RPG-Palm: Realistic Pseudo-data Generation for Palmprint Recognition

Lei Shen*  
Youtu Lab, Tencent  
Shanghai, China  
shenlei1996@gmail.com

Jianlong Jin*  
Hefei University of Technology  
Hefei, China  
2022111048@mail.hfut.edu.cn

Ruixin Zhang  
Youtu Lab, Tencent  
Shanghai, China  
ruixinzhang@tencent.com

Jianlong Jin*  
Hefei University of Technology  
Hefei, China  
2022111048@mail.hfut.edu.cn

Yuqi Li  
Hefei University of Technology  
Hefei, China  
huayuqiplat@163.com

Kai Zhao  
UCLA  
California, USA  
kz@kaizhao.net

Huaen Li  
Hefei University of Technology  
Hefei, China  
huaenli@mail.hfut.edu.cn

Yang Zhao†  
Hefei University of Technology  
Hefei, China  
YZhao@hfut.edu.cn

Jingyun Zhang  
WeChat Pay Lab33, Tencent  
Shenzhen, China  
naskyzhang@tencent.com

Shouhong Ding†  
Youtu Lab, Tencent  
Shanghai, China  
ericshding@tencent.com

Wei Jia†  
Hefei University of Technology  
Hefei, China  
jiawei@hfut.edu.cn

Yingyi Zhang  
Youtu Lab, Tencent  
Shanghai, China  
sherlyzhang@tencent.com

Ruixin Zhang  
Youtu Lab, Tencent  
Shanghai, China  
ruixinzhang@tencent.com

Huaen Li  
Hefei University of Technology  
Hefei, China  
huaenli@mail.hfut.edu.cn

Kai Zhao  
UCLA  
California, USA  
kz@kaizhao.net

Abstract

Palmprint recently shows great potential in recognition applications as it is a privacy-friendly and stable biometric. However, the lack of large-scale public palmprint datasets limits further research and development of palmprint recognition. In this paper, we propose a novel realistic pseudo-palmprint generation (RPG) model to synthesize palmprints with massive identities. We first introduce a conditional modulation generator to improve the intra-class diversity. Then an identity-aware loss is proposed to ensure identity consistency against unpaired training. We further improve the Bézier palm creases generation strategy to guarantee identity independence. Extensive experimental results demonstrate that synthetic pretraining significantly boosts the recognition model performance. For example, our model improves the state-of-the-art BézierPalm by more than 5% and 14% in terms of TAR@FAR=1e-6 under the 1 : 1 and 1 : 3 Open-set protocol. When accessing only 10% of the real training data, our method still outperforms ArcFace with 100% real training data, indicating that we are closer to real-data-free palmprint recognition.

1. Introduction

Palmprint is an excellent biometric in terms of privacy, willingness, convenience, and security. Since the palmprint locates on the inner side of the palm, obtaining the palmprint without one’s permission is nearly impossible, which is more privacy-friendly than the face. Compared to iris and fingerprints, palmprint has fewer usage restrictions, which makes palmprint more user-friendly. Furthermore, Palmprints and palm veins can be collected simultaneously to form a highly secure dual-modal system.

Due to these advantages, big companies such as Amazon [1] and Tencent [69] begin to apply palmprint recognition in their payment services. However, for the same reason, palmprints are rare in public and expensive to collect. To our best knowledge, publicly available datasets [2, 21, 29, 30, 36, 41, 52, 57, 68] only contain thousands of identities and tens of thousands of images in total. Meanwhile, face recognition has several million-level publicly available datasets [4, 9, 27, 38, 44, 62] that contain tens to hundreds of thousands of identities. The lack of large-scale public dataset seriously inhibits the research on palmprint recognition.

