusion

Supplementary Material

This supplementary material provides more methodological and experimental details that were streamlined in the main text due to space limitations, which we hope is of interest to our readers. It mainly includes:

(A) Detailed experimental setup;
(B) Methodological details;
(C) Further explanations to metrics used in the main text;
(D) Additional experimental studies;
(E) Visualizations;
(F) Miscellaneous.

A. Detailed Experimental Setup

A.1. Implementation Details

We use publicly released FR model $\mathcal{F}$ from ElasticFace [2], unconditional DM $\mathcal{G}_{id}$ from DCFace [9], and encoder and decoder ϕ_e, ϕ_d from the VAE of LDM [16]. For the style encoder $\mathcal{E}$, we employ a simple network as depicted in Fig. 11. We train our generator $\mathcal{G}$ for 250K steps, using an Adam optimizer [10], an initial learning rate of 1e-4, and a total batch size of 512. To incorporate context blending, during training, we replace $\mathbf{c}_{id}$ and $\mathbf{c}_{sty}$ with learnable empty contexts $\mathbf{c}_{id}^{\emptyset}$ and $\mathbf{c}_{sty}^{\emptyset}$ with a probability of 0.1; during inference time, we employ CFG by choosing t_0=500 and w=0.5. To evaluate our 0.5/1.2M synthetic datasets, we train an IR-50 [5] FR model $\mathcal{F}_{syn}$ for 40 epochs using an SGD optimizer [18], an ArcFace [4] loss, a total batch size of 256, and an initial learning rate of 0.1. We employ random horizontal flipping as following the *de facto* standard in FR, and random cropping with a probability of 0.2 as recommended by [8]. We do not use other forms of data augmentation. We run all experiments on 8 NVIDIA RTX 3090 GPUs and use fixed random seed across all experiments.

A.2. Datasets

We train our LDM $\mathcal{G}$ on CASIA-WebFace [21], a dataset that consists of 490k face images of varied qualities from 10575 identities. We benchmark FR model $\mathcal{F}_{syn}$ trained on our synthetic images on 5 widely used test datasets, LFW [12], CFP-FP [20], AgeDB [14], CPLFW [22], and CALFW [23]. CFP-FP and CPLFW are designed to measure the FR in cross-pose variations, and AgeDB and CALFW are for cross-age variations.

A.3. Critical Feature Shapes

$\mathcal{G}, \mathcal{G}_{id}$ produces 3×128×128 images. Each of the 3DMM feature maps (surface normals, albedo, Lambertian rendering) is 3×128×128, and their concatenation by channel is 9×128×128. The latent representation of $\mathcal{G}$ is 3×32×32. For the training of FR model $\mathcal{F}_{syn}$, we resize the synthetic images into 3×112×112 to match $\mathcal{F}_{syn}$'s input shape. The lengths of $\mathbf{c}_{id}$ and $\mathbf{c}_{sty}$ are 512.

B. Methodological Details

B.1. Style Extraction

Our proposed approach uses an off-the-shelf DECA 3DMM $\mathcal{M}$ to extract style attributes $\mathbf{p}$ and render them into feature maps $\mathbf{m}$. We briefly digest these attributes and feature maps from the DECA paper [6] to help explain their details.

Style attribute extraction. Given input face image $\mathbf{x} \in \mathbb{R}^{3\times128\times128}$, DECA uses a trained encoder to infer 6 attribute groups that entirely describe the face's style: (1) **Shape** $p_s \in \mathbb{R}^{100}$, representing facial geometry features decomposed via principal component analysis (PCA). Each dimension controls a specific geometric aspect, *e.g.*, the width of facial contours; (2) **Expression** $p_e \in \mathbb{R}^{50}$, describing facial expression features extracted through PCA; (3&4) **Pose and Camera** $p_p \in \mathbb{R}^{9}$. Pose is represented in 3D coordinates, while the camera models the projection from the 3D facial mesh to 2D space. Since the image's pose is jointly determined by both 3D pose and camera information, we collectively refer to them as "pose" for simplicity; (5) **Texture** $p_t \in \mathbb{R}^{50}$, modeling facial textures such as wrinkles, derived via PCA; (6) **Illumination** $p_i \in \mathbb{R}^{27}$, describing lighting conditions on the facial 3D mesh using spherical harmonics. For simplicity, we represent these attributes together as a unified style vector $\mathbf{p} \in \mathbb{R}^{236}$ in our main text.

