Joint Finger Valley Points-Free ROI Detection and Recurrent Layer Aggregation for Palmprint Recognition in Open Environment

Tingting Chai, *Member, IEEE*, Xin Wang, Ru Li, *Member, IEEE*, Wei Jia, *Member, IEEE*, and Xiangqian Wu, *Senior Member, IEEE*

***Abstract*—Cooperative palmprint recognition, pivotal for civilian and commercial uses, stands as the most essential and broadly demanded branch in biometrics. These applications, often tied to financial transactions, require high accuracy in recognition. Currently, research in palmprint recognition primarily aims to enhance accuracy, with relatively few studies addressing the automatic and flexible palm region of interest (ROI) extraction (PROIE) suitable for complex scenes. Particularly, the intricate conditions of open environment, alongside the constraint of human finger skeletal extension limiting the visibility of Finger Valley Points (FVPs), render conventional FVPs-based PROIE methods ineffective. In response to this challenge, we propose an FVPs-Free Adaptive ROI Detection (FFARD) approach, which utilizes cross-dataset hand shape semantic transfer (CHSST) combined with the constrained palm inscribed circle search, delivering exceptional hand segmentation and precise PROIE. Furthermore, a Recurrent Layer Aggregation-based Neural Network (RLANN) is proposed to learn discriminative feature representation for high recognition accuracy in both open-set and closed-set modes. The Angular Center Proximity Loss (ACPLoss) is designed to enhance intra-class compactness and inter-class discrepancy between learned palmprint features. Overall, the combined FFARD and RLANN methods are proposed to address the challenges of palmprint recognition in open environment, collectively referred to as RDRLA. Experimental results on four palmprint benchmarks HIT-NIST-V1, IITD, MPD and BJTU_PalmV2 show the superiority of the proposed method RDRLA over the state-of-the-art (SOTA) competitors. The code of the proposed method is available at https://github.com/godfatherwang2/ RDRLA.**

***Index Terms*—Palmprint recognition, palm ROI detection, recurrent layer aggregation, angular center proximity loss.**

Received 9 June 2024; revised 25 November 2024; accepted 2 December 2024. Date of publication 12 December 2024; date of current version 27 December 2024. This work was supported in part by the Natural Science Foundation of Shandong Province under Grant ZR2023QF030 and Grant ZR2024QF064; and in part by the National Natural Science Foundation of China under Grant 62402136, Grant 62076086, and Grant 62476077. The associate editor coordinating the review of this article and approving it for publication was Dr. Naser Damer. *(Corresponding authors: Ru Li; Xiangqian Wu.)*

Tingting Chai, Ru Li, and Xiangqian Wu are with the Faculty of Computing, Harbin Institute of Technology, Harbin 150001, China (e-mail: ttchai@hit.edu.cn; liru@hit.edu.cn; xqwu@hit.edu.cn).

Xin Wang is with the Faculty of Computing, Harbin Institute of Technology, Harbin 150001, China, and also with the School of Cyberspace Security, University of Science and Technology of China, Hefei 230026, China (e-mail: sa24221049@mail.ustc.edu.cn).

Wei Jia is with the School of Computer and Information, Hefei University of Technology, Hefei 230009, China (e-mail: jiawei@hfut.edu.cn).

Digital Object Identifier 10.1109/TIFS.2024.3516539

I. Introduction

PALMPRINT recognition, from verification on payment applications to identification on surveillance streams, plays an important role on enhancing the security and convenience of processes involving access control and financial services [1], [2], [3]. Given its flexibility and hygienic benefits, palmprint recognition is showing significant promise for broader implementation in open environment, as seen in applications like Amazon One[1] and the Beijing subway system.[2] The typical pipeline of a palmprint recognition system contains three core steps: PROIE, feature extraction, and feature matching (including one-to-one verification and one-to-all identification). PROIE is devoted to clipping the key part of the palm including abundant texture pattern from the raw image. It also adjusts the located sub-image into canonical orientation for consistent processing. Feature extraction involves mapping the palm ROI into a feature vector (embedding). The comparison of two palmprint images is conducted by assessing their embeddings that evaluates the level of identity resemblance between the two palms. Specially, such embeddings should exhibit compact intra-class variations and separable inter-class differences. Therefore, the main challenges in open-environment palmprint recognition are centered around the development of flexible PROIE and the learning of discriminative palmprint features, as shown in Figure 1.

The increasing need for robust security measures has propelled palmprint recognition to evolve, focusing on adaptable and trustworthy capabilities in complex scenarios. The remarkable success of deep learning (DL) has paved the way for widespread application [4], making DL-based methods the predominant approach in the field. Due to factors such as changing illumination, complex backgrounds, unrestricted hand placement, and heterogeneous palmprint images in open environment, cooperative palmprint recognition aimed at civilian and commercial use still has room for further improvement. It involves a series of steps where the effectiveness of feature extraction depends on the quality of PROIE. The accuracy largely relies on the neural network's ability to learn and

[1]https://www.aboutamazon.com/news/retail/amazon-one-whole-foods-market-palm-scanning

[2]https://www.globaltimes.cn/page/202305/1290687.shtml

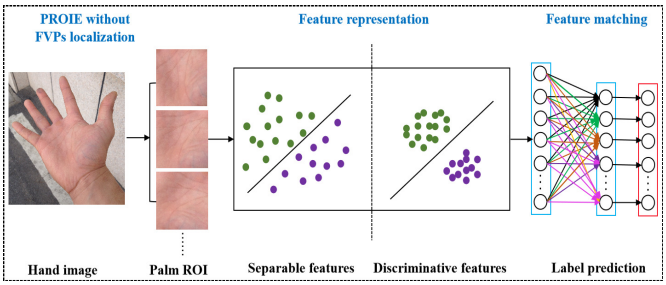


Fig. 1. Illustration of the challenges in palmprint recognition. Within the context of open environment, enhancements are required in the processes of PROIE, feature representation, and feature matching for palmprint recognition.

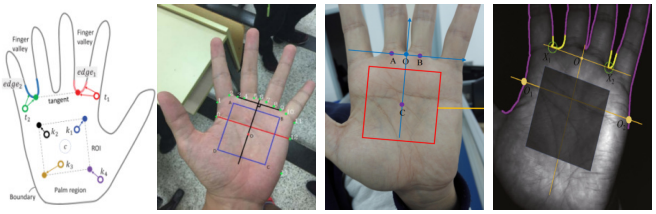


Fig. 2. The commonly used FVPs-based PROIE methods. The methods displayed above, from left to right, are taken from Refs. [5], [6], [7], and [8].

distinguish palmprint features effectively. While there has been a significant push to refine neural network architecture for better feature learning, the enhancement of PROIE efficiency and the formulation of specialized loss function for training palmprint images have not received comparable attention [5].

As shown in Figure 2, present PROIE methods are heavily contingent upon precise FVPs (also known as Key Reference Points, KRPs) localization. The ability to detect FVPs frequently suffers due to unconstrained hand placement or constraints in the human finger skeleton extension, making FVPs challenging to identify. Such limitations notably detract from the efficiency of PROIE techniques. The palmprint recognition accuracy is seriously affected by the poor PROIE quality. It should be noted that the terms KRPs and FVPs are nearly synonymous. Yet, the KRPs identified in the literature often encompass both the FVPs and certain points along the palm's perimeter [6]. For simplicity, these two cases are collectively referred to as FVPs in this work.

Our primary focus is centered on the discussion around the FVPs. To tackle the challenges outlined, we revisit the open-environment palmprint recognition solution. The main contributions can be summarized as follows.

- FFARD accomplishes two key tasks: Firstly, it facilitates unsupervised hand segmentation through CHSST. Secondly, it streamlines PROIE in an automated manner via the constrained inscribed circle search, eliminating the reliance on FVPs, which was a common requirement in previous methods.
- RLANN reuses the feature maps of convolutional layers through RLA, forming a powerful deep feature representation. Also, ACRLoss is proposed to enable higher inter-class separability and intra-class convergence by incorporating angular margin constraints and class center proximity.
- FFARD can be easily combined with current palmprint recognition methods. Exhaustive experiments on four palmprint benchmarks illustrate the superiority of the proposed method. Additionally, ablation study confirms the effectiveness of FFARD, RLANN architecture, ACPLoss and parameter setting.

The remainder of this work is outlined as follows. Section II reviews current existing techniques in the field of PROIE, DL-based palmprint recognition and angular margin-based loss. Section III elaborates on the proposed palmprint recognition method including hand segmentation, adaptive PROIE, RLANN and ACPLoss design. Comparative experiments and ablation study are discussed in Section IV. Section V gives conclusion to this work.

II. Related Work

A. PROIE Methods

The groundbreaking study of Zhang et al. [9] has meticulously defined the palm ROI, leading to a paradigm where most contemporary approaches employ FVPs, to establish a coordinate system to segment palm ROI based on empirical parameters. To cater to the requirements of open-environment applications, the academic sector has developed a suite of optimization algorithms for PROIE, grounded in the guidance information provided by these FVPs.

Izadpanahkakhk et al. [10] improved the Chatfield's design by including a unit with four neurons, dedicated to generating a bounding box. This modification allowed the four parameters that determined the center, height and width of the box to effectively encapsulate the palm region. ELSayed et al. [11] introduced a technique utilizing blob analysis along with straightforward morphological and geometrical processes to extract palm and knuckle ROI images, circumventing the need for training or parameter tuning. The described method, when applied to palmprint images against a black background, greatly simplifies hand segmentation and FVPs localization. Yan et al. [12] proposed a hand image composition method combining ROI harmonization and palm blending. The harmonization phase used a modified style transfer technique based on the pretrained Contrastive Arbitrary Style Transfer (CAST) network to adapt the attacking ROI's appearance to the carrier hand, achieving effective results in a few iterations.

Genuine efforts to investigate PROIE in open environment include the approaches proposed in [5], [6], and [7]. Liang et al. [5] proposed a Keypoint Coordinate Regression (KCR) module with an edge min-distance loss to predict FVPs' positions. The automatic annotation toolkit PalmKit required manual hand skin extraction to train a binary classifier for palm segmentation. Utilizing 14 manually annotated FVPs, Shao et al. [6] adopted a combination of sliding widow and regression tree for palm detection and PROIE. Through meticulous image annotation, Zhang et al. [7] utilized Tiny-YOLOv3 [13] for the detection of three FVPs. These points facilitated the creation of a point pair and a palm center, which were instrumental in establishing a coordinate system for PROIE.

While the previous methods have achieved commendable results, they all rely on segmenting the palm ROI after forming

a coordinate system based on FVPs, inevitably involving challenging manual annotation. This work seeks to address this limitation by developing an automatic and adaptive PROIE method that eliminates the necessity to identify FVPs.

B. DL-Based Palmprint Recognition Methods

Over the last decade, the swift evolution of DL models has significantly advanced palmprint recognition technology, addressing a host of challenges for practical applications in open environment. DL methods have surpassed hand-crafted methods to become the dominant approach in this field.

For the cross-dataset recognition issue, Shen et al. [14] developed a Progressive Target Distribution loss (PTD loss) to incrementally reduce the discrepancy in the representation of palmprint samples captured by different devices, thereby improving the compatibility and accuracy across varying datasets. Shao and Zhong [4] focused on multi-target cross-dataset palmprint recognition via teacher-student feature extractors. The former captured adaptive knowledge of source-target dataset pairs, and the latter was used to learn adaptive knowledge from the former. Rong et al. [15] proposed CGDNet, a whole palm-based recognition network with a trunk branch and a part exciting branch. The GS module enhanced global feature representation, while the CGD module refined fine-grained features within channel groups, using group-wise drop masks to boost feature learning and discrimination.