To solve this problem, one way is to collect a large-scale palmprint dataset. However, this way is very time-
Figure 1: Synthetic Bézier palm creases are used as the identity control condition to guide the generator to produce diverse synthetic data. Recognition models are pretrained with pseudo-data and then finetuned on real data, achieving significant performance boosts. Besides, even if only accessing 10% of the real training data for generation and recognition model, our method still gets comparable performance to 100% of real data direct training.

consuming and expensive. In addition, collecting biometric data causes more and more privacy and ethical concerns [7] and is strictly restricted by legislation in many countries. Another way is to augment the training dataset with synthetic palmprints. To better benefit palmprint recognition applications, the synthetic method should support generating massive identities with inter-class and intra-class diversity, and the gap between synthetic palmprints and real palmprints should be small.

Recently, BézierPalm [69] synthesizes fake palmprint images by generating palmprint creases with several parameterized Bézier curves for recognition model pretraining. BézierPalm first shows the ability to output massive new identities without using real palmprints and significantly improves the performance of the palmprint recognition models. However, BézierPalm has some problems unsolved. Firstly, synthetic palmprints have a large gap to real ones, resulting in non-neglectable finetune data requirements in real-world applications. Secondly, the curve parameter difference of each identity cannot ensure inter-class independence when generating massive data. Thirdly, the intra-class diversity only contains small curve deformation in BézierPalm, while real palmprints have diverse textures, lighting and et al.

In this paper, we propose a realistic pseudo-palmprint generation (RPG) model. As shown in Fig.1, the RPG model takes synthetic Bézier palm creases as an identity (ID) condition and outputs realistic pseudo-palmprints with a bidirectional mapping from the Gaussian noise domain to the palmprints domain. Since there is no correspondence between synthetic Bézier palm creases and real palmprints, the generation model can only use unpaired data and will lose intra-class identity consistency easily during training. To solve this problem, we introduce an identity-aware loss that restrains the identity consistency between palmprints generated from the same Bézier palm creases. In addition, we design a conditional modulation generator to generate diversified intra-class textures and lighting conditions using a latent control vector encoded from random noises. To further reduce the distribution gap between synthetic palm creases and real-world palmprints, we refined the synthetic palmprint creases generation strategy with a more reasonable parameter design and identity independence check.

The contributions of this paper are as follows:

- We propose a realistic pseudo-palmprint generation (RPG) model with a conditional modulation generator to improve the intra-class diversity and an ID-aware loss to help the RPG model ensure identity consistency under unpaired training.
- We improve the Bézier palm creases synthetic method to get more reasonable palm creases and independent identities.
- Extensive experimental results on 13 public datasets demonstrate that recognition models pretrained with our synthetic pseudo-palmprints achieve state-of-the-art recognition accuracy.
- With our RPG pretraining, even if accessing only 10% of the real data, the recognition model performance still outperforms 100% real data direct training. Showing we are closer to real-data-free palmprint recognition.

2. Related Work

2.1. Palmprint Recognition Methods

Palmprint recognition methods can be divided into two categories: traditional-based and deep-learning based [18]. Traditional methods extract various kinds of local or global features to make different palmprints more discriminative. Local-based methods [19, 28, 34, 40, 43, 57, 70] manually design effective local features for recognition. Global-based methods [3, 20, 32, 42, 51, 59] extract low-dimensional features from the whole image to distinguish different palmprints. Deep Learning based methods [16, 25, 35, 58] train
modified neural networks to extract features with classification or pair-wise loss [14, 55, 71]. However, the lack of large-scale public palmprint datasets limits the potential of existing palmprint recognition methods.

2.2. Generation Model for Image Transformation

With the development of image generation models, such as generative adversarial network (GAN)-based models [26, 37] and diffusion-based models [47, 48], image-to-image generation/translation methods have achieved impressive performance [11, 12, 33]. However, many typical models, e.g., conditional generation model pix2pixHD [61], multi-domain transformation model StarGAN [12], dual-domain mapping model BicycleGAN [74] and recent conditional diffusion model [50, 60], rely on paired training data, which is unavailable in many applications. Therefore, some unpaired image-to-image translation models have been proposed [39, 54, 64, 72], which usually adopt cycle consistency loss to train the models without paired data. However, these models are not designed for recognition tasks, and thus ignore the ID preservation in the generation process. Overall, there is still a lack of research on generating diversified and realistic palmprints with ID consistency from limited unpaired samples.

