SynFace: Face Recognition with Synthetic Data

Haibo Qiu[1]*, Baosheng Yu[2]*, Dihong Gong[3], Zhifeng Li[3], Wei Liu[3], Dacheng Tao[1,2]

[1] JD Explore Academy, China   [2] The University of Sydney, Australia

[3] Tencent Data Platform, China

qiuhaibo1@jd.com, baosheng.yu@sydney.edu.au, gongdihong@gmail.com,

michaelzfli@tencent.com, wl2223@columbia.edu, dacheng.tao@gmail.com

Abstract

With the recent success of deep neural networks, remarkable progress has been achieved on face recognition. However, collecting large-scale real-world training data for face recognition has turned out to be challenging, especially due to the label noise and privacy issues. Meanwhile, existing face recognition datasets are usually collected from web images, lacking detailed annotations on attributes (e.g., pose and expression), so the influences of different attributes on face recognition have been poorly investigated. In this paper, we address the above-mentioned issues in face recognition using synthetic face images, i.e., SynFace. Specifically, we first explore the performance gap between recent state-of-the-art face recognition models trained with synthetic and real face images. We then analyze the underlying causes behind the performance gap, e.g., the poor intra-class variations and the domain gap between synthetic and real face images. Inspired by this, we devise the SynFace with identity mixup (IM) and domain mixup (DM) to mitigate the above performance gap, demonstrating the great potentials of synthetic data for face recognition. Furthermore, with the controllable face synthesis model, we can easily manage different factors of synthetic face generation, including pose, expression, illumination, the number of identities, and samples per identity. Therefore, we also perform a systematically empirical analysis on synthetic face images to provide some insights on how to effectively utilize synthetic data for face recognition.

1. Introduction

In the last few years, face recognition has achieved extraordinary progress in a wide range of challenging problems including pose-robust face recognition [5, 24, 62], matching faces across ages [15, 17, 55, 59], across modalities [13, 14, 16, 30, 31], and occlusions [40, 48, 70].

_____
*Equal contribution

Figure 1. Examples of real/synthetic face images. The first row indicates real face images from CASIA-WebFace, and the second row shows synthetic face images generated by DiscoFace-GAN [11] with the proposed identity mixup module.

Among these progresses, not only the very deep neural networks [22, 25, 29, 47] and sophisticated design of loss functions [10, 23, 32, 56, 60], but also large-scale training datasets [20, 26, 27] played important roles. However, it has turned out to be very difficult to further boost the performance of face recognition with the increasing number of training images collected from the Internet, especially due to the severe label noise and privacy issues [20, 54, 58]. For example, several large-scale face recognition datasets are struggling with the consent of all involved person/identities, or even have to close the access of face data from the website [20]. Meanwhile, many face training datasets also suffer from the long-tailed problem, *i.e.*, head classes with a large number of samples and tail classes with a few number of samples [34, 37, 71]. To utilize these datasets for face recognition, people need to carefully design the network architectures and/or loss functions to alleviate the degradation on model generalizability brought by the long-tailed problem. Furthermore, the above-mentioned issues also make it difficult for people to explore the influences of different attributes (*e.g.*, expression, pose and illumination).

To address the aforementioned issues, we explore the potentials of synthetic images for face recognition in this paper. Recently, face synthesis using GANs [18] and 3DMM [3] have received increasing attention from the computer vision community, and existing methods usually

focus on generating high-quality identity-preserving face images [2, 46, 64]. Some synthetic and real face images are demonstrated in Figure 1. However, the problem of face recognition using synthetic face images has not been well-investigated [28, 52]. Specifically, Trigueros *et al*. [52] investigated the feasibility of data augmentation with photorealistic synthetic images. Kortylewski *et al*. [28] further explored the pose-varying synthetic images to reduce the negative effects of dataset bias. Lately, disentangled face generation has become popular [11], which can provide the precise control of targeted face properties such as identity, pose, expression, and illumination, thus making it possible for us to systematically explore the impacts of facial properties on face recognition. Specifically, with a controllable face synthesis model, we are then capable of 1) collecting large-scale face images of non-existing identities without the risk of privacy issues; 2) exploring the impacts of different face dataset properties, such as the depth (the number of samples per identity) and the width (the number of identities); 3) analyzing the influences of different facial attributes (*e.g*., expression, pose, and illumination).

However, there is usually a significant performance gap between the models trained on synthetic and real face datasets. Through the empirical analysis, we find that 1) the poor intra-class variations in synthetic face images and 2) the domain gap between synthetic and real face datasets are the main reasons of the performance degradation. To address the above issues, we introduce identity mixup (IM) into the disentangled face generator to enlarge the intra-class variations of generated face images. Specifically, we use a convex combination of the coefficients from two different identities to form a new intermediate identity coefficient for synthetic face generation. Experimental results in Sec. 4 show that the identity mixup significantly improves the performance of the model trained on synthetic face images. Furthermore, we observe a significant domain gap via cross-domain evaluation: 1) training on synthetic face images and testing on real face images; 2) training on real face images and testing on synthetic face images (see more details in Sec. 3.2). Therefore, we further introduce the domain mixup (DM) to alleviate the domain gap, *i.e*., by using a convex combination of images from a large-scale synthetic dataset and a relatively small number of real face images during training. With the proposed identity mixup and domain mixup, we achieve a significant improvement over the vanilla SynFace, further pushing the boundary of face recognition performance using synthetic data. The main contributions of this paper are as follows:

- We observe a performance gap between the models trained on real and synthetic face images, which can be effectively narrowed by 1) enlarging the intra-class variations via identity mixup; 2) leveraging a few real face images for domain adaption via domain mixup.

- We discuss the impacts of synthetic datasets with different properties for face recognition, *e.g*., depth (the number of samples per identity) and width (the number of identities), and reveal that the width plays a more important role.

- We analyze the influences of different facial attributes on face recognition (*e.g*., facial pose, expression, and illumination).

2. Related Work

We first briefly introduce visual tasks using synthetic data. Then recent face synthesis and recognition methods are reviewed. Lastly, we discuss the mixup and its variants to indicate their similarities/differences with the proposed identity mixup and domain mixup.

