Image Enhancement Optimization of ResNet Convolutional Neural Network for Palmprint Recognition

Yunfei Qu*[a]
[a]Institute of Automation, Chinese Academy of Science, Beijing, China;
[*] Corresponding author: 2015yunfei@gmail.com

ABSTRACT

Palmprint recognition serves as a biometric identification technology, involving the analysis and comparison of ridge patterns on an individual's palm for identity verification or individual recognition. This technique relies on the unique patterns present on the skin of each person's palm, encompassing features such as wrinkles, grooves, and skin texture. Utilizing image processing techniques and convolutional neural networks (CNNs), palmprint features can be extracted, and classification tasks can be achieved even with a limited training dataset. This study explores the impact of various data augmentation methods on the improvement of classification accuracy for ResNet convolutional neural networks. Experimenting with alterations in brightness, contrast, noise addition, and image flipping, as well as exploring combinations of these augmentation techniques, resulted in diverse experimental outcomes. Significantly, the repetitive adjustment of brightness and contrast, along with their combined effects, notably contributed to enhancing accuracy in the ResNet convolutional neural network.

Keywords: Palmprint recognition, data augmentation, brightness transformation, contrast transformation, convolutional neural network

1. INTRODUCTION

Palmprint recognition, a key facet of biometric identification, has evolved from ancient manual methods to contemporary automated systems. Initially employed for identity verification thousands of years ago, palmprints have transitioned to digital formats with advancements in computer science and image processing. The advent of automated systems, particularly utilizing Convolutional Neural Networks (CNNs) and large-scale datasets, has significantly enhanced the accuracy and reliability of palmprint recognition[1]. Its unique features, even among identical twins, contribute to its high precision and non-repudiation, making it a preferred choice for biometric authentication in applications like smartphone unlocking, attendance systems, border security, and forensics.

The rich feature information inherent in palmprints sets them apart from other biometric features, such as fingerprints or irises. Their reliability in individual identity verification, coupled with the advantages of security and convenience, has established palmprint recognition as an integral component in various domains of modern life[2].

The use of feature extraction techniques allows for a clearer observation of the specific details of palmprints, as shown in the following figure: The left image is a grayscale capture of palmprints from an individual, the middle image depicts the result after applying low-pass Gaussian filtering to the grayscale image, and the right image illustrates the outcome of Gabor filtering applied to the image[3].

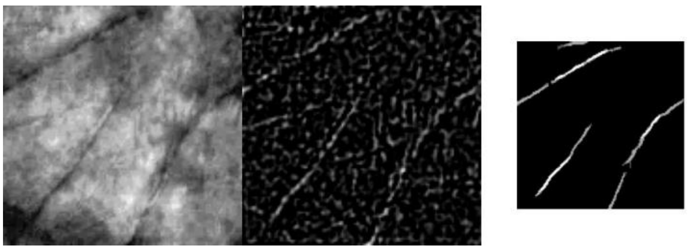

Figure 1. Examples

In recent years, numerous scholars have made innovative research contributions to palmprint recognition. David conducted a comprehensive survey of the latest palmprint recognition methods[4], performing comparative studies to evaluate the performance of state-of-the-art approaches. Concurrently, they compared various methods based on holistic and local features for recognition performance. Youssef conducted an extensive comparative study on the performance of various recent texture descriptors in palmprint recognition[5]. They introduced local texture descriptors for the first time in palmprint recognition, achieving satisfactory results. To further enhance the accuracy of palmprint recognition, Zhang employed structured light imaging technology to acquire three-dimensional palmprint data, extracting various unique features[6]. Their experimental results indicate that three-dimensional palmprint recognition possesses higher anti-counterfeiting capabilities and is more robust to illumination variations and severe abrasions on the palm surface, although the accuracy of three-dimensional palmprint recognition is slightly lower than that of two-dimensional palmprint recognition. Michele applied the MobileNet V2 deep convolutional neural network to palmprint recognition, achieving outstanding results on datasets with a substantial number of samples[7]. However, it is noteworthy that in small-scale palmprint training datasets, there are scarcely any recognition methods with high accuracy, which is a more prevalent scenario in real-life situations. To improve recognition accuracy on small-scale datasets, in addition to proposing superior recognition algorithms, data augmentation is also a direction worth considering.

