# DigiFace-1M: 1 Million Digital Face Images for Face Recognition

Gwangbin Bae
University of Cambridge
gb585@cam.ac.uk

Martin de La Gorce
Microsoft
madelago@microsoft.com

Tadas Baltrušaitis
Microsoft
tabaltru@microsoft.com

Charlie Hewitt
Microsoft
chewitt@microsoft.com

Dong Chen
Microsoft
doch@microsoft.com

Julien Valentin
Microsoft
juvalen@microsoft.com

Roberto Cipolla
University of Cambridge
rc10001@cam.ac.uk

Jingjing Shen
Microsoft
jinshen@microsoft.com

## Abstract

*State-of-the-art face recognition models show impressive accuracy, achieving over 99.8% on Labeled Faces in the Wild (LFW) dataset. Such models are trained on large-scale datasets that contain millions of real human face images collected from the internet. Web-crawled face images are severely biased (in terms of race, lighting, make-up, etc) and often contain label noise. More importantly, the face images are collected without explicit consent, raising ethical concerns. To avoid such problems, we introduce a large-scale synthetic dataset for face recognition, obtained by rendering digital faces using a computer graphics pipeline[1]. We first demonstrate that aggressive data augmentation can significantly reduce the synthetic-to-real domain gap. Having full control over the rendering pipeline, we also study how each attribute (e.g., variation in facial pose, accessories and textures) affects the accuracy. Compared to Syn-Face, a recent method trained on GAN-generated synthetic faces, we reduce the error rate on LFW by 52.5% (accuracy from 91.93% to 96.17%). By fine-tuning the network on a smaller number of real face images that could reasonably be obtained with consent, we achieve accuracy that is comparable to the methods trained on millions of real face images.*

## 1. Introduction

Learning-based face recognition models [29, 23, 33, 35, 8, 15, 24, 18] use Deep Neural Networks (DNNs) to encode the given face image into an embedding vector of fixed di-

mension (e.g., 512). These embeddings can then be used for various tasks, such as face identification (who is this person) and verification (are they the same person). To learn diverse, discriminative embeddings, the training dataset should contain a large number of unique identities. To learn robust embeddings, i.e., which are not sensitive to the changes in pose, expression, accessories, camera and lighting, the dataset should also contain a sufficient number of images per identity with these variations.

Publicly available face recognition datasets satisfy both. MS1MV2 [8] contains 5.8M images of 85K identities (approx. 68 images per ID). Recently released Web-Face260M [43] contains 260M images of 4M identities (approx. 65 images per ID). While such datasets have driven recent advances in face recognition models, there are several problems associated with them.

**(1) Ethical issues.** Large-scale face recognition datasets are often criticized for ethical issues including privacy violation and the lack of informed consent. For example, datasets like [39, 12, 8, 43] are obtained by crawling web images of celebrities *without consent*. To increase the number of identities, some datasets exploited the term "celebrities" to include anyone with online presence. Datasets like [17, 26] collected face images of the *general public* (including children) from Flickr [3]. Projects like MegaPixels [4] are exposing the ethical problems of such web-crawled face recognition datasets. Following severe criticism, public access to several datasets has been removed [2].

**(2) Label noise.** Web images collected by searching the names of celebrities often contain label errors. For example, the Labeled Faces in the Wild (LFW) dataset [14] contains several known errors including: (1) mislabeled images; (2) distinct persons with the same name labeled as the same per-

---

[1]DigiFace-1M dataset can be downloaded from https://github.com/microsoft/DigiFace1M

Figure 1. Examples of synthetic face images in our dataset. Our dataset captures a wide variety of facial geometry, pose, textures, expressions, accessories and environments.

son; and (3) the same person that goes by different names labeled as different persons.

**(3) Data bias.** Face recognition models are generally trained and tested on celebrity faces, many of which are taken with strong lighting and make-up. Celebrity faces also have imbalanced racial distribution (e.g., 84.5% of the faces in CASIA-WebFace [39] are Caucasian faces [34]), leading to poor recognition accuracy for the under-represented racial groups [34].

In order to circumvent all these issues that affect the existing real face datasets, we introduce a new large-scale face recognition dataset consisting only of photo-realistic *digital face images* rendered using a computer graphics pipeline and make this dataset available to the community. Specifically, we build upon the face generation pipeline introduced by Wood et al. [36], tailoring the amount of variability for each attribute (e.g., pose and accessories) for our recognition task, and generate 1.22M images with 110K unique identities. Each identity is generated by randomizing the facial geometry and texture as well as the hair style. The generated face is then rendered with different poses, expressions, hair color, hair thickness and density, accessories (including clothes, make-ups, glasses, and head/face wear), cameras and environments, to encourage the network to learn a robust embedding. Figure 1 shows examples of synthetic face images in this new dataset. We generated 1.22M images, but in practice the number of identities and images you can generate with synthetics pipeline is only limited by the cost of generating and storing these images.

Digital synthetic faces can solve the aforementioned problems associated with the real face datasets. Firstly, the generated faces are free of label noise. Secondly, the bias in lighting, make-up and skin color can be reduced as we have full control over those attributes. Most importantly, the face generation pipeline does not rely on any privacy-sensitive data obtained without consent.