Synthetic Data. Synthetic data for computer vision tasks has been widely explored, *e.g*., crowd counting [57], vehicle re-identification [51], semantic segmentation [6, 43, 44], 3D face reconstruction [42] and face recognition [28, 52]. According to the motivation, existing methods can be categorized into three groups: (1) It is time-consuming and expensive to collect and annotate large-scale training data [6, 42, 43, 44]; (2) It can be used to further improve the model trained on a real dataset [28, 52]; (3) It can be used to systematically analyze the impacts of different dataset attributes [28]. Among these works, [28] is the most related one to our work, while it only discusses the impacts of different head poses. Apart from facial attributes (*e.g*., pose, expression, and illumination), we also explore the impacts of the width and the depth of training dataset. Furthermore, we introduce identity mixup (IM) and domain mixup (DM) to increase the intra-class variations and narrow down the domain gap, leading to a significant improvement.

Face Synthesis. With the great success of GANs [1, 7, 18, 35, 36, 39, 41], face synthesis has received increasing attention and several methods have been proposed to generate identity-preserving face images [2, 46, 64]. Specifically, FF-GAN [64] utilizes 3D priors (*e.g*., 3DMM [3]) for high-quality face frontalization. Bao *et al*. [2] first disentangled identity/attributes from the face image, and then recombined different identities/attributes for identity-preserving face synthesis. FaceID-GAN [46] aims to generate identity-preserving faces by using a classifier (C) as the third player, competing with the generator (G) and cooperating with the discriminator (D). However, unlike exploring the identity-preserving property, generating face images from multiple disentangled latent spaces (*i.e*., different facial attributes) have not been well-investigated. Recently, DiscoFaceGAN [11] introduces a novel disentangled learning scheme for face image generation via an imitative-contrastive paradigm using 3D priors. Thus, it further enables precise control of targeted face properties such as un-
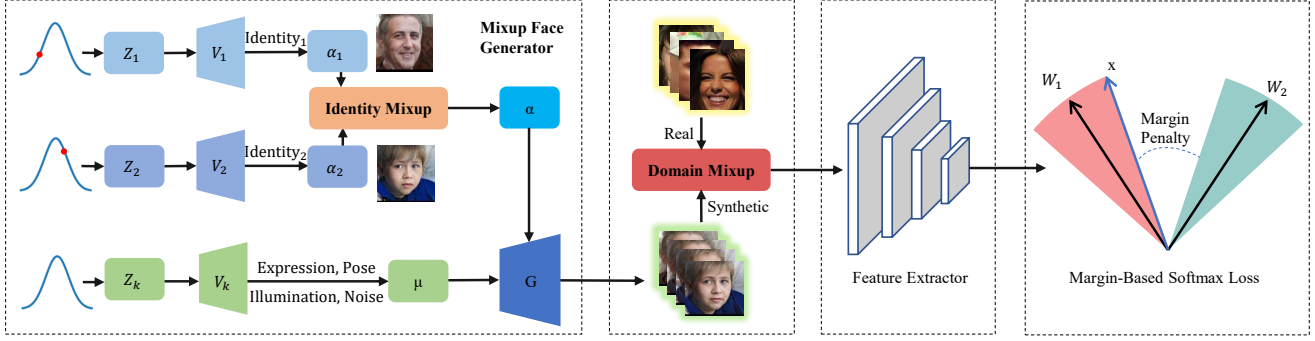
Figure 2. An overview of the proposed SynFace. Firstly, the identity mixup is introduced into DiscoFaceGAN [11] to form the Mixup Face Generator, which can generate face images with different identities and their intermediate states. Next, the synthetic face images are cooperating with a few real face images via domain mixup to alleviate the domain gap. Then, the feature extractor takes the mixed face images as input and extracts the corresponding features. The extracted features are either utilized to calculate the margin-based softmax loss (where W_1, W_2 are the center weight vectors for two different classes and x is the feature vector) for model training, or employed as the face representations to perform face identification and verification tasks.

known identities, pose, expression, and illumination, yielding the flexible and high-quality face image generation.

Deep Face Recognition. Recent face recognition methods mainly focus on delivering novel loss functions for robust face recognition in the wild. The main idea is to maximize the inter-class variations and minimize the intra-class variations. For example, 1) contrastive loss [8, 21] and triplet loss [23, 65] are usually utilized to increase the Euclidean margin for better feature embedding; 2) center loss [60] aims to learn a center for each identity and then minimizes the center-aware intra-class variations; 3) Large-margin softmax loss [32, 33] and its variants such as Cos-Face [56] and ArcFace [10] improve the feature discrimination by adding marginal constraints to each identity.

Mixup. Mixup [67] uses the convex combinations of two data samples as a new sample for training, regularizing deep neural networks to favor a simple linear behavior in-between training samples. Vanilla mixup is usually employed on image pixels, while the generated data samples are not consistent with the real images, *e.g.*, a mixup of two face images in the pixel level does not always form a proper new face image. Inspired by this, we introduce identity mixup to face generator via the identity coefficients, where a convex combination of two identities forms a new identity in the disentangled latent space. With the proposed identity mixup, we are also able to generate high-fidelity face images correspondingly. Recently, several mixup variants have been proposed to perform feature-level interpolation [19, 49, 50, 53], while [61] further leverages domain mixup to perform adversarial domain adaptation. Inspired by this, we perform domain adaption via domain mixup between real and synthetic face images, while the main difference is that [61] uses the mixup ratio to guide the model training, but we utilize the identity labels of both synthetic and real face images as the supervision for face recognition.

3. Method

In this section, we introduce face recognition with synthetic data, *i.e.*, SynFace, and the overall pipeline is illustrated in Figure 2. We first introduce deep face recognition using margin-based softmax loss functions. We then explore the performance gap between the models trained on synthetic and real datasets (SynFace and RealFace). Lastly, we introduce 1) identity mixup to enlarge the intra-class variations and 2) domain mixup to mitigate the domain gap between synthetic and real faces images.

3.1. Deep Face Recognition

With the great success of deep neural networks, deep learning-based embedding learning has become the mainstream technology for face recognition to maximize the inter-class variations and minimize the intra-class variations [8, 21, 23, 33]. Recently, margin-based softmax loss functions have been very popular in face recognition due to their simplicity and excellent performance, which explicitly explore the margin penalty between inter- and intra-class variations via a reformulation of softmax-based loss function [10, 32, 56, 66]. Similar to [10], we use a unified formulation for margin-based softmax loss functions as follows:

$$\mathcal{L}_{margin} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{s\cdot\delta}}{e^{s\cdot\delta}+\sum_{j\neq y_i}^{n}e^{s\cos\theta_j}}, \quad (1)$$

where $\delta = \cos(m_1\theta_{y_i} + m_2) - m_3$, $m_{1,2,3}$ are margins, N is the number of training samples, θ_j indicates the angle between the weight W_j and the feature x_i, y_i represents the ground-truth class, and s is the scale factor. Specifically, for SphereFace [32], ArcFace [10] and CosFace [56], we have the coefficients $(m_1, 0, 0)$, $(0, m_2, 0)$, and $(0, 0, m_3)$, respectively, and we use ArcFace [10] as our baseline.

