CompNet: Competitive Neural Network for Palmprint Recognition Using Learnable Gabor Kernels

Xu Liang, Jinyang Yang, Guangming Lu, and David Zhang, *Life Fellow, IEEE*

***Abstract*—Contactless palmprint recognition has recently made significant progress in palm-scanning payment and social security. However, most existing methods are based on handcrafted kernels and are sensitive to illumination and scale variations. To address this problem, a competitive convolutional neural network (CompNet) with constrained learnable Gabor filters is proposed for contactless palmprint recognition. The proposed CompNet is built on multisize competitive blocks, which are applied to effectively exploit the rich direction ordering information of the palmprint patterns by means of the ad-hoc softmax and channel-wise convolution operations. Compared to the current deep neural networks, the backbone of the proposed network contains only very few parameters, making it quite easy to train, especially on small-scale datasets. Experimental results obtained on four popular contactless palmprint datasets demonstrate that the proposed CompNet achieves the lowest equal error rate compared to the most commonly used methods.**

***Index Terms*—Biometrics, palmprint recognition, competitive feature encoder, competitive block, learnable Gabor filter.**

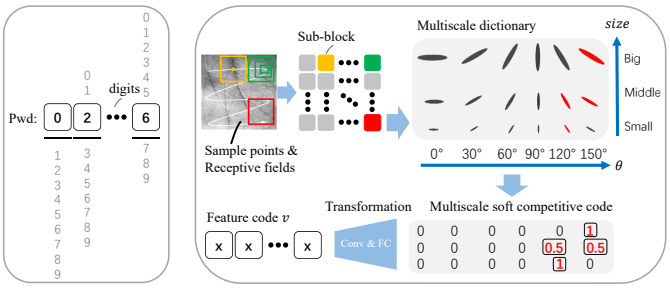
I. Introduction

In recent years, with the increasing consideration of social security and user privacy issues, palmprint recognition has attracted increasing attention. Palmprint recognition considers rich features including principal lines, wrinkles, and skin textures, and has achieved high recognition accuracy [1]. Multimodal palm images can also be utilized to further increase recognition accuracy [2]–[4]. However, several challenges remain. For example, in the contactless mode, palmprint images are captured in an open environment; thus, they can suffer interference from scale variations and unstable illuminance. Therefore, designing a feature extraction method that is robust against such factors is important for real-world applications.

This work was supported in part by the Guangdong Basic and Applied Basic Research Foundation under Grant 2019B1515120055, in part by the Shenzhen Key Technical Project under Grant 2020N046, in part by the Shenzhen Fundamental Research Fund under Grant JCYJ20180306172023949 and Grant JCYJ20170412170438636, in part by Natural Science Foundation of China under Grant 61772296, in part by the Open Project Fund (AC01202005018) from Shenzhen Institute of Artificial Intelligence and Robotics for Society, in part by the Shenzhen Research Institute of Big Data, and in part by the Medical Biometrics Perception and Analysis Engineering Laboratory, Shenzhen, China. *(Corresponding authors: Guangming Lu; David Zhang.)*

Xu Liang, Jinyang Yang, and Guangming Lu are with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, Shenzhen 518055, China (e-mail: xuliangcs@gmail.com; 18s051015@stu.hit.edu.cn; luguangm@hit.edu.cn).

David Zhang is with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, Shenzhen 518055, China, and the Shenzhen Institute of Artificial Intelligence and Robotics for Society, Shenzhen 518172, China, and The Chinese University of Hong Kong, Shenzhen, Shenzhen 518172, China (e-mail: davidzhang@cuhk.edu.cn).



(a) (b)

Fig. 1: Comparison of the (a) traditional password and (b) palmprint two-dimensional (2D) password.