This is a critical difference from the GAN-generated synthetic faces; face GANs rely (either directly or indirectly) on large-scale real face datasets to train some components of their pipeline, leaving unresolved ethical problems. For ex-

ample, a recent method called SynFace [28] was trained on synthetic faces generated using DiscoFaceGAN [9]. While the generated face images are free of label noise, millions of real face images were used for training DiscoFaceGAN. The GANs may also inherit any bias that exists in the real face images used to train them. For our dataset, only 511 face scans, *obtained with consent*, were used to build a parametric model of face geometry and texture library [36]. From this limited source data, we can generate infinite number of identities, making our approach easily scalable.

Our contributions can be summarized as below:

- We release a new large-scale *synthetic* dataset for face recognition that is free from privacy violations and lack of consent. To the best of our knowledge, our dataset, containing 1.22M images of 110K identities, is the largest public synthetic dataset for face recognition.

- Compared to SynFace [28], which is trained on GAN-generated faces, we reduce the error rate on LFW by 52.5% (accuracy from 91.93% to 96.17%). For five popular benchmarks [14, 30, 41, 25, 42], the average error rate is reduced by 46.0% (accuracy from 74.75% to 86.37%).

- We demonstrate how the proposed synthetic dataset can be used in conjunction with a small number of real face images to substantially improve the accuracy. This simulates a scenario where a small number of curated (i.e., no label noise and reduced bias) real face images are collected with consent. By fine-tuning our network with only 120K real face images (i.e., 2% of the commonly-used MS1MV2 dataset [8]), we achieve 99.33% accuracy on LFW and 93.61% on average across the five benchmarks, which is comparable to the methods trained on millions of real face images.

- Having full control over the rendering pipeline, we perform extensive experiments to study how each attribute (e.g., variation in facial pose, accessories and textures) affects the face recognition accuracy.

## 2. Related Work

**Face recognition datasets with real face images.** Major tech companies can utilize private data to train their face recognition models. Google used 100M-200M images of 8M identities to train FaceNet [29], and Facebook used 500M images of 10M identities [31]. It is challenging to construct datasets of comparable size using face images that are publicly available. Public datasets generally rely on celebrity images [14, 39, 12, 43] or web images that are posted with Creative Commons license [17, 26]. As discussed in section 1, such datasets have ethical issues and suffer from label noise and data bias.

**Synthetic faces generated using deep generative models.** Deep generative models such as GANs [11] can produce photo-realistic images and have been used to generate synthetic data to train face recognition [32, 28]. While traditional generators (e.g., [16]) generate a face image from a single latent vector that changes both the identity and its appearance, DiscoFaceGAN [9] learned *disentangled* latent representations for identity, pose, expression and illumination. SynFace [28] used DiscoFaceGAN to generate a synthetic dataset for face recognition, consisting of 10K identities and 500K images. SynFace achieved 91.93% accuracy on LFW dataset [14], and by mixing the synthetic dataset with 2K real identities (20 images each), the accuracy was pushed up to 97.23%. However, their performance is poor for large-pose-variation datasets (e.g., 75.03% on CFP-FP [30] and 70.43% on CPLFW [41]). This is mainly because it is challenging to train a 2D GAN to produce images that preserve 3D geometric consistency [10].

**Synthetic faces generated using 3D parametric models.** Classical 3D parametric face models such as morphable models [5] explicitly model the identity independently from other parameters which makes them well suited for generating face recognition datasets. However, previous results obtained with this kind of synthetic images have shown limited performance [20, 19] unless combined with a large number of real images. This can be due to the lack of realism and variability in the models that have been used to generate the faces. Wood et al. [36] introduced a pipeline for generating and rendering diverse and photo-realistic 3D face models. A generative face model, learned from the 3D scans of 511 individuals, is used to generate a random 3D face. The face is then combined with artist-created assets (e.g., texture, hair, accessories) and is rendered under a random environment (simulated with HDRIs - high dynamic range images). The rendered synthetic face images (and the corresponding auto-generated ground truth annotations) were used to learn various face analysis tasks such as face parsing [36], landmark localization [36, 37] and face reconstruction [37], demonstrating state-of-the-art performance. In this paper, we aim to demonstrate that such photo-realistic rendered synthetic faces can be used to tackle face recognition.
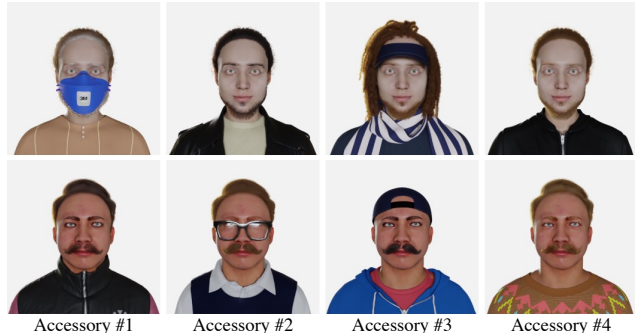


Figure 2. Each row shows the same identity rendered with different accessory setups. Accessories include clothes, glasses, make-up (e.g., eyeshadow and eyeliner), face-wear and head-wear. The color, density and thickness of facial and head hair are also randomized. The hair style is modified only when the sampled accessory conflicts with the original hair style.
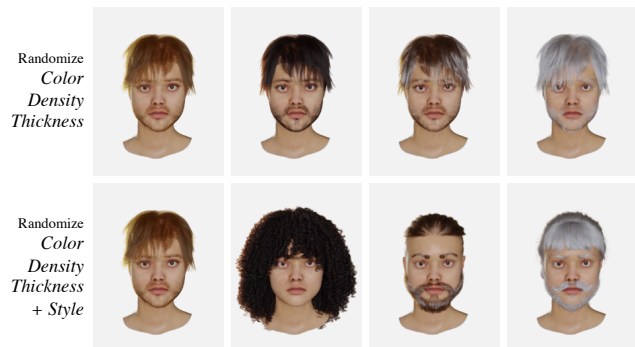


Figure 3. Randomizing the hair style makes the problem unnecessarily difficult (see the bottom row), as most people maintain similar hair styles. Therefore, we only randomize the color, density and thickness of the hair as shown in the top row (the hair is also randomly flipped horizontally).