3.2. SynFace vs. RealFace

Method	Training Dataset	LFW	Syn-LFW
RealFace	CASIA-WebFace	99.18	98.85
SynFace	Syn_10K_50	88.98	99.98

Table 1. The cross-domain evaluation of SynFace and RealFace using the metric of face verification accuracy (%).

To explore the performance gap between SynFace and RealFace, as well as the underlying causes, we perform experiments on real-world face datasets and synthetic face datasets generated by DiscoFaceGAN [11]. Specifically, for real-world face datasets, we use CASIA-WebFace [63] for training and LFW [26] for testing. For the fair comparison, we generate the synthetic version of the LFW dataset, Syn-LFW, using the same parameters (the number of samples, the number of identities, distributions of expression, pose, and illumination). For synthetic training data, we generate $10K$ different identities with 50 samples per identity to form a comparable training dataset to CASIA-WebFace (containing 494,414 images from 10,575 subjects) and we refer to it as Syn_10K_50. More details of synthetic dataset construction can be found in Sec. 4.1. With both synthetic and real face images, we then perform the cross-domain evaluation as follows. We train two face recognition models on CASIA-WebFace and Syn_10K_50, and test them on LFW and Syn-LFW, respectively. As shown in Table 1, there is a clear performance gap (88.98% vs. 99.18%) when testing on LFW, while SynFace outperforms RealFace on Syn-LFW (99.98% vs. 98.85%). These observations suggest that the domain gap between synthetic and real face images contributes to the performance gap between Syn-Face and RealFace.

We compare the face images between Syn_10K_50 and CASIA-WebFace, and find that the synthetic face images usually lack the intra-class variations, which may be one of the reasons for the performance degradation (please refer to the supplementary materials for more illuminations). Furthermore, we also visualize the distributions of feature embeddings by using multidimensional scaling (MDS [4]) to convert the 512-dimensional feature vector into 2D space. As shown in Figure 3, we randomly select 50 samples from two different classes of Syn_10K_50 and CASIA-WebFace, respectively. In particular, we observe that the cyan triangles have a much more compact distribution than the green pentagons, suggesting the poor intra-class variations in Syn_10K_50.

3.3. SynFace with Identity Mixup

To increase the intra-class variations of synthetic face images, we incorporate the identity mixup into DiscoFace-GAN [11] to form a new face generator for face recognition,
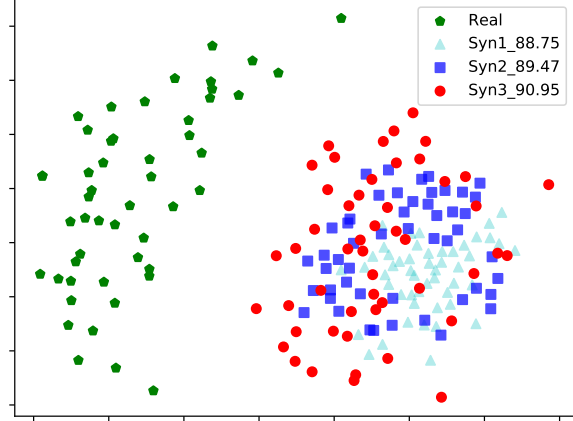

Figure 3. Visualization of the feature distributions (using MDS [4]) for the samples from three different synthetic datasets (Syn1, Syn2 and Syn3) and CASIA-WebFace, which are illustrated by the cyan triangles, blue square, red circle and green pentagon, respectively. Note that the intra-class variations of Syn1, Syn2 and Syn3 are increasing, which lead to the consistent improvements on accuracy ($88.75\% \rightarrow 89.47\% \rightarrow 90.95\%$). Best viewed in color.

i.e., the Mixup Face Generator, which is capable of generating different identities and their intermediate states. In this subsection, we first briefly discuss the mechanism of DiscoFaceGAN, and we then introduce how to incorporate the proposed identity mixup into the face generator.

Face Generator. DiscoFaceGAN [11] can provide the disentangled, precisely-controllable latent representations for the identity of non-existing people, expression, pose, and illumination to generated face images. Specifically, it generates realistic face images x from random noise z, which consists of five independent variables $z_i \in \mathbb{R}^{N_i}$ and each of them follows a standard normal distribution. The above five independent variables indicate independent factors for face generation: identity, expression, illumination, pose, and random noise accounting for other properties such as the background. Let $\lambda \doteq [\alpha, \beta, \gamma, \theta]$ denote the latent factors, where α, β, γ and θ indicate the identity, expression, illumination, and pose coefficient, respectively. Four simple VAEs [9] of α, β, γ and θ are then trained for z-space to λ-space mapping, which enables training the generator to imitate the rendered faces from 3DMM [3]. The pipeline of generating a face image is to 1) first randomly sample latent variables from the standard normal distribution, 2) then feed them into the trained VAEs to obtain α, β, γ and θ coefficients, and 3) the corresponding face image is synthesized by the generator using these coefficients.