Over the last few decades, studies have proposed different methods to address these problems. For example, Kong and Zhang [5] proposed a competitive coding scheme (CompCode) to encode the palmprint features with several Gabor filters in different directions. Inspired by CompCode, Sun *et al.* [6] designed orthogonal Gaussian filters to extract ordinal codes of the palmprint, and Jia *et al.* [7] utilized principal lines to extract the palmprint feature by applying modified finite Radon transformation (MFRAT). Guo *et al.* [8] proposed the binary orientation co-occurrence vector algorithm to extract multidirections for a single palmprint block. Similarly, different local line-based methods have been proposed, such as DOC [9], ALDC [10], and LLDP [11]. However, traditional coding-based algorithms are highly dependent on handcrafted convolution filters. Consequently, various machine learning-based methods including DDBC [12], SDDLM [13], and LRR [14] have been designed to improve the performance of feature extraction. After obtaining the histograms of the Gabor responses, in [15], CR_CompCode is calculated via sparse representation [16]. To learn more model parameters, CNN-based methods have been utilized for both palmprint recognition and image enhancement [17]–[24]. Unfortunately, the traditional CNN framework tends to utilize the convolution responses of the image patterns to make decisions. However, palmprint images are low-resolution images. Due to variations in the ambient light and defocus blur, the captured palmprint images are generally not sufficiently stable in brightness, scale, and sharpness. Thus, their responses are not stable, and a framework to extract robust features from low-quality images is required (see Fig. 1).

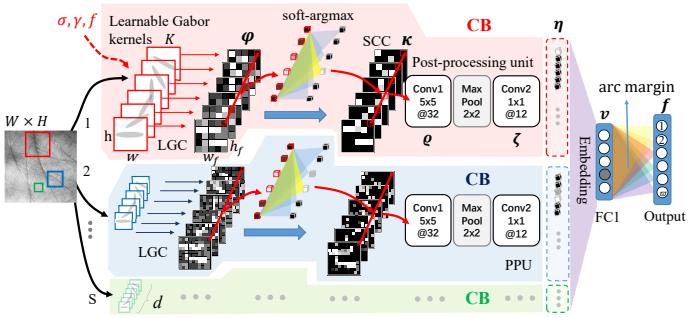


Fig. 2: Architecture of the proposed CompNet.

Motivations: (1) The conventional competitive code attempts to extract features from the index of the Gabor filter rather than the magnitude response. Consequently, it may be more resistant to changes in lighting. However, because this method lacks learning ability, it is sensitive to scale variations. To that end, we use an ad-hoc softmax layer to combine the competitive code's robustness with the advantages of CNN. (2) Furthermore, conventional competitive code only employs the winner-take-all strategy to extract features. Each sampling point's directional responses yield only a single value. On the one hand, winner-take-all is only one relationship between channel responses. On the other hand, because the operation only occurs at one sampling point, the correlation between spatial sampling points is ignored. Therefore, we aim to improve this strategy by incorporating more channel-wise and spatial information to exploit high-level features, such as palmprint line intersections, to achieve higher robustness and accuracy[1].

Contributions: (1) We introduced a soft-argmax layer to extract the ordering relation between channels based on the learnable Gabor convolution (LGC) layer responses, making the network be resistant to illumination changes. (2) Furthermore, a channel-wise feature extractor is designed as a postprocessing unit (PPU) to further process the ordering features, which can improve recognition accuracy via high-level knowledge extraction. (3) The multisize competitive blocks (CBs) are proposed to learn more scale information of different palmprint patterns.

II. Proposed Method

A. Architecture of the Proposed CompNet

The proposed CompNet utilizes CBs to extract the distribution of the main orientations of each image patch (see Fig. 1). As shown in Fig. 2, the orientation responses of the constrained LGC layer are encoded using the channel-wise soft-argmax operation to generate a sparse vector. The vector represents their direction-based ordering relationship. In the vector, the value of the direction, from which the filter response reached the highest, tends to one, whereas the other elements tend to zero. Then, the ordering relationships are further processed using a PPU module, comprising of two convolutional layers and a Maxpool layer, for extracting high-level features (denoted $\boldsymbol{\zeta}$). After postprocessing, the output of each CB is flattened and concatenated sequentially as a high-dimensional feature vector $\boldsymbol{\eta}$, similar to a palmprint password. However, this password contains noise and variations due to the complex capture conditions; thus, an effective dimensionality reduction layer is implemented to finely remap the feature vector $\boldsymbol{\eta}$. In the output layer, an angular margin [25] is applied to ensure that the remapped feature vectors (denoted $\boldsymbol{v}$) are sufficiently compact for intraclass samples. Note that a large receptive field is robust against misalignment, and a small receptive field is effective for describing the details of the palmprint. Thus, CBs with receptive fields of different sizes are employed in the proposed CompNet.