## 3. Digital Faces for Face Recognition

This section explains how the proposed dataset is generated. We first explain how digital faces are controlled, rendered and aligned to create the dataset (subsection 3.1). After providing the dataset statistics (subsection 3.2), we introduce the data augmentation details which help in minimizing the synthetic-to-real domain-gap (subsection 3.3).

### 3.1. Face Rendering

We build upon the face generation and rendering pipeline introduced by Wood et al. [36]. In this section, we explain the modifications we made to the original pipeline to create a large-scale dataset for face recognition.

We define identity as a unique combination of facial geometry, texture (albedo and displacement), eye color and hair style. For each identity, we render a number of im-

Neutral View              Rendered Views

Figure 4. Examples of images rendered for the same identity and accessory setup. The same face can look very different depending on the pose, expression, environment (lighting and background) and camera, encouraging the network to learn robust embedding.

ages where all other parameters are varied to encourage the network to learn robust embeddings. While hair style *can* change for an individual, most people maintain similar hair style (for both facial and head hair) which makes hair style an important cue for the person's identity. Consequently, for the same identity, we randomize only the color, density and thickness of the hair (see Figure 3 for examples), and the hair style is only changed when the added head-wear is not compatible with the original hair style to avoid intersection (e.g., third image of top row in Figure 2). For sampling facial geometry, texture and eye color we follow [36].

For a given identity, we sample different accessories including clothing, make-up, glasses, face-wear (e.g., face masks) and head-wear (e.g., hats). After selecting the clothing randomly from the digital wardrobe, other accessories are added with probability $p = \{0.15, 0.15, 0.01, 0.15\}$ respectively. We also add hands and secondary faces with a small probability ($p = 0.01$) to simulate the case when (1) the face is occluded by hands and when (2) there are multiple faces in the image. Figure 2 shows examples of the sampled identities rendered with different sets of accessories.

For each accessory setup, we vary the pose, expression, camera and environment (lighting and background) to render multiple images. The camera is rotated around the face, both horizontally and vertically. Horizontal angle is sampled from a truncated zero-mean normal distribution with support $\theta_{\text{hori}} \in [-90°, 90°]$. The variance is set such that the probability density $p(\theta_{\text{hori}} = 90°)$ equals to $10^{-3} \times p(\theta_{\text{hori}} = 0°)$. Vertical angle is sampled from a similar truncated normal distribution with support $\theta_{\text{vert}} \in [-30°, 30°]$ and $p(\theta_{\text{vert}} = 30°) = 10^{-3} \times p(\theta_{\text{vert}} = 0°)$. This allows us to render a wide range of poses while making sure that frontal views are rendered more often. Lastly, the face is randomly translated within the viewing frustum to add additional perspective distortion. For pose, expression, and environment sampling, we follow [36]. Figure 4 shows the impact of varying the pose, expression, environment and camera for the same identity and accessory setup.

**Face alignment.** The input to the face embedding network should be an aligned crop around the face. Instead of de-



Rendered Image   Landmarks   Aligned Image   Rendered Image   Landmarks   Aligned Image

Figure 5. For synthetic faces, it is trivial to extract the locations of ground-truth facial landmarks (e.g., eyes, nose-tip and mouth corners) and align the crop around the face. This enables robust face alignment, even when some of the landmarks are not visible.

tecting facial landmarks using pre-trained DNNs (such as MTCNN [40] and RetinaFace [7]), we align the faces using the *ground truth* landmarks (see Figure 5), which enable robust alignment even when some landmarks are not visible.

**Limitations.** The face generation pipeline [36] we build upon has a number of limitations resulting in domain-gap to real face images. Particularly relevant to face recognition is that we cannot generate the same person at different ages. While we simulate aging to some extent by randomizing the color, density and thickness of the hair (as hair typically becomes grayer, sparser and thinner during aging), more work should be done to faithfully simulate aging. Lack of coverage (e.g., no jewelry and tattoos) may also mean that the distribution of the synthetic data does not match reality.

### 3.2. Dataset Statistics

The proposed dataset consists of two parts. The **first part** contains 720K images with 10K identities. For each identity, 4 different sets of accessories are sampled and 18 images are rendered for each set (i.e., 72 images-per-identity). Since many views of the same face are available, the network can learn embedding that is robust to the changes in accessories, camera, pose, expression, and environment. The **second part** contains 500K images with 100K identities. For each identity, only one set of accessories is sampled and only 5 images are rendered. This part was added to substantially increase the total number of identities with small rendering cost. Ensuring sufficient number of identities is important since the network should learn to distinguish between similar-looking faces of different identities. We show in the experiments that mixing the two parts leads to better accuracy than using one of them (Table 3).