Identity Mixup (IM). Inspired by the reenactment of face images [68], we propose to enlarge the intra-class variations by interpolating two different identities as a new intermediate one with changing the label correspondingly. Recalling that the coefficient α controls the identity characteristic, we thus interpolate two different identity coeffi-
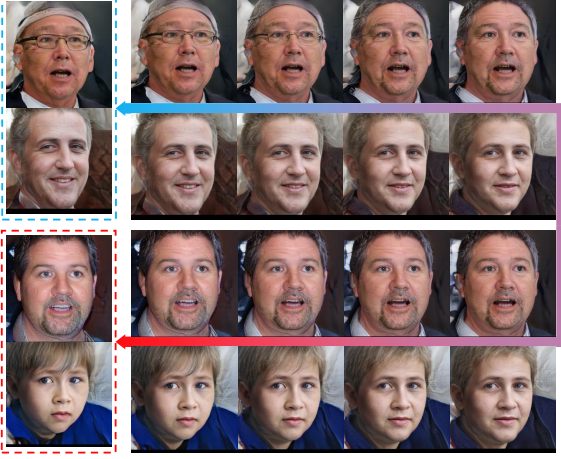
Figure 4. Examples of an identity gradually and smoothly varying to another identity as the weighted ratio φ varies from 0 to 1.

cients to generate a new intermediate identity coefficient. Mathematically, it can be formulated as follows:

$$\begin{aligned}
\alpha &= \varphi \cdot \alpha_1 + (1 - \varphi) \cdot \alpha_2, \\
\eta &= \varphi \cdot \eta_1 + (1 - \varphi) \cdot \eta_2,
\end{aligned} \quad (2)$$

where α_1, α_2 are two random identity coefficients from λ-space, and η_1, η_2 are the corresponding class labels. Note that the weighted ratio φ is randomly sampled from the linear space which varies from 0.0 to 1.0 with interval being 0.05 (*i.e.*, $np.linspace(0.0, 1.0, 21)$). Comparing to the vanilla mixup [67] which is employed at the pixel level, the proposed mixup is operating on the identity coefficient latent space, denoted as identity mixup (IM), which enlarges the intra-class variations by linearly interpolating different identities, forming the Mixup Face Generator. However, both of them can regularize the model to favor the simple linear behavior in-between training samples.

As illustrated in Figure 2, the pipeline of Mixup Face Generator is first randomly sampling two different identity latent variables from the standard normal distribution, and then feeding them to the trained VAEs to obtain α_1, α_2 coefficients. The mixed identity coefficient α is obtained by identity mixup with α_1, α_2 according to Eq. (2), the corresponding face image is finally synthesized by the generator with α, μ coefficients (where $\mu \doteq [\beta, \gamma, \theta]$). We also visualize two groups of identity interpolation with identity mixup in Figure 4. As we can see, one identity gradually and smoothly transforms to another identity as the weighted ratio φ varies from 0 to 1. Besides, from Figure 4, it is obvious that the face images generated with intermediate identity coefficients are also high-quality.

To evaluate the identity mixup for enlarging the intra-class variations, as illustrated in Figure 3, we visualize the feature embedding distributions of the same class in three
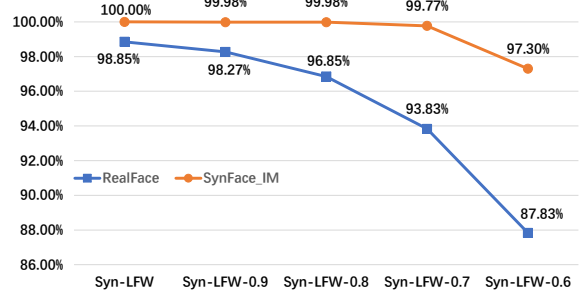

Figure 5. Face verification accuracy comparison between RealFace and SynFace_IM (*i.e.*, SynFace with Identity Mixup) on five different synthetic testing datasets. Syn-LFW is the synthetic version of the LFW dataset, while Syn-LFW-R (with R $\in [0.6, 0.7, 0.8, 0.9]$) indicates introducing identity mixup with ratio R into Syn-LFW.

synthetic datasets (containing $5K$ different identities with 50 samples per identity) with different levels of identity mixup (IM) by using multidimensional scaling (MDS [4]). Note that Syn1, Syn2 and Syn3 represent the weighted ratio φ is 1.0 (*i.e.*, no IM), 0.8 and randomly sampled from the linear space which varies from 0.6 to 1.0 with the interval being 0.05 (*i.e.*, $np.linspace(0.6, 1.0, 11)$). It is clear that the cyan triangles (Syn1) have the smallest variations, while the red circles (Syn3) have the largest one and the blue squares (Syn2) are in the middle position. Accordingly, the accuracy is in an increasing trend (*i.e.*, $88.75\% \to 89.47\% \to 90.95\%$). Besides, 88.98% (as in Table 1) is boosted to 91.97% (as in Table 2) after utilizing identity mixup. In particular, when the baseline is weaker, the improvement brought by identity mixup is larger, which are shown in Table 3 and Figure 7.

In addition to utilizing identity mixup in the training process, we also make an attempt of employing identity mixup on the synthetic testing dataset to evaluate the model's robustness on the identity coefficient noises. Specifically, both RealFace (trained on CASIA-WebFace) and SynFace_IM (trained on Syn_10K_50 with identity mixup) are evaluated on five different synthetic testing datasets, as illustrated in Figure 5. Note that Syn-LFW is the synthetic version of the LFW dataset, while Syn-LFW-R (with R $\in [0.6, 0.7, 0.8, 0.9]$) indicates employing the identity mixup with the weighted ratio R during the generation of Syn-LFW. Specifically, we mix the primary class with a random secondary class using the ratio R according to Eq. (2), but we keep the original label unchanged. Apparently, when R is smaller (*i.e.*, the weight of the primary class is smaller), the corresponding testing dataset is more difficult to recognize because the secondary class impacts the identity information more heavily.

From the results of Figure 5, we can find that our SynFace_IM achieves nearly perfect accuracy when R is larger than 0.6, and also obtains an impressive 97.30% result

which remarkably outperforms the 87.83% accuracy by RealFace when R is 0.6. On the other hand, the accuracy of RealFace drops significantly on Syn-LFW-R when R becomes small, which suggests that the domain gap between real and synthetic face data is still large even after employing the identity mixup. Another interesting conclusion is that the current state-of-the-art face recognition model (*i.e.*, RealFace) cannot handle the identity mixup attack. In other words, if a face image is mixup with another identity, the model cannot recognize it well. However, the proposed SynFace with identity mixup can nearly keep the accuracy under the identity mixup attack. We prefer to explore how to make the RealFace handle such an attack in future work.

