Biometrics and Security: Implementation of Siamese Neural Networks for Embedded Systems

Heriberto Márcio da Silva Júnior
Department of Computer Systems
UFPB (Federal University of Paraiba)
João Pessoa - PB, Brazil
heribertomarcio@gmail.com

Carmonizia da Silva Freire
Mechanical Eng. Graduate Program
UFPB (Federal University of Paraiba)
João Pessoa - PB, Brazil
carmoniziafreire@gmail.com

Alisson Vasconcelos de Brito
Department of Computer Systems
UFPB (Federal University of Paraiba)
João Pessoa - PB, Brazil
alisson@ci.ufpb.br

Verônica Maria Lima Silva
Department of Computer Systems
UFPB (Federal University of Paraiba)
João Pessoa - PB, Brazil
veronica.lima@ci.ufpb.br

João Janduy Brasileiro Primo
Department of Research and Innovation
Vsoft Tecnologia
João Pessoa - PB, Brazil
jjanduy@vsoft.com.br

***Abstract*—This paper presents an application of the Siamese Neural Network in palm print biometrics using embedded systems. The use of Siamese neural networks is a promising solution to overcome challenges faced by the hardware limitations of a microcontroller, especially in restricted scenarios. This article presents the development of a system, with low computational cost, capable of performing biometric processing using the comparison between two images. The database used in the tests was the CASIA Multi-Spectral Palmprint, which includes images of palms captured in different light spectrums, aiming for a robust and comprehensive analysis of the unique characteristics of palm prints. The proposed approach aims to improve the efficiency and robustness of palm print recognition, offering an alternative with low computational cost, capable of being implemented in embedded systems. The developed network achieved a 99.99% rank-1 metric, 86.67% rank-5 metric, and an equal error rate of 1.5%.**

***Index Terms*—Biometrics, Palmprint, Siamese Neural Network, Embedded Systems.**

I. INTRODUCTION

These days, biometrics is an indispensable technique when it comes to security and authentication of people. Using unique physiological and behavioral characteristics, such as iris pattern, retina, fingerprint, palmprint, face, voice, gestures and signature, it is possible to recognize and authenticate people with a level of security and practicality incomparable to traditional authentication methods (token, card and password) [1]. Palmprint biometrics is based on the analysis of the unique pattern present on the inner surface of the human palm [2]. According to the literature, recognition based on palmprints presents a low rate of false correspondence and a high rate of accuracy for recognition due to the richness and stability of the characteristics present in the human palm [3]. Furthermore, the ability to combine a database of multiple light spectrums, such as near-infrared and visible, has improved the robustness of systems, making them less susceptible to environmental variations and adverse conditions [12].

The main characteristics of the palmprint that can be extracted and used for authentication and identification are three: main lines, wrinkles and creases [5]. Existing methods for representing palmprints depend on some factors (feature types, feature extraction type and feature expression form), thus, they can be divided into six categories: texture-based, line-based, subspace learning-based , based on correlation filters, based on local descriptors and based on convolutional neural networks (CNN) [6].

Biometrics techniques based on palmprints have evolved a lot, but there are still challenges to be overcome. For methods that use CNN networks, an obstacle is restricted or semi-restricted scenarios, making training the network with subsampling per person a difficult task. [7]. This obstacle becomes even greater in applications involving embedded systems, where hardware limitations make the use of the deep learning model a challenging task.

A promising solution to this dilemma is the use of Siamese neural networks. These networks are designed to learn the similarity between pairs of images, allowing for effective training even with smaller datasets. By focusing on learning discriminative features, Siamese networks can help mitigate the overfitting problem and improve model generalization, offering a more efficient and robust approach to palmprint recognition.

Once the network is trained, it makes it possible to carry out the biometric recognition process by storing just one image, or its vector representation, of the users in the device's internal memory, since the computational cost involved is limited to vector transformation and similarity calculation. between the image presented to the device and those present in its memory. Therefore, the integration of Siamese neural networks in embedded systems for palm print recognition provides an efficient and robust solution for biometric authentication. This combination allows the creation of advanced security systems that are accurate and practical for use in a variety of applications.