For the multi-modal features fusion issue, Fei et al. [16] proposed a unified spectrum-invariant feature representation method, which aimed to address the problem of low accuracy caused by different spectra of the gallery and the probe. Yang et al. [17] proposed a coordinate-aware contrastive competitive neural network (CO_3Net) designed to learn multiscale textures, with the contrastive loss employed to jointly optimize the network, thereby enhancing the accuracy of palmprint recognition. Zhao et al. [1] proposed TMLA_RHR, a robust hand-print recognition method using tensorized multi-view low-rank approximation and aligned structure regression loss for compact feature representation and reduced redundancy.

For palmprint data enhancement issue, Zhu et al. [18] proposed a self-paced CycleGAN with self-attention to generate missing training data by mining the structural correlation among cross-device samples. Wang et al. [3] proposed a dense hybrid attention (DHA) network for palmprint image super-resolution (SR). The DHA network extracts high-dimensional shallow features with a convolution layer and jointly learns local and global features using parallel convolutional neural network (CNN) and transformer-based branches. The data scarcity in palmprint recognition research results in insufficient fitting during DL model training. Consequently, research on pseudo-palmprint generation has emerged [19], [20]. Extensive experimental results show that synthetic pre-training significantly enhances recognition model performance.

Researchers have built upon prior work by developing methods that integrate palmprint-specific information with neural networks, utilize multi-modal features, and generate palmprint data to advance the field. Although these methods have produced promising experimental results, further research is needed to explore lightweight feature representation, network architectures with improved learning capability, and more efficient loss function design for easier training.

C. Angular Margin Penalty-Based Loss

Softmax loss [21], a common multi-class classification loss, faces limitations in enhancing feature representations for verification due to the lack of constraints on class center distances. To overcome this, angular margin penalty-based softmax loss (L_{AS}) was introduced as a superior alternative, aiming for better inter-class and intra-class distance optimization. Margin is added to the softmax loss to learn features more discriminative. Different works introduce different forms of margin functions. The general L_{AS} can be written as follows:

$$L_{AS} = \frac{1}{N}\sum_{i\in N} -log\frac{e^{s(\cos(m_1\theta_{y_i}+m_2)-m_3)}}{e^{s(\cos(m_1\theta_{y_i}+m_2)-m_3)}+\sum\limits_{j=1,j\neq y_i}^{c} e^{s(\cos\theta_j)}}, \quad (1)$$

where N represents the total number of training samples, while s serves as a scaling parameter for the feature space. The feature representation of the x_ith sample, along with its class label y_i, is denoted by $x_i \in R_d$. Furthermore, W_j corresponds to the jth column in the weight matrix of the final fully-connected (FC) layer. m_1, m_2 and m_3 denote the margin penalty parameters proposed by SphereFace [22], CosFace [23] and ArcFace [24], respectively. In SphereFace [22], the parameters are set as $m_1 = \alpha$, $m_2 = 0$ and $m_3 = 0$ (α¿1.0), leading to a decision boundary defined by $cos(m_1\theta_{y_i}) - cos(\theta_j) = 0$. Contrastingly, CosFace [23] specifies $m_1 = 1$, $m_2 = 0$ and $m_3 = \alpha$ (0‘!α‘!1 $- cos(\frac{\pi}{4})$), resulting in the decision boundary $cos(\theta_{y_i}) - cos(\theta_j) - m_3 = 0$. ArcFace [24] modifies the approach by setting $m_1 = 1$, $m_2 = \alpha$ and $m_3 = 0$ (0‘!α‘!1.0), which adjusts the decision boundary to $cos(\theta_{y_i} + m_2) - cos(\theta_j) = 0$.

The success of these loss functions has led to their variants being effectively utilized in face recognition [25], [26], [27], and expansion into palmprint recognition [28], showcasing their wide-ranging influence in the field of biometrics. Our ACPLoss incorporates an additional term to ArcFace for improving intra-class similarity, aiming to enhance proximity for positive samples towards their class center and reduce it for negative samples.

III. Proposed Method

To address the challenges of palmprint recognition in open environment, RDRLA is introduced, incorporating FFARD and RLANN. FFARD includes hand segmentation and adaptive PROIE.

A. Hand Segmentation

In open environment, hand images often showcase complex backgrounds and varied lighting conditions, leading

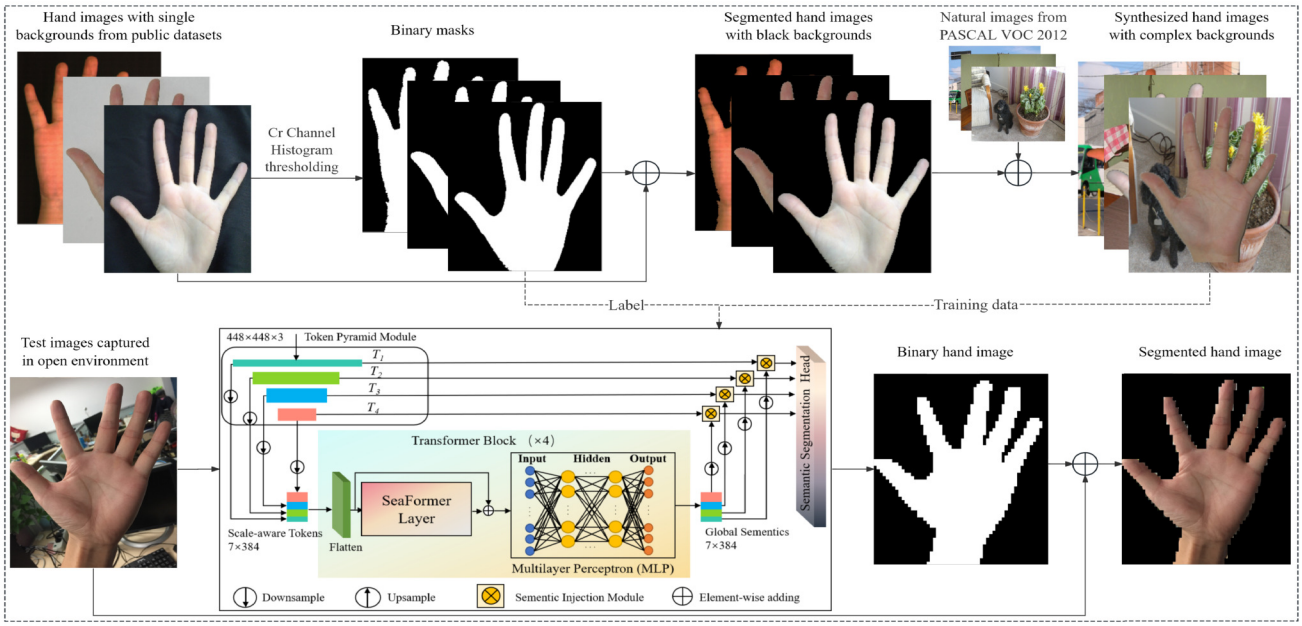


Fig. 3. Schematic representation of the proposed hand segmentation method CHSST.

to failures in thresholding segmentation methods and presenting a challenge for accurately distinguishing the hand foreground from the background. This issue is the primary motivation behind the development of CHSST. A flowchart illustrating the CHSST process is presented in Figure 3.

1) Automatic Training Data Annotation: Hand segmentation aims to label pixels corresponding to the hand shape region within the initial palmprint image. Relying on data annotation is impractical due to the significant amount of manual work required. Therefore, an automatic annotation approach is introduced, which operates as described below.

- The hand images with single backgrounds from five public palmprint datasets PolyU2D/3D1.0 [29], REST [30], COEP [31], BJTU_PalmV1 [32], NTU-CP-v1 [33] established in well-controlled environment are merged to be used as data basis.
- The YCbCr color space is produced through a linear transformation of the RGB color space. Following this, the Cr channel image is extracted from the YCbCr image. To create binary masks, identified as B, Otsu's method [34] is utilized for histogram thresholding of Cr channel. As a result, the segmented hand images with black background are obtained by selecting the relevant pixels from the original hand images.
- For the segmented hand images and the binary mask B, data augmentation is explored using an affine transformation with randomly set parameter. Here, (x, y) represents the pixel coordinates, while (x', y') indicates the amount of coordinates translation corresponding to (x, y). The transformation parameters are represented by W.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = W \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \omega_{11} & \omega_{12} & \omega_{13} \\ \omega_{21} & \omega_{22} & \omega_{23} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}. \quad (2)$$

The parameters $\omega_{13} \in [-\frac{w}{4}, \frac{w}{4}]$ and $\omega_{23} \in [-\frac{h}{4}, \frac{h}{4}]$ denote the horizontal and vertical translation pixels of each image, respectively. w and h separately indicate the width and the height of a hand image. The rotation angle θ and the constant parameter q are set to $[-\pi, \pi]$ and $[0.5, 1.5]$, respectively. The relationship of ω_{11}, ω_{12}, ω_{21} and ω_{22} can be formulated as follows.

$$\begin{aligned} \omega_{11} &= \omega_{22} = q\cos\theta, \\ \omega_{12} &= -q\sin\theta, \\ \omega_{21} &= q\sin\theta. \end{aligned} \quad (3)$$

- Following data augmentation, background images are randomly chosen from PASCAL VOC 2012 dataset [35] and combined with the segmented hand images to create new images featuring intricate backgrounds. These binary masks and the newly synthesized hand images serve as labels and training data, respectively, for the hand segmentation network. Specifically, the test data comprises the hand images captured in complex environments sourced from public palmprint datasets.

2) CHSST: Vision Transformers (ViTs) rapidly became the leading model for image classification, surpassing ConvNets as the top choice in the 2020s [36]. To address the high computational cost of ViTs, we employ the lightweight TopFormer [37], a ViT Token Pyramid Vision Transformer, which delivers better performance than MobileNets [38] with reduced latency for the hand segmentation task. To strike a balance between model complexity and inference efficiency, as well as to address the issue of diminished network feature learning ability due to the simplification of the attention computation operation, we substitute the Multi-Head Attention module in TopFormer with the squeeze-enhanced Axial Transformer (SeaFormer) layer [39]. CHSST is the first hand segmentation

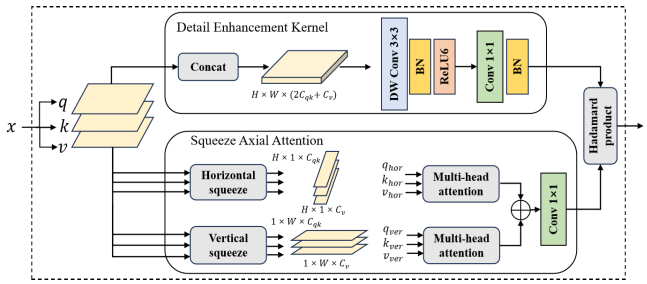


Fig. 4. Schematic illustration of SeaFormer Layer.

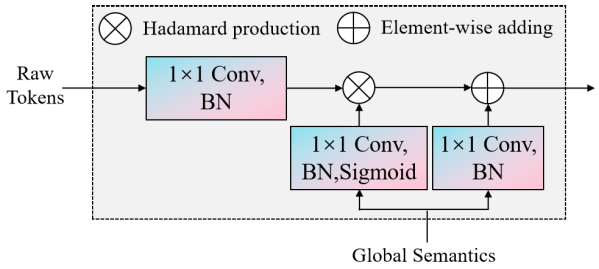


Fig. 5. The architecture of Semantic Injection Module.