2.3. Data generation for Recognition Tasks

In order to improve the recognition performance, data generation can be used to expand the depth (diversity of each identity) and width (total number of identities) of training data. For example, in face recognition field, several facial image generation methods [15, 24, 45, 46, 65, 22, 23] have been proposed to generate facial samples based on the priors of facial attributes, facial structures and 3D faces [6]. Also, in field of fingerprint recognition, some methods based on hand-crafted or learning-based approaches [5, 17, 63] have been proposed to generate high-fidelity fingerprint images.

For palm recognition tasks, BézierPalm [69] use Bézier curves to synthesize fake palm creases by changing the Bézier control parameters. PalmGAN [53] improves the cross-domain recognition performances by using CycleGAN to transfer the styles between normal and multispectral palmprints. However, PalmGAN doesn’t create virtual identities, and the BézierPalm suffers from the domain gap between palmprint images and geometry curves. By introducing the new generation model and ID-aware loss, our method can expand the inter-class diversity and intra-class diversity of palmprint dataset simultaneously.

3. Method

3.1. Overall Framework

Fig.2 illustrates the whole framework of the proposed realistic pseudo-palmprint generation (RPG) model, which includes a training stage and a forward palmprint generation stage. In the training phase, palmprint \( B \) is first mapped from the palmprint image domain to the latent vector \( Q(z|B) \) in the Gaussian noise domain through the encoder \( E \). Then, the generator \( G \) utilizes the unpaired palm creases \( A \) as condition, and remaps the encoded noise vector \( Q(z|B) \) back to the palmprint image domain. In order to increase the diversity and randomness of the generated palmprints, a conditional modulation structure is designed for \( G \), which uses the input noise vector to control the modulation of intermediate features.

In order to constrain the ID consistency of the generated pseudo palmprints, an ID-aware loss is presented to enforce the generator to maintain the ID information of input palm creases. As shown in Fig.2 (a), a siamese generator \( G \) is used to produce another palmprint \( B' \) with the same palm creases \( A \). Then, a pretrained palmprint recognition discriminator is used to measure the ID consistency between \( B' \) and \( B'' \).

In the forward pseudo-palmprint generation stage, synthetic Bézier creases are input into the generator \( G \) to obtain corresponding palmprint images for virtual identities. Motivated by BézierPalm [69], we improve the random Bézier curves generation strategy to obtain a more reasonable layout of principal lines and wrinkle lines. Then, a classic palmprint recognition method RLOC [34] is applied to ensure the inter-class difference.

In the following, we will introduce the details of generator \( G \), encoder \( E \), ID-aware loss, and the improved Bézier palm creases synthesis strategy.

3.2. Conditional Modulation Palmprint Generator

The proposed generator \( G \) takes Gaussian noise vector as input, palm creases image \( A \) as condition, and reproduces the pseudo-palmprint image \( B' \). As a typical image generation task, the common UNet [49] architecture is adopted. The detailed structure of \( G \) is illustrated in Fig.3 (a). To generate diversified results, a conditional adaptive instance normalization module (CAdaIN) is introduced to modulate the generated details in each down-block and up-block.

In the CAdaIN module, the input noise vector \( N(z) \) is first encoded into a latent control vector \( w(z) \) through two fully-connected (FC) layers. Note that the parameters of these two FC layers are shared for the whole generator \( G \), so that the generated style can be consistent by means of the same \( w(z) \). Then, two other FC layers are used to modulate the mean and variance of intermediate feature maps.
respectively, which can be calculated as,
\[ X_o = f_{c1}(w(z))X_i + f_{c2}(w(z)) + n_0, \] (1)
where \( X_i \) and \( X_o \) denote the input and output features of CAdaIN, and \( f_{c1}, f_{c2} \) represent two FC layers. By using a latent control vector encoded from random noise, the CAdaIN can modulate the features to produce different distributions, resulting in diversified palmprint images. In addition, in order to further improve the diversity, random noise \( n_0 \) with the same spatial resolution as the feature map is added to the modulated features to inject more randomness.