3.4. SynFace with Domain Mixup

The lack of intra-class variation is an observable cause of the domain gap between synthetic and real faces, and SynFace can be significantly improved by the proposed identity mixup. To further narrow the performance gap between SynFace and RealFace, we introduce the domain mixup as a general domain adaptation method to alleviate the domain gap for face recognition. Specifically, we utilize large-scale synthetic face images with a small number of real-world face images with labels as the training data. When training, we perform mixup within a mini-batch of synthetic images and a mini-batch of real images, where the labels changed accordingly as the supervision. Mathematically, the domain mixup can be formulated as follows:

$$\begin{aligned} X &= \psi \cdot X_S + (1 - \psi) \cdot X_R, \\ Y &= \psi \cdot Y_S + (1 - \psi) \cdot Y_R, \end{aligned} \quad (3)$$

where X_S, X_R indicate the synthetic and real face images, respectively, and Y_S, Y_R indicate their corresponding labels. Note that ψ is the mixup ratio which is randomly sampled from the linear space distribution from 0.0 to 1.0 with the interval being 0.05 (*i.e.*, $np.linspace(0.0, 1.0, 21)$). For the large-scale synthetic data, we synthesize the Syn_10K_50 dataset that has $10K$ different identities with 50 samples per identity. For a small set of real-world data, we utilize the first $2K$ identities of CASIA-WebFace. The experimental results are shown in Table 2. Specifically, the first row, Syn_10K_50, indicating the baseline method without using any real face images, achieves the accuracy 91.97% using identity mixup. "Real_N_S" means the use of only real images, N identities with S samples per identity during training, while "Mix_N_S" indicates a mixture of N real identities with S samples per identity with Syn_10K_50 during training. Both identity mixup and domain mixup are employed on all the 'Mix_N_S' datasets. As demonstrated in Table 2, domain mixup brings a significant and consistent improvement over the baseline methods under different settings. For example, Mix_2K_10 obtains 95.78% accuracy, which significantly surpasses 91.97% achieved by

Method	R_ID	Samples per R_ID	Accuracy
Syn_10K_50	0	0	91.97
Real_1K_10	1K	10	87.50
Mix_1K_10	1K	10	**92.28**
Real_1K_20	1K	20	92.53
Mix_1K_20	1K	20	**95.05**
Real_2K_10	2K	10	91.22
Mix_2K_10	2K	10	**95.78**

Table 2. Face verification accuracies (%) of models trained on synthetic, real and mixed datasets on LFW. R_ID means the number of real identities.

Syn_10K_50 and 91.22% achieved by Real_2K_10. We conjecture that mixup with the real images can bring the real-world appearance attributes (*e.g.*, blur and illumination) to synthetic images, which alleviate the domain gap. If we continue to increase the number of real images for training, *e.g.*, Mix_2K_20, the performance can be further boosted from 95.78% to 97.65%.

4. Experiments

With the introduced Mixup Face Generator, we are able to generate large-scale face images with controllable facial attributes, including the identity, pose, expression, illumination, and other dataset characteristics such as the depth and the width. In this section, we perform empirical analysis using synthetic face images. Specifically, we first introduce the datasets (Sec. 4.1) and the implementation details (Sec. 4.2). Then the long-tailed problem is mitigated by employing the balanced synthetic face dataset and identity mixup (Sec. 4.3). Lastly, we analyze the impacts of depth, width (Sec. 4.4), and different facial attributes (Sec. 4.5).

4.1. Datasets

Real Datasets. We employ the CASIA-WebFace [63] and LFW [26] for training and testing, respectively. The CASIA-WebFace dataset contains around 500,000 web images, *i.e.*, 494,414 images from 10,575 subjects. The LFW dataset is a widely-used benchmark for face verification, which contains 13,233 face images from 5,749 identities. Following the protocol in [10], we report the verification accuracy on 6,000 testing image pairs.

Synthetic Datasets. We first generate a synthetic version of LFW, in which all synthetic face images share the same properties with LFW images, *e.g.*, expression, illumination, and pose. Specifically, for each image in LFW, we first use the 3D face reconstruction network in [12] to obtain the attribute coefficients $\mu \doteq [\beta, \gamma, \theta]$, which indicate the expression, illumination and pose coefficient, respectively. We then adopt the DiscoFaceGAN [11] to generate the face images according to these attribute coefficients with a random identity coefficient. Finally, we obtain a new dataset
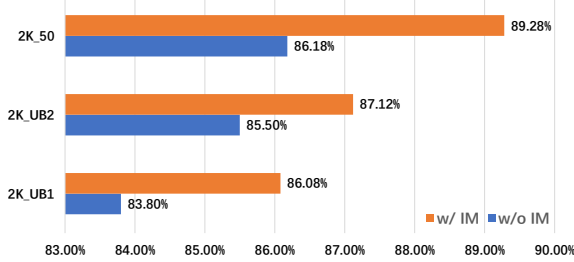
Figure 6. Face verification accuracies (%) on LFW using the training datasets with decreasing imbalance, *i.e.*, "2K_UB1", "2K_UB2", and "2K_50", where we assign N defined in Eq.(4) as $[2, 2, 6, 40, 200]$, $[4, 16, 30, 80, 120]$, and $[50, 50, 50, 50, 50]$, respectively. w/ IM and w/o IM indicate whether identity mixup (IM) is used during training.

and refer to it as Syn-LFW, which has the same statistics as LFW with unknown identities (non-existing people). For synthetic training dataset (*e.g.*, Syn_10K_50), we construct it by randomly sampling latent variables from the standard normal distribution for identity, expression, pose and illumination coefficients, respectively, which leads to the same person with different expressions, poses and illuminations in the same class. Note that the identities of Syn-LFW do not have the overlap with any synthetic training datasets.

4.2. Implementation Details

We use the MTCNN [69] to detect face bounding boxes and five facial landmarks (two eyes, nose and two mouth corners). All face images are then cropped, aligned (similarity transformation), and resized to 112×96 pixel as illustrated in Figure 1. Similar to [10, 56], we normalize the pixel values (in $[0, 255]$) in RGB images to $[-1.0, 1.0]$ for training and testing. To balance the trade-off between the performance and computational complexity, we adopt the variant of ResNet [22], LResNet50E-IR, as our backbone framework, which is devised in ArcFace [10]. All models are implemented with PyTorch [38] and trained from scratch using Eight NVIDIA Tesla V100 GPUs. We use the additive angular margin loss defined in Eq. (1), *i.e.*, with $(m_1, m_2, m_3) = (0, 0.5, 0)$ and $s = 30$. If not mentioned, we always set the batch size to 512. We use SGD with a momentum of 0.9 and a weight decay of 0.0005. The learning rate starts from 0.1, and is divided by 10 at the 24, 30 and 36 epochs, with 40 epochs in total.