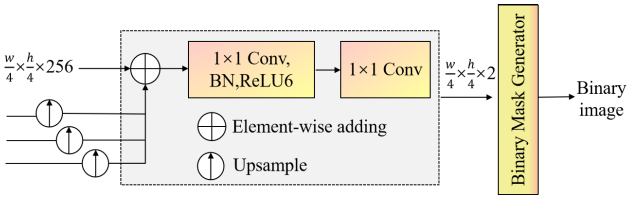


Fig. 6. The architecture of Segmentation Head.

method specifically designed for open-environment palmprint recognition. Without it, extracting hand information from a complex background becomes extremely challenging, rendering the subsequent PROIE infeasible. As a result, other palmprint recognition methods cannot effectively function for comparison purpose.

- The Token Pyramid Module takes a hand image as input and generates a token pyramid with dimensions 7×7×384. Within this module, the output of each scale block is treated as a raw token. The raw tokens are then combined to create scale-aware Tokens following a down-sampling operation.
- ViT is adopted as a semantic extractor, considering the token pyramid as input to produce scale-aware semantics. Subsequently, these semantics are injected into tokens of the corresponding scale to enhance the representation, a process facilitated by the Semantics Injection Module. Notably, global semantics are derived from the ViT-extracted semantics and are up-sampled to match the dimensions of the corresponding raw tokens.
- As for SeaFormer Layer (refer to Figure 4), in the Squeeze Axial Attention (SAA) module, by performing a linear mapping on the original feature maps, three tensors $q \in \mathbb{R}^{H\times W\times C_{qk}}$, $k \in \mathbb{R}^{H\times W\times C_{qk}}$ and $v \in \mathbb{R}^{H\times W\times C_v}$ can be obtained. Squeezing q horizontally and vertically yields q_{hor} and q_{ver}, respectively. Similarly, $k_{\text{hor}} \in \mathbb{R}^{H\times C_{qk}}$ and $k_{\text{ver}} \in \mathbb{R}^{W\times C_{qk}}$, as well as $v\text{hor} \in \mathbb{R}^{H\times C_v}$ and $v_{\text{ver}} \in \mathbb{R}^{W\times C_v}$, can be obtained in the same manner. The Detail Enhancement Kernel (DEK) module with a structure comprising a 3×3 depth-wise separable convolution (DW Conv), a batch normalization (BN), a ReLU6 activation, a 1×1 convolution (Conv), and a BN layer to refine spatial details. The final output of SeaFormer is derived by merging DEK's processed feature map with SAA's output via the Hadamard product, ensuring global information retention and local detail enhancement.
- In Semantic Injection Module (refer to Figure 5), the raw tokens denoted as $T = [T_1, T_2, T_3, T_4]$ undergo processing through a 1×1 Conv layer and a BN layer. Simultaneously, global semantics are processed using a 1×1 Conv layer, a BN layer and Sigmoid activation. The raw tokens and global semantics are then operated via Hadamard product and element-wise adding to generate the output map.
- The augmented token pyramid is employed by the Segmentation Head for hand segmentation. In the Segmentation Head (refer to Figure 6), the outputs of the Semantic Injection Modules are up-sampled to a consistent scale of $\frac{w}{4} \times \frac{h}{4}$ and aggregated through element-wise summation. Subsequently, the feature map undergoes a series of transformations, including a 1×1 Conv layer, a BN layer, a ReLU6 layer [40] and another 1×1 Conv layer, resulting in the generation of a segmentation output with dimensions $\frac{w}{4} \times \frac{h}{4} \times 2$.
- The '2' channels represent two probability values, signifying whether pixels belong to the hand region or the background. The binary mask B is derived through up-sampling to dimensions $w \times h$, where the class label corresponding to the higher probability value is assigned as the predicted class for each pixel. The RGB hand image against black background can be generated by considering the pixel correspondence between the original hand image with a complex background and the binary image. Binary Cross-Entropy (BCE) Loss [41] is used as loss function for training, which can be expressed as

$$L_{BCE}(a, \hat{a}) = -(alog(\hat{a}) + (1 - a)log(1 - \hat{a})), \quad (4)$$

where a is the true value and $\hat{a}$ is the predicted value by the Segmentation Head.

B. Adaptive PROIE

In open environment, hands often assume various poses, with fingers not necessarily always spread apart and may sometimes be close together. Consequently, the rotation angles of the palms vary, making it challenging for traditional FVPs localization methods to achieve satisfactory PROIE. To tackle this issue, we propose promising methods for palm alignment and ROI detection. Here, the hand images with single backgrounds from PolyU2D/3D1.0 [29], REST [30], COEP [31], CASIA [42], CASIA-MS-PalmprintV1 [43] and NTU-CP-v1 [33] are merged to be used as data basis.

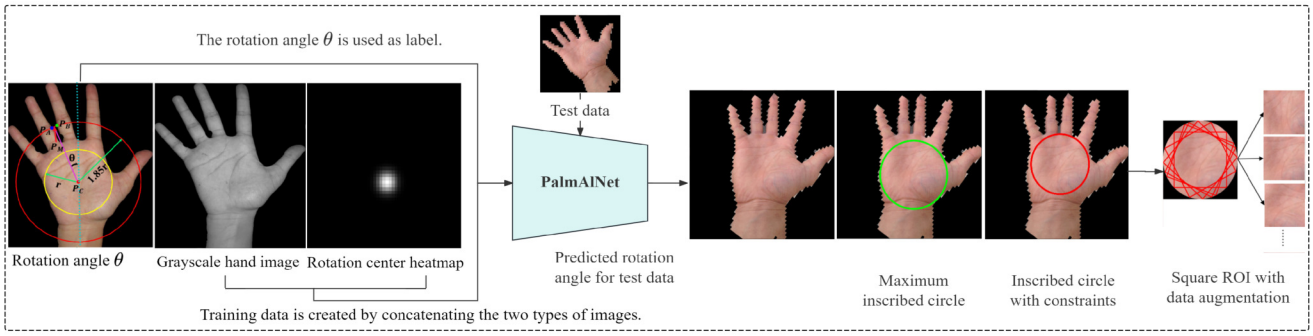


Fig. 7. Schematic representation of palm alignment and ROI detection.

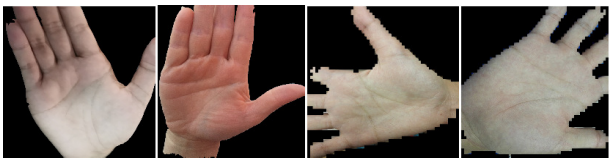


Fig. 8. Fingers closed together and missing palm pixels result in the absence of P_A and P_B.

1) Rotation Angle Learning-Based Palm Alignment: Figure 3 illustrates that by applying histogram thresholding, the segmented images of hands against a black background are produced, sized at $128 \times 128 \times 3$. As shown in Figure 7, the hand's largest inscribed circle, located in the palm area, is identified by its center $P_C(x_0, y_0)$ and a radius of r. A larger circle, with a radius of $1.85r$, is then drawn, intersecting with the hand's finger segments. This circle meets the finger section at its fourth and fifth points from the left in the hand image's upper section, labeled as P_A and P_B, respectively. The midpoint between P_A and P_B is defined as P_M. Following this, the angle θ is calculated between line $P_C P_M$ and the vertical y-axis, allowing for the palm region to be rotated around P_C by θ for alignment.

However, when high degrees of freedom in image capture cause fingers to come together or missing pixels in the palm, these intersections may fail to materialize. (refer to Figure 8), complicating the identification of P_A and P_B due to significant rotation or missing fingers. To overcome these issues, the palm alignment network (PalmAlNet) is introduced to predict the rotation angle needed to orient the input hand image around a given center point based on the center point heatmap. For test images, the center point is the center of the largest inscribed circle within the corresponding binary image.

PalmAlNet incorporates the initial three blocks of the VGG16 model, a 1×1 Conv layer paired with a ReLU layer, a layer to flatten the data, and a FC layer. The Mean Absolute Error (MAE) metric evaluates the discrepancy between the input angle θ and PalmAlNet's output. Initially, the hand image with black background is converted to a grayscale image for processing. Then, a center point heatmap H showing the rotation center is generated by

$$H(x, y) = \exp\left(-\frac{(x - x_0)^2 + (y - y_0)^2}{2\sigma^2}\right), \tag{5}$$

where $H(x, y)$ means the intensity value of the heatmap at coordinate (x, y), and the variance σ is set to 2. The training data consists of a grayscale hand image merged with a heatmap, where the target variable is the rotation angle. The network predicts the rotation angle by scaling its output using a tanh function and a linear mapping to fit within the range $(-\pi, \pi)$. This predicted angle is then applied to adjust the orientation of test hand images. For data augmentation, the method employs rotations and cropping adjustments. Each hand image undergoes a single rotation by a random integer degree from 0 to 360. Post-rotation, the cropped image's side length is determined by a random factor between 0.7 and 1, relative to the original image's dimensions. Additionally, the positions of P_A, P_B, P_C, and the angle θ are modified in accordance with these transformations. The heatmap used during training is generated by the transformed P_C.

2) Adaptive ROI Detection With Constraints: We compute the distance from each pixel within the hand region to the closest zero pixel, resulting in the distance matrix $D_c \in \mathbb{R}^{w \times h}$ with w and h representing the hand image's width and height, respectively. The center of the largest inscribed circle, $P_C(x_0, y_0)$, with a radius $r = \max(D_c)$ is identified by $P_C = \text{argmax}_{i,j}(D_c(i, j))$. In well-controlled environment, P_C is considered the ideal palm center. In the context of civil and commercial purposes, even when participants are cooperative, contactless hand positioning often results in a tilted angle. This tilt can cause the lower part of the palm to appear enlarged in some cases, while in others, the upper part may be more pronounced (as depicted in Figure 7 regarding the Maximum inscribed circle). To tackle this variability, the adaptive detection with constraints is proposed, facilitating data augmentation within a predetermined square ROI.

- In certain images, the circular ROI centered at $P_C(x_0, y_0)$ might be positioned lower than anticipated, which could result in the omission of crucial details and unnecessary noise. A direct comparison illustrating the differences between utilizing the largest inscribed circle and applying an inscribed circle with specific constraints is provided in Figure 7. Initially, we determine a set of acceptable regions $S = \{(x_i, y_i) \mid D_c(x_i, y_i)\} > t_1 r$, with t_1 being an empirically determined threshold set at 0.85. Within this acceptable region, pixel values are assigned a value of 1, while all other pixels are designated a value of 0.

- Within S, the distance from each nonzero pixel to the closest zero pixel is calculated, resulting in the distance matrix D_s. To refine the feasible region further, we derive S' by

$$S' = S \cap T, \quad (6)$$
$$T = \{(x_i, y_i) \mid \|(x_i, y_i) - (x_0, y_0)\|_2 > D_s(x_0, y_0)\},$$
$$y_i >= y_0.$$

  The center of S' is indicated by $P'_C(x'_0, y'_0)$, with its radius determined through the expression $t_2 D_c(x'_0, y'_0)$.
- The hand image is rotated around its center, $P'_C(x'_0, y'_0)$, at angle intervals of α, within the range from $-\gamma$ to γ. Following each rotation, we select the largest rectangle that can be inscribed within the rotated image, ensuring its edges are parallel to the coordinate axes, to define our square ROI with a side length L_R, which are then used for training the recognition model RLANN. This process, iterating through the specified angle range, generates $\frac{2\gamma}{\alpha}$ augmented data samples. It can be observed that a single palmprint image generates multiple palm ROI images, leading to data augmentation. This process helps enhance the training performance of the recognition model RLANN. The values for t_2, α, and γ are set at 1.1, 3, and 30, respectively, for these operations.