Feature map rending. DECA renders 3 feature maps based on the extracted style attributes. First, it generates a 3D facial mesh using FLAME [13], combining shape, expression, and pose attributes. The 3D mesh contains 5023 vertices. It then renders the mesh into the following feature maps: (1) **Surface Normals** $m_s \in \mathbb{R}^{3\times128\times128}$, representing facial geometry as the normal vectors of each vertex in the mesh; (2) **Albedo** $m_a \in \mathbb{R}^{3\times128\times128}$, capturing facial texture without lighting effects, derived by combining the mesh with texture attributes; (3) **Lambertian Rendering** $m_l \in \mathbb{R}^{3\times128\times128}$, a coarse rendering that incorporates both texture and illumination attributes. These three feature maps provide a detailed description of facial styles and are concatenated along the channel dimension into $\mathbf{m} \in \mathbb{R}^{9\times128\times128}$. This consolidated representation

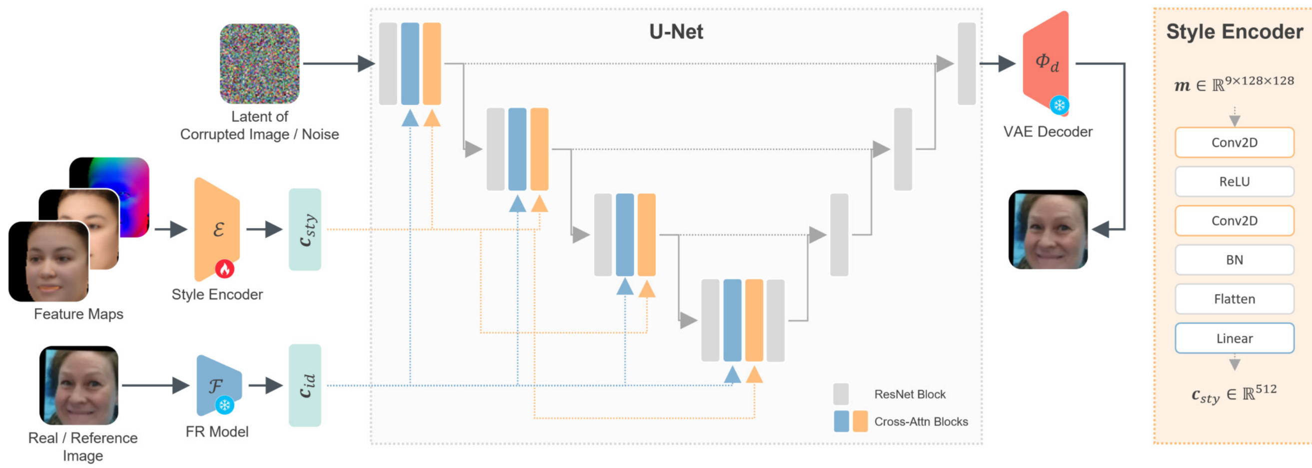

Figure 11. A detailed look at MorphFace generator $\mathcal{G}$ and style encoder $\mathcal{E}$. The main body of the generator is a U-Net [17] noise estimator. The identity and style contexts $\mathbf{c}_{id}, \mathbf{c}_{sty}$ are incorporated into the model via cross-attention layers after the U-Net's ResNet blocks. By cross-attention, we follow the same practice described in Sec. 3.3 of the LDM fundamental paper [16]. The style encoder $\mathcal{E}$ is a rather simple module consisting of 2 convolution layers plus a linear layer.