For producing realistic results, the loss \( \mathcal{L}_1 \) and adversarial loss \( \mathcal{L}_D \) are used to restrain the learning of \( G \), as follows,
\[ \mathcal{L}_G = \lambda_1 \mathcal{L}_1(B, B') + \lambda_2 \mathcal{L}_D(B, B'), \] (2)
where \( \lambda_1 \) and \( \lambda_2 \) are weights. The \( \mathcal{L}_1 \) is a commonly used pixel-wise loss, which can ensure the numerical similarity between the generated image and real palmprint. But note that the palm creases image \( A \) is not matched with palmprint \( B \), so that too strong pixel-wise constraint may cause wrong overfitting to the details in \( B \). Hence, adversarial loss \( \mathcal{L}_D \) is also used to relax the constraint and restrain the semantic similarity between \( B \) and \( B' \). For the discriminator in \( \mathcal{L}_D \),
we adopt the PatchGAN [13], which determines patch-wise authenticity by mapping the image to a 70 × 70 grid.

3.3. Palmprint Encoder

The encoder $E$ is used to map the palmprint image to the Gaussian noise domain. Its structure is shown in Fig.3 (b), which is a straightforward ResNet structure. The original size of the feature map is 256 × 256, and it gradually decreases to 16 × 16 through several residual blocks (RB). Then, a FC layer is used to estimate the target mean $\mu_Q$ and variance $\sigma_Q^2$ from flattened features. Finally, the noise vector $Q(z|B)$ is sampled from Gaussian space $\mathcal{N}(\mu_Q, \sigma_Q^2)$ via reparameterization trick [74].

In the training stage, we constrain the KL divergence between $Q(z|B) \sim \mathcal{N}(\mu_Q, \sigma_Q^2)$ and noise vector $N(z) \sim \mathcal{N}(0, 1)$ sampled from Gaussian space, so that the target domain of $E$ can keep approximating to the standard normal distribution. This loss can be computed as,

$$L_{KL} = \frac{1}{2}(1 + \log \sigma_Q^2 - \sigma_Q^2 - \mu_Q^2).$$  \hspace{1cm} (3)

In this paper, $E$ and $G$ are combined to learn a domain-to-domain mapping from unpaired training data, instead of using a fully supervised manner. Firstly, this structure can avoid the dependence on manual labeling of paired data. Secondly, conditional bidirectional domain-to-domain mapping can reproduce random and realistic images and keep the information of palm creases condition as well.

3.4. ID-aware Loss

For training recognition models, the generated palmprints not only need to be diversified, but also have to preserve ID information. For a synthetic palm creases image $A$, the generated palmprints should have the same ID information. Therefore, an ID-aware loss $L_{ID}$ is added to restrain the generator. As shown in Fig.2 (a), a siamese generator $G$ is used to produce another palmprint image $B''$ with the same creases $A$ and random noise vector $N(z)$. Then the $L_{ID}$ restricts the ID consistency of two generated results of $B'$ and $B''$, as follows,

$$L_{ID} = 1 - \frac{D_{MB}(B') \cdot D_{MB}(B'')}{\|D_{MB}(B')\| \times \|D_{MB}(B'')\|},$$ \hspace{1cm} (4)

where “:” denotes the vector dot product operation, $D_{MB}$ is a pretrained palmprint recognition model using MobileFaceNet [10] and extracts the 512 × 1 feature of $B'$ and $B''$. That is, $L_{ID}$ calculates cosine similarity between the extracted features of two generated images for the same ID. Owing to the ID-aware loss, increasing the randomness and diversity of the generator will not destroy the intra-class ID consistency.

The total loss of entire model, as follows,

$$L_{total} = \lambda_D L_D + \lambda_1 L_1 + \lambda_{KL} L_{KL} + \lambda_{ID} L_{ID},$$ \hspace{1cm} (5)

Figure 4: Examples of some synthetic palm creases in BézierPalm [69] and our method.

where $L_D$ denotes GAN loss, $L_1$ denotes absolute error loss, $L_{KL}$ denotes KL divergence loss, and $L_{ID}$ denotes ID-aware loss. We will add the total loss in final version.