[1]Supplementary Materials at https://github.com/xuliangcs/compnet

B. Learnable Gabor Convolution Kernel

Many filters have been developed to extract line patterns, such as the neurophysiology-based Gabor [5], orthogonal Gaussian [6], and MFRAT [11] filters. In the proposed CompNet, we modify a learnable Gabor filter from the literature [26]. The Gabor function employed in the proposed CompNet is modified as follows:

$$G(x, y; \sigma, \gamma, f, \psi, \theta) = -\exp\{-\frac{\gamma^2 x'^2 + y'^2}{2(2\sqrt{2}\sigma)^2}\}\cos(2\pi f x' + \psi) \tag{1}$$

where $x' = xcos\theta + ysin\theta$, and $y' = -xsin\theta + ycos\theta$. f, θ, ψ, σ, and γ are the parameters of the Gabor function. Here σ and γ are the scale factor and spatial aspect ratio of the Gaussian function, respectively. f and ψ are the frequency and phase offset of the cosinusoidal carrier wave, respectively. In addition, θ is the orientation of the Gabor function, x and y are coordinate variables, and $2\sqrt{2}$ is a constant factor. From Eq. (1), when $f \to 0, \psi \to 0$, the Gabor function becomes a Gaussian function, and when $\gamma \to 0, f \to 0, \psi \to 0$ the shape of the Gabor filter can be a smooth version of MFRAT. Thus, with different parameters, the Gabor function can have different kinds of basic shapes, which is the reason why the Gabor function is employed as the basic pattern extractor in the CB in the proposed CompNet. The Gabor kernel K in the proposed CompNet is defined as follows:

$$\begin{cases} J(i,j) = G(i - \dfrac{h-1}{2}, j - \dfrac{w-1}{2}), \\ K(i,j) = J(i,j) - \dfrac{1}{w * h}\displaystyle\sum_{m=0}^{h-1}\sum_{n=0}^{w-1} J(m,n), \end{cases} \tag{2}$$

where i and j are pixel indices, and $i \in \{0, 1, ..., h-1\}, j \in \{0, 1, ..., w-1\}$. h and w are the height and width of the Gabor kernel, respectively. Considering the feature map generated by the previous layer (in Fig. 2, the previous layer is the input layer), the LGC layer first establishes kernels K according to Eq. (1) and (2) based on the learnable parameters σ, γ, f, ψ, and θ, and then generates its output using traditional 2D convolutions. The parameter update process is described in detail in the literature [26]. Taking σ as an example, after obtaining $\partial K/\partial\sigma$, using the chain rule the traditional backpropagation approch can be applied to update σ.

Constraints: the LGC layer is designed to realize optimal orientation filters to extract the direction ordering information of palmprint blocks. Each learnable Gabor filter can learn its own parameters (σ, γ, f, ψ, and θ) to improve the effectiveness of feature extraction and classification; however, overly flexible parameter learning will break the inherent competitive characteristics of the LGC. For example, according to the experimental results, using learnable θ will reduce the final accuracy. Thus, only constrained Gabor filters are utilized in the block. The LGC module comprises d directional Gabor filters denoted $G(x, y; \theta_k), k = 0, 1, ..., d-1$. Except for orientation angles, the filters in each LGC share the same receptive field $h \times w$, the same scale σ, and the same filter shape (γ, f, σ). Additionally, ψ is set to 0, and the fixed orientation angle of the k-th filter is $\theta_k = \pi \cdot k/d$.