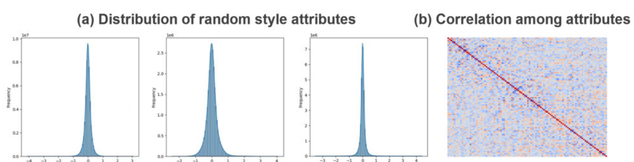

Figure 12. Distribution of style attributes. (a) We find Gaussian distributions in randomly chosen attribute dimensions from $\mathbf{P}$. (b) Sample correlation matrix of $\mathbf{P}$'s shape dimensions.

effectively supports style control.

Correction. We correct a minor mistake in our pipeline figure (Fig. 2): The "3D mesh" should appear after "style attributes", as part of DECA's rendering process.

B.2. Architecture of LDM Generator

Figure 11 provides a detailed look at MorphFace generator $\mathcal{G}$ and style encoder $\mathcal{E}$. The generator's main body is a U-Net [17] noise estimator. The identity and style contexts $\mathbf{c}_{id}, \mathbf{c}_{sty}$ are incorporated into the model via cross-attention layers after the U-Net's ResNet blocks. By cross-attention, we follow the same practice described in Sec. 3.3 of the LDM paper [16]. The style encoder $\mathcal{E}$ is a simple module including 2 convolution layers plus a linear layer.

B.3. Distribution of Style Attributes

In Sec. 3.3, we approximate the distribution of real-world style attributes by a multiplicative Gaussian distribution, *i.e.*, $\mathbb{D}(\mathbf{P}) \sim \mathcal{N}(\mu, \mathbf{\Sigma})$. We here explain its rationale: (1) Each attribute dimension of shape, expression and texture follows a Gaussian distribution as a natural outcome of DECA [6]. In DECA, these attributes are derived from PCA, and Gaussian distribution is part of PCA's assumption. (2) Previous findings [1, 15] suggest that facial attributes including pose and illumination can be modeled via Gaussian distributions. As each attribute dimension can be considered as an approximation of Gaussian distribution, their multiplication holds $\mathbb{D}(\mathbf{P}) \sim \mathcal{N}(\mu, \mathbf{\Sigma})$.

We also empirically validate the assumption. We find Gaussian distributions in randomly chosen attribute dimensions from $\mathbf{P}$, as shown in Fig. 12(a). Here, we can also infer each dimension's mean μ_i and variance ϵ_i. In Fig. 12(b), we visualize the correlation matrix of $\mathbf{P}$'s shape dimensions. Knowing each dimension's mean and variance, and the correlation matrix allows concretizing $\mathcal{N}(\mu, \mathbf{\Sigma})$.

B.4. Classifier-Free Guidance

In Sec. 3.4, we employ CFG [7] for context blending. CFG is a common technique in generative models, particularly DMs, to strengthen the generated samples' adherence to conditioning contexts without an explicit classifier.

In the training phase, CFG requires the model to be trained to predict the noise added to data for two scenarios, (1) conditional, when conditioning context (*i.e.*, $\mathbf{c}_{id}$ and $\mathbf{c}_{sty}$ in our case) is provided, and (2) unconditional, when the context is null or a placeholder. We achieve unconditional training by probabilistically replacing $\mathbf{c}_{id}$ and $\mathbf{c}_{sty}$ with learnable empty contexts $\mathbf{c}_{id}^{\emptyset}$ and $\mathbf{c}_{sty}^{\emptyset}$, *i.e.*, the placeholders. In the inference phase, the predicted noise ϵ_{cfg} is computed as a weighted combination of conditional and unconditional predictions as

$$\epsilon_{cfg} = (1+w)\epsilon_\theta(\mathbf{z}_t, t, \mathbf{c}) - w\epsilon_\theta(\mathbf{z}_t, t, \mathbf{c}^{\emptyset}), \tag{10}$$