3.5. Improved Bézier Palm Creases Synthesis

In the forward stage, only the generator $G$ is used to produce palmprint images from synthetic palm creases, as shown in Fig.2 (b). Motivated by BézierPalm [69], we also adopt the two-level Bézier curves to synthesize palm creases image that contains three principal lines and random wrinkle lines. The parametric control points of the Bézier curve can be adjusted randomly within a preset range to obtain a large number of fake palm creases for virtual ID.

However, we have observed that randomly adjusted Bézier curves may lead to some unreasonable results. As shown in Fig.4, the layout of some randomly generated creases is quite different from that of the real palmprints.

As a result, this paper improves the Bézier curves synthesis strategy based on real palmprint prior from the following three aspects. Firstly, we observe and adjust the rough range of the parameter points (i.e., start point, control point, and end point) of three principal lines according to real palmprints. So that the layout of synthetic principal lines becomes more similar to that of real palms. Secondly, we adjust the synthesis rules to generate more wrinkle lines with moderate length and more uniform distribution. Thirdly, a similarity constraint based on RLOC [34] is added, which makes the principal lines of different IDs sufficiently distinguishable. RLOC is a classic palmprint recognition method that can measure the similarity of two creases images. In order to avoid very similar principal lines being generated for different IDs, we filter out some creases images which exceed the RLOC inter-class similarity threshold. We ablate the threshold of RLOC with 0.1, 0.5, 0.9, 0.95 and experimentally set it as 0.9. More details of improved Bézier palm creases can be found in supplementary materials.

Finally, the improved random Bézier curves are input into the generator $G$ to obtain corresponding palmprint images. This generation process can be repeated until a large-
scale realistic pseudo-palmprint dataset is established.

4. Experimental Settings

We adopt the same experimental datasets and Open-set evaluation protocol following BézierPalm [69]. The identities of the training and test set are isolated. TAR(True Acceptance Rate) @FAR(False Accept Rate) is used to evaluate model performance. Details about Open-set and evaluation protocol can be found in supplementary materials.

4.1. Datasets

We adopt 13 public datasets in our experiments as in BézierPalm [69], including 3,268 IDs and 59,162 palmprint images. Detailed descriptions of these datasets are shown in Tab.1. We follow the detect-then-crop protocol in [67] to extract the Region Of Interests (ROIs).

<table>
<thead>
<tr>
<th>Name</th>
<th>#IDs</th>
<th>#Images</th>
<th>Device</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPD [68]</td>
<td>400</td>
<td>16,000</td>
<td>Phone</td>
</tr>
<tr>
<td>TCD [68]</td>
<td>600</td>
<td>12,000</td>
<td>Contactless</td>
</tr>
<tr>
<td>IITD [41]</td>
<td>460</td>
<td>2,601</td>
<td>Contactless</td>
</tr>
<tr>
<td>CASIA [57]</td>
<td>620</td>
<td>5,502</td>
<td>Contactless</td>
</tr>
<tr>
<td>CASIA-MS [29]</td>
<td>200</td>
<td>7,200</td>
<td>Contactless</td>
</tr>
<tr>
<td>COEP [2]</td>
<td>167</td>
<td>1,344</td>
<td>Digital camera</td>
</tr>
<tr>
<td>MOHI [30]</td>
<td>200</td>
<td>3,000</td>
<td>Phone</td>
</tr>
<tr>
<td>WEHI [30]</td>
<td>200</td>
<td>3,000</td>
<td>Web cam</td>
</tr>
<tr>
<td>GPDS [21]</td>
<td>100</td>
<td>2,000</td>
<td>Web cam</td>
</tr>
<tr>
<td>XJTU_U [52]</td>
<td>200</td>
<td>30,000</td>
<td>Phone</td>
</tr>
<tr>
<td>XJTU_A [52]</td>
<td>114</td>
<td>1,130</td>
<td>CMOS camera</td>
</tr>
<tr>
<td>PolyU-MS [66]</td>
<td>500</td>
<td>24,000</td>
<td>Contactless</td>
</tr>
<tr>
<td>PolyU(2D+3D) [36]</td>
<td>400</td>
<td>8,000</td>
<td>Web cam</td>
</tr>
</tbody>
</table>

Table 1: Details of the 13 public palmprint datasets.