4.3. Long-tailed Face Recognition

Experimental Setup. To explore the long-tailed problem, we construct multiple synthetic datasets with the purpose that each dataset has the same number of identities ($2K$) and total images ($100K$) but different degrees of un-

balance. Face images are generated using the equation:

$$N = [N_1, N_2, N_3, N_4, N_5], \\ ID = [400, 400, 400, 400, 400], \quad (4)$$

where ID indicates the number of identities in each of the five groups, and N means the number of samples of the five groups. For example, if $N = [30, 40, 50, 60, 70]$, the corresponding synthetic dataset has 400 identities with 30 samples per identity, and the rest 1600 identities with $40, 50, 60, 70$ samples per identity, respectively. We construct three different synthetic datasets by assigning N to be $[2, 2, 6, 40, 200]$, $[4, 16, 30, 80, 120]$ and $[50, 50, 50, 50, 50]$, which are denoted as "2K_UB1", "2K_UB2" and "2K_50", respectively. The detailed construction process can be found in Sec. 4.1. Note that all the three datasets have average 50 samples per identity, while the first two have unbalanced distributions with the standard deviations 76.35 and 43.52, and the last one is the perfectly balanced dataset.

Empirical Analysis. We train face recognition models on the above three different synthetic datasets and the experimental results are illustrated in Figure 6. We see that the model trained on the "2K_UB1" achieves the worst performance (83.80%), suggesting that the long-tailed problem or the unbalanced distribution leads to the degradation of the model performance. Comparing with the models trained on "2K_UB1" and "2K_UB2", we discover that decreasing the degree of unbalance leads to the improvement on the performance. Finally, when the model is trained on "2K_50", *i.e.*, the perfectly balanced dataset, the accuracy is significantly improved to 86.18%. Therefore, with balanced synthetic data, the long-tailed problem can be intrinsically avoided. Besides, introducing the identity mixup for training can consistently and significantly improve the performance over all the settings.

4.4. Effectiveness of "Depth" and "Width"

Experimental Setup. We synthesize multiple face datasets with different width (the number of identities) and depth (the number of samples per identity). Let "N_S" denote the synthetic dataset containing N identities with S samples per identity, *e.g.*, $1K\_50$ indicates the dataset having $1K$ different identities and 50 samples per identity. Obviously, N and S represent the dataset's width and depth. The details of dataset construction can be found in Sec. 4.1.

Empirical Analysis. We train the same face recognition model on these synthetic datasets, and the experimental results (both w/wo identity mixup) are shown in Table 3. Firstly, we analyze the influence of the width of the dataset by comparing the results of (a), (b), (c), (i). From (a) to (c), we see that the accuracy dramatically increases from 83.85% to 88.75%. However, the improvement is marginal from (c) to (i), which implies that the synthetic data may suffer from the lack of inter-class variations. Observing the

Method	ID	Samples	LFW	LFW(w/ IM)
(a) 1K_50	1K	50	83.85	**87.53**
(b) 2K_50	2K	50	86.18	**89.28**
(c) 5K_50	5K	50	88.75	**90.95**
(d) 10K_2	10K	2	78.85	**80.30**
(e) 10K_5	10K	5	88.22	**88.32**
(f) 10K_10	10K	10	89.48	**90.28**
(g) 10K_20	10K	20	89.90	**90.87**
(h) 10K_30	10K	30	89.73	**91.17**
(i) 10K_50	10K	50	88.98	**91.97**

Table 3. Face verification accuracies (%) on LFW [63]. "N_S" implies that the corresponding dataset has N identities with S samples per identity, *i.e.*, N and S indicate the width and depth. LFW (w/ IM) means employing the identity mixup (IM) for training.
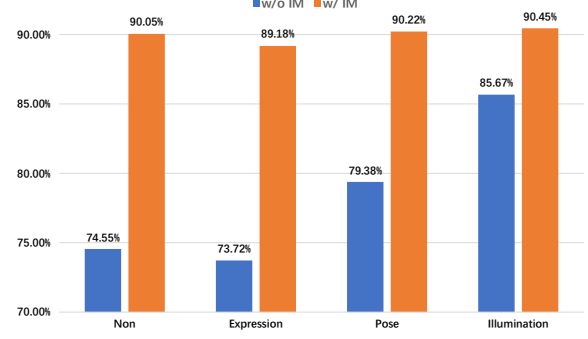

Figure 7. Face verification accuracies (%) on LFW using the training datasets with variations in different facial attributes. Specifically, "Expression", "Pose", and "Illumination" indicate that we separately introduce variations in expression, pose, and illumination while keeping the other attributes unchanged. w/ IM and w/o IM indicate whether identity mixup (IM) is used during training.

results of (d), (e), (f), (g), (h), (i), we conclude that the accuracy significantly increases with the increasing of dataset depth, but it is quickly saturated when the depth is larger than 20, which is in line with the observation on real data made by Schroff *et al*. [45]. Lastly, we see that (a) and (e) have the same number of total images (50K), while (a) outperforms (e) with a large margin, *i.e.*, 4.37%, which reveals that the dataset width plays as the more important role than the dataset depth in term of the final face recognition accuracy. Similar observation can be found by comparing (b) and (f). Importantly, employing the identity mixup (IM) for training consistently improves the performance over all the datasets, which confirms the effectiveness of IM. The best accuracy 91.97% brought by IM significantly outperforms the original 88.98%.

4.5. Impacts of Different Facial Attributes

Experimental Setup. We explore the impacts of different facial attributes for face recognition (*i.e.*, expression, pose and illumination) by controlling face generation process. We construct four synthetic datasets that have $5K$ identities and 50 samples per identity. The difference between the four datasets is the distribution of different facial attributes. Specifically, the first dataset is referred to as "Non", since it fixes all the facial attributes. The rest three datasets are referred to as "Expression", "Pose", and "Illumination", respectively, which indicates the only changed attribute while keeping other attributes unchanged.

Empirical Analysis. As shown in Figure 7, "Non" and "Expression" achieve the worst two performances 74.55% and 73.72%. Specifically, we find that "Expression" is limited to poor diversity, *i.e.*, the generated face images mainly have the expression of "smiling" (see more demo images in the supplementary materials). Hence, there is basically only one valid sample per identity for "Non" and "Expression", causing the poor performances. Experimental results on "Pose" and "Illumination" demonstrate significant im-

provements over "Non", possibly due to their more diverse distributions and the testing dataset (*i.e.*, LFW) also has similar pose and illumination. Lastly, we find that all of four settings are significantly improved with the proposed identity mixup, especially for "Non". A possible reason is that identity mixup can be regarded as a strong data augmentation method for face recognition, reducing the influences of different facial attributes on the final recognition accuracy.