In the field of computer image processing, data augmentation is a technique employed to augment the training dataset by applying transformations, aiming to enhance the model's robustness and generalization performance. Common data augmentation techniques include brightness transformation, contrast transformation, adding random noise, flipping images, rotating images, and random cropping. Based on these fundamental image processing methods, I conducted experiments using individual or a combination of multiple augmentation techniques to explore the most beneficial approaches for improving the recognition accuracy of ResNet-18 networks[8].

In the following subsection, we will provide an overview of existing work and studies relevant to the current research. Firstly, a brief introduction to the convolutional neural network employed in this experiment will be presented, as it serves the purpose of palmprint recognition. Subsequently, we will elucidate the data augmentation techniques utilized and their corresponding effects. The third subsection will provide a concise description of the experimental setup, including the computational environment. In the fourth subsection, we will delineate the criteria for grouping and present detailed experimental results. The fifth subsection will encompass a comprehensive summary of all obtained data and explain some of the rules obtained from the experimental data.

2. RELATED WORK

2.1 Deep residual network

We employed the ResNet-18 deep residual network for palmprint recognition, integrating predefined data augmentation and preprocessing operations, including random cropping, rotation, normalization, etc., to enhance the model's generalization capability. Simultaneously, the dataset was transformed into an iterable data loader for subsequent training and validation purposes. The pre-trained ResNet-18 model, initialized with weights pre-trained on the ImageNet dataset, was loaded. All parameters of the model were frozen to prevent the updating of pre-trained weights during the training process. Modifications were made to the model's final fully connected layer to accommodate the task-specific output class count (in this case, 99 classes), and an additional hidden layer with ReLU activation and dropout was added[9].

The model employs the cross-entropy loss function and the Adam optimizer. During the training process, it iterates through the training data, computes the loss and gradients, performs backward propagation, and optimizes the parameters. Model performance is evaluated on the validation set by calculating validation loss and accuracy[10]. Throughout the training procedure, the model that achieves the highest accuracy on the validation set is saved.

In convolutional neural networks, the deep residual network (ResNet) addresses the issue of vanishing gradients by introducing residual blocks. This innovation facilitates the training of deep structures more effectively. Through skip connections, the residual block directly adds the input to the output, enabling the model to learn residuals rather than complete feature mappings. This contributes to a better capture of information in images, thereby enhancing the network's performance[11].

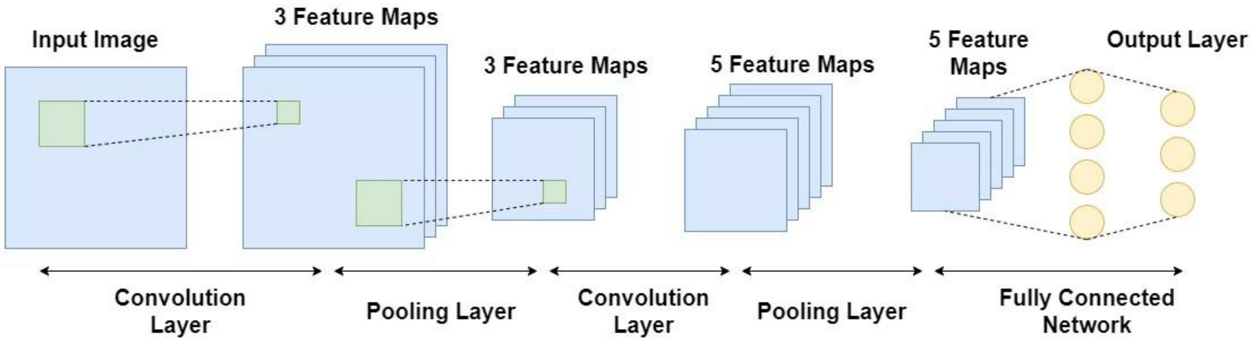

Figure 2. The basic structure of a typical convolutional neural network.