4.2. Implementation Details

**Generation Model Training.** We generate 4000 identities and 100 samples for each identity by default following BézierPalm [69]. For training the generation model, the weights of \( L_1 \), \( L_D \) and \( L_{KL} \) are set as 10.0, 1.0 and 0.01 according to [74]. We ablate the weight of \( L_{ID} \) with 1.0, 5.0, 10.0 and experimentally set it as 5.0. The resolution of input and output images is 256 × 256. The learning rate is 0.0002 in the first 30 epochs and linearly decays to 1e-8 in the last 30 epochs. The generation model is trained using Adam optimizer with parameters (0.5, 0.99). For comparative experiment, we use the source codes of pix2pixHD [61], CycleGAN [73] and BicycleGAN [74] in their original papers.

**Recognition Model Training.** For our recognition model, we use ResNet50 [31] and MobileFaceNet [10] as the backbone with the input resolution of 224 × 224. The model is first pretrained on synthesized data for 25 epochs and then finetuned on real datasets for 50 epochs. For the baseline model, we train the model on real datasets for 50 epochs. The feature extractor in ID-aware Loss uses the same training setting as the baseline. We use Arcface [14] with margin \( m = 0.5 \) and scale factor \( s = 48 \) for the pretraining, finetuning and baseline training supervision. We use the cosine learning rate schedule with a warmup start for one epoch. The maximum learning rate is 1e-2 and the minimum learning rate is 1e-6 for pretraining and finetuning. All recognition models are trained with a mini-batch SGD optimizer. We use 4 NVIDIA Tesla V100 GPUs for training with total batch size of 128.

It should be emphasized that the generation model, the feature extractor in ID-aware loss, and the recognition model all ensure that the training and test sets are completely isolated.

5. Experimental Results

5.1. Open-set Palmprint Recognition

We first test our method under the open-set protocol with two different training and test ratios 1:1 and 1:3 (trainIDs:testIDs=1634:1632, 818:2448). Details about the "Open-set" protocol can be found in supplementary materials. The quantitative results are shown in Tab.2. The TAR v.s. FAR curves of the 1:1 setting are in Fig.5. Our method outperforms BézierPalm by 5.09% and 14.73% under 1:1 and 1:3 settings @FAR=1e-6 using ‘MB’, which establishes a new state-of-the-art. Our method achieves more significant improvement under 1:3 setting than 1:1 setting, which demonstrates the effectiveness of our method in scenarios with only a small amount of real training data.

5.2. Palmprint Recognition with Limited Identities

To further verify the performance of our method with limited real training IDs, we test our method with different sizes of real training IDs and fix the test set under the 1:1
Table 2: Quantitative performances under the open-set protocol where the performances are evaluated in terms of TAR@FAR. ‘MB’ represents the MobileFaceNet [10] backbone and ‘R50’ is resnet50 [31] backbone.