### 3.3. Data Augmentation

The quality of in-the-wild face images can vary significantly. Certain parts of the face may be occluded, and the images are subject to distortion and noise that are specific to each camera. As our synthetic faces are rendered with controlled quality using a perfect pinhole camera, aggressive data augmentation is needed to reduce the synthetic-to-real domain-gap. We first apply random horizontal flipping and cropping, following [18]. Then, we apply two sets of augmentations - appearance and warping. Figure 6 shows train-

Figure 6. Synthetic face images at different stages of data augmentation. Aggressive augmentation helps to simulate effects such as motion blur and distortion common in real-world images and thus improve the robustness of DNNs trained on synthetic images.

ing images with these augmentations. Note that we apply the data augmentation on-the-fly during training, i.e., each epoch sees different random augmentations. For each type of augmentation, we indicate its probability $p$ to be applied on a sample image.

**Appearance augmentation.** We apply random Gaussian blur ($p = 0.05$) and Gaussian noise ($p = 0.035$). By applying the Gaussian blur along a random direction using an anisotropic covariance, we also simulate motion blur ($p = 0.05$). Brightness, contrast, hue and saturation are randomized with $p = \{0.15, 0.3, 0.1, 0.1\}$. Images are converted into grayscale with $p = 0.01$. Lastly, the image quality is randomized by downsampling-and-upsampling ($p = 0.01$) and JPEG compression ($p = 0.05$).

**Warping augmentation.** Warping is performed by randomly shifting the four corners of the image. Firstly, the aspect ratio is randomized with $p = 0.1$. Then, all images undergo random scaling, rotation and shift. Lastly, the four corners are shifted differently for additional distortion.

## 4. Experimental Setup

**Implementation details.** Synthetic faces are rendered using Cycles renderer [1], with 256 samples per pixel. The rendering of the full dataset took approximately 10 days, using 300 NVIDIA M60 GPUs. The images are rendered at $256 \times 256$ resolution, and the aligned crop around the face is resized into $112 \times 112$. We use ResNet-50 [13] backbone for the experiments in subsection 5.1, 5.2 and 5.3. For comparison against the state-of-the-art methods in subsection 5.4, we use their encoder architecture to ensure fair comparison. For all experiments, the networks are implemented with PyTorch [27] and are trained for 40 epochs using SGD. The batch size is set to 256 and the networks are trained on four NVIDIA P100 GPUs. We follow the learning rate scheduling of [28], and use the training loss from [18]. Note that all

networks are trained from scratch (not pre-trained on, e.g., ImageNet [6]), to make sure that no real images are used.

**Evaluation protocol.** Following state-of-the-art methods [15, 21, 24, 22, 18], we report the face verification accuracy on five benchmark datasets - LFW [14], CFP-FP [30], CPLFW [41], AgeDB [25] and CALFW [42]. LFW contains 6,000 pairs of in-the-wild face images. CFP-FP and CPLFW have *larger pose variation* (CFP-FP specifically compares frontal views to profile views). AgeDB and CALFW have *larger age variation*.

## 5. Experiments

We run a series of experiments to demonstrate the usefulness of the proposed dataset. Subsection 5.1 compares different data augmentations. In subsection 5.2, we train the network on various different subsets of the full dataset to understand how each attribute sampling in rendering affects the accuracy. In subsection 5.3, we show that our synthetic faces can be used in conjunction with a small number of real faces to substantially improve the accuracy. Lastly, we provide comparison against the state-of-the-art methods in subsection 5.4.

### 5.1. Data Augmentation

In subsection 3.3, we introduced *appearance* and *warping* augmentations. As shown in Table 1, both lead to significant improvement across all datasets. We also compare against the augmentation used by AdaFace [18], which includes horizontal flipping, cropping and *mild* color augmentation. For our synthetic face images which are free of imperfection, more aggressive data augmentation is needed to reduce the domain-gap. Notice that the warping augmentation improves the performance especially for the large-pose-variation datasets (CFP-FP and CPLFW).

### 5.2. Dataset Composition

Having full control over the rendering pipeline, we can create a dataset with desired statistics to study how each attribute affects the face recognition accuracy. The results are provided in Table 2.

**Accessory sampling.** For 10K synthetic identities, we sampled 4 accessory setups and rendered 18 images for each setup (i.e., 720K images in total). These 18 images have variations in pose, expression, camera, and environment (see Figure 4). From this, we can create a subset of 180K images by selecting 18 images per ID with *fixed accessory*. Similarly, we can select 18 images *randomly* so that images with different accessories are used during training. When randomizing the accessories, we also randomized the color, thickness and density of the hair to simulate aging (Figure 3). As a result, the accuracy is improved especially for the large-age-variation datasets (AgeDB and CALFW). For CFP-FP and CPLFW, which has smaller age gap (i.e.,

| Experiment | Method | LFW | CFP-FP | CPLFW | AgeDB | CALFW | Avg |
|---|---|---|---|---|---|---|---|
| Data augmentation | No augmentation | 88.07 | 70.99 | 66.73 | 60.92 | 69.23 | 71.19 |
| | Augmentation from AdaFace [18] | 90.12 | 76.41 | 71.33 | 67.17 | 74.13 | 75.83 |
| | Ours (appearance) | 94.32 | 80.00 | 74.83 | 75.82 | 76.92 | 80.38 |
| | Ours (appearance + warping) | **94.55** | **84.86** | **77.08** | **76.97** | **77.20** | **82.13** |