2.2 Common data enhancement methods

Data augmentation is a technique vital for expanding the training dataset, enhancing deep learning model performance through various transformations[12]. In computer vision and image processing, common techniques include brightness and contrast adjustments, image flipping, rotation, and random noise introduction.

Early methods focused on linear transformations for brightness and contrast, achieved through pixel value operations. Research progressed to non-linear methods, such as logarithmic transformation and gamma correction, improving image detail capture. With the rise of deep learning, brightness transformations in neural networks have become prevalent, enabling flexible adaptation to different brightness conditions[13].

Contrast enhancement evolved from linear to non-linear methods like histogram equalization. In deep learning, neural networks learn non-linear relationships, improving adaptability. Contrast transformation in this experiment adjusts the difference between lightest and darkest values, enhancing or reducing contrast for improved visibility.

Random noise introduction simulates real-world variability, enhancing the model's generalization ability. Image flipping, initially horizontal and vertical, now includes random flipping, diversifying datasets and aiding recognition tasks. Early fixed-angle rotations evolved to multi-angle rotations for improved adaptation.

Another technique, random cropping, extracts random regions from images, promoting translation invariance and focusing on different object parts, contributing to better localization and recognition[14].

These augmentation techniques play a crucial role in preventing overfitting, enhancing generalization, and improving performance across diverse datasets in computer vision tasks like classification, detection, and segmentation. The goal is to expose the model to a broad range of variations, making it robust in real-world scenarios.

During each epoch of the training phase, the training process involves comparing the model's predictions on the training data to the corresponding ground truth labels. For each sample, the model identifies the maximum predicted value and its associated class index, indicating the class deemed most probable by the model. Simultaneously, a Boolean tensor is generated to signify whether each sample's prediction is correct. Subsequently, the proportion of correctly predicted samples within each batch is computed. In essence, the computation of accuracy involves consolidating information on prediction correctness across batches, achieved by comparing the predicted class indices to the true labels. The overall accuracy is then obtained[15]. Notably, accuracy calculations in the code occur at the conclusion of each epoch. The specific calculation formula is as follows:

$$average\ validation\ accuracy = \frac{valid\_acc}{valid\_data\_size} \tag{1}$$

The formula for calculating average validation accuracy is expressed as the ratio of valid_acc (the sum of correctly predicted samples on the validation set) to valid_data_size (the total number of samples in the validation set). The computation of average validation accuracy involves dividing the number of correctly predicted samples by the total number of validation samples.

3. EXPERIMENTAL DETAIL

All experiments were conducted on GPU servers at the Beijing Super Cloud Computing Center. The server was equipped with 32GB V100 computing cards. Each machine had 8 GPUs of the NVIDIA Tesla V100-SXM2 32GB VRAM model. Each GPU card was allocated 10 CPU cores and 38GB of memory by default.

4. EXPERIMENTAL DESIGN

4.1 Based on a variety of data enhancement methods

Brightness transformation and contrast transformation, as the most fundamental data augmentation techniques, serve as the foundation for our experiment. We will investigate whether the ResNet convolutional neural network can achieve better recognition accuracy with the assistance of other data augmentation methods.

In the first experiment, we applied a single-mode data augmentation process to each image in the training dataset to observe whether we could achieve better accuracy. The specific transformation methods are as follows:

1. Random rotation
2. Random flipping
3. Random brightness transformation
4. Random contrast transformation
5. Random noise addition, with a noise factor of 0.05

After the aforementioned transformations, the original dataset, initially comprising three images, has been augmented to a total of eighteen images. The recognition accuracy of the model on the augmented dataset has improved to 0.8215 (with a batch size of 32) and 0.8013 (with a batch size of 64).

Considering the noticeable impact of black borders generated by rotation on model training, the second experiment directly excludes the rotation transformation. Instead, it utilizes the remaining four data augmentation methods to expand the dataset.

After the rotation transformation is removed, the accuracy is further improved by 4% to 5%.

