Diff-Palm: Realistic Palmprint Generation with Polynomial Creases and Intra-Class Variation Controllable Diffusion Models

Supplementary Material

This supplementary materials provide the following contents:

- an overview of datasets used: both public datasets and anonymous datasets.
- evaluation metrics.
- additional experimental results.
- discussions on proposed sampling methods and validation approaches.

1. Datasets

1.1. Public Datasets

We utilize seven publicly available palmprint datasets, including, CASIA [S11], PolyU [S13], Tongji [S14], MPD [S16], XJTU-UP [S10], IITD [S7], and NTU-CP-v1 [S9], with detailed information provided in Tab.1 and example images shown in Fig.1. Following the open-set protocol, we divide the first five palmprint datasets into training and testing sets in a 1:1 ratio based on the number of IDs, ensuring no overlap between the IDs in the training and testing sets. Due to the limited number of images in the other two datasets, we used them exclusively for training.

1.2. Collected Anonymous Dataset

We employ keywords such as "hand," "palm," and "palm print" to search for images on the Internet using search engines. After obtaining these images, we utilize Mediapipe [S8] to detect the presence and completeness of palms in the images. Following this filtering process, we apply the detect-then-crop protocol in [S15] to extract the Region of Interest (ROI) of the palmprints. We have acquired 48,000 complete palmprint ROI images, which are then used to train our generative model. Example images are shown in Fig.2. Due to relevant privacy protection regulations, we will release the URLs to these images.

Datasets	#ID	#Images	Devices
CASIA	620	5502	Digital camera
PolyU	388	7738	Scanner
Tongji	600	12,000	Digital camera
MPD	400	16,000	Mobile phone
XJTU-UP	200	7900	Mobile phone
IITD	460	2601	Digital camera
NTU-CP-v1	652	2390	Digital camera

Table 1. Details of the seven public palmprint datasets.

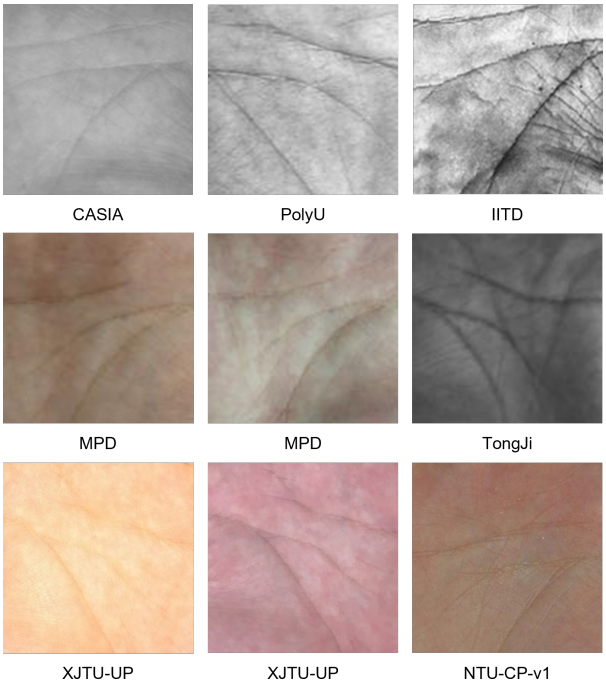

Figure 1. Example images of public palmprint datasets.

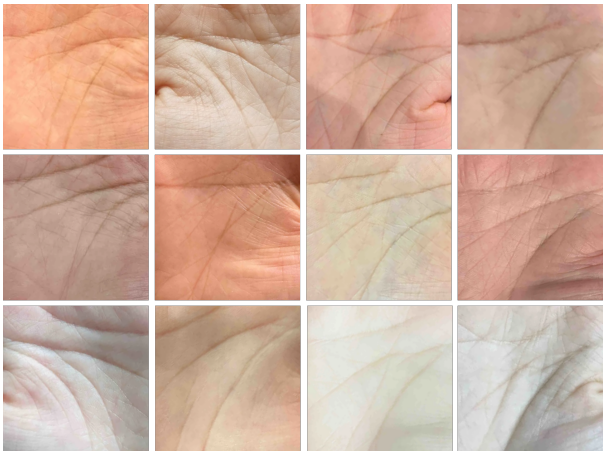

Figure 2. Example images of collected anonymous datasets.