5. Conclusion

In this paper, we explored the potentials of synthetic data for face recognition, *i.e.*, SynFace. We performed a systematically empirical analysis and provided novel insights on how to efficiently utilize synthetic face images for face recognition: 1) enlarging the intra-class variations of synthetic data consistently improves the performance, which can be achieved by the proposed identity mixup; 2) both the depth and width of the training synthetic dataset have significant influences on the performance, while the saturation first appears on the depth dimension, *i.e.*, increasing the number of identities (width) is more important; 3) the impacts of different attributes vary from pose, illumination and expression, *i.e.*, changing pose and illumination brings significant improvements, while generated face images suffer from a poor diversity on expression; 4) a small subset of real-world face images can greatly boost the performance of SynFace via the proposed domain mixup.

Acknowledgement

References

[1] Martin Arjovsky, Soumith Chintala, and Leon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017. 2

[2] Jianmin Bao, Dong Chen, Fang Wen, Houqiang Li, and Gang Hua. Towards open-set identity preserving face synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6713–6722, 2018. 2

[3] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual Conference on Computer graphics and interactive techniques*, pages 187–194, 1999. 1, 2, 4

[4] Ingwer Borg and Patrick JF Groenen. *Modern multidimensional scaling: Theory and applications*. Springer Science & Business Media, 2005. 4, 5

[5] Kaidi Cao, Yu Rong, Cheng Li, Xiaoou Tang, and Chen Change Loy. Pose-robust face recognition via deep residual equivariant mapping. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5187–5196, 2018. 1

[6] Yuhua Chen, Wen Li, Xiaoran Chen, and Luc Van Gool. Learning semantic segmentation from synthetic data: A geometrically guided input-output adaptation approach. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1841–1850, 2019. 2

[7] Zhuo Chen, Chaoyue Wang, Bo Yuan, and Dacheng Tao. Puppeteergan: Arbitrary portrait animation with semantic-aware appearance transformation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[8] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 539–546. IEEE, 2005. 3

[9] Bin Dai and David Wipf. Diagnosing and enhancing vae models. *arXiv preprint arXiv:1903.05789*, 2019. 4

[10] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4690–4699, 2019. 1, 3, 6, 7

[11] Yu Deng, Jiaolong Yang, Dong Chen, Fang Wen, and Xin Tong. Disentangled and controllable face image generation via 3d imitative-contrastive learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5154–5163, 2020. 1, 2, 3, 4, 6

[12] Yu Deng, Jiaolong Yang, Sicheng Xu, Dong Chen, Yunde Jia, and Xin Tong. Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 0–0, 2019. 6

[13] Zhongying Deng, Xiaojiang Peng, Zhifeng Li, and Yu Qiao. Mutual component convolutional neural networks for heterogeneous face recognition. *IEEE Transactions on Image Processing (TIP)*, 28(6):3102–3114, 2019. 1

[14] Dihong Gong, Zhifeng Li, Weilin Huang, Xuelong Li, and Dacheng Tao. Heterogeneous face recognition: A common encoding feature discriminant approach. *IEEE Transactions Image Process (TIP)*, 26(5):2079–2089, 2017. 1

[15] Dihong Gong, Zhifeng Li, Dahua Lin, Jianzhuang Liu, and Xiaoou Tang. Hidden factor analysis for age invariant face recognition. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2872–2879, 2013. 1

[16] Dihong Gong, Zhifeng Li, Jianzhuang Liu, and Yu Qiao. Multi-feature canonical correlation analysis for face photo-sketch image retrieval. In *Proceedings of the 21th ACM International Conference on Multimedia*, pages 617–620, 2013. 1

[17] Dihong Gong, Zhifeng Li, Dacheng Tao, Jianzhuang Liu, and Xuelong Li. A maximum entropy feature descriptor for age invariant face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5289–5297, 2015. 1

[18] 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)*, pages 2672–2680, 2014. 1, 2

[19] Hongyu Guo, Yongyi Mao, and Richong Zhang. Mixup as locally linear out-of-manifold regularization. In *AAAI Conference on Artificial Intelligence (AAAI)*, volume 33, pages 3714–3722, 2019. 3

[20] 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 (ECCV)*, pages 87–102. Springer, 2016. 1

[21] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 1735–1742. IEEE, 2006. 3

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 1, 7

[23] Elad Hoffer and Nir Ailon. Deep metric learning using triplet network. In *International Workshop on Similarity-Based Pattern Recognition*, pages 84–92. Springer, 2015. 1, 3

[24] Fu Jie Huang, Zhihua Zhou, Hong-Jiang Zhang, and Tsuhan Chen. Pose invariant face recognition. In *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition*, pages 245–250, 2000. 1

[25] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4700–4708, 2017. 1

[26] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. 2008. 1, 4, 6

[27] Ira Kemelmacher-Shlizerman, Steven M Seitz, Daniel Miller, and Evan Brossard. The megaface benchmark: 1 million faces for recognition at scale. In *IEEE Conference*

on Computer Vision and Pattern Recognition (CVPR), pages 4873–4882, 2016. 1

[28] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 0–0, 2019. 2

[29] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017. 1

[30] Zhifeng Li, Dihong Gong, Qiang Li, Dacheng Tao, and Xuelong Li. Mutual component analysis for heterogeneous face recognition. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 7(3):1–23, 2016. 1

[31] Zhifeng Li, Dihong Gong, Yu Qiao, and Dacheng Tao. Common feature discriminant analysis for matching infrared face images to optical face images. *IEEE Transactions Image Process (TIP)*, 23(6):2436–2445, 2014. 1

[32] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 212–220, 2017. 1, 3

[33] Weiyang Liu, Yandong Wen, Zhiding Yu, and Meng Yang. Large-margin softmax loss for convolutional neural networks. In *International Conference on Machine Learning (ICML)*, volume 2, page 7, 2016. 3

[34] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2537–2546, 2019. 1

[35] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. 2

[36] Youssef Mroueh, Tom Sercu, and Vaibhava Goel. Mcgan: Mean and covariance feature matching gan. *arXiv preprint arXiv:1702.08398*, 2017. 2

[37] Wanli Ouyang, Xiaogang Wang, Cong Zhang, and Xiaokang Yang. Factors in finetuning deep model for object detection with long-tail distribution. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 864–873, 2016. 1