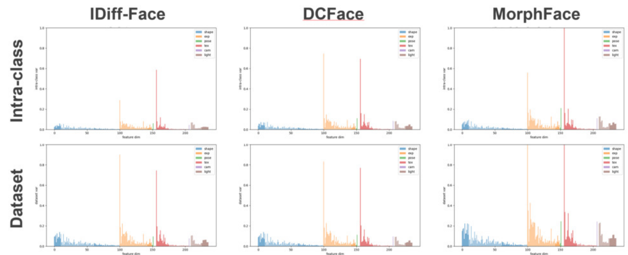

Figure 13. Variances of DECA-extracted style attributes. The same result is streamlined in Fig. 1. Larger intra-class and dataset-wise variance represent better intra-class style variation and dataset variability. It can be inferred that IDiff-Face is inadequate in style variation and MorphFace has diverse varied styles.

where $w{>}0$ strengthens the condition. We concretize $\epsilon_\theta(\mathbf{z}_t, t, \mathbf{c}^\emptyset)$ as $\epsilon_\theta(\mathbf{z}_t, t, \mathbf{c}_{id}, \mathbf{c}^\emptyset_{sty})$ and $\epsilon_\theta(\mathbf{z}_t, t, \mathbf{c}^\emptyset_{id}, \mathbf{c}_{sty})$ to incorporate dual conditions, to augment style and identity, respectively.

C. Metrics Explained

We explain the details of 4 metrics we used in the main text.

C.1. Cosine Similarity

It is the fundamental metric in SOTA FR systems to measure the similarity between two identity embeddings representing face images. Formally, let $\mathbf{x}_1, \mathbf{x}_2$ denote two face images, $\mathcal{F}$ denote a pre-trained FR model, and $\mathbf{d}_1, \mathbf{d}_2$ denote the identity embeddings extracted as $\mathbf{d} = \mathcal{F}(\mathbf{x})$. $\mathbf{d}_1, \mathbf{d}_2$ are 512-dim feature vectors in our case. The cosine similarity between $\mathbf{d}_1, \mathbf{d}_2$ is

$$\mathrm{cossim}(\mathbf{d}_1, \mathbf{d}_2) = \frac{\mathbf{d}_1 \cdot \mathbf{d}_2}{|\mathbf{d}_1||\mathbf{d}_2|}. \tag{11}$$

A larger $\mathrm{cossim}(\mathbf{d}_1, \mathbf{d}_2)$ indicates that $\mathbf{x}_1, \mathbf{x}_2$ are more likely the same person. To train an effective FR model, we expect the training face images to have high intra-class cosine similarity (*i.e.*, identity consistency within each subject) and low inter-class cosine similarity (*i.e.*, unique subjects). *In our main text:* (1) In Fig. 1, we depict curves of intra-class and inter-class cosine similarities, hence more separated curves indicate better FR efficacy. (2) In Fig. 7 and Tab. 2, we report the average intra-class cosine similarity. (3) In Sec. 3.3, we only enroll reference images with cosine similarity below 0.3 to filter those less distinct subjects.

C.2. DECA Attribute Variance

In Sec. 3.2, we use a pre-trained DECA 3DMM to infer the style attributes from an input image, $\mathbf{p}{=}\mathcal{M}(\mathbf{x})$. The style attributes can be considered a 236-dim vector, where its 100, 50, 9, 50, and 27 dimensions represent the image $\mathbf{x}$'s facial shape, expression, pose, texture, and illumination, respectively. As the image's style is solely parameterized by

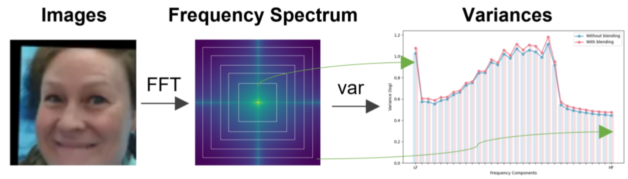

Figure 14. The calculation of frequency variances. Images are converted into frequency spectrum via FFT and partitioned into different frequency components, where their variances are measured. Higher variances reflect more informative frequency components.

style attributes, the intra-class and dataset-wise variances of these attributes demonstrate the dataset's intra-class style variation and dataset variability. *In our main text:* In Fig. 1, we depict the average style variance of facial shape, expression, pose, texture, and illumination, hence larger shaded areas represent better intra-class style variation and dataset variability. We supplement the detailed attribute-wise variance in Fig. 13. It can be inferred that IDiff-Face is inadequate in style variation and MorphFace has diverse styles.