C. Soft Competitive Code

As shown in Fig. 2, a CB comprises a LGC layer, a soft-argmax layer, and a PPU. The soft competitive code (SCC) is generated by the LGC and soft-argmax layers. For the LGC layer, each channel of the input feature maps is convoluted with the d directional filters and generates d corresponding feature maps (denoted $\boldsymbol{\varphi}$). The outputs of the LGC layer are expressed as follows:

$$\boldsymbol{\varphi}(c, i, j) = \sum_{y=-\frac{h}{2}}^{h/2} \sum_{x=-\frac{w}{2}}^{w/2} K(c, \frac{h}{2} + y, \frac{w}{2} + x) \cdot I(c, \frac{h}{2} + pi + y, \frac{w}{2} + pj + x), \tag{3}$$

where I is the input image. $c \in \{0, 1, \cdots, d-1\}$ is the channel index, and (i, j) are pixel indices of the output feature map. p is the convolution stride, and (x, y) are the shifts along the horizental and vertical directions, respectively. Differing from competitive code [5] in which only the index of the maximum response is recorded, the proposed CB reserves responses of all directions. Note that $\mathrm{softmax}$ is a continuous approximation of the $\mathrm{onehot}(\mathrm{argmax})$ operation; thus, all responses of the LGC are fed into a channel-wise $\mathrm{softmax}$ layer, *i.e.*, the soft-argmax layer, to generate $\mathrm{onehot}(\mathrm{argmax})$ vectors $\boldsymbol{\kappa}(:, i, j)$. The corresponding feature maps (as shown in Fig. 2) are named SCC maps. The SCC value at location (c, i, j) is defined as follows:

$$\boldsymbol{\kappa}(c, i, j) = \frac{e^{a \cdot (\boldsymbol{\varphi}(c,i,j) - b)}}{\sum_{c=1}^{d} e^{a \cdot (\boldsymbol{\varphi}(c,i,j) - b)}}, \tag{4}$$

where c is the index of the channel, and d is the number of channels the LGC contains. Considering the representation range of the machine's word length, in (4), learnable factors a and b are introduced to control the distribution of $\boldsymbol{\varphi}$.

D. Postprocessing Unit

After obtaining the SCC map, the PPU is implemented to discover high-level features of the competitive responses, such as the intersections of lines. After postprocessing, the output feature map $\boldsymbol{\zeta}$ is defined as: $\boldsymbol{\zeta} = \mathcal{F}(\boldsymbol{\varrho}) \circledast \mathbf{W_2} + \boldsymbol{b_2}$, where $\mathcal{F}(\cdot)$ is the Maxpool function with 2×2 filters, $\circledast$ is the convolution operation, and $\boldsymbol{\varrho} = \boldsymbol{\kappa} \circledast \mathbf{W_1} + \boldsymbol{b_1}$. Here, $\mathbf{W_1}$, $\boldsymbol{b_1}$, and $\mathbf{W_2}$, $\boldsymbol{b_2}$ are the parameters of layers Conv_1 and Conv_2, respectively. The flattened vector of the CB is denoted $\boldsymbol{\eta}$.

E. Feature Embedding and Matching

FC1 is a linear layer that maps $\boldsymbol{\eta}$ to a low-dimensional feature vector $\boldsymbol{\upsilon}$, which can be obtained by $\boldsymbol{\upsilon} = (\mathbf{W}^{\mathrm{T}}\boldsymbol{\eta} + \boldsymbol{b})/\|\mathbf{W}^{\mathrm{T}}\boldsymbol{\eta} + \boldsymbol{b}\|$, where $\mathbf{W}$ and $\boldsymbol{b}$ are parameters of FC1. If multisize CBs are used, the concatenated vector is $\boldsymbol{\eta} = [\boldsymbol{\eta_1}; \boldsymbol{\eta_2}; ...; \boldsymbol{\eta_s}; ...; \boldsymbol{\eta_S}]$, where s is the index of the CB, and S is the total number of CBs. Then, according to the arc-loss [25], the output of the proposed CompNet is $\boldsymbol{f} = \varepsilon \cdot \mathbf{W_o}^{\mathrm{T}} \cdot \boldsymbol{\upsilon} = \varepsilon \cdot [\cos\theta_0, \cos\theta_1, \cdots, \cos\theta_{\varpi-1}]^{\mathrm{T}}$, where $\mathbf{W_o}$ is the normalized weights of the output layer, ϖ is the number of classes and ε is a hyperparameter. In the *training* phase, FC1 is followed by a dropout layer, and an angular margin m is added to θ_{y_i}, where $y_i \in \{0, 1, ..., \varpi - 1\}$ is the class label of sample i. Here, softmax loss is utilized to train the proposed CompNet. Note that only $\boldsymbol{\upsilon}$ is utilized in the *testing* phase, and the matching distance between two feature vectors $\boldsymbol{\upsilon_1}$ and $\boldsymbol{\upsilon_2}$ are there arc distance: $dis(\boldsymbol{\upsilon_1}, \boldsymbol{\upsilon_2}) = \arccos(\boldsymbol{\upsilon_1} \cdot \boldsymbol{\upsilon_2})/\pi$.