2. Evaluation Metrics

2.1. Performance Metrics

we adopt the TAR(True accept ratio)@FAR(False accept ratio) metric, which is a widely used metric in open-set recognition tasks. It quantifies the system's ability to correctly ac-

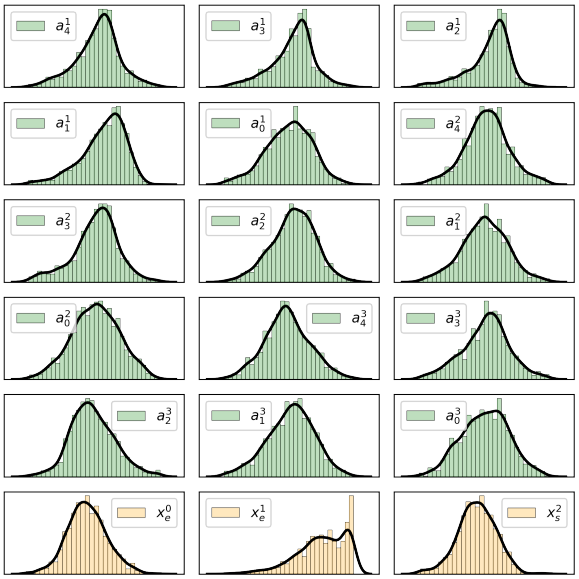

Figure 3. Histograms of the 15 polynomial coefficients and the x-coordinates for 3 endpoints

cept genuine instances while controlling the rate of falsely accepted impostors. To compute TAR@FAR, one first determines the threshold t under the specific FAR, which represents the proportion of non-genuine instances incorrectly accepted as genuine. At this threshold t, the TAR is calculated as the proportion of genuine instances correctly accepted, as follows,

$$\mathrm{TAR}(t) = \frac{\text{Number of True Acceptances}}{\text{Total Number of Genuine Attempts}}.$$

2.2. Datasets Evaluation Metrics

We utilize three metrics derived from[S6] to evaluate the synthesized dataset. Specifically, we employ a recognition model F_{eval}, pre-trained on a real dataset to extract features from each image in the dataset. For the image X_i^c, i-th sample within c-th label, we denote its feature as $f_i^c = F_{eval}(X_i^c)$. We use cosine similarity to measure the distance between two samples. Additionally, we denote the feature of the center of each class as $\bar{f^c}$ for $c \in \{1, \dots C\}$, which is also the spherical mean of the samples within the same label.

Class Uniqueness. We first define U_c as follows,

$$U_c = \{\bar{f^c} : d(\bar{f^{c_n}}, \bar{f^{c_m}}) > r, m < n, n, m \in \{1, \dots C\}\},$$

where $d(\cdot,\cdot)$ is the cosine distance. The U_c is the set of unique subjects determined by the threshold r. For this metric, we define $U_{class} := \mid U_c \mid /C$, the ratio between the number of unique subjects and the number of total labels.

Identity Consistency. To measure how consistent the synthesized samples are in adhering to the label condition, we define $C_{identity}$ as

$$C_{identity} = \frac{1}{C}\sum_{c=1}^{C}\frac{1}{N_c}\sum_{i=1}^{N_c} d(f_i^c, \bar{f^c}) < r,$$

which is the ratio of individual features f_i^c being close to the class center f^c. For a given threshold r, higher values of $C_{identity}$ mean the samples under the same label are more likely to be the same subject.

Intra-class Diversity. We aim to measure how diverse the generated samples are under the same label condition, as well as the diversity is in the style of an image, not in the subject's identity. In the original paper [S6], the Inception Network pre-trained on ImageNet is utilized to extract the style information of the images. However, since palmprints are significantly different from the images in ImageNet, and are often simple and relatively uniform, we employ a pixel-based diversity measure. Specifically, we adopt $\bar{X^c} = \frac{1}{N_c}\sum_{i=0}^{N_c} X_i^c$ denotes the mean image of c class, and diversity is defined as:

$$D_{intra} = \frac{1}{C}\sum_{c=1}^{C}\frac{1}{N_c}\sum_{i=1}^{N_c} \|X_i^c - \bar{X^c}\|_1,$$

where $\|\cdot\|_1$ denotes $L1$ norm. We take the D_{intra} value of datasets generated by Diff-Palm with $K = 0$ as the baseline, normalizing it to 1.0, and adjusting all other values accordingly.