Table 1. The proposed aggressive data augmentation significantly improves the accuracy across all datasets.

| Experiment | Method | LFW | CFP-FP | CPLFW | AgeDB | CALFW | Avg |
|---|---|---|---|---|---|---|---|
| Accessory sampling | Fix accessory | 93.50 | **82.16** | **75.75** | 73.05 | 73.83 | 79.66 |
| | Randomize accessory | **94.23** | 82.04 | 75.18 | **76.43** | **77.22** | **81.02** |
| Pose sampling | Minimize horizontal angle | 93.42 | 67.19 | 66.48 | **76.78** | **77.22** | 76.22 |
| | Minimize vertical angle | 93.67 | 81.13 | 74.57 | 76.57 | 76.68 | 80.52 |
| | Random pose | **94.23** | 82.04 | 75.18 | 76.43 | **77.22** | **81.02** |
| Texture sampling (# textures to select from) | 50 | 89.63 | **75.04** | 69.72 | 69.47 | 70.10 | 74.79 |
| | 100 | **90.83** | 74.84 | **70.30** | 70.62 | **70.57** | **75.43** |
| | 150 | 90.03 | 73.01 | 69.63 | **71.48** | 70.27 | 74.89 |
| | 200 | 89.82 | 73.37 | 69.37 | 71.45 | 70.50 | 74.90 |

Table 2. Dataset composition experiments to study how the sampling of each attribute affects the accuracy.



Figure 7. Left: 40 textures selected randomly from the texture library. The library covers diverse skin color and age. Right top row: various identities (facial geometry and hair style) sampled with the same texture. Right bottom row: same identity with the same texture under different environments (taken from [36]). With large variations in geometry, hair style and environments, rich appearance variation could be achieved with limited textures.

positive pairs capture the identity at similar age), fixing the accessory and hair leads to slightly better accuracy.

**Pose sampling.** Similar to the accessory sampling, we can select 18 images for each of the 10K identities by selecting the ones with the smallest horizontal/vertical angles. Then, we can compare them against the 18 images selected randomly. For the randomly selected images, the standard deviation in horizontal and vertical angles were $(\sigma_{hori}, \sigma_{vert}) = (24.13°, 9.20°)$. For the images with the smallest horizontal/vertical angles, they were $(4.71°, 8.06°)$ and $(22.02°, 1.72°)$ respectively. As shown in Row 3-5 in Table 2, increasing the variation in horizontal and vertical angles improved the accuracy especially for the large-pose-variation datasets (CFP-FP and CPLFW). For AgeDB and CALFW, which consists mainly of *frontal* faces, the accuracy was similar.

**Texture sampling.** While we can create infinite number of unique facial geometries, the texture is sampled from a library built from 208 scans of real human faces (obtained with consent). Since we generated 110K identities in total, many of them share the same texture. To see how the number of textures affects the accuracy, we created a dataset of 1200 identities with $N$ textures, by generating $1200/N$ identities for each texture. As shown in Row 6-9 of Table 2, increasing the number of textures did not lead to a meaningful improvement in the accuracy. This is contrary to the intuition that small number of textures and lack of texture generative model are limitations of synthetic data for face recognition. We believe that the appearance variability is a combination of geometry, texture, hair, accessories, environment and image quality. In Figure 7, we show that (1) the texture library already covers diverse skin color and age, (2) an arbitrary number of unique identities can be generated with the same texture, and (3) skin appearance is greatly affected by the environment. Also, the image quality for face recognition task is in general limited due to low resolution and data augmentation. Thus, the contribution of texture variation is likely less important than that of geometry and environment.

**Balance between # IDs and # images/ID.** Ensuring large number of IDs is important for learning diverse discriminative embedding. On the other hand, large number of images per ID (referred to as *images/ID*) is needed for learning robust embedding (that is not affected by the changes in pose, accessories, expressions, camera and environment). Mixing the two datasets with different number of images/ID can be considered as an efficient way of getting the best of both. This also simulates the long-tailed distribution of real face datasets (i.e., most identities have small number of images). The result in Table 3 shows that mixing the two datasets leads to better accuracy than using one of them.
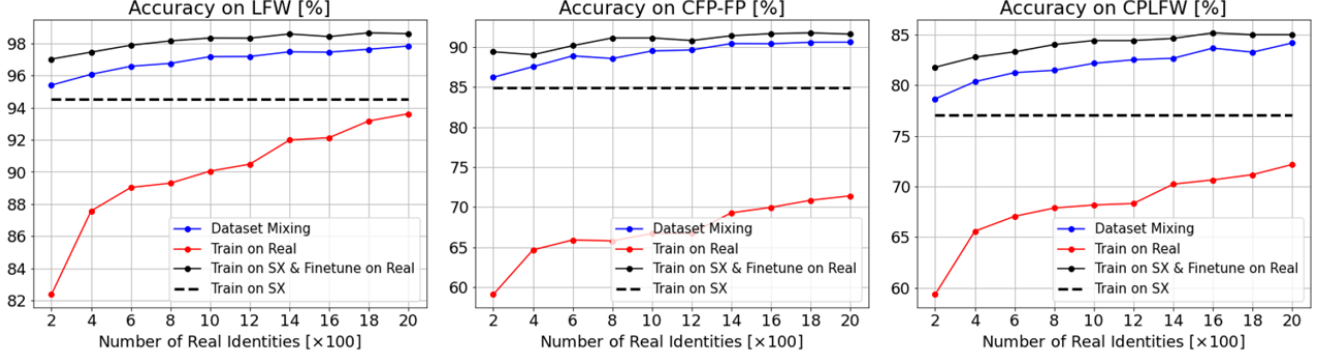
Figure 8. Comparison between training with our synthetic data only (black dashed line), with small amount of real data only (red line), with the mixture of the two (blue line), and pre-training on synthetics and fine-tuning with the real data (black line). The number of real identities varies from 200 to 2000, and 20 images are sampled for each identity. When only a small number of real face images are available (e.g., due to ethical issues), the proposed synthetic dataset can substantially improve the accuracy.