C.3. Extended Improved Recall (eIR)

It is proposed by DCFace [9] as an extension of Improved Recall [11] to measure the style diversity of synthetic images. The images $\mathbf{x}$ are first mapped into a style latent space via an Inception Network [19] trained on ImageNet [3] to obtain inception vectors $\mathbf{v}$. To calculate eIR, for a set of real (*i.e.*, CASIA) and synthetic inception vectors $\{\mathbf{v}^c_i\}, \{\hat{\mathbf{v}}^c_j\}$ under the same label condition c, define the k-nearest feature distance r_k as $r_k{=}d(\hat{\mathbf{v}}^c_j - \mathrm{NN}_k(\hat{\mathbf{v}}^c_j, \{\hat{\mathbf{v}}^c_j\})$ where NN_k returns the k-nearest vectors in $\{\hat{\mathbf{v}}^c_j\}$ and

$$\mathbf{I}(\mathbf{v}^c_i, \{\hat{\mathbf{v}}^c_j\}) = \begin{cases} 1, & \exists \hat{\mathbf{v}}^c_j \in \{\hat{\mathbf{v}}^c_j\} \text{ s.t. } d(\mathbf{v}^c_i, \hat{\mathbf{v}}^c_j) < r_k, \\ 0, & \text{otherwise}, \end{cases} \tag{12}$$

$d(\cdot)$ is l_2 distance. The eIR is defined as

$$\mathrm{eIR} = \frac{1}{C}\frac{1}{\sum_c N_c}\sum_{c=1}^{C}\sum_{i=1}^{N_c}\left(\mathbf{I}(\mathbf{v}^c_i, \{\hat{\mathbf{v}}^c_j\})\right), \tag{13}$$

which is the fraction of real image styles manifold covered by the synthetic image style manifold as defined by k-nearest neighbor ball. If the style variation is small, then r_k becomes small, reducing the chance of $d(\mathbf{v}^c_i, \hat{\mathbf{v}}^c_j) < r_k$. *In our main text:* (1) In Fig. 7, we measure intra-class style variation by intra-class eIR and dataset variability by dataset eIR. (2) In Tab. 3, we report intra-class eIR for each setting.

C.4. Frequency Variances

It measures the diversity across different frequency components. Figure 14 explains its calculation, where images

Figure 15. Comparison of synthetic images with and without replacing their shape attributes with ground-truth shape during style sampling. Shape replacement offers synthetic images with improved visual similarity to the reference image.

Figure 16. Alternative methods for style conditioning. Directly using style attributes as context fails to control style due to lacking pixel-aligned details. Concatenating style feature maps to image channels produces artifacts when incorporating blending.

are converted into frequency spectrum via FFT and partitioned into different frequency components, and the variance of each component is measured. Higher variances reflect (though not in a decisive manner) more informative frequency components, hence better identity consistency and style variation. *In our main text*, it is exhibited in Fig. 9(a).

D. Additional Experimental Studies

D.1. Shape Attribute Replacement

As discussed in Sec. 3.3, we sample style attributes (*i.e.*, facial shape, expression, pose, texture and illumination) from a real-world prior distribution. Then, we replace the intra-class mean of facial shape attributes with the reference image's ground-truth shape. In Fig. 15, we compare synthetic images with and without replacing their shape attributes during style sampling. Shape replacement offers synthetic images with better visual similarity to the reference image. Experimentally, this improves average FR accuracy by 0.22.

D.2. Alternation for Style Conditioning

We proposed to condition style from 3DMM renderings using cross-attention. We study 2 alternatives: (1) Directly using style attributes $\mathbf{p}$ as $\mathbf{c}_{sty}$, and (2) Concatenating style feature maps $\mathbf{m}$ to the image channels. From Fig. 16, we observe that the first approach provides ineffective style control due to a lack of pixel-aligned details. Though the second approach controls style, we find it incompatible with context blending (as CFG ineffectively learns empty feature maps) and may introduce artifacts.