3. Additional Experimental Results

3.1. Histogram of Polynomial Coefficients

We use three polynomial curves to mark the three main lines of the palmprint. Each polynomial curve contains five coefficients. Therefore, we plot the histograms for all 15 coefficients, as shown in Fig.3. Additionally, we conduct a statistical analysis of the x-coordinates for the endpoints of 3 palm lines.

3.2. Performance on Individual Public Datasets

We conduct recognition experiments on individual public datasets. The experimental results are presented in the first section of Tab.2. The performance achieved on individual public datasets is significantly lower compared to that on mixed public datasets.

3.3. Further Fine-Tuning Experiments

We adopt Diff-Palm to generate datasets with a large number of IDs. These datasets are used to pre-train the recognition model, which is subsequently fine-tuned using real datasets. As shown in the second section of Tab.2, the performance of our method consistently improves after fine-tuning.

Methods	Configs			Performance (TAR@FAR=1e-6) ↑					
	#IDs	#Images	FT w/ Real	CASIA	PolyU	TongJi	MPD	XJTU-UP	*Avg.*
CASIA	310	2510	✗	0.7243	0.7198	0.5952	0.0964	0.1963	0.4664
PolyU	194	3869	✗	0.5235	0.7574	0.4322	0.0746	0.1402	0.3856
TongJi	300	6000	✗	0.7344	0.7115	0.8032	0.1476	0.0977	0.4989
MPD	200	8000	✗	0.8839	0.8254	0.8599	0.3745	0.2939	0.6455
XJTU-UP	100	3950	✗	0.7741	0.7518	0.7183	0.1805	0.4549	0.5759
IITD	460	2601	✗	0.3895	0.5205	0.1024	0.0397	0.0681	0.2240
NTU-CP-v1	652	2390	✗	0.6652	0.8330	0.7970	0.1628	0.2588	0.5434
Diff-Palm	5k	100k	✓	0.9783	0.9850	0.9848	0.6754	0.8966	0.9040
Diff-Palm	10k	200k	✓	0.9857	0.9910	0.9905	0.7400	0.9442	0.9303
Diff-Palm	20k	400k	✓	0.9832	0.9943	0.9920	0.7922	0.9484	0.9420
Diff-Palm	30k	600k	✓	0.9870	0.9945	0.9941	0.8044	0.9570	0.9474
Diff-Palm	40k	800k	✓	0.9827	0.9949	0.9936	0.8179	0.9593	0.9497
Diff-Palm	50k	1M	✓	0.9843	0.9965	0.9933	0.8498	0.9700	0.9588
Diff-Palm(10k)	2k	40k	✗	0.8598	0.9472	0.9113	0.4124	0.6133	0.7488
Diff-Palm(48k)	2k	40k	✗	0.8782	0.9601	0.9460	0.4643	0.6161	0.7729
Real data(R50)	2.2k	29.3k	✗	0.9429	0.8927	0.9449	0.3974	0.6111	0.7577
IDiff-Face(R50) [S1]	2k	40k	✗	0.7944	0.7404	0.6208	0.1793	0.2734	0.5217
PCE-Palm(R50) [S5]	2k	40k	✗	0.5749	0.7188	0.5835	0.2738	0.4347	0.5171
Diff-Palm(R50)	2k	40k	✗	0.8708	0.9558	0.9577	0.4472	0.6266	0.7716
Real data(MBF)	2.2k	29.3k	✗	0.9323	0.9071	0.9247	0.3616	0.6067	0.7466
IDiff-Face(MBF) [S1]	2k	40k	✗	0.7732	0.7445	0.6507	0.1667	0.2259	0.5120
PCE-Palm(MBF) [S5]	2k	40k	✗	0.6112	0.7379	0.5266	0.2695	0.4013	0.5093
Diff-Palm(MBF)	2k	40k	✗	0.8546	0.9489	0.9409	0.4297	0.6177	0.7584
Real data(ViT)	2.2k	29.3k	✗	**0.8279**	0.6909	0.7390	0.2132	0.2826	0.5505
IDiff-Face(ViT) [S1]	2k	40k	✗	0.6676	0.5968	0.5304	0.1234	0.1783	0.4193
Vec2Face(ViT) [S12]	2k	40k	✗	0.6967	0.5075	0.4406	0.0962	0.1108	0.3704
PCE-Palm(ViT) [S5]	2k	40k	✗	0.6012	0.4918	0.4604	0.1303	0.1515	0.3670
Diff-Palm(ViT)	0.6814	**0.7920**	**0.7792**	**0.2798**	**0.3754**	**0.5816**			