| # IDs × # images/ID | LFW | CFP-FP | CPLFW | AgeDB | CALFW | Avg |
|---|---|---|---|---|---|---|
| 10K × 50 + 0 × 5 | 94.38 | 84.07 | 76.53 | 75.93 | 76.72 | 81.53 |
| 8K × 50 + 20K × 5 | 94.80 | 84.79 | 77.52 | 76.47 | 77.65 | 82.24 |
| 6K × 50 + 40K × 5 | 95.22 | **85.24** | 77.15 | 77.52 | 78.32 | 82.69 |
| 4K × 50 + 60K × 5 | **95.45** | 84.83 | 77.70 | **77.68** | **79.10** | **82.95** |
| 2K × 50 + 80K × 5 | 94.82 | 84.09 | **77.75** | 77.55 | 78.37 | 82.51 |
| 0 × 50 + 100K × 5 | 94.45 | 83.34 | 76.77 | 76.33 | 77.28 | 81.64 |

Table 3. Number of IDs and number of images/ID should both be high to learn diverse and robust embedding. Mixing two datasets with large/small number of images/ID can be an efficient way of satisfying both. The overall accuracy becomes higher than relying on one of the two datasets.

## 5.3. Mixing with Real Faces

The main problems associated with large-scale real face datasets are *ethical issues*, *label noise* and *data bias*. In this study, we assume a scenario where a small number of real face images are collected with consent. For small number of images, it would also be possible to remove (or reduce) the label noise and data bias.

For synthetic data, we used 10K identities with 72 images per identity. For real face images, we varied the number of identities from 200 to 2000, with 20 images sampled for each identity (the identities and images were sampled randomly from CASIA-WebFace [39]).

We first tried training only on the synthetic data. Secondly, we tried training only on the real data. Then, we explored two different strategies for using both real and synthetic images: (1) dataset mixing and (2) pre-training on synthetic data and fine-tuning on the real data. For fine-tuning, we reduced the learning rate by 1/10 for the prediction head, and 1/100 for the encoder to avoid catastrophic forgetting. The results are provided in Figure 8.

When the network is trained only on a small number of real face images, the accuracy is worse than the network trained only on our synthetic dataset. Both dataset mixing and pre-training can lead to significantly higher accuracy, especially for the large-pose-variation datasets (CFP-FP and CPLFW). Compared to dataset mixing, pre-training on synthetics followed by fine-tuning on real images led to better accuracy. This can be due to the imbalance between the number of images (we use 720K synthetic images, and a lot fewer real images).

## 5.4. Comparison to the State-of-the-Art

**Comparison to SynFace.** SynFace [28] is the current state-of-the-art for *face recognition model trained on synthetic faces*. They used DiscoFaceGAN [9] to generate 500K synthetic faces (10K identities & 50 images/ID). To ensure a fair comparison, we trained the same encoder (LResNet50E-IR) with same number of images. We also trained using our full dataset (1.22M images). The results are provided in Row 1-3 of Table 4. In the second scenario, we additionally used 40K real face images from CASIA-WebFace [39]. While SynFace *mixed* their synthetic dataset with the real faces, we instead adopted the two-stage method of pre-training and fine-tuning as discussed in subsection 5.3. The results are provided in Row 4-6 of Table 4.

For both scenarios, we significantly outperform SynFace across all datasets. This suggests that our *rendered* synthetic faces are better than *GAN-generated* faces for learning face recognition. While GANs like [9] can generate realistic face images, the data they generate is not ideal for face recognition, due to following reasons: **(1) Identity change.** While [9] is *encouraged* to preserve the identity when changing other latent variables, there is no guarantee that the identity will be preserved during data generation. **(2) Geometric inconsistency.** As pointed out by [10], the images generated

| Method | # Synthetic images (# IDs × # imgs/ID) | # Real images (# IDs × # imgs/ID) | LFW | CFP-FP | CPLFW | AgeDB | CALFW | Avg | Avg† |
|---|---|---|---|---|---|---|---|---|---|
| SynFace [28] | 500K (10K×50) | 0 | 91.93 | 75.03 | 70.43 | 61.63 | 74.73 | 74.75 | 79.13 |
| Ours | 500K (10K×50) | 0 | 95.40 | 87.40 | 78.87 | 76.97 | 78.62 | 83.45 | 87.22 |
| Ours | 1.22M (10K×72+100K×5) | 0 | **95.82** | **88.77** | **81.62** | **79.72** | **80.70** | **85.32** | **88.74** |
| SynFace [28] | 500K (10K×50) | 40K (2K×20) | 97.23 | 87.68 | 80.32 | 81.42 | 85.08 | 86.35 | 88.41 |
| Ours | 500K (10K×50) | 40K (2K×20) | 99.05 | 94.01 | 87.27 | 89.77 | 90.08 | 92.04 | 93.44 |
| Ours | 1.22M (10K×72+100K×5) | 40K (2K×20) | **99.17** | **94.63** | **88.10** | **90.50** | **90.97** | **92.67** | **93.97** |