<table>
<thead>
<tr>
<th>Method</th>
<th>Backbone</th>
<th>train : test = 1 : 1</th>
<th>TAR@FAR=1e-3</th>
<th>TAR@FAR=1e-4</th>
<th>TAR@FAR=1e-5</th>
<th>TAR@FAR=1e-6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CompCode [40]</td>
<td>N/A</td>
<td>0.4800 0.4292 0.3625 0.2103</td>
<td>0.4501 0.3932 0.3494 0.2648</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LLDP [43]</td>
<td>N/A</td>
<td>0.7382 0.6762 0.5222 0.1247</td>
<td>0.7372 0.6785 0.6171 0.2108</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BOCV [28]</td>
<td>N/A</td>
<td>0.4930 0.4515 0.3956 0.2103</td>
<td>0.4527 0.3975 0.3527 0.2422</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RLOC [34]</td>
<td>N/A</td>
<td>0.6490 0.5884 0.4475 0.1443</td>
<td>0.6482 0.5840 0.5224 0.3366</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DOC [19]</td>
<td>N/A</td>
<td>0.4975 0.4409 0.3712 0.1667</td>
<td>0.4886 0.4329 0.3889 0.2007</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PalmNet [25]</td>
<td>N/A</td>
<td>0.7174 0.6661 0.5992 0.1069</td>
<td>0.7217 0.6699 0.6155 0.2877</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C-LMCL [71]</td>
<td>MB</td>
<td>0.9290 0.8554 0.7732 0.6239</td>
<td>0.8509 0.7554 0.7435 0.5932</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ArcFace [14]</td>
<td>MB</td>
<td>0.9292 0.8568 0.7812 0.7049</td>
<td>0.8516 0.7531 0.6608 0.5825</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BézierPalm [69]</td>
<td>MB</td>
<td>0.9640 0.9438 0.9102 0.8437</td>
<td>0.9407 0.8861 0.7934 0.7012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td>MB</td>
<td>0.9802 0.9714 0.9486 0.8946</td>
<td>0.9496 0.9267 0.8969 0.8485</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C-LMCL [71]</td>
<td>R50</td>
<td>0.9545 0.9027 0.8317 0.7534</td>
<td>0.8601 0.7701 0.6821 0.6254</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ArcFace [14]</td>
<td>R50</td>
<td>0.9467 0.8925 0.8252 0.7462</td>
<td>0.8709 0.7884 0.7156 0.6580</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BézierPalm [69]</td>
<td>R50</td>
<td>0.9671 0.9521 0.9274 0.8956</td>
<td>0.9424 0.8950 0.8217 0.7649</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td>R50</td>
<td>0.9821 0.9732 0.9569 0.9347</td>
<td>0.9533 0.9319 0.9016 0.8698</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Performance under different real training identities. The generation model, feature extractor in ID-aware loss, and the recognition model access the same number of real training identities. The backbone is MobileFaceNet.

<table>
<thead>
<tr>
<th>Method</th>
<th>#ID</th>
<th>TAR@FAR=1e-3</th>
<th>TAR@FAR=1e-4</th>
<th>TAR@FAR=1e-5</th>
<th>TAR@FAR=1e-6</th>
</tr>
</thead>
<tbody>
<tr>
<td>ArcFace</td>
<td>1,600</td>
<td>0.9292 0.8568 0.7812 0.7049</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BézierPalm</td>
<td>800</td>
<td>0.9534 0.9390 0.9025 0.8164</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td>80</td>
<td>0.8102 0.7050 0.6668 0.3320</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td>400</td>
<td>0.9189 0.8497 0.7542 0.6899</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ArcFace</td>
<td>600</td>
<td>0.6761 0.5294 0.4783 0.2437</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BézierPalm</td>
<td>160</td>
<td>0.8179 0.6998 0.5826 0.4832</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td>80</td>
<td>0.9753 0.9324 0.8836 0.8162</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ArcFace</td>
<td>800</td>
<td>0.5384 0.4682 0.3249 0.1173</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BézierPalm</td>
<td>80</td>
<td>0.6547 0.5511 0.4490 0.3743</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td>80</td>
<td>0.8974 0.8092 0.6947 0.5824</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5.3. Palmprint Recognition at Million Scale

In order to verify the effectiveness of our method on large-scale real-world datasets, we test our method on our internal dataset with millions of palmprint images. Our dataset contains 19,286 training identities with 2,871,073 images and 1,000 test identities with 182,732 images. For a fair comparison with BézierPalm, we generate 20,000 IDs with 100 samples in each ID for pretraining. Quantitative results are shown in Tab.4. Our method outperforms ArcFace and BézierPalm with a clear margin, showing its practical application value on large-scale datasets.

6. Ablation Study

In this section, we conduct ablation studies to verify different components and design choices of our method. The MobileFaceNet is used as the backbone for all experiments under the same Open-set protocol.