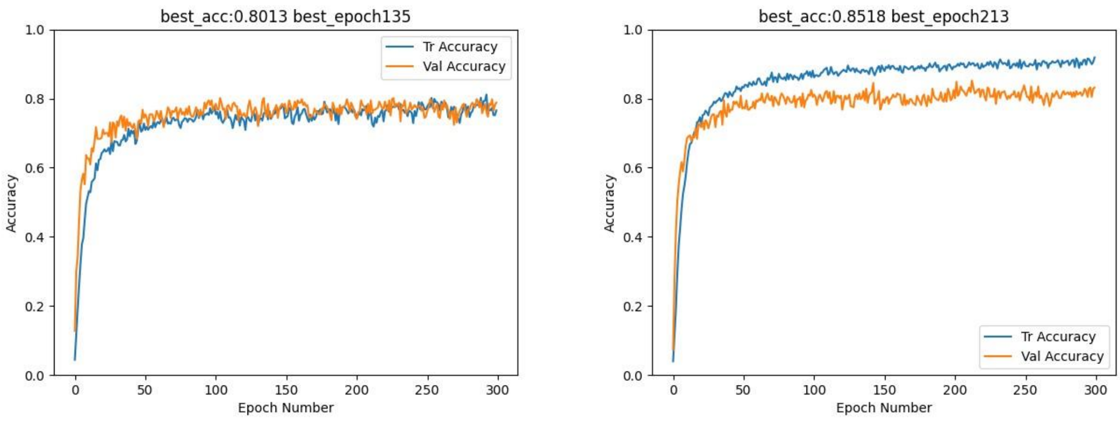

Figure 3. The left image shows the results of Experiment 1, while the right image displays the results of Experiment 2.

The first two sets of experiments involve transforming the original images using a single method to obtain new images. Starting from the third set of experiments, while processing the original images, random flipping is applied before the same transformations. The specific transformation methods are as follows (with a noise factor of 0.05) :

1. Random brightness transformation
2. After performing brightness transformation, introduce random noise
3. Random contrast transformation
4. After conducting contrast transformation, introduce random noise
5. Perform brightness and contrast transformations
6. After applying brightness and contrast transformations, introduce random noise
7. Random flipping
    a) After random horizontal and vertical flipping, perform brightness transformation
    b) After random horizontal and vertical flipping, apply brightness transformation and introduce random noise
    c) After random horizontal and vertical flipping, perform contrast transformation
    d) After random horizontal and vertical flipping, perform contrast transformation and introduce random noise
    e) After random horizontal and vertical flipping, apply brightness and contrast transformations.
    f) After random horizontal and vertical flipping, apply brightness and contrast transformations and introduce random noise.

Unfortunately, after adding random flipping to the images, the model's recognition accuracy declined. Upon simple analysis, I believe it may be due to the introduction of multiple other data augmentation methods after a single random flip. This resulted in a disproportionately high percentage of images with a singular flip in the entire training set, affecting the model's generalization ability and reducing recognition accuracy on an unknown test set.

Due to the results of the third group of experiments, the fourth group made two adjustments. One was to adjust the random noise factor to 0.03, and the other was to modify the compound transformation for the original images. Before applying brightness transformation, contrast transformation, or adding random noise, the images were randomly flipped (including both horizontal and vertical flips). And the results of each image flip were independent of others. The specific transformations are as follows:

1. Random brightness transformation
2. After performing brightness transformation, introduce random noise
3. Random contrast transformation
4. After conducting contrast transformation, introduce random noise
5. Perform brightness and contrast transformations
6. After applying brightness and contrast transformations, introduce random noise
7. Random flipping
8. After random horizontal and vertical flipping, perform brightness transformation
9. After random horizontal and vertical flipping, perform brightness transformation and introduce noise
10. After random horizontal and vertical flipping, perform contrast transformation
11. After random horizontal and vertical flipping, perform contrast transformation and introduce noise
12. After random horizontal and vertical flipping, apply brightness and contrast transformations

13. After random horizontal and vertical flipping, apply brightness and contrast transformations and introduce random noise

Through experiments, it was found that such data augmentation methods did not significantly improve the model's accuracy, and the possible reason is that random flipping itself does not provide much help for palmprint recognition. Additionally, this data augmentation method may lead to a decrease in the model's generalization ability since the orientation of all palmprint images (whether in the training or test dataset) is the same.