Table 2. Comparsion performance of recognition models trained on various datasets. Results are reported in TAR@FAR=1$e-6$. 'R50', 'MBF' and 'ViT' represent ResNet-50[S3], MobileFaceNet[S2] and ViT-t [S4], respectively

3.4. Polynomial Creases Similarity Control

We generate a polynomial creases dataset and control the overall similarity using different γ. As illustrated in Fig.4, we can observe that when γ is less than 1.0, the similarity of the generated polynomial creases increases. Conversely, when γ is greater than 1.0, the overall similarity decreases, and the generated creases become more random.

3.5. Additional Ablation Experiments

Smaller Anonymous Datasets We have collected an anonymous dataset containing 48,000 images, which we used to train the generative model. We also experiment with training Diff-Palm using a smaller dataset of 10,000 images. The experimental results are presented in the third section of Tab.2. We observe that the performance of Diff-Palm trained with 10,000 images is inferior to that trained with 48,000 images. However, it still achieves results comparable to those obtained with real datasets.

Different Recognition Backbone We conduct comparative experiments using different recognition backbones (modified Resnet-50 [S3], MobileFaceNet [S2] and ViT-t [S4]). The experimental results are shown in the last three sections of Tab.2. We arrive at the same conclusions as those in the main paper.

4. More Discussion

4.1. K-Step Noise-Sharing Sampling

To verify that K-step noise-sharing sampling can be applied to other diffusion-based methods, we use the offi-

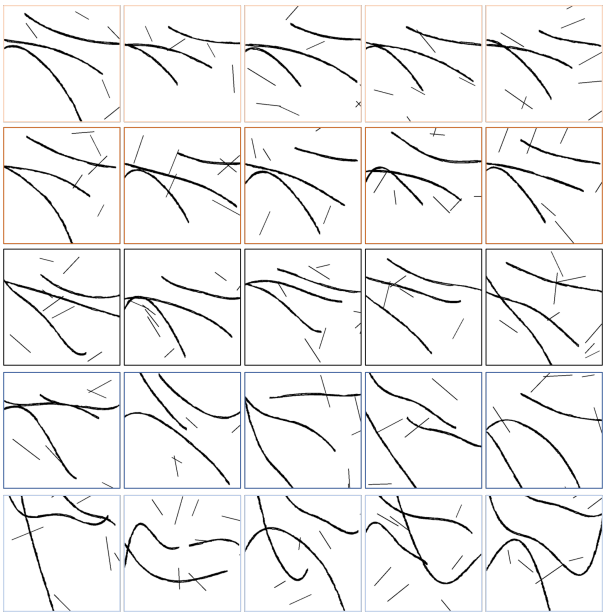

Figure 4. Synthesized polynomial crease images with varying γ. From top to bottom, the γ is set to 0.25, 0.5, 1.0, 2.0, and 4.0, respectively

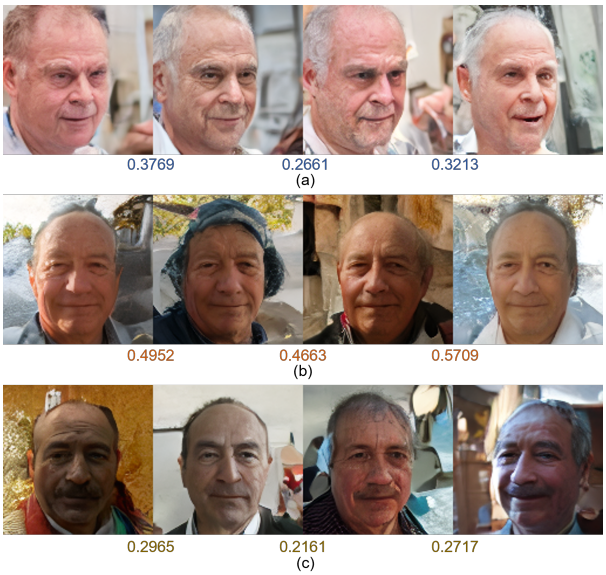

Figure 5. Synthesized face images by IDiff-Face[S1], (a) with noise-sharing in first 500 sampling steps, (b) with noise-sharing in last 500 sampling steps, (c) without noise-sharing, as well as cosine similarity calculated between adjacent face images

cially released pre-trained IDiff-Face model [S1] and apply our K-step noise-sharing sampling to obtain facial images, as shown in Fig.5. We set $K = 500$ with a total step of $T = 1000$. Each row of images is generated from the same ID condition. Moreover, we employ a pre-trained facial recognition model to extract features from each image and calculate the cosine similarity between adjacent face images. It is evident that applying K-step noise-sharing sampling significantly enhances the identity consistency of the generated results. Additionally, applying noise-sharing in the last K steps further improves the identity consistency of the generated outcomes.