6.1. Components and design choices

The main components and design choices of our method are ID-aware loss, conditional modulation generator and improved strategy for Bézier palm creases synthesis. Tab.5 shows the results of recognition models with or without these components. ‘I’, ‘C’, ‘S’ represent the three components respectively. For baseline without ‘C’, the normal UNet [49] as in [74] is used to take place of ‘C’. ID-aware loss achieves the greatest improvement by 11.72%@FAR=1e-6 against the baseline, which reflects its advantage for preserving intra-class consistency. The con-
Figure 6: Generated palmprint images of different methods, (a) pix2pixHD [61], (b) CycleGAN [73], (c) BicycleGAN [74], (d) the proposed method.

<table>
<thead>
<tr>
<th>Method</th>
<th>Backbone</th>
<th>TAR@FAR=</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1e-6</td>
<td>1e-7</td>
</tr>
<tr>
<td>ArcFace</td>
<td>0.9770</td>
<td>0.9550</td>
</tr>
<tr>
<td>BézierPalm</td>
<td>MB</td>
<td>0.9803</td>
</tr>
<tr>
<td>Ours</td>
<td></td>
<td>0.9871</td>
</tr>
</tbody>
</table>

Table 4: Palmprint recognition performance on million scale dataset.

Figure 7: Diverse synthetic palmprints can be generated by adjusting the input noise vector $N(z)$.

Additional modulation generator also significantly improves the performance by 4.49%@FAR=1e-6, which reflects its advantage for improving the intra-class diversity. The improved Bézier palm creases bring an improvement by 1.15%@FAR=1e-6.

6.2. Comparison of Generation Methods

Three generation methods pix2pixHD [61], CycleGAN [73], BicycleGAN [74] are used for comparison, and all of them are retrained on the same training set with unpaired data. Fig.6 illustrates the generated palmprints of different generation methods. It can be found that pix2pixHD and CycleGAN tend to synthesize blurred results. As marked in blue rectangle, BicycleGAN may generate some creases that are inconsistent with the input Bézier curves. Our method can synthesize clearer and sharper principal lines and faithfully preserve the ID information of the input Bézier curves. For intra-class diversity, as shown in Fig.7, our method is able to randomly generate high-fidelity palmprints with diverse lighting and skin types by adjusting the input noise vector $N(z)$.

Quantitative results are shown in Tab.6. Our method substantially outperforms the existing methods in terms of TAR@FAR. Also, we use FID [8] to evaluate the quality of generated results and our method also effectively decreases the FID score by more than 50% against other methods. Besides, we reproduced some other generation methods [54, 56] for unpaired image-to-image transfer, including diffusion-based models, but the generated results are unsat-
isfying. Related details and more subjective results can be found in supplementary material.

6.3. Number of Synthesized Identities and Images

In this ablation, we investigate the influence of the number of synthesized identities and images. Specifically, we generate 4000 identities and 100 samples by default, and fix one number and vary the other under the open-set protocol (train:test=1:1). The results are shown in Fig.8. With the increase of “width” and “depth” of synthetic data, our method can continuously improve the performance of the recognition model and achieve a higher upper bound than BézierPalm. The performance reaches the upper bound with 80k synthetic identities and 160 samples per identity.

Figure 8: TAR@FAR=1e-6 of recognition models pre-trained with different numbers of synthetic identities and samples. The backbone is MobileFaceNet.

7. Conclusion

This paper proposed an ID-aware conditional modulation generation model which can produce realistic and diversified palmprint images. Specifically, a conditional modulation generator was designed, which adopted synthetic palm creases as condition, and used encoded Gaussian noise vector to modulate the generated diversity. An ID-aware loss was proposed to preserve the identity information of input palm creases during the unpaired training process. In the forward pseudo-palmprint generation stage, we improved the Bézier curves generation strategy to produce more realistic synthetic palm creases. From experimental results, we can obtain the following findings. Firstly, the generated pseudo-palmprint samples can effectively improve the performance of palmprint recognition models. Secondly, by using the synthetic palmprints, our method can effectively reduce the dependence of real data by 90%. In future work, we hope to implement complete real-data-free palmprint recognition with synthetic data.

8. Acknowledgements

This work is partly supported by the grants of the National Natural Science Foundation of China, Nos.62076086, 62272142 and 61972129.

References