C. RLANN

1) Architecture Design of RLANN: Layer aggregation studies network designs for feature reuse, exemplified by DenseNet [44]. Dense connections lead to a quadratic increase in parameters, causing redundancy and limiting information storage [45]. A lightweight RLA module with fewer parameters is introduced, using recurrent connections to maintain a depth-independent parameter count [46], which can be added to existing CNNs for better feature extraction.

Indeed, as palmprint feature maps traverse the network, they tend to blur rapidly. But creating inter-layer connections within the network significantly boosts its capacity for feature learning [47]. Leveraging information aggregated from earlier layers emerges as a logical strategy. Motivated by this insight, RLANN has been crafted to excel in learning palmprint features, with its structure detailed in Table I. Where, LRN represents the local response normalization operation, and Maxpool represents Max Pooling operation.

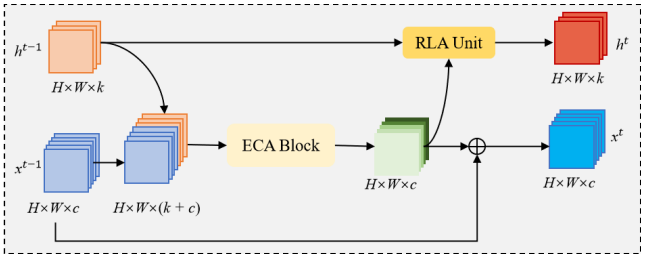
The RLA module's structure is illustrated in Figure 9, with the specifics of the RLA unit and efficient channel attention (ECA) block. The RLA mechanism applied in this study is characterized by

$$h^t = g^t(h^{t-1}, x^{t-1}),$$
$$x^t = f^t(h^{t-1}, x^{t-1}), \quad (7)$$

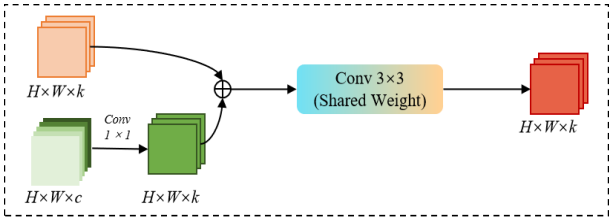
where h^t represents the recurrently aggregated information up to $(t-1)th$ layer, and x^t represents the main features learned by $(t-1)th$ layer. f^t represents ResNet bottleneck with ECA mechanism [48]. The recurrent unit, denoted as g^t, functions analogously to a recurrent neural network by accumulating information over time. Moreover, implementing weight sharing within these modules can serve as a regularization mechanism for layer aggregation.

TABLE I
THE ARCHITECTURE OF RLANN

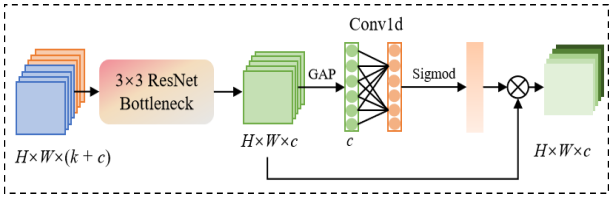
Layers	Input	Kernel size	No. of filters	Stride	Padding
Conv_1	128×128×3	Conv2d, 7×7	64	2	3
Maxpool	64×64×3	Maxpool2d, 3×3	64	2	1
RLABlk_1	32×32×64	RLA Bottleneck, 3×3	256	1	1
	32×32×256	RLA Bottleneck, 3×3	256	1	1
	32×32×256	RLA Bottleneck, 3×3	256	1	1
RLABlk_2	32×32×512	RLA Bottleneck, 3×3	512	2	1
	16×16×512	RLA Bottleneck, 3×3	512	1	1
	16×16×512	RLA Bottleneck, 3×3	512	1	1
RLABlk_3	16×16×512	RLA Bottleneck, 3×3	1024	2	1
	8×8×1024	RLA Bottleneck, 3×3	1024	1	1
	8×8×1024	RLA Bottleneck, 3×3	1024	1	1
	8×8×1024	RLA Bottleneck, 3×3	1024	1	1
	8×8×1024	RLA Bottleneck, 3×3	1024	1	1
	8×8×1024	RLA Bottleneck, 3×3	1024	1	1
RLABlk_4	8×8×1056	LRN, 2×2	1056	-	-
	8×8×1056	SSAP Layer	1056	-	-
	8×8×1024	Conv2d, 8×8	512	1	0
Conv_2	512	FC layer	N	-	-



(a) RLA Module



(b) RLA Unit



(c) ECA Block

Fig. 9. Schematic diagram of RLA module.

The initial hidden state of the RLA, h^0, starts at 0. RLANN is segmented into four distinct stages, classified according to the resolution of the feature maps. Within each stage, the Conv layer weights in the RLA unit are reused across all RLA modules. To adjust the spatial dimensions of the RLA hidden state across different stages, average pooling is employed. Ultimately, the 32-dimensional RLA hidden state is merged with the 1024-dimensional feature map from the main

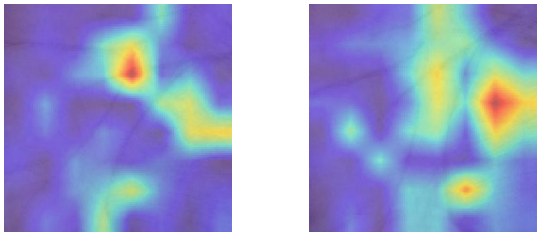


Fig. 10. Visual impact of $W_{spatial}$ across two distinct inputs.

pathway, forming a unified input for RLABlk_4. A dropout operation with a rate of 0.4 is applied after RLABlk_4.

CNNs function as a series of filters across channel dimension, crucial for processing palmprint images affected by environmental variations. Not every filter is equally effective for all images, hence the channel-wise attention mechanisms is necessary. These mechanisms dynamically adjust channel feature weights, essentially prioritizing features from the most effective filters, thus enhancing focus on significant features and diminishing the less relevant ones. Therefore, ECA Block is adopted as the structure on the main learning path.

Global Average Pooling (GAP), preferred for its simplicity and lack of parameters, is ill-suited for palmprint recognition due to its low spatial sensitivity. Recognizing the need for heightened focus on crucial palmprint regions, we creatively propose Spatial Self-Attentive Pooling (SSAP) used in RLABlk_4. This method transforms the final RLA module's feature map, $X_{final} \in \mathbb{R}^{8\times8\times1056}$, into a flattened format, $X_{flatten} \in \mathbb{R}^{64\times1056}$, treating each pixel as an individual feature token. Spatial weights are then calculated to enhance feature extraction,

$$W_{spatial} = \text{rescale}\left(\text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)\right), \tag{8}$$

where Q and K denote query and key vectors, respectively, both projected linearly from $X_{flatten}$, preserving the same spatial dimensions. The variance in QK^T is captured by d_k, drawing inspiration from Scaled Dot-Product Attention [49]. This weighting method not only analyzes individual pixel data but also their interrelations, enabling RLANN to adapt its focus per instance. Spatial weights, $W_{spatial}$, resized to 8×8, multiplying by X_{final} to form a weighted map. Subsequently, a 8×8.2D convolution with 512 filters creates the final vector. Despite $W_{spatial}$ offering instance-specific spatial emphasis, the final convolution layer is essential to globally weights different regions. Figure 10 shows SSAP's role on fostering discriminative learning. The texture regions that significantly impact the discriminative results vary across different input images, each having a distinct high weight.

2) ACPLoss Design: Combination of angular margin penalty-based Loss and metric-based loss can effectively enhance recognition accuracy [14]. Driven by this, ACPLoss is proposed for RLANN training, which can be expressed as

$$L_{ACP} = L_{AS'} + \lambda L_{CP}. \tag{9}$$

Equation (1) provides the formula for L_{AS}, wherein the parameters m_1, m_2, and m_3 within L_{AS} are individually assigned values of 1, m, and 0, respectively, to formulate $L_{AS'}$.

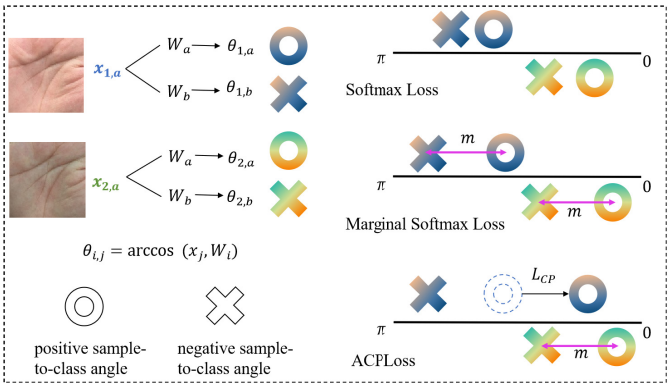


Fig. 11. Illustrate of the sample-to-class similarities learned from normalized softmax loss, marginal softmax loss, and ACPLoss.

In computing $L_{AS'}$ for a given x_i, it's possible to determine the angles θ_j between x_i and each W_j. An angular penalty margin m is applied to the angle between deep features and corresponding weights to adjust the decision boundary nearer to the class center W_{y_i}. This margin enhances similarity within classes and reduces similarity between different classes.

$$L_{AS'} = \frac{1}{N}\sum_{i\in N} -log\frac{e^{s(cos(\theta_{y_i}+m))}}{e^{s(cos(\theta_{y_i}+m))} + \sum_{j=1,j\neq y_i}^{c} e^{s\cos(\theta_j)}}. \tag{10}$$

However, $L_{AS'}$ aims to maximize angular similarity between each training sample and its own class center, it does not explicitly constrain that all the positive sample-to-class similarities are larger than all negative sample-to-class similarities, as highlighted by [27].

Consider two samples, x_1 and x_2, from class a with center W_a, and another class center W_b. While optimizing $L_{AS'}$ ensures $\theta_{1,a} < \theta_{1,b}$ and $\theta_{2,a} < \theta_{2,b}$, it's possible that $\theta_{1,a} > \theta_{2,b}$, as depicted in Figure 11. This implies that the discriminative capability of features refined by $L_{AS'}$ might be insufficient for classification during testing. We introduce L_{CP} to draw samples nearer to their class centers, thereby improving the training methodology and boosting intra-class cohesion.

$$L_{CP} = \frac{1}{N}\sum_{i=1}^{N} max(0, \theta_{y_i}^{\beta} - \theta_t^{\beta}), \tag{11}$$

where as the distance from the class center increases, samples further away are assigned a greater weight through the hyper-parameter β. A margin value, θ_t, is utilized to prevent excessive compression of sample distribution in the feature space, which helps in avoiding over-fitting throughout the training of the model. The value of β is established at 1.5, and θ_t is dynamically determined using the Exponential Moving Average (EMA) method [50],

$$\theta_t^{(k)} = \alpha\theta_r^{(k)} + (1-\alpha)\theta_t^{(k)}, \tag{12}$$

where $\theta_t^{(k)}$ represents the iteration-specific value of θ_t during the kth training cycle. The term $\theta_r^{(k)} = \frac{1}{N}\sum_i^N \theta_{y_i}$ calculates the mean angle between positive samples and their class center in the kth batch. The momentum parameter α, assigned a value of 0.99, influences this calculation. By adjusting λ appropriately, the ability of deep features to distinguish between classes can be greatly improved.