III. Experiments and Discussions

A. Datasets and Experimental Settings

The popular Tongji [15], IITD [27], [28], REST [29], [30], and XJTU-UP [31], [32] datasets were used to evaluate the effectiveness of the proposed CompNet. Note that, the XJTU-UP database contains four datasets, *i.e.* IF, IN, HF, and HN. Same with [15], for the Tongji contactless palmprint dataset, Session-1 is set as the training set, and Session-2 is the test set. For the IITD, REST, IF, and IN datasets, the first three images of each palm were selected as the training set, and the remaining images were used as the test set. Here, the equal error rate (EER) and rank-1 recognition rate were utilized to measure the performance of different algorithms. Each sample in the probe set matches with each sample in the gallery set and generates a matching score. Then, the EER and rank-1 were calculated based on the matching scores. Note that the class-based score aggregation [11], [17] was not performed when calculating the EER in these experiments to reserve the bad matching scores caused by hard samples.

TABLE I: Rank-1 Recognition Rate (%) on the REST Dataset Obtained by Different CompNet Frameworks.

CB_{35}	CB_{17}	CB_7	CB_3	soft-argmax	*ab*	PPU	FC1	Dropout	FC[a]	Rank-1
✓				✓	✓	✓	✓	✓		83.87
✓	✓			✓	✓	✓	✓	✓		90.44
✓	✓	✓		✓	✓	✓	✓	✓		**92.51**
✓	✓	✓	✓	✓	✓	✓	✓	✓		91.59
✓	✓	✓			✓	✓	✓	✓		91.82
✓	✓	✓		✓		✓	✓	✓		91.13
✓	✓	✓		✓	✓		✓	✓		87.10
✓	✓	✓		✓	✓	✓	✓			91.93
✓	✓	✓		✓	✓	✓			✓	85.25

[a] Here FC=$\mathrm{FC1}_{512}$+BN+ReLU+$\mathrm{FC2}_{512}$+BN+ReLU+Dropout

TABLE II: EER (%) and the Rank-1 Recognition Rate (%) of Different Algorithms on Different Databases.

Refs.	Methods	EER					Rank-1				
		Tongji	IITD	REST	IF	IN	Tongji	IITD	REST	IF	IN
[5]	CompCode	0.177	4.969	10.491	1.433	1.765	**100**	97.87	88.48	98.91	98.55
[6]	OrdinalCode	0.280	5.515	11.518	1.693	2.399	99.98	97.87	86.64	98.32	97.75
[7]	RLOC	0.154	4.750	11.722	1.533	1.861	**100**	98.03	88.13	98.98	98.55
[11]	LLDP	0.991	5.951	13.556	1.491	1.861	99.65	97.87	85.83	99.27	98.69
[15]	CR_CompCode	0.442	3.931	10.829	1.665	2.969	98.78	90.75	68.89	95.55	92.68
[17]	PalmNet	0.630	4.477	12.711	1.336	1.402	99.97	98.28	90.09	**99.78**	**99.64**
Proposed	CompNet	**0.018**	**0.628**	**3.211**	**0.146**	**0.193**	**100**	**98.77**	**92.51**	99.56	**99.64**