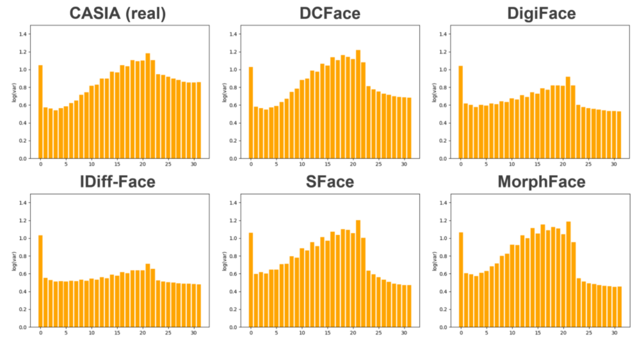

Figure 17. Comparison with SOTAs on frequency variances. Our proposed method, DCFace and SFace exhibit informative frequency components similar to CASIA. The analyses match the quantitative eIR results in Fig. 7(a).

	Strategy	eIR	cos-sim	FR Avg.
	W/o blending	0.608	0.37	93.11
t_0	750	0.617	**0.48**	93.23
	250	**0.675**	0.38	93.18
	500 (Proposed)	0.642	0.45	**93.32**
w	1	**0.684**	**0.51**	93.05
	0.25	0.613	0.37	93.14
	0.5 (Proposed)	0.642	0.45	**93.32**

Table 4. Choices of shifting timesteps t_0 and CFG weight w.

D.3. Comparison on Frequency Variance

In Fig. 17, we measure the intra-class variances of frequency components for the real-world CASIA dataset, several SOTAs, and our proposed MorphFace. This is an extension of Fig. 9(a) of our main text. We highlight: (1) MorphFace, DCFace and SFace exhibit informative frequency components similar to CASIA, while DigiFace and SFace exhibit less informative components. (2) The frequency analyses match the quantitative eIR results in Fig. 7(a), where MorphFace, DCFace and SFace have higher intra-class eIR. This also demonstrates the reasonableness of frequency analyses.

D.4. Choice of Blending Parameters

We study the impact of choosing different shifting timesteps t_0 and CFG weight w during context blending on synthesizing quality and FR efficacy. Results are summarized by eIR, cosine similarity and average FR accuracy in Tab. 4. We highlight: (1) Choosing larger/smaller t_0 strengthens the impact of identity/style contexts, leading to increased cosine similarity/eIR, respectively. They both suffer a slight accuracy drop, suggesting the importance of balancing be-

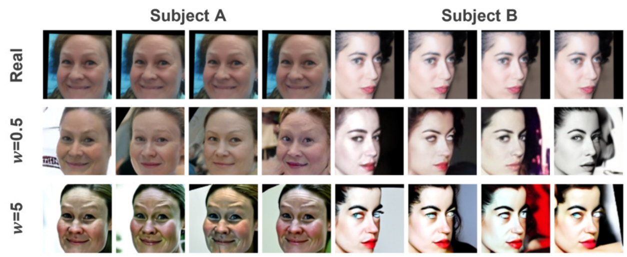

Figure 18. Sample images from different CFG weight w. A too-intensive weight (*e.g.*, 5) could produce less realistic images that downgrade FR efficacy.

Figure 19. Sample DECA feature maps and their synthetic images.

tween contexts. The drop however is slight and both settings outperform the non-blending baseline, demonstrating the effectiveness of our proposed technique. (2) By choosing a smaller w=0.25, context blending is too inconspicuous to affect performance. (3) Choosing a larger w=1 increases both eIR and cosine similarity. Interestingly, this negatively impacts FR efficacy. In Fig. 18, we find an intensive w (*e.g.*, a very large 5) could generate less realistic images, which explains the accuracy downgrade. This suggests that a moderate w should be chosen for context blending. We leave its improvement in future studies.

E. Visualizations

In Fig. 19, we provide sample DECA feature maps $\mathbf{m}$ and their synthetic images. As discussed in Secs. 3.2 and 3.3, the feature maps are rendered from style attributes $\mathbf{p}'$, whose expression, pose, texture and illumination are randomly sampled from a real-world prior distribution and shape comes from the reference image. Figure 19 is an extension of Fig. 3 in our main text.