Due to the unsatisfactory results of the fourth set of experiments, the fifth set of experiments eliminated the data augmentation method of random flipping. It retained only brightness transformation, contrast transformation, random noise, and their various combinations, while also incorporating the common data augmentation method of image cropping. It's worth noting that the original palmprint images already have a relatively small resolution of 128x128 pixels. For the sake of training efficiency, in this set of experiments, only a 100x100 region is cropped as an extension of the original dataset. It can be speculated that image cropping might not have a significant impact on the model's recognition accuracy since the original images are already relatively small. This hypothesis will be validated in the next set of experiments. The specific methods are as follows:

1. Random brightness transformation
2. After performing brightness transformation, introduce random noise
3. Random contrast transformation
4. After conducting contrast transformation, introduce random noise
5. Perform brightness and contrast transformations
6. After applying brightness and contrast transformations, introduce random noise
7. Extract a new 100x100 image by cropping from the center of each original image.

After eliminating the data augmentation method of random flipping, the model's recognition accuracy experienced a significant improvement, increasing by approximately 3%. In order to further investigate how to combine known data augmentation techniques to achieve the best accuracy, the following experiments will be conducted.

Considering that adding random noise to already relatively small (128x128) images might have a more significant impact than anticipated, in the sixth set of experiments, I removed the data augmentation method of random noise. Additionally, I increased the number of times each transformation is applied to obtain different results under the same transformation. The specific data augmentation methods are as follows:

1. Four sets of brightness transformations
2. Four sets of contrast transformations
3. Four sets of brightness and contrast transformations
4. Extracting four new 100x100-pixel images from the center of each original image

Under a batch size of 32, the model's recognition accuracy has significantly improved.

4.2 Based on brightness and contrast

The results of the fifth and sixth sets of experiments have demonstrated that adding random noise as a means of data augmentation is not very effective. Considering the relatively small size of the images, in the seventh set of experiments, I removed the transformation of image cropping to observe if it would lead to better accuracy. The specific data augmentation methods are as follows:

1. Four sets of brightness transformations
2. Four sets of contrast transformations
3. Four sets of brightness and contrast transformations

With a batch size of 32, the results of this set of experiments showed almost no change compared to the previous one. However, when the batch size is increased to 64, the accuracy improves by 1% to 2%. This indicates that the image cropping data augmentation method has minimal impact on the training of the palmprint recognition model.

From the eighth set of experiments onward, we adjusted the number of times each data augmentation method is applied to the original image. The aim is to observe the optimal recognition accuracy obtained by varying the frequency of data augmentation for a single original image. The specific data augmentation methods are as follows:

1. Two sets of brightness transformations
2. Two sets of contrast transformations
3. Two sets of brightness and contrast transformations

The experimental results showed no significant changes.

In the ninth experiment, we kept the data augmentation methods unchanged and only increased the number of times we applied data augmentation to the original images. We then observed whether better experimental results could be obtained. The specific data augmentation methods are as follows:

1. Six sets of brightness transformations
2. Six sets of contrast transformations
3. Six sets of brightness and contrast transformations

As of now, we have achieved the best results in all experiments. The model's performance has further improved by approximately 3% compared to the previous results. To investigate whether increasing the number of processing times for each original image further enhances recognition accuracy, we will continue with additional experiments.

In the tenth experiment, we will increase the number of times for data augmentation on the original images to 8 and then conduct the experiment. The specific data augmentation methods are as follows:

1. Eight sets of brightness transformations
2. Eight sets of contrast transformations
3. Eight sets of brightness and contrast transformations