B. Ablation Experiments

To test the necessity of each module in the proposed CompNet, we conducted several ablation experiments on the REST dataset. In Table I, $\mathrm{CB_n}$ implies that the filter size of the LGC is $n \times n$, and ab means whether a and b were introduced in Eq (4). In this experiment, different CB combinations were tested. As shown in Table I, $\mathrm{CB}_{35\times35} + \mathrm{CB}_{17\times17} + \mathrm{CB}_{7\times7}$ achieved the best rank-1. By analyzing the results, we found that the soft-argmax, PPU, simple FC1, and dropout layers were all necessary. Without the soft-argmax layer, the rank-1 decreased from 92.51% to 91.82%. When the simple FC1 layer was replaced by the complex FC module, the rank-1 accuracy decreased considerably (from 92.51% to 85.25%).

For orientation angles in the LGC layer, we found that when the learnable θs were used, the final accuracy worsened slightly (from 92.51% to 91.24%). Furthermore, when the angular margin was omitted, the rank-1 decreased from 92.51% to 89.52%. In summary, the selected framework for the proposed CompNet was the third one shown in Table I.

C. Recognition Performance

We compared our model to the most commonly used methods, including CR_CompCode [15] and PalmNet [17], and common coding-based algorithms like CompCode [5], OrdinalCode [6], RLOC [7], and LLDP [11]. Table II shows the palmprint recognition performance of different conventional algorithms on the datasets mentioned above, and the corresponding receiver operating characteristic (ROC) curves and genuine-impostor distributions are plotted in Figs. 3b-d. In this experiment, the feature template translation range [5] of CompCode, OrdinalCode, and RLOC is [-3:1:3]. We conclude that the proposed method achieved the best EER on all the datasets, whereas its rank-1 recognition rates was slightly lower than PalmNet on the IF dataset. As shown in Figs. 3c and 3d, the proposed method can greatly improve the distributions of the intraclass and interclass matching scores. The genuine matching distance is further reduced.

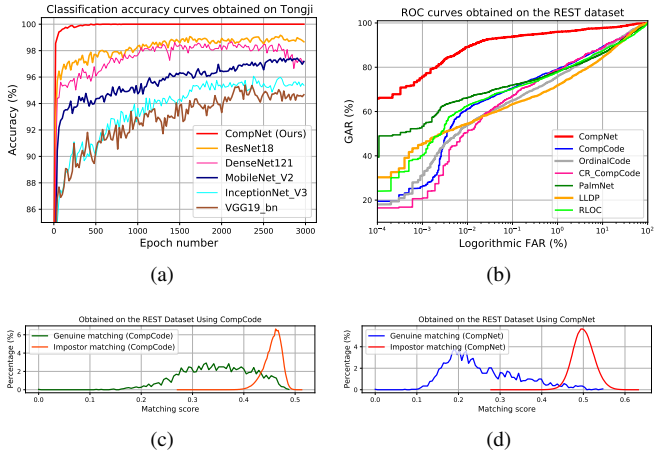


Fig. 3: Comparisons of different methods: (a) classification accuracy obtained at different numbers of epochs on the Tongji testing dataset; (b) ROC curves obtained through conventional methods; (c) matching score distributions obtained using CompCode on the REST dataset; and (d) distributions obtained using CompNet on the REST dataset.

The results of the DCNN-based models on the Tongji dataset are shown in Table III and Fig. 3a. Note that all experiments were implemented using PyTorch and conducted on a personal computer with a Nvidia GTX-2080Ti GPU. We trained all the DCNNs for 3000 epochs. Except the proposed CompNet, the other DCNN models were all pretrained on ImageNet. Fig. 3a shows CompNet outperforms the other DCNNs in convergence speed and classification accuracy.

TABLE III: Comparison of the CompNet and Different DCNN-based Methods on the Tongji Dataset with 600 Classes.