TABLE II
AN OVERVIEW OF THE CONSIDERED PALMPRINT DATASETS

Datasets	Task(s)	Classes	Images	Close-set		Open-set	
				Tr images	Ts images	Tr classes	Ts classes
PolyU2D/3D1.0 [29]	CHSST/PalmAlNet	347	2431	2431	–	347	–
REST [30]	CHSST/PalmAlNet	296	2663	2663	–	296	–
COEP [31]	CHSST/PalmAlNet	167	1305	1305	–	167	–
CASIA [42]	PalmAlNet	312	5502	5502	–	312	–
CASIA-MS-PalmprintV1 [43]	PalmAlNet	200	7200	7200	–	200	–
BJTU_PalmV1 [32]	CHSST	244	2434	2434	–	244	–
NTU-CP-v1 [33]	PalmAlNet	656	2481	2481	–	656	–
HIT-NIST-V1 [51]	RLANN	321	3016	1619	1397	161	160
MPD [7]	RLANN	400	16000	8000	8000	200	200
IITD [52]	RLANN	460	2601	1380	1221	230	230
BJTU_PalmV2 [32]	RLANN	296	2675	1344	1331	148	148

IV. EXPERIMENTS AND ANALYSIS

This section offers an overview of considered public palmprint datasets and outlines a structured evaluation method. In the close-set paradigm, for datasets with two sessions, the first is used for training and the second for testing. Single-session datasets are split in half by class, assigning extra images to training if necessary. Training uses the first set of images, and testing uses the latter. In the open-set paradigm, classes are equally divided for training and testing, with extra classes added to training if the count is odd.

Left palms are mirrored to simulate right palms, treating each hand as a separate class. Training leverages the Adam optimizer (momentum of 0.9, weight decay of $1e^{-4}$), with a batch size of 32. Learning rates for hand segmentation, and RLANN are set at $5e^{-4}$, and $1e^{-4}$, respectively. For assessing identification results, Rank-1 accuracy and Cumulative Match Characteristic (CMC) curve are utilized, where higher values indicate better performance. Conversely, Equal Error Rate (EER) and Receiver Operating Characteristic (ROC) curve are used for evaluating verification results, with lower values denoting improved accuracy.

Experiments were conducted on a system equipped with an Intel(R) Core(TM) i7-9700K CPU, 16GB RAM, and an NVIDIA Quadro GV100 32GB GPU. We optimized the model and data pipelines to enhance hardware efficiency. Through extensive preliminary experiments, a batch size of 32 was identified as optimal for the hardware configuration. The data pre-processing pipeline was streamlined with on-the-fly augmentation techniques, including random cropping, contrast and brightness adjustments, and Gaussian blur, reducing storage overhead and accelerating training. Additionally, mixed precision training was employed to fully utilize GPU capabilities, significantly reducing computational demands and training time.

A. The Used Public Palmprint Datasets

Total eleven palmprint datasets were selected for different purposes: PolyU2D/3D1.0 [29], REST [30], COEP [31], CASIA [42], CASIA-MS-PalmprintV1 [43], BJTU_PalmV1 [32], NTU-CP-v1 [33], HIT-NIST-V1 [51], MPD [7], IITD [52], and BJTU_PalmV2 [32]. The PROIE task including CHSST and rotation angle regression, used the first seven datasets, while the palmprint recognition task utilized the latter four.

The first seven datasets were created in well-controlled environments with single backgrounds, uniform illumination, and similar hand postures, making them suitable for automated learning of hand shape features. In contrast, HIT-NIST-V1 [51], MPD [7], and BJTU_PalmV2 [32] were developed in open environments with complex backgrounds, varying illumination, and diverse hand postures, making them ideal for validating the effectiveness of palmprint recognition methods.

Although the IITD dataset [52] was established in a controlled environment, it features significant variations in lighting conditions, hand scales, and hand postures, along with ornaments such as rings on fingers, adding complexity similar to the latter group, making it also suitable for palmprint recognition experiments. Besides, only 2D images from PolyU2D/3D1.0 dataset were employed in this work. Table II outlines the datasets' details, with 'Tr' and 'Ts' denoting 'Training' and 'Test', respectively, and '–' indicating datasets exclusively used for PROIE.

B. Palmprint Recognition Experiments

This section presents the performance evaluation of the proposed method, RDRLA, against both hand-crafted and DL approaches. The hand-crafted approaches such as HOL [53], LLDP [54], CR_CompCode [8], LSIR [16], and DL approaches like CGDNet [15], PalmNet [55], FERnet [33], CompNet [56], CCNet [57]. The comparative results, showcased in Table III and Table IV, stem from our own experiments, conducted under the parameters reported in the original studies for a fair assessment. The palmprint recognition experiments here are all based on the proposed PROIE method FFARD. The tables highlight the **best** performance in bold and the second-best in underline.

After RLANN is trained properly, the last 'FC' layer is removed for testing. For the identification task, cosine similarity is measured between each test image and all training images. The label of the image with the highest similarity is assigned to the test image. A correct classification occurs if the predicted label matches the ground truth label. For the

TABLE III

COMPARISON OF RANK-1 ACCURACY AND EER FOR CLOSE-SET ISSUE USING THE PROPOSED PROIE METHOD FFARD (%)

Methods	HIT-NIST-V1		IITD		MPD		BJTU_PalmV2	
	Rank-1	EER	Rank-1	EER	Rank-1	EER	Rank-1	EER
HOL [53]	79.90	3.59	95.82	1.70	68.69	14.10	87.38	4.44
LLDP [54]	92.15	25.28	95.58	12.29	72.56	34.89	85.34	22.60
CR_Compcode [8]	80.70	8.08	90.78	3.90	36.22	28.72	66.21	13.42
LSIR [16]	88.09	3.71	98.46	0.65	78.48	7.57	86.77	4.25
CGDNet [15]	86.16	3.53	**98.85**	**0.27**	73.35	7.90	83.74	4.51
CCNet [57]	83.63	6.39	92.14	3.71	70.72	13.42	84.43	6.36
CompNet [56]	78.24	10.32	90.99	4.31	70.62	13.81	81.55	7.77
FERNet [33]	91.15	1.96	98.44	0.57	80.67	7.06	92.74	1.64
PalmNet [55]	86.09	27.73	95.16	12.91	77.21	32.70	88.96	22.33
RDRLA (ours)	**95.48**	**1.51**	98.77	0.54	**88.43**	**5.16**	**95.24**	**1.37**

TABLE IV

COMPARISON OF RANK-1 ACCURACY AND EER FOR OPEN-SET ISSUE USING THE PROPOSED PROIE METHOD FFARD (%)

Methods	HIT-NIST-V1		IITD		MPD		BJTU_PalmV2	
	Rank-1	EER	Rank-1	EER	Rank-1	EER	Rank-1	EER
HOL [53]	58.44	11.13	93.59	3.32	71.61	10.62	85.91	**4.46**
LLDP [54]	80.23	29.06	94.23	11.66	72.02	34.27	87.10	20.93
CR_Compcode [8]	58.01	16.74	83.33	8.28	40.71	26.55	66.55	13.89
LSIR [16]	88.85	11.51	98.29	2.55	90.88	17.41	89.30	10.97
CGDNet [15]	89.55	8.64	**99.48**	**0.85**	97.73	8.20	88.37	8.34
CCNet [57]	77.85	22.11	94.38	8.56	93.36	22.43	78.76	19.60
CompNet [56]	76.18	22.56	92.68	8.84	95.27	21.51	76.59	20.89
FERNet [33]	91.64	10.79	98.29	2.55	99.57	6.91	92.40	8.98
PalmNet [55]	82.68	32.53	95.83	10.09	79.61	32.49	89.13	21.81
RDRLA (ours)	**94.29**	**8.43**	99.15	1.53	**99.62**	**6.13**	**95.50**	6.84

verification task, cosine similarity is calculated between any two test images with the same ground truth labels.

1) Close-Set Palmprint Recognition: Close-set issue usually indicates the same classes for classification in the source set and target set. Table III presents the Rank-1 accuracy and EER for RDRLA and the comparison methods in a close-set paradigm. Figure 12 and Figure 13 illustrate the CMC and ROC curves, respectively, highlighting close-set palmprint recognition analysis. Overall, RDRLA surpasses both hand-crafted and DL methods, achieving the highest Rank-1 accuracy values of 95.48%, 98.77%, 88.43%, and 95.24% on HIT-NIST-V1, IITD, MPD, and BJTU_PalmV2 datasets, respectively. It also records the lowest EER values of 1.51%, 0.54% 5.16%, and 1.37% on HIT-NIST-V1, IITD, MPD, and BJTU_PalmV2 datasets, respectively, demonstrating a significant performance improvement and establishing a new benchmark for palmprint biometric recognition. The CMC and ROC curves further verify the superior performance of RDRLA in palmprint recognition.

2) Open-Set Palmprint Recognition: Unlike close-set issue, open-set recognition involves a target set with few or no overlapping classes with the source. While the open-set testing procedure is similar to close-set testing, the key distinction is that the gallery set is comprised of different classes rather than the test set.

Table IV details Rank-1 accuracy and EER for RDRLA and the comparison methods in the open-set paradigm. Figure 14 and Figure 15 show the CMC and ROC curves corresponding to the considered datasets for comparison and analysis. As shown in Table IV, RDRLA achieves the best identification performance, achieving Rank-1 accuracy values of 94.29%, 99.15%, 99.62%, and 95.50% on the HIT-NIST-V1, IITD, MPD, and BJTU_PalmV2 datasets, respectively. In the case of verification, RDRLA also has the best performance with the lowest EER values of 8.43%, 1.53%, and 6.13% on HIT-NIST-V1, IITD, and MPD datasets, respectively, and a competitive second-best performance on BJTU_PalmV2 dataset. RDRLA remains the best performing solution for open-set palmprint recognition. CMC and ROC curves demonstrate the consistent superior performance for the majority of TPR and FPR values.

TABLE V

COMPARISON OF COMPUTATIONAL COMPLEXITY AMONG DEEP PALMPRINT RECOGNITION MODELS

Methods	Params/MB	Flops/G	Training speed (img/sec)	Inference speed (img/sec)
CGDNet [15]	29.32	2.70	285	40
CCNet [57]	62.31	0.88	183	56
CompNet [56]	4.93	0.29	1280	298
FERNet [33]	27.58	4.61	516	54
PalmNet [55]	12.28	1.16	37	14
RLANN (ours)	35.85	1.17	464	51

3) Comparison of Computational Complexity: The primary objective of palmprint recognition is to achieve high individual recognition accuracy. However, considering the practical requirements of experimental applications, it is equally important to account for computational complexity and memory efficiency during model design. To strike a balance between optimal recognition performance and resource constraints, RLANN incorporates techniques such as Max pooling, dropout, weight sharing, and on-the-fly data augmentation.

To facilitate a fair comparative analysis, we compare the computational complexity of RLANN with other deep palmprint recognition models. As shown in Table V, the proposed method delivers optimal palmprint recognition performance while keeping computational complexity, training speed, and inference speed within acceptable bounds.

C. Ablation Study

As mentioned in Section III-C2, the ACPLoss, the parameter λ as well as the SSAP plays an important role on the performance. This section presents ablation studies to investigate the effects of these factors on the experimental results.