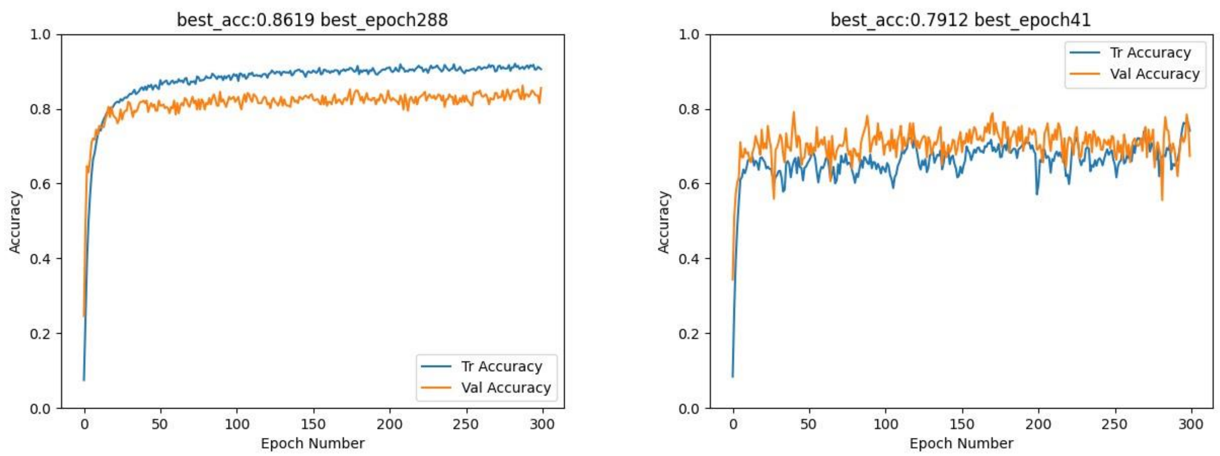

Figure 4. The left image shows the results of Experiment 9, the right image displays the results of Experiment 10.

From the experimental results, it can be observed that the accuracy has experienced a significant decline. Thus, we can conclude that performing 6 augmentations for each original image yields the highest recognition accuracy.

The summarized results of the ten groups are presented in the following table:

Table 1. Results of all ten experiments.

Group number	Batch size is 32	Batch size is 64
Group1	0.8215	0.8013
Group2	0.8384	0.8519
Group3	0.7980	0.8081
Group4	0.8047	0.8013
Group5	0.8384	0.8384
Group6	0.8485	0.8350
Group7	0.8249	0.8418
Group8	0.8249	0.8451
Group9	0.8519	0.8620
Group10	0.8114	0.7912

5. CONCLUSION

Almost all common data augmentation techniques can enhance the precision of the ResNet-18 model in palmprint recognition and detection. Brightness and contrast transformations are the most frequently employed augmentation methods, proving to be effective means of expanding the original palmprint dataset to improve recognition accuracy. However, image flipping and rotation operations are not suitable for augmenting palmprint images, as the relative orientation and position between the palm and the data-capturing device remain fixed. Performing flipping or rotation does not contribute to enhancing the model's generalization ability.

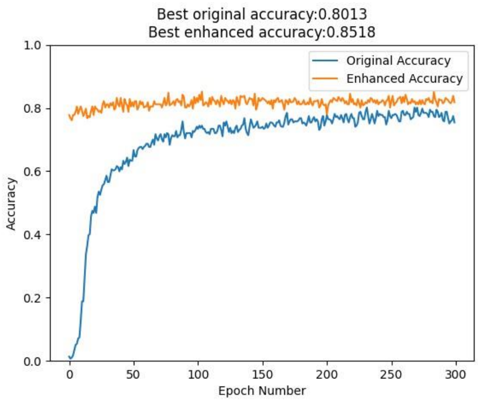

Figure 5. Comparison between the original results and the results after data augmentation.

Furthermore, it is noteworthy that subjecting each image to six independent brightness transformations and six separate contrast transformations, followed by six independent compound transformations of brightness and contrast, yields the optimal results. This signifies that a single palmprint image, after undergoing data augmentation procedures, is expanded to a total of 19 images (including the original). This augmentation strategy elevates the accuracy of the ResNet-18 network in palmprint recognition from 80% to 85%.

REFERANCE

[1] Zhang, M. Li, J. Wang, Y. Xu, G. (2021) CECNet: Coordinate Encoding Competitive Neural Network For Palm Vein Recognition by Soft Large Margin Centralized Cosine Loss. IEEE Access, vol.11:pp.141329-141342. https://ieeexplore.ieee.org/abstract/document/9514931.