In Fig. 20, we provide additional sample images from MorphFace, where intra-class style variation and subject distinctiveness can both be observed. The 0.5/1.2M synthetic datasets will be later released for public access.

F. Miscellaneous

Code and dataset. The code and synthetic datasets will be available at https://github.com/Tencent/TFace/.

Figure 20. Additional sample images from MorphFace.

References

[1] James Booth, Anastasios Roussos, Allan Ponniah, David Dunaway, and Stefanos Zafeiriou. Large scale 3d morphable models. *International Journal of Computer Vision*, 126(2): 233–254, 2018. 5, 2

[2] Fadi Boutros, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Elasticface: Elastic margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 1578–1587, 2022. 2, 4, 1

[3] 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. 3

[4] 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. 2, 1

[5] Ionut Cosmin Duta, Li Liu, Fan Zhu, and Ling Shao. Improved residual networks for image and video recognition. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 9415–9422. IEEE, 2021. 1

[6] Yao Feng, Haiwen Feng, Michael J Black, and Timo Bolkart. Learning an animatable detailed 3d face model from in-the-wild images. *ACM Transactions on Graphics (ToG)*, 40(4): 1–13, 2021. 2, 3, 4, 1

[7] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 5, 2

[8] Minchul Kim, Anil K. Jain, and Xiaoming Liu. Adaface: Quality adaptive margin for face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18750–18759, 2022. 2, 1

[9] Minchul Kim, Feng Liu, Anil Jain, and Xiaoming Liu. Dcface: Synthetic face generation with dual condition diffusion model. In *Proceedings of the ieee/cvf conference on com-*

puter vision and pattern recognition, pages 12715–12725, 2023. 1, 2, 3, 4, 5, 6, 7, 8

[10] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 1

[11] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in neural information processing systems*, 32, 2019. 7, 3

[12] Gary B. Huang Erik Learned-Miller. Labeled faces in the wild: Updates and new reporting procedures. Technical Report UM-CS-2014-003, University of Massachusetts, Amherst, 2014. 6, 1

[13] Tianye Li, Timo Bolkart, Michael J Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4d scans. *ACM Trans. Graph.*, 36(6):194–1, 2017. 2, 3, 5, 1

[14] Stylianos Moschoglou, Athanasios Papaioannou, Christos Sagonas, Jiankang Deng, Irene Kotsia, and Stefanos Zafeiriou. Agedb: the first manually collected, in-the-wild age database. In *proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 51–59, 2017. 2, 6, 1

[15] Pascal Paysan, Reinhard Knothe, Brian Amberg, Sami Romdhani, and Thomas Vetter. A 3d face model for pose and illumination invariant face recognition. In *2009 sixth IEEE international conference on advanced video and signal based surveillance*, pages 296–301. Ieee, 2009. 5, 2

[16] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2, 3, 1

[17] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015. 3, 2

[18] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016. 1

[19] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016. 3

[20] Soumyadip Sengupta, Jun-Cheng Chen, Carlos Castillo, Vishal M Patel, Rama Chellappa, and David W Jacobs. Frontal to profile face verification in the wild. In *2016 IEEE winter conference on applications of computer vision (WACV)*, pages 1–9. IEEE, 2016. 2, 6, 1

[21] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li. Learning face representation from scratch. *arXiv preprint arXiv:1411.7923*, 2014. 2, 5, 6, 8, 1

[22] Tianyue Zheng and Weihong Deng. Cross-pose lfw: A database for studying cross-pose face recognition in unconstrained environments. *Beijing University of Posts and Telecommunications, Tech. Rep*, 5(7):5, 2018. 2, 6, 1

[23] Tianyue Zheng, Weihong Deng, and Jiani Hu. Cross-age lfw: A database for studying cross-age face recognition in unconstrained environments. *arXiv preprint arXiv:1708.08197*, 2017. 2, 6, 1