Table 4. Comparison to SynFace using the same encoder architecture (LResNet50E-IR [28]). For both scenarios - training only on synthetic faces & using a small number of real faces - we significantly outperform SynFace across all datasets. Avg† shows average of LFW, CFP-FP and CPLFW, excluding the large-age-variation datasets.

| Method | # Synthetic images | # Real images | LFW | CFP-FP | CPLFW | AgeDB | CALFW | Avg | Avg† |
|---|---|---|---|---|---|---|---|---|---|
| Ours (SX best) | 1.22M | 0 | 96.17 | 89.81 | 82.23 | 81.10 | 82.55 | 86.37 | 89.40 |
| Ours (SX+Real best) | 1.22M | 120K | 99.33 | 95.93 | 89.47 | 91.55 | 91.78 | 93.61 | 94.91 |
| SV-AM-Softmax [35] | | | 99.50 | 95.10 | 89.48 | 95.68 | 94.38 | 94.83 | 94.69 |
| SphereFace [23] | | | 99.67 | 96.84 | 91.27 | 97.05 | 95.58 | 96.08 | 95.93 |
| CosFace [33] | 0 | 5.8M | 99.78 | 98.26 | 92.18 | **98.17** | **96.18** | 96.91 | 96.74 |
| ArcFace [8] | | | 99.81 | 98.40 | 92.72 | 98.05 | 95.96 | 96.99 | 96.98 |
| MagFace [24] | | | **99.83** | 98.46 | 92.87 | **98.17** | 96.15 | 97.10 | 97.05 |
| AdaFace [18] | | | 99.82 | **98.49** | **93.53** | 98.05 | 96.08 | **97.19** | **97.28** |

Table 5. Comparison to the state-of-the-art methods trained on real face images (MS1MV2 [8]). We use the same backbone (ResNet100) for fair comparison. By only using 120K real face images (2% of MS1MV2 [8]), we achieve accuracy that is comparable to the methods trained on millions of real face images. Since we do not model aging explicitly, our accuracy is worse for large-age-variation datasets (AgeDB and CALFW). Avg† shows average of LFW, CFP-FP and CPLFW, and on these, we outperform [35] and are similar to [23].

by [9] for same identity and different poses lack 3D consistency. **(3) Lack of accessory change.** [9] cannot randomize accessories. **(4) Unresolved ethical concerns.** Training the GAN model itself requires large-scale real face dataset. For example, 70K images are used to train [9]. To learn to preserve identity, they also used a perceptual loss based on [38], which is trained on 3M real face images.

In Row 2 and 3 of Table 4, we increase our synthetics dataset size from 500K to 1.22M, and achieve better accuracy. This indicates that the accuracy may not have converged yet and could be improved further by generating more synthetic data.

**Comparison to methods trained on real faces.** Lastly, we compare the accuracy against the methods that are trained on real face images. In Table 5, we provide the accuracy of six methods that use ResNet100 as the embedding network and MS1MV2 [8] as the training data. We trained the same architecture on our synthetic dataset (Row 1). We also tried fine-tuning the network on a small number of real face images (Row 2). When trained only with the proposed synthetic dataset, the network can achieve 96.17% on LFW. For LFW, CFP-FP and CPLFW (excluding the high-age-variation datasets), the average accuracy is 89.40%. By fine-tuning the network on just 120K images (2.0% of MS1MV2), the accuracy becomes comparable to the methods trained on MS1MV2 (e.g., average accuracy on LFW, CFP-FP and CPLFW becomes higher than that of SV-AM-

Softmax [35]).

The performance of our method on AgeDB [25] and CALFW [42] has a significantly larger gap than for the other datasets evaluated. This is expected given the lack of aging simulation in our synthetic data. We suspect that other causes of domain-gap, as described at the end of subsection 3.1, are the primary reason for the remaining performance gap for other evaluation datasets. Reducing this domain-gap remains an area of ongoing work for our synthetic data and is likely to result in improved performance for all downstream tasks, including face recognition. We leave this as future work.

# 6. Conclusion

In this paper, we introduced a new large-scale synthetic dataset for face recognition by rendering digital faces using a graphics pipeline. We ran extensive experiments to study how data augmentation and various other attributes affect the accuracy. We demonstrated that our synthetic faces are significantly better than the GAN-generated faces for learning face recognition. With a small number of real face images, we achieve accuracy that is comparable to the methods trained on millions of web-crawled face images. We hope this dataset would be a meaningful step towards developing socially responsible face recognition models that do not depend on privacy-sensitive data obtained without consent.

# References

[1] Blender foundation. cycles renderer. `https://www.cycles-renderer.org/`. Accessed: 2022-10-03.

[2] The ethical questions that haunt facial-recognition research. `https://www.nature.com/articles/d41586-020-03187-3`. Accessed: 2022-10-03.

[3] flickr creative commons. `https://www.flickr.com/creativecommons/`. Accessed: 2022-10-03.

[4] Megapixels. `https://ahprojects.com/megapixels-glassroom/`. Accessed: 2022-10-03.

[5] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In Proc. SIGGRAPH, 1999.

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009.