Methods	ACC (%)	GPU time (ms)	Model size (MB)
DenseNet-121 [33]	97.05	14.732	30.8
MobileNet-V2 [34]	97.13	5.784	**12.2**
Inception-V3 [35]	95.58	12.164	92.3
VGG-19bn [36]	94.95	3.273	568.2
ResNet-18 [37]	98.73	2.548	46.0
CompNet (Ours)	**100**	**2.153**	21.2

IV. Conclusion

In this paper, we have proposed a competitive convolution neural network that contains very few model parameters in the backbone due to a well-designed CB. Compared to the popular DCNN-based methods, the proposed CompNet converged considerably quickly and obtained the best classification accuracy on the Tongji dataset. Compared to conventional methods, the proposed CompNet obtained the best verification EER on the Tongji, IITD, REST, and XJTU-UP datasets, and the best rank-1 on the Tongji, IITD, and REST datasets.

REFERENCES

[1] D. Zhang, W.-K. Kong, J. You, and M. Wong, "Online palmprint identification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 9, pp. 1041–1050, Sept. 2003.

[2] S. Zhao, B. Zhang, and C.L. Philip Chen, "Joint deep convolutional feature representation for hyperspectral palmprint recognition," *Inf. Sci.*. vol. 489, pp. 167–181, Jul. 2019.

[3] X. Liang, D. Zhang, G. Lu, Z. Guo, and N. Luo, "A novel multicamera system for high-speed touchless palm recognition," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 51, no. 3, pp. 1534–1548, Mar. 2021.

[4] Z. Guo, D. Zhang, L. Zhang, and W. Liu, "Feature band selection for online multispectral palmprint recognition," *IEEE Trans. Inf. Forensics and Secur.*, vol. 7, no. 3, pp. 1094–1099, Jun. 2012.

[5] A. W.-K. Kong and D. Zhang, "Competitive coding scheme for palmprint verification," *in Proc. Int. Conf. Pattern Recognit.*, Aug. 2004, pp. 520–523.

[6] Z. Sun, T. Tan, Y. Wang, and S. Li, "Ordinal palmprint represention for personal identification," *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, pp. 279–284.

[7] W. Jia, D.-S. Huang, and D. Zhang, "Palmprint verification based on robust line orientation code," *Pattern Recognit.*, vol. 41, no. 5, pp. 1504–1513, May 2008.

[8] Z. Guo, D. Zhang, L. Zhang, and W. Zuo, "Palmprint verification using binary orientation co-occurrence vector," *Pattern Recognit. Lett.*, vol. 30, no. 13, pp. 1219–1227, Oct. 2009.

[9] L. Fei, Y. Xu, W. Tang, and D. Zhang, "Double-orientation code and nonlinear matching scheme for palmprint recognition," *Pattern Recognit.*, vol. 49, pp. 89–101, Jan. 2016.

[10] L. Fei, B. Zhang, W. Zhang, and S. Teng, "Local apparent and latent direction extraction for palmprint recognition," *Inf. Sci.*, vol. 473, pp. 59–72, Jan. 2019.

[11] Y.-T. Luo *et al.*, "Local line directional pattern for palmprint recognition," *Pattern Recognit.*, vol. 50, pp. 26–44, Feb. 2016.

[12] L. Fei, B. Zhang, and Y. Xu, "Learning discriminant direction binary palmprint descriptor," *IEEE Trans. Image Process.*, vol. 28, no. 8, pp. 3808–3820, Aug. 2019.

[13] S. Zhao and B. Zhang, "Learning salient and discriminative descriptor for palmprint feature extraction and identification," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 31, no. 12, pp. 5219–5230, Dec. 2020.

[14] L. Fei, Y. Xu, B. Zhang, X. Fang, and J. Wen, "Low-rank representation integrated with principal line distance for contactless palmprint recognition," *Neurocomputing*, vol. 218, pp. 264–275, Dec. 2016.

[15] L. Zhang, L. Li, A. Yang, *et al.*, "Towards contactless palmprint recognition: A novel device, a new benchmark, and a collaborative representation based identification approach," *Pattern Recognit.*, vol. 69, pp. 199–212, Sept. 2017.

[16] Z. Lai, Y. Xu, J. Yang, J. Tang, and D. Zhang, "Sparse tensor discriminant analysis," *IEEE Trans. Image Process.*, vol. 22, no. 10, pp. 3904–3915, Oct. 2013.