4.2. Validation Set

In PCE-Palm, they first split several public datasets into training and testing sets in a 1:1 ratio and then mixed all the testing sets from the public datasets for evaluation with the trained recognition model. However, due to different collection devices, environments, etc., the various public datasets have significant style differences. When the recognition model is tested on the mixed testing set, it is easy to distinguish an identity from one dataset from identities in other datasets. In contrast, we adopt a more general validation approach. After splitting the public data into training and testing sets in a 1:1 ratio, we validate each dataset separately. Subsequently, we average the validation results from each dataset.

References

[S1] Fadi Boutros, Jonas Henry Grebe, Arjan Kuijper, and Naser Damer. Idiff-face: Synthetic-based face recognition through fizzy identity-conditioned diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19650–19661, 2023. 3, 4

[S2] Sheng Chen, Yang Liu, Xiang Gao, and Zhen Han. Mobilefacenets: Efficient cnns for accurate real-time face verification on mobile devices. In *Chinese Conference on Biometric Recognition*, pages 428–438. Springer, 2018. 3

[S3] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*, pages 4690–4699, 2019. 3

[S4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ArXiv*, abs/2010.11929, 2020. 3

[S5] Jianlong Jin, Lei Shen, Ruixin Zhang, Chenglong Zhao, Ge Jin, Jingyun Zhang, Shouhong Ding, Yang Zhao, and Wei Jia. Pce-palm: Palm crease energy based two-stage realistic pseudo-palmprint generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 2616–2624, 2024. 3

[S6] Minchul Kim, Feng Liu, Anil Jain, and Xiaoming Liu. Dcface: Synthetic face generation with dual condition diffusion model. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 12715–12725, 2023. 2

[S7] Ajay Kumar. Incorporating cohort information for reliable palmprint authentication. In *2008 Sixth Indian conference on*

computer vision, graphics & image processing, pages 583–590. IEEE, 2008. 1

[S8] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. Mediapipe: A framework for building perception pipelines, 2019. 1

[S9] Wojciech Michal Matkowski, Tingting Chai, and Adams Wai Kin Kong. Palmprint recognition in uncontrolled and uncooperative environment. *IEEE TIFS*, 2019. 1

[S10] Huikai Shao, Dexing Zhong, and Xuefeng Du. Effective deep ensemble hashing for open-set palmprint recognition. *Journal of Electronic Imaging*, 29(1):013018, 2020. 1

[S11] Zhenan Sun, Tieniu Tan, Yunhong Wang, and Stan Z Li. Ordinal palmprint represention for personal identification [represention read representation]. In *CVPR*, volume 1, pages 279–284. IEEE, 2005. 1

[S12] Haiyu Wu, Jaskirat Singh, Sicong Tian, Liang Zheng, and Kevin W. Bowyer. Vec2face: Scaling face dataset generation with loosely constrained vectors. *ICLR*, 2025. 3

[S13] David Zhang, Wai-Kin Kong, Jane You, and Michael Wong. Online palmprint identification. *IEEE Transactions on pattern analysis and machine intelligence*, 25(9):1041–1050, 2003. 1

[S14] Lin Zhang, Lida Li, Anqi Yang, Ying Shen, and Meng Yang. Towards contactless palmprint recognition: A novel device, a new benchmark, and a collaborative representation based identification approach. *Pattern Recognition*, 69:199–212, 2017. 1

[S15] Yingyi Zhang, Lin Zhang, Xiao Liu, Shengjie Zhao, Ying Shen, and Yukai Yang. Pay by showing your palm: A study of palmprint verification on mobile platforms. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 862–867, 2019. 1

[S16] Yingyi Zhang, Lin Zhang, Ruixin Zhang, Shaoxin Li, Jilin Li, and Feiyue Huang. Towards palmprint verification on smartphones. *arXiv preprint arXiv:2003.13266*, 2020. 1