1) Effect of the Proposed PROIE Method.: Table VI and Table VII demonstrate that, using traditional FVPs-based PROIE methods, our proposed RLANN consistently outperforms in terms of palmprint recognition for both close-set and open-set scenarios. However, a comparison with the results in Table III and Table IV reveals a minor decrease in accuracy and an increase in EER, aligning with our theoretical expectations. Unlike traditional methods, FFARD offers an automatic and adaptive approach for ROI detection that does not require manual annotation or FVPs localization. Although there is a slight performance drop, the ability to detect ROI without FVPs holds substantial value for the practical deployment of palmprint recognition in open environment, as evidenced

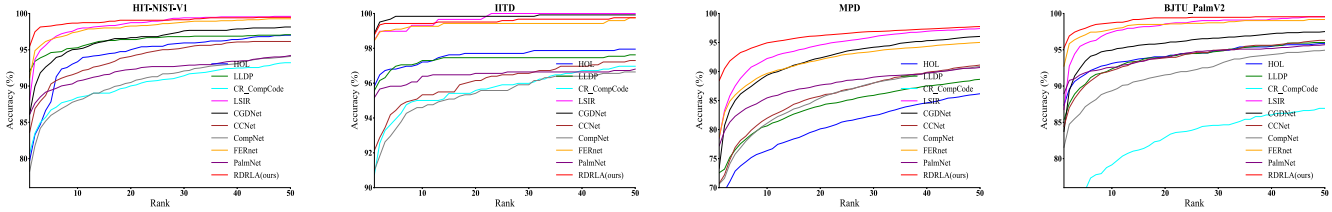


Fig. 12. CMC curves under close-set evaluation criterion.

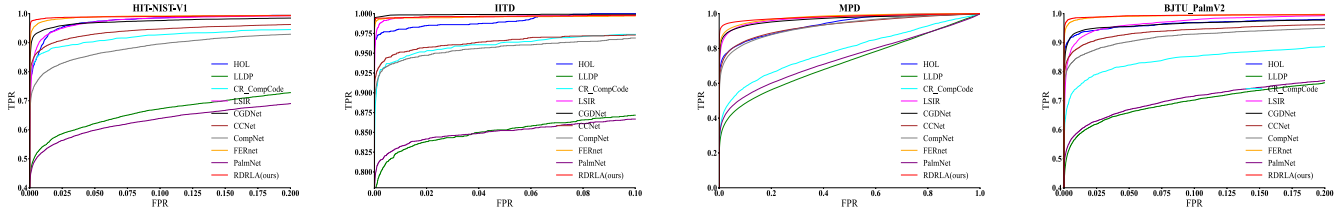


Fig. 13. ROC curves under close-set evaluation criterion.

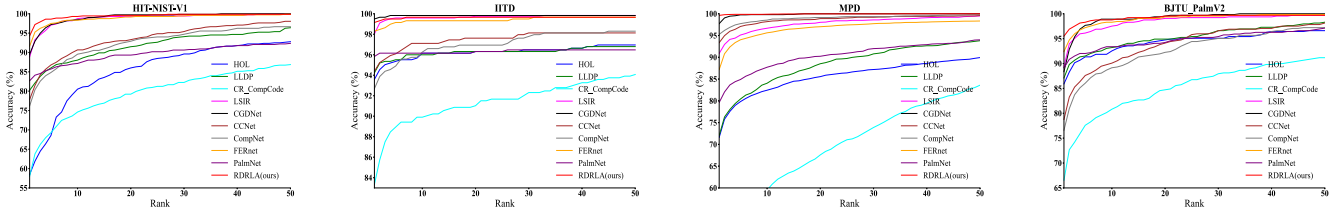


Fig. 14. CMC curves under open-set evaluation criterion.

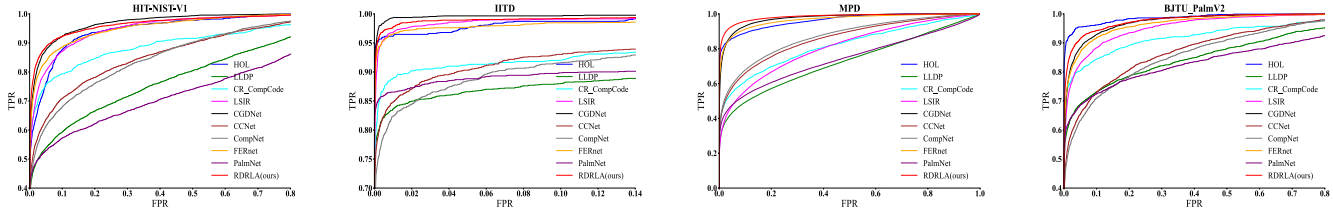


Fig. 15. ROC curves under open-set evaluation criterion.

TABLE VI

COMPARISON OF RANK-1 ACCURACY AND EER FOR CLOSE-SET ISSUE USING TRADITIONAL FVPS-BASED PROIE METHOD (%)

Methods	HIT-NIST-V1		IITD		MPD		BJTU_PalmV2	
	Rank-1	EER	Rank-1	EER	Rank-1	EER	Rank-1	EER
HOL [53]	87.79	2.09	96.14	0.32	93.79	**1.08**	91.98	1.20
LLDP [54]	96.60	11.47	99.01	4.56	94.92	9.63	96.43	9.91
CR_Compcode [8]	85.32	4.99	94.18	2.72	66.43	11.02	85.31	5.76
LSIR [16]	92.48	2.73	98.57	0.46	91.94	7.85	92.42	6.52
CGDNet [15]	94.52	1.98	99.62	0.24	94.76	2.13	94.03	1.45
CCNet [57]	91.19	3.16	97.52	2.62	79.82	7.65	90.32	4.09
CompNet [56]	87.05	6.48	97.49	3.88	78.53	8.35	86.05	7.35
FERNet [33]	97.46	0.76	99.56	0.25	99.51	1.47	**96.92**	0.93
PalmNet [55]	97.39	12.63	99.59	4.20	95.97	9.88	96.77	10.60
RLANN (ours)	**98.39**	**0.41**	**99.68**	**0.21**	**99.61**	1.51	96.85	**0.90**

by our experimental outcomes which often see improvements with human involvement adjustments.

2) Effect of the RLANN's Architecture Design: To highlight the advantages of RLANN, we conducted a comparison among the proposed ResBNeck+ECA+RLA architecture and other configurations: ResNet bottleneck (ResBNeck), ResBNeck+ECA, and ResBNeck+RLA. According to Table VIII, our designed architecture consistently delivers superior performance, with ResBNeck+RLA ranking as the second best across all tests. This underscores the critical contribution of the RLA module to palmprint recognition effectiveness.

3) Effect of the RLABlk_4 Design: To demonstrate the efficacy of the RLABlk_4 design, we first conducted a comparative analysis of normalization operations, including LRN, *w/o* normalization, BN, and Layer Normalization (LN), as detailed in Table IX. While BN is widely adopted in modern architectures, our specific application and experimental observations justify the use of LRN for its unique advantages in palmprint recognition. The choice was validated through empirical comparisons, ensuring it aligns with the model's performance and efficiency requirements.

Furthermore, we conducted a comparative analysis of different flattening layer configurations, including Conv,

TABLE VII

COMPARISON OF RANK-1 ACCURACY AND EER FOR OPEN-SET ISSUE USING TRADITIONAL FVPS-BASED PROIE METHOD (%)

Methods	HIT-NIST-V1		IITD		MPD		BJTU_PalmV2	
	Rank-1	EER	Rank-1	EER	Rank-1	EER	Rank-1	EER
HOL [53]	87.79	5.12	99.69	**0.17**	93.85	**1.48**	93.10	4.95
LLDP [54]	93.30	14.51	99.18	4.71	95.12	10.27	96.02	7.80
CR_Compcode [8]	79.86	8.88	95.41	2.77	71.32	10.89	84.51	5.45
LSIR [16]	91.92	9.35	98.21	2.15	91.82	**2.36**	91.81	9.29
CGDNet [15]	92.34	7.82	**99.78**	**0.73**	98.24	3.05	93.72	5.03
CCNet [57]	90.42	17.00	94.06	7.51	99.34	9.76	89.79	16.10
CompNet [56]	90.28	17.63	91.26	10.02	99.10	11.43	87.65	17.68
FERNet [33]	97.46	6.11	99.02	2.19	99.73	2.73	96.34	5.64
PalmNet [55]	94.34	17.14	99.67	4.09	96.32	10.16	96.21	10.08
RLANN (ours)	**98.12**	**4.94**	99.76	1.03	**99.92**	2.73	**96.54**	**4.16**

TABLE VIII

THE EFFECT OF THE RLANN ARCHITECTURE DESIGN (%)

Criteria	Architectures	HIT-NIST-V1		IITD		MPD		BJTU_PalmV2	
		Rank-1	EER	Rank-1	EER	Rank-1	EER	Rank-1	EER
Close-set	ResBNeck	93.74	2.64	97.7	0.60	83.25	16.43	93.72	1.72
	ResBNeck+ECA	94.23	2.32	98.12	0.57	84.96	14.39	94.38	1.69
	ResBNeck+RLA	95.02	1.64	98.54	0.55	88.02	6.32	94.96	1.51
	ResBNeck+ECA+RLA	**95.48**	**1.51**	**98.77**	**0.54**	**88.43**	**5.16**	**95.24**	**1.37**
Open-set	ResBNeck	91.22	10.47	98.98	2.83	98.69	9.43	92.24	9.02
	ResBNeck+ECA	92.43	10.23	99.03	2.21	98.92	8.21	93.12	8.36
	ResBNeck+RLA	93.97	8.89	99.14	1.56	99.37	6.74	95.07	7.51
	ResBNeck+ECA+RLA	**94.29**	**8.43**	**99.15**	**1.53**	**99.62**	**6.13**	**95.50**	**6.84**

TABLE IX

THE EFFECT OF NORMALIZATION OPERATION IN RLABLK_4 (%)

Criteria	Normalization operations	HIT-NIST-V1		IITD		MPD		BJTU_PalmV2	
		Rank-1	EER	Rank-1	EER	Rank-1	EER	Rank-1	EER
Close-set	*w/o* normalization	95.34	**1.47**	98.60	0.61	87.54	6.05	95.16	1.59
	BN	95.02	1.64	98.77	**0.42**	87.94	5.51	95.15	1.42
	LN	95.21	1.79	**98.85**	0.47	88.20	5.43	94.97	1.48
	LRN (ours)	**95.48**	1.51	98.77	0.54	**88.43**	**5.16**	**95.24**	**1.37**
Open-set	*w/o* normalization	93.89	8.79	98.98	1.92	98.24	9.47	94.19	7.51
	BN	93.47	9.08	98.96	2.15	98.79	8.00	95.01	8.42
	LN	94.17	**8.22**	99.04	1.87	98.59	7.74	94.57	7.96
	LRN (ours)	**94.29**	8.43	**99.15**	**1.53**	**99.62**	**6.13**	**95.50**	**6.84**

TABLE X

THE EFFECT OF FLATTENING LAYER IN RLABLK_4 (%)