[38] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems (NeurIPS)*, pages 8024–8035. Curran Associates, Inc., 2019. 7

[39] Guo-Jun Qi. Loss-sensitive generative adversarial networks on lipschitz densities. *International Journal of Computer Vision (IJCV)*, 128(5):1118–1140, 2020. 2

[40] Haibo Qiu, Dihong Gong, Zhifeng Li, Wei Liu, and Dacheng Tao. End2end occluded face recognition by masking corrupted features. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, pages 1–1, 2021. 1

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

[42] Elad Richardson, Matan Sela, and Ron Kimmel. 3d face reconstruction by learning from synthetic data. In *2016 fourth International Conference on 3D vision (3DV)*, pages 460–469, 2016. 2

[43] Christos Sakaridis, Dengxin Dai, Simon Hecker, and Luc Van Gool. Model adaptation with synthetic and real data for semantic dense foggy scene understanding. In *European Conference on Computer Vision (ECCV)*, pages 687–704, 2018. 2

[44] Swami Sankaranarayanan, Yogesh Balaji, Arpit Jain, Ser Nam Lim, and Rama Chellappa. Learning from synthetic data: Addressing domain shift for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3752–3761, 2018. 2

[45] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823, 2015. 8

[46] Yujun Shen, Ping Luo, Junjie Yan, Xiaogang Wang, and Xiaoou Tang. Faceid-gan: Learning a symmetry three-player gan for identity-preserving face synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 821–830, 2018. 2

[47] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[48] Lingxue Song, Dihong Gong, Zhifeng Li, Changsong Liu, and Wei Liu. Occlusion robust face recognition based on mask learning with pairwise differential siamese network. In *IEEE International Conference on Computer Vision (ICCV)*, pages 773–782, 2019. 1

[49] Cecilia Summers and Michael J Dinneen. Improved mixed-example data augmentation. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1262–1270. IEEE, 2019. 3

[50] Ryo Takahashi, Takashi Matsubara, and Kuniaki Uehara. Ricap: Random image cropping and patching data augmentation for deep cnns. In *Asian Conference on Machine Learning (ACML)*, pages 786–798. PMLR, 2018. 3

[51] Zheng Tang, Milind Naphade, Stan Birchfield, Jonathan Tremblay, William Hodge, Ratnesh Kumar, Shuo Wang, and Xiaodong Yang. Pamtri: Pose-aware multi-task learning for vehicle re-identification using highly randomized synthetic data. In *IEEE International Conference on Computer Vision (ICCV)*, pages 211–220, 2019. 2

[52] Daniel Sáez Trigueros, Li Meng, and Margaret Hartnett. Generating photo-realistic training data to improve face recognition accuracy. *arXiv preprint arXiv:1811.00112*, 2018. 2

[53] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Na-jafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning (ICML)*, pages 6438–6447. PMLR, 2019. 3

[54] Fei Wang, Liren Chen, Cheng Li, Shiyao Huang, Yanjie Chen, Chen Qian, and Chen Change Loy. The devil of face recognition is in the noise. In *European Conference on Computer Vision (ECCV)*, pages 765–780, 2018. 1

[55] Hao Wang, Dihong Gong, Zhifeng Li, and Wei Liu. Decorrelated adversarial learning for age-invariant face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3527–3536, 2019. 1

[56] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5265–5274, 2018. 1, 3, 7

[57] Qi Wang, Junyu Gao, Wei Lin, and Yuan Yuan. Learning from synthetic data for crowd counting in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8198–8207, 2019. 2

[58] Xiaobo Wang, Shuo Wang, Jun Wang, Hailin Shi, and Tao Mei. Co-mining: Deep face recognition with noisy labels. In *IEEE International Conference on Computer Vision (ICCV)*, pages 9358–9367, 2019. 1

[59] Yitong Wang, Dihong Gong, Zheng Zhou, Xing Ji, Hao Wang, Zhifeng Li, Wei Liu, and Tong Zhang. Orthogonal deep features decomposition for age-invariant face recognition. In *European Conference on Computer Vision (ECCV)*, pages 738–753, 2018. 1

[60] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *European Conference on Computer Vision (ECCV)*, pages 499–515. Springer, 2016. 1, 3

[61] Minghao Xu, Jian Zhang, Bingbing Ni, Teng Li, Chengjie Wang, Qi Tian, and Wenjun Zhang. Adversarial domain adaptation with domain mixup. In *AAAI Conference on Artificial Intelligence (AAAI)*, volume 34, pages 6502–6509, 2020. 3

[62] Xiaolong Yang, Xiaohong Jia, Dihong Gong, Dong-Ming Yan, Zhifeng Li, and Wei Liu. Larnet: Lie algebra residual network for face recognition. In *International Conference on Machine Learning (ICML)*, pages 11738–11750. PMLR, 2021. 1

[63] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li. Learning face representation from scratch. *arXiv preprint arXiv:1411.7923*, 2014. 4, 6, 8

[64] Xi Yin, Xiang Yu, Kihyuk Sohn, Xiaoming Liu, and Manmohan Chandraker. Towards large-pose face frontalization in the wild. In *IEEE International Conference on Computer Vision (ICCV)*, pages 3990–3999, 2017. 2

[65] Baosheng Yu, Tongliang Liu, Mingming Gong, Changxing Ding, and Dacheng Tao. Correcting the triplet selection bias for triplet loss. In *European Conference on Computer Vision (ECCV), Munich, Germany*, pages 71–87, September 08-14, 2018. 3

[66] Baosheng Yu and Dacheng Tao. Deep metric learning with tuplet margin loss. In *IEEE International Conference on Computer Vision (ICCV), Seoul, Korea*, pages 6490–6499, October 27-November 02, 2019. 3

[67] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 3, 5

[68] Jiangning Zhang, Xianfang Zeng, Mengmeng Wang, Yusu Pan, Liang Liu, Yong Liu, Yu Ding, and Changjie Fan. Freenet: Multi-identity face reenactment. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5326–5335, 2020. 4

[69] 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. 7

[70] Wenchao Zhang, Shiguang Shan, Xilin Chen, and Wen Gao. Local Gabor binary patterns based on Kullback - Leibler divergence for partially occluded face recognition. *IEEE Signal Processing Letters*, 14(11):875–878, 2007. 1

[71] Xiangxin Zhu, Dragomir Anguelov, and Deva Ramanan. Capturing long-tail distributions of object subcategories. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 915–922, 2014. 1