[17] A. Genovese, V. Piuri, K. N. Plataniotis and F. Scotti, "PalmNet: Gabor-PCA convolutional networks for touchless palmprint recognition," *IEEE Trans. Inf. Forensics Secur.*, vol. 14, no. 12, pp. 3160–3174, Dec. 2019.

[18] J. Svoboda, J. Masci and M. M. Bronstein, "Palmprint recognition via discriminative index learning," *in Proc. Int. Conf. Pattern Recognit.*, Dec. 2016, pp. 4232–4237.

[19] L. Zhang, Z. Cheng, Y. Shen, and D. Wang, "Palmprint and palmvein recognition based on DCNN and a new large-scale contactless palmvein dataset," *Symmetry*, vol. 10, no. 4, pp. 78, Mar. 2018.

[20] Y. Liu and A. Kumar, "Contactless palmprint identification using deeply learned residual features," *in IEEE Trans. Biom. Behav. Identity Sci.*, vol. 2, no. 2, pp. 172–181, Apr. 2020.

[21] S. Zhao and B. Zhang, "Deep discriminative representation for generic palmprint recognition," *Pattern Recognit.* vol. 98, Feb. 2020.

[22] W. M. Matkowski, T. Chai, and A. W.-K. Kong, "Palmprint recognition in uncontrolled and uncooperative environment," *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp 1601–1615, Oct. 2019.

[23] C. Tian, L. Fei, W. Zheng, Y. Xu, W. Zuo, and C.-W. Lin, "Deep learning on image denoising: An overview," *Neural Networks*, vol. 131, pp. 251–275, Nov. 2020.

[24] C. Tian, Y. Xu, W. Zuo, C.-W. Lin, and D. Zhang, "Asymmetric CNN for image super-resolution," *IEEE Trans. Syst., Man, Cybern. Syst.*, *Early Access*, May 2021. doi: 10.1109/TSMC.2021.3069265.

[25] J. Deng, J. Guo, N. Xue and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face recognition," *in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 4690–4699.

[26] P. Chen, W. Li, L. Sun, X. Ning, L. Yu, and L. Zhang, "LGCN: Learnable Gabor convolution network for human gender recognition in the wild," *IEICE Trans. Inf. Syst.*, vol. E102D no. 10, pp. 2067–2071, Oct. 2019.

[27] Indian Institute of Technology Delhi, "IIT Delhi Touchless Palmprint Database (IITD), version 1.0," 2008. [Online]. Available: https://www4.comp.polyu.edu.hk/~csajaykr/IITD/Database_Palm.htm.

[28] A. Kumar, "Incorporating cohort information for reliable palmprint authentication" *in Proc. Indian Conf. Comput. Vis. Graph. & Image Process.*, Dec. 2008, pp. 583–590.

[29] Research Groups in Intelligent Machines, ENIS, University of Sfax, Tunisia, "REgim Sfax Tunisian hand database (REST database)," 2016. [Online]. Available: https://ieee-dataport.org/open-access/rest-database.

[30] N. Charfi, H. Trichili, A. M. Alimi, B. Solaiman, "Local invariant representation for multi-instance toucheless palmprint identification" *in Proc. IEEE Int. Conf. Syst. Man Cybern.*, Oct. 2016, pp. 3522–3527.

[31] H. Shao, D. Zhong, and X. Du, "Cross-Domain Palmprint Recognition Based on Transfer Convolutional Autoencoder," *in Proc. IEEE Int. Conf. Image Process.*, Sept. 2019, pp. 1153–1157.

[32] H. Shao, D. Zhong, and X. Du, "Deep distillation hashing for unconstrained palmprint recognition," *IEEE Trans. Instrum. Meas.*, vol. 70, Jan. 2021. doi: 10.1109/TIM.2021.3053991

[33] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4700–4708.

[34] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," *in Prof. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.

[35] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2818–2826.

[36] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," Apr. 2015. [Online]. Available: arXiv:1409.1556v6.

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition" *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.