Criteria	Layers	HIT-NIST-V1		IITD		MPD		BJTU_PalmV2	
		Rank-1	EER	Rank-1	EER	Rank-1	EER	Rank-1	EER
Close-set	Conv	93.74	1.92	98.77	0.64	87.98	6.19	94.63	1.72
	GAP	92.41	6.70	99.01	1.06	85.15	14.74	91.53	9.22
	GDC	93.47	1.67	98.19	0.51	86.41	6.03	93.76	1.95
	SSAP+GDC	94.54	1.71	99.42	**0.47**	87.22	**4.51**	94.25	1.44
	SSAP+GAP	92.88	3.91	**99.83**	0.65	86.54	16.39	91.92	8.09
	SSAP+Conv (ours)	**95.48**	**1.51**	98.77	0.54	**88.43**	5.16	**95.24**	**1.37**
Open-set	Conv	93.17	8.81	99.48	1.89	99.51	7.43	95.03	7.11
	GAP	89.41	8.49	98.81	0.92	94.52	21.78	89.41	8.49
	GDC	91.78	8.78	99.31	1.87	99.43	7.12	92.24	6.81
	SSAP+GDC	94.01	8.52	99.31	1.34	99.49	7.01	93.79	**6.31**
	SSAP+GAP	89.27	8.86	**99.65**	**0.79**	93.86	26.54	89.45	8.08
	SSAP+Conv (ours)	**94.29**	**8.43**	99.15	1.53	**99.62**	**6.13**	**95.50**	6.84

TABLE XI

THE EFFECT OF DOWNSAMPLING OPERATION IN THE SECOND LAYER OF RLANN (%)

Criteria	Downsampling operations	HIT-NIST-V1		IITD		MPD		BJTU_PalmV2	
		Rank-1	EER	Rank-1	EER	Rank-1	EER	Rank-1	EER
Close-set	*w/o* pooling	93.61	2.48	97.62	0.92	87.54	6.85	94.10	2.43
	Average pooling	94.76	2.08	98.52	0.79	86.87	7.10	**95.41**	1.48
	Stochastic pooling [58]	89.82	3.75	97.21	1.06	83.16	10.97	88.30	5.67
	Conv2d (stride=2)	90.26	3.38	98.03	0.87	83.24	10.18	89.76	4.53
	Max pooling (ours)	**95.48**	**1.51**	**98.77**	**0.54**	**88.43**	**5.16**	95.24	**1.37**
Open-set	*w/o* pooling	93.45	11.14	98.56	2.60	98.84	11.43	92.78	9.96
	Average pooling	92.78	10.76	98.63	2.49	99.02	8.64	94.71	7.39
	Stochastic pooling [58]	87.54	14.74	97.61	4.28	97.32	16.16	87.51	11.24
	Conv2d (stride=2)	85.07	16.85	97.96	3.14	97.43	15.82	85.43	12.68
	Max pooling (ours)	**94.29**	**8.43**	**99.15**	**1.53**	**99.62**	**6.13**	**95.50**	**6.84**

GAP, Global Depthwise Convolution (GDC), SSAP+GDC, SSAP+GAP, SSAP+Conv. The results, presented in Table X, demonstrate that our proposed architecture (SSAP+Conv) consistently outperforms the alternatives across most scenarios. These findings highlight the effectiveness of the proposed SSAP mechanism in significantly enhancing palmprint recognition performance.

4) Effect of Max Pooling Operation in RLANN.: To validate the superior performance of Max pooling over alternative downsampling operations, we conducted a comprehensive comparison against *w/o* pooling, Average pooling, Stochastic pooling [58], and Conv2d (stride=2). As presented in Table XI, the results highlight the distinct advantages of Max pooling in enhancing palmprint recognition performance.

Max pooling compresses the spatial dimensions of feature maps by selecting the maximum value within local regions, effectively retaining critical information while discarding irrelevant details. Furthermore, it enhances robustness against noise introduced by variations in lighting, skin texture, and capture conditions, while providing translation invariance, enabling the model to effectively manage slight positional shifts during acquisition. Additionally, max pooling indirectly increases the network's receptive field, facilitating the capture of broader contextual information, and significantly reduces computational complexity.

5) Effect of the Loss Function: ACPLoss with $m = 0.65$ and $\lambda = 3e^{-3}$ is evaluated against Cross Entropy, ArcFace Loss [24] with $m = 0.65$ and $s = 64$, CurricularFace Loss ($L_{AS'}$) [25] with $m = 0.65$ and $s = 64$, and C-LMCL [28] with $m = 0.65$ and $s = 48$. Table XII clearly shows that RDRLA performs relatively stable behavior and superior performance on the considered datasets, with Cross Entropy and ArcFace Loss showing the worst performing methods.

6) Effect of the Parameter λ of ACPLoss.: Table XIII demonstrates how varying λ values influence feature space distributions. As for close-set setting, an increase in λ from 0 to $3e^{-3}$ boosts Rank-1 accuracy from 95.14% to 95.48% on HIT-NIST-V1 dataset and from 88.21% to 88.43% on MPD dataset, while EER decreases from 1.54% to 1.51% on HIT-NIST-V1 dataset and from 7.19% to 5.16% on MPD dataset. As for open-set setting, an increase in λ from 0 to $3e^{-3}$ leads to Rank-1 accuracy improvements from 99.38% to 99.62% on MPD dataset and from 95.34% to 95.50% on BJTU_PlamV2 dataset, while EER decreases from 6.40% to 6.13% on MPD dataset and from 8.76% to 6.84% on BJTU_PlamV2 dataset. However, a too large λ value (e.g., $\lambda = 1e^{-1}$) causes a noticeable performance degradation, although it may yield superior open-set Rank-1 accuracy on HIT-NIST-V1 dataset. Considering overall performance across different datasets, $\lambda = 3e^{-3}$ emerges as the optimal setting.

TABLE XII
THE EFFECT OF LOSS FUNCTION (%)

Criteria	Loss functions	HIT-NIST-V1		IITD		MPD		BJTU_PalmV2	
		Rank-1	EER	Rank-1	EER	Rank-1	EER	Rank-1	EER
Close-set	Cross Entropy	93.47	1.56	97.95	**0.38**	87.64	7.21	93.87	1.59
	ArcFace [24]	94.74	2.59	98.68	0.50	87.98	6.12	93.72	2.56
	CurricularFace [25]	95.14	1.54	98.36	0.65	88.21	7.19	95.16	1.41
	C-LMCL [28]	94.67	1.79	98.61	0.57	87.23	6.07	94.93	1.40
	ACPLoss (ours)	**95.48**	**1.51**	**98.77**	0.54	**88.43**	**5.16**	**95.24**	**1.37**
Open-set	Cross Entropy	93.59	**8.05**	99.14	1.76	99.03	9.64	94.72	7.27
	ArcFace [24]	92.89	9.90	**99.31**	2.32	99.19	10.57	94.26	8.05
	CurricularFace [25]	**94.37**	8.49	98.97	1.92	99.38	6.40	95.34	8.76
	C-LMCL [28]	94.28	8.72	**99.31**	1.81	**99.62**	6.39	95.35	8.15
	ACPLoss (ours)	94.29	8.43	99.15	**1.53**	**99.62**	**6.13**	**95.50**	**6.84**

TABLE XIII
THE EFFECT OF λ OF ACPLOSS (%)

Criteria	Values	HIT-NIST-V1		IITD		MPD		BJTU_PalmV2	
		Rank-1	EER	Rank-1	EER	Rank-1	EER	Rank-1	EER
Close-set	$\lambda = 0$	95.14	1.54	98.36	0.65	88.21	7.19	95.16	1.41
	$\lambda = 1e^{-4}$	95.23	1.65	**99.18**	**0.24**	88.02	5.30	95.38	1.29
	$\lambda = 5e^{-4}$	95.23	1.61	98.85	0.53	87.89	5.27	**95.76**	1.63
	$\lambda = 1e^{-3}$	95.47	1.72	98.79	0.54	88.09	5.12	95.01	1.45
	$\lambda = 3e^{-3}$	**95.48**	**1.51**	98.77	0.54	**88.43**	5.16	95.24	1.37
	$\lambda = 5e^{-3}$	95.12	1.65	98.71	0.63	88.31	5.07	95.54	1.47
	$\lambda = 8e^{-3}$	94.93	1.56	98.32	0.66	88.33	5.12	95.23	**1.24**
	$\lambda = 1e^{-2}$	94.71	1.67	98.21	0.66	88.19	5.12	95.46	1.32
	$\lambda = 1e^{-1}$	94.71	1.67	98.18	0.67	88.39	**4.72**	95.69	1.25
Open-set	$\lambda = 0$	94.37	8.49	98.97	1.92	99.38	6.40	95.34	8.76
	$\lambda = 1e^{-4}$	94.43	8.67	99.15	1.81	99.40	6.13	94.72	7.41
	$\lambda = 5e^{-4}$	94.57	8.96	**99.66**	1.64	99.57	6.39	95.03	7.51
	$\lambda = 1e^{-3}$	94.24	9.02	99.35	**1.31**	99.59	6.40	95.19	7.25
	$\lambda = 3e^{-3}$	94.29	8.43	99.15	1.53	**99.62**	6.13	**95.50**	**6.84**
	$\lambda = 5e^{-3}$	94.28	**8.29**	99.49	1.53	**99.62**	**6.03**	95.43	6.92
	$\lambda = 8e^{-3}$	94.28	8.40	99.65	1.87	99.57	6.39	95.21	6.87
	$\lambda = 1e^{-2}$	94.29	8.93	99.32	1.51	99.57	6.69	94.98	7.09
	$\lambda = 1e^{-1}$	**94.70**	8.67	99.14	1.66	99.21	7.21	94.92	7.26

V. CONCLUSION

In this study, we introduce RDRLA to tackle the primary challenges of PROIE and palmprint recognition in open environment, marking a departure from previous efforts with the first-ever proposal of a FVPs-free PROIE method. This method adeptly addresses the issue of FVPs occlusion, which arises from the high degree of freedom in contactless acquisition. Specifically, CHSST is leveraged for hand segmentation, enhancing its performance through the integration of publicly available palmprint datasets from closed environments. PalmAlNet is developed to standardize hand orientation, while an adaptive PROIE method (FFARD) based on maximum inscribed circle searching with constraints is introduced. Additionally, RLANN is investigated to achieve outperforming close-set and open-set palmprint recognition results, thereby establishing a new baseline for the biometrics community. The effectiveness of RDRLA is demonstrated through comparative or superior results on public datasets such as HIT-NIST-V1, IITD, MPD, and BJTU_PalmV2, showcasing its potential against SOTA methods.

Future work will focus on extending the proposed method RDRLA to other biometric modalities, enhancing its versatility and impact. To better meet practical application requirements, the model will be further optimized to reduce complexity, creating a more lightweight structure. This optimization aims to shorten training time and improve inference speed. Moreover, controllable image generation algorithms will be investigated for palmprint dataset augmentation, further enhancing the model's performance in palmprint recognition.

ACKNOWLEDGMENT

The authors are grateful to Nanyang Technological University, The Hong Kong Polytechnic University, University of Sfax, College of Engineering, Pune 411005, India, COEP Technological University, Tongji University, Beijing Jiaotong University, Chinese Academy of Sciences' Institute of Automation, and IIT Delhi for sharing their palmprint datasets.

REFERENCES

[1] S. Zhao, L. Fei, B. Zhang, J. Wen, and P. Zhao, "Tensorized multi-view low-rank approximation based robust hand-print recognition," *IEEE Trans. Image Process.*, vol. 33, pp. 3328–3340, 2024.

[2] Z. Yang, A. B. J. Teoh, B. Zhang, L. Leng, and Y. Zhang, "Physics-driven spectrum-consistent federated learning for palmprint verification," *Int. J. Comput. Vis.*, vol. 132, no. 10, pp. 4253–4268, Oct. 2024.