[2] Al-Taie, S. A. M., & Khaleel, B. I. (2023). Palm Print Recognition Using Intelligent Techniques: A review. Jurnal Ilmiah Teknik Elektro Komputer dan Informatika, 9(1), 156-164. http://scholar.googleusercontent.com/scholar?q=cache:G5U5vuhcIEcJ:scholar.google.com/+palm+print+review&hl=zh-CN&as_sdt=0,5.

[3] Wu, X. Zhao, Q. Bu W. (2014) A SIFT-based contactless palmprint verification approach using iterative RANSAC and local palmprint descriptors. Pattern Recognition, Volume 47, Issue 10: Pages 3314-3326. https://doi.org/10.1016/j.patcog.2014.04.008.

[4] Zhang, D. Zuo, W. Feng, Y. (2012) A Comparative Study of Palmprint Recognition Algorithms. ACM Computing Surveys, Vol. 44, No. 1. https://dl.acm.org/doi/10.1145/2071389.2071391.

[5] El merabet, Y. Ruichek, Y. (2020) Palmprint recognition using state-of-the-art local texture descriptors: a comparative study. IET Biometrics. https://doi.org/10.1049/iet-bmt.2019.0103.

[6] Zhang, D. Lu, G. Li, W. Zhang, L. Luo, N. (2009) Palmprint Recognition Using 3-D Information. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 39, no. 5, pp. 505-519, Sept. 2009. doi: 10.1109/TSMCC.2009.2020790.

[7] Michele, A. Colin, V. Diaz, D. (2019) MobileNet Convolutional Neural Networks and Support Vector Machines for Palmprint Recognition. Procedia Computer Science, Volume 157, 2019, Pages 110-117. https://doi.org/10.1016/j.procs.2019.08.147.

[8] Janani, P. Premaladha, J. Ravichandran, K. S. (2015). Image enhancement techniques: A study. Indian Journal of Science and Technology, vol 8(22), DOI: 10.17485/ijst/2015/v8i22/79318, September 2015

[9] Bartunek, J. S. Nilsson, M. Sallberg, B. Claesson, I. (2013) Adaptive Fingerprint Image Enhancement With Emphasis on Preprocessing of Data. IEEE Transactions on Image Processing, vol. 22, no. 2, pp. 644-656, Feb. 2013, doi: 10.1109/TIP.2012.2220373.

[10] Wang, L. Zhang, Q. Qian, Q. Wang, J. Pan, Y. Yang, R. Cheng, W. (2022) Multispectral Palm Print and Palm Vein Acquisition Platform and Recognition Method Based on Convolutional Neural Network. The Computer Journal, Volume 65, Issue 6, June 2022, Pages 1461–1471, https://doi.org/10.1093/comjnl/bxaa190

[11] Fan, Y., Li, J., Song, S. et al. (2022) Palmprint Phenotype Feature Extraction and Classification Based on Deep Learning. Phenomics 2, 219–229. https://doi.org/10.1007/s43657-022-00063-0.

[12] Yin, Y. Ghasedi, K. Wu, H. Yang, J. Tong, X. Fu, Y. (2023) NeRFInvertor: High Fidelity NeRF-GAN Inversion for Single-Shot Real Image Animation. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.8539-8548, 2023. doi: 10.1109/BTAS46853.2019.9186003

[13] Yadav, G. Yadav, D. Mouli, P. (2022) Chapter 4 - Statistical measures for Palmprint image enhancement. In Cognitive Data Science in Sustainable Computing, Machine Learning for Biometrics, Academic Press, 2022, Pages 65-85. https://doi.org/10.1016/B978-0-323-85209-8.00010-9.

[14] Yin, Y. Ghasedi, K. Wu, H. Yang, J. Tong, X. Fu, Y. (2023) NeRFInvertor: High Fidelity NeRF-GAN Inversion for Single-Shot Real Image Animation. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.8539-8548, 2023. doi: 10.1109/BTAS46853.2019.9186003

[15] Prasanna, Y. S. D. L., & Deepika, R. M. (2020). Palm print recognition using inner finger deep learning using neural network. International Journal, 5. http://www.ijasret.com/VolumeArticles/FullTextPDF/615_9.PALM_PRINT_RECOGNITION_USING_INNER_FINGER_DEEP_LEARNING_USING_NEURAL_NETWORK.pdf