[7] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-shot multi-level face localisation in the wild. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[8] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[9] Yu Deng, Jiaolong Yang, Dong Chen, Fang Wen, and Xin Tong. Disentangled and controllable face image generation via 3d imitative-contrastive learning. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[10] Yu Deng, Jiaolong Yang, Jianfeng Xiang, and Xin Tong. Gram: Generative radiance manifolds for 3d-aware image generation. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2022.

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Advances in Neural Information Processing Systems (NeurIPS), 2014.

[12] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In Proc. European Conference on Computer Vision (ECCV), 2016.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[14] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In Workshop on faces in 'Real-Life' Images: detection, alignment, and recognition, 2008.

[15] Yuge Huang, Yuhan Wang, Ying Tai, Xiaoming Liu, Pengcheng Shen, Shaoxin Li, Jilin Li, and Feiyue Huang. Curricularface: adaptive curriculum learning loss for deep face recognition. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[16] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In International Conference on Learning Representations (ICLR), 2018.

[17] Ira Kemelmacher-Shlizerman, Steven M Seitz, Daniel Miller, and Evan Brossard. The megaface benchmark: 1 million faces for recognition at scale. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[18] Minchul Kim, Anil K Jain, and Xiaoming Liu. Adaface: Quality adaptive margin for face recognition. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2022.

[19] Adam Kortylewski, Bernhard Egger, Andreas Morel-Forster, Andreas Schneider, Thomas Gerig, Clemens Blumer, Corius Reyneke, and Thomas Vetter. Can synthetic faces undo the damage of dataset bias to face recognition and facial landmark detection? arXiv preprint arXiv:1811.08565, 2018.

[20] Adam Kortylewski, Bernhard Egger, Andreas Schneider, Thomas Gerig, Andreas Morel-Forster, and Thomas Vetter. Analyzing and reducing the damage of dataset bias to face recognition with synthetic data. In Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2019.

[21] Bi Li, Teng Xi, Gang Zhang, Haocheng Feng, Junyu Han, Jingtuo Liu, Errui Ding, and Wenyu Liu. Dynamic class queue for large scale face recognition in the wild. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2021.

[22] Shen Li, Jianqing Xu, Xiaqing Xu, Pengcheng Shen, Shaoxin Li, and Bryan Hooi. Spherical confidence learning for face recognition. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2021.

[23] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[24] Qiang Meng, Shichao Zhao, Zhida Huang, and Feng Zhou. Magface: A universal representation for face recognition and quality assessment. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2021.

[25] Stylianos Moschoglou, Athanasios Papaioannou, Christos Sagonas, Jiankang Deng, Irene Kotsia, and Stefanos Zafeiriou. Agedb: the first manually collected, in-the-wild age database. In Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2017.

[26] Aaron Nech and Ira Kemelmacher-Shlizerman. Level playing field for million scale face recognition. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[27] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In Advances in Neural Information Processing Systems (NeurIPS), 2019.

[28] Haibo Qiu, Baosheng Yu, Dihong Gong, Zhifeng Li, Wei Liu, and Dacheng Tao. Synface: Face recognition with synthetic data. In Proc. IEEE International Conference on Computer Vision (ICCV), 2021.

[29] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[30] 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 Proc. IEEE Winter Conference on Applications of Computer Vision (WACV), 2016.

[31] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Web-scale training for face identification. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[32] Daniel Sáez Trigueros, Li Meng, and Margaret Hartnett. Generating photo-realistic training data to improve face recognition accuracy. Neural Networks, 134:86–94, 2021.

[33] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[34] Mei Wang, Weihong Deng, Jiani Hu, Xunqiang Tao, and Yaohai Huang. Racial faces in the wild: Reducing racial bias by information maximization adaptation network. In Proc. IEEE International Conference on Computer Vision (ICCV), 2019.

[35] Xiaobo Wang, Shuo Wang, Shifeng Zhang, Tianyu Fu, Hailin Shi, and Tao Mei. Support vector guided softmax loss for face recognition. arXiv preprint arXiv:1812.11317, 2018.

[36] Erroll Wood, Tadas Baltrušaitis, Charlie Hewitt, Sebastian Dziadzio, Thomas J Cashman, and Jamie Shotton. Fake it till you make it: face analysis in the wild using synthetic data alone. In Proc. IEEE International Conference on Computer Vision (ICCV), 2021.

[37] Erroll Wood, Tadas Baltrušaitis, Charlie Hewitt, Matthew Johnson, Jingjing Shen, Nikola Milosavljevic, Daniel Wilde, Stephan Garbin, Toby Sharp, Ivan Stojiljkovic, et al. 3d face reconstruction with dense landmarks. In Proc. European Conference on Computer Vision (ECCV), 2022.

[38] Jiaolong Yang, Peiran Ren, Dongqing Zhang, Dong Chen, Fang Wen, Hongdong Li, and Gang Hua. Neural aggregation network for video face recognition. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[39] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li. Learning face representation from scratch. arXiv preprint arXiv:1411.7923, 2014.

[40] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. IEEE signal processing letters, 23(10):1499–1503, 2016.

[41] 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, 2018.

[42] 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.

[43] Zheng Zhu, Guan Huang, Jiankang Deng, Yun Ye, Junjie Huang, Xinze Chen, Jiagang Zhu, Tian Yang, Jiwen Lu, Dalong Du, et al. Webface260m: A benchmark unveiling the power of million-scale deep face recognition. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2021.