[3] Y. Wang et al., "Dense hybrid attention network for palmprint image super-resolution," *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 54, no. 4, pp. 2590–2602, Apr. 2024.

[4] H. Shao and D. Zhong, "Multi-target cross-dataset palmprint recognition via distilling from multi-teacher," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–14, 2023.

[5] X. Liang, D. Fan, J. Yang, W. Jia, G. Lu, and D. Zhang, "PKLNet: Keypoint localization neural network for touchless palmprint recognition based on edge-aware regression," *IEEE J. Sel. Topics Signal Process.*, vol. 17, no. 3, pp. 662–676, May 2023.

[6] H. Shao, D. Zhong, and X. Du, "Efficient deep palmprint recognition via distilled hashing coding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 714–723.

[7] Y. Zhang, L. Zhang, R. Zhang, S. Li, J. Li, and F. Huang, "Towards palmprint verification on smartphones," 2020, *arXiv:2003.13266*.

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

[9] 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, Sep. 2003.

[10] M. Izadpanahkakhk, S. Razavi, M. Taghipour-Gorjikolaie, S. Zahiri, and A. Uncini, "Deep region of interest and feature extraction models for palmprint verification using convolutional neural networks transfer learning," *Appl. Sci.*, vol. 8, no. 7, p. 1210, Jul. 2018.

[11] A. S. ELSayed, H. M. Ebeid, M. Roushdy, and Z. T. Fayed, "Robust palm and knuckle ROI extraction in unconstrained environment," *Pattern Anal. Appl.*, vol. 22, no. 4, pp. 1537–1559, Nov. 2019.

[12] L. Yan, L. Leng, A. B. J. Teoh, and C. Kim, "A realistic hand image composition method for palmprint ROI embedding attack," *Appl. Sci.*, vol. 14, no. 4, p. 1369, Feb. 2024.

[13] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

[14] L. Shen, Y. Zhang, K. Zhao, R. Zhang, and W. Shen, "Distribution alignment for cross-device palmprint recognition," *Pattern Recognit.*, vol. 132, Dec. 2022, Art. no. 108942.

[15] W. Rong, Z. Yang, and L. Leng, "Channel group-wise drop network with global and fine-grained-aware representation learning for palm recognition," in *Proc. IEEE Int. Joint Conf. Biometrics (IJCB)*, Oct. 2022, pp. 1–9.

[16] L. Fei, W. K. Wong, S. Zhao, J. Wen, J. Zhu, and Y. Xu, "Learning spectrum-invariance representation for cross-spectral palmprint recognition," *IEEE Trans. Syst. Man, Cybern., Syst.*, vol. 53, no. 6, pp. 3868–3879, Jun. 2023.

[17] Z. Yang et al., "CO_3Net: Coordinate-aware contrastive competitive neural network for palmprint recognition," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–14, 2023.

[18] Q. Zhu et al., "Contactless palmprint image recognition across smartphones with self-paced CycleGAN," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 4944–4954, 2023.

[19] K. Zhao et al., "Bézierpalm: A free lunch for palmprint recognition," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2022, pp. 19–36.

[20] L. Shen et al., "RPG-palm: Realistic pseudo-data generation for palmprint recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 19548–19559.
[21] W. Liu, Y. Wen, Z. Yu, and M. Yang, "Large-margin softmax loss for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 507–516.
[22] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "SphereFace: Deep hypersphere embedding for face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 212–220.
[23] H. Wang et al., "CosFace: Large margin cosine loss for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5265–5274.
[24] 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. (CVPR)*, Jun. 2019, pp. 4690–4699.
[25] Y. Huang et al., "CurricularFace: Adaptive curriculum learning loss for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 5901–5910.
[26] F. Boutros, N. Damer, F. Kirchbuchner, and A. Kuijper, "ElasticFace: Elastic margin loss for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, New Orleans, LA, USA, Jun. 2022, pp. 1578–1587.
[27] J. Zhou, X. Jia, Q. Li, L. Shen, and J. Duan, "UniFace: Unified cross-entropy loss for deep face recognition," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 20673–20682.
[28] D. Zhong and J. Zhu, "Centralized large margin cosine loss for open-set deep palmprint recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 6, pp. 1559–1568, Jun. 2020.
[29] V. Kanhangad, A. Kumar, and D. Zhang, "A unified framework for contactless hand identification," *IEEE Trans. Inf. Forensics Security*, vol. 20, no. 5, pp. 1415–1424, Sep. 2011.
[30] N. Charfi, H. Trichili, A. Alimi, and B. Solaiman, "Local invariant representation for multi-instance touchless palmprint identification," in *Proc. IEEE Int. Conf. Syst. Man Cybern. Syst.*, Oct. 2016, pp. 3522–3527.
[31] COEP Technological University.(2022). *COEP Palm Print Database*. [Online]. Available: https://www.coep.org.in/resources/coeppalmprintdatabase
[32] T. Chai, S. Prasad, and S. Wang, "Boosting palmprint identification with gender information using DeepNet," *Future Gener. Comput. Syst.*, vol. 99, pp. 41–53, Oct. 2019.
[33] W. M. Matkowski, T. Chai, and A. W. K. Kong, "Palmprint recognition in uncontrolled and uncooperative environment," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 1601–1615, 2020.
[34] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst. Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.
[35] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes challenge: A retrospective," *Int. J. Comput. Vis.*, vol. 111, no. 1, pp. 98–136, Jan. 2015.
[36] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Sep. 2022, pp. 11976–11986.
[37] W. Zhang et al., "TopFormer: Token pyramid transformer for mobile semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 12083–12093.
[38] A. G. Howard et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.
[39] Q. Wan, Z. Huang, J. Lu, G. Yu, and L. Zhang, "SeaFormer: Squeeze-enhanced axial transformer for mobile semantic segmentation," in *Proc. 11th Int. Conf. Learn. Represent. (ICLR)*, 2023, pp. 1–19.
[40] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.
[41] S. Jadon, "A survey of loss functions for semantic segmentation," in *Proc. IEEE Conf. Comput. Intell. Bioinf. Comput. Biol. (CIBCB)*, Oct. 2020, pp. 1–7.
[42] The Institute of Automation, Chinese Academy of Sciences. (2005).*CASIA Palm Print Database*. [Online]. Available: http://www.cbsr.ia.ac.cn/english/index.asp
[43] Y. Hao, Z. Sun, T. Tan, and C. Ren, "Multispectral palm image fusion for accurate contact-free palmprint recognition," in *Proc. 15th IEEE Int. Conf. Image Process.*, Oct. 2008, pp. 281–284.
[44] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput Vis. Pattern Recognit.*, Jun. 2017, pp. 4700–4708.
[45] Y. Chen, J. Li, H. Xiao, X. Jin, S. Yan, and J. Feng, "Dual path networks," in *Proc. NIPS*, 2017, pp. 1–9.
[46] J. Zhao, Y. Fang, and G. Li, "Recurrence along depth: Deep convolutional neural networks with recurrent layer aggregation," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Jan. 2021, pp. 10627–10640.
[47] W. Jia, Q. Ren, Y. Zhao, S. Li, H. Min, and Y. Chen, "EEPNet: An efficient and effective convolutional neural network for palmprint recognition," *Pattern Recognit. Lett.*, vol. 159, pp. 140–149, Jul. 2022.
[48] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-Net: Efficient channel attention for deep convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11534–11542.
[49] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 30, Jun. 2017, pp. 5998–6008.
[50] B. Li, Y. Liu, and X. Wang, "Gradient harmonized single-stage detector," in *Proc. AAAI Conf. Artif. Intell.*, Sep. 2019, vol. 33, no. 1, pp. 8577–8584.
[51] T. Chai, S. Prasad, J. Yan, and Z. Zhang, "Contactless palmprint biometrics using DeepNet with dedicated assistant layers," *Vis. Comput.*, vol. 39, no. 9, pp. 4029–4047, Sep. 2023.
[52] A. Kumar and S. Shekhar, "Personal identification using rank-level fusion," *IEEE Trans. Syst. Man, Cybern. C*, vol. 41, no. 5, pp. 743–752, Jan. 2011.
[53] W. Jia, R.-X. Hu, Y.-K. Lei, Y. Zhao, and J. Gui, "Histogram of oriented lines for palmprint recognition," *IEEE Trans. Syst. Man, Cybern., Syst.*, vol. 44, no. 3, pp. 385–395, Mar. 2014.
[54] Y.-T. Luo et al., "Local line directional pattern for palmprint recognition," *Pattern Recognit.*, vol. 50, pp. 26–44, Feb. 2016.
[55] A. Genovese, V. Piuri, K. N. Plataniotis, and F. Scotti, "PalmNet: Gabor-PCA convolutional networks for touchless palmprint recognition," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 12, pp. 3160–3174, Dec. 2019.
[56] X. Liang, J. Yang, G. Lu, and D. Zhang, "CompNet: Competitive neural network for palmprint recognition using learnable Gabor kernels," *IEEE Signal Process. Lett.*, vol. 28, pp. 1739–1743, 2021.
[57] Z. Yang, H. Huangfu, L. Leng, B. Zhang, A. B. J. Teoh, and Y. Zhang, "Comprehensive competition mechanism in palmprint recognition," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 5160–5170, 2023.
[58] M. D. Zeiler and R. Fergus, "Stochastic pooling for regularization of deep convolutional neural networks," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Jan. 2013, pp. 1–9.

Tingting Chai (Member, IEEE) received the B.S. degree in information security and the M.S. degree in computer technology from China University of Mining and Technology, Beijing, China, in 2012 and 2014, respectively, and the Ph.D. degree from the Institute of Information Science, Beijing Jiaotong University, Beijing, in 2020. Currently, she works as an Assistant Professor with the Faculty of Computing, Harbin Institute of Technology, Harbin, China. Her research interests include biometrics, computer vision, and pattern recognition.

Xin Wang received the B.S. degree in cyberspace security from Harbin Institute of Technology, Weihai, China, in 2024. He is currently pursuing the M.S. degree in information security with the School of Cyberspace Security, University of Science and Technology of China, Hefei, China. His research interests include biometrics, computer vision, and image processing.

Ru Li (Member, IEEE) received the B.E. degree in electronic information engineering from China University of Petroleum, Qingdao, China, in 2016, and the Ph.D. degree from the School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu, China, in 2022. She was a Visiting Student Researcher with the University of Oxford. She is currently a Lecturer with the Faculty of Computing, Harbin Institute of Technology, Harbin, China. Her research interests include image processing and computer vision.

Wei Jia (Member, IEEE) received the B.Sc. degree in informatics from Central China Normal University, Wuhan, China, in 1998, the M.Sc. degree in computer science from Hefei University of Technology, Hefei, China, in 2004, and the Ph.D. degree in pattern recognition and intelligence systems from the University of Science and Technology of China, Hefei, in 2008. He is currently a Professor with the School of Computer and Information, Hefei University of Technology. His research interests include computer vision, biometrics, pattern recognition, and image processing.

Xiangqian Wu (Senior Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees in computer science from Harbin Institute of Technology (HIT), Harbin, China, in 1997, 1999, and 2004, respectively. He works as a Lecturer (2004–2006), an Associate Professor (2006–2009), and a Professor (since 2009) with the School of Computer Science and Technology, HIT. He is a Principal Investigator of dozens of research projects, including the projects of the Natural Science Foundation of China (NSFC) and the National 863 Plan Project. He held dozens of patents of China, USA, and Hong Kong. His current research interests include image processing, biomedical image analysis, and computer vision.