II. Related Work

As Siamese network models are a work in progress, obtaining an evaluation metric for a Siamese network applied to palm biometrics becomes a challenging task. Therefore, we chose to choose works related to palm biometrics, regardless of the technique used, in order to compare the results obtained with what exists in the literature. The publications chosen are from works that developed a new model of biometrics, some articles did not name the approaches used, so the reference of the article was repeated for later consultation. Thus, the metrics presented by them were unified and then the results of this work were compared. The comparison for the EER Equal Error Rate can be seen in T.I, and for Rank-1 in Fig.II.

COMPARATIVE ERR			
Ref.	DATASET	APPROACHES	ERR (%)
[14] Pereira, et al, 2018	CASIA	HOG (BOTH HANDS)	0
[20] Srinivas, 2009	POLYU	SURF	0.0155
[17] Khan 2011	CASIA	Khan, Mian e Hu (2011)	0.3
[13] Khan 2014	CASIA	ContCode-ATM	0.433
[15] Hao et al 2008	CASIA	Hao et al. (2008)	0.5
[14] Pereira, et al, 2018	CASIA	HOG (RIGHT HAND)	0.544
[13] Khan 2014	CASIA	OrdCode	0.5667
[12] Jaswal, 2019	CASIA	PALMCODE	0.6
[13] Khan 2014	CASIA	ContCode-STM	0.6279
[13] Khan 2014	CASIA	CompCode	0.8667
[14] Pereira, et al, 2018	CASIA	HOG (LEFT HAND)	1
[12] Jaswal, 2019	GPDS-CLI	palm print system.	1.022
-	CASIA	This Paper	1.5
[13] Khan 2014	CASIA	DoGCode	1.9667
[13] Khan 2014	CASIA	Wavelet fusion with ACO	3.125
[16] Kisku, 2010	CASIA	Kisku et al (2010)	3.125
[12] Jaswal, 2019	GPDS-CLI	Palm print (without using DIC, CS-LBP)	4.22
[12] Jaswal, 2019	BOSPHORUS	hand shape system	7.55
[18] Canestraro 2015	CASIA	Canestraro(2015)	9
[12] Jaswal, 2019	BOSPHORUS	hand geometry system	10.62
[19] Gupta 2016	SELF	GUPTA ET AL	11.93

TABLE I
Work developed and its results for EER.

COMPARATIVE RANK-1			
Ref.	DATASET	APPROACHES	Rank-1 (%)
[13] Khan 2014	PolyU	ContCode-ATM	100
-	CASIA	this paper	99.99
[13] Khan 2014	PolyU	DoGCode	99.97
[13] Khan 2014	PolyU	CompCode	99.97
[13] Khan 2014	CASIA	ContCode-ATM	99.88
[13] Khan 2014	CASIA	CompCode	99.52
[13] Khan 2014	PolyU	OrdCode	99.03
[13] Khan 2014	CASIA	OrdCode	99.02
[13] Khan 2014	CASIA	DoGCode	95.08

TABLE II
Work developed and its results for Rank-1.

As can be seen, the model developed in this work obtained results close to the best works developed, highlighting the rank-1 metrics in which the difference to first place was 0.01

III. Methodology

A. Database definition

The database chosen for the development of this work was the *CASIA Multi-Spectral Palmprint*, which includes images of palmprints captured in different light spectrums, including the visible spectrum and non-visible spectrums, such as near infrared [11].

The database contains *72* images for each individual, *36* of the left hand and *36* of the right hand, captured in *6* different light spectrums. Includes palmprints from *100* individuals, spanning different ages, genders and ethnicities. This ensures that algorithms developed using this database can generalize well to diverse populations.

B. Preprocessing

To optimize training time, the size of the images was reduced to *25%* of the original value, significantly reducing the number of network parameters.

As the database originally does not distinguish between right and left hands, the images were manually divided, creating *200* classes in total, *100* users with left and right hands. Then, the images were separated into training and testing data. The training data consists of *2* images from each of the *6* spectrum variations, separating the data into *12* images for testing and *24* images for training for each class.

From the separated data, image pairs were formed for the Siamese network. Each pair contains an anchor image and a validation image that can be positive, belonging to the same class, or negative, belonging to a different class, as shown in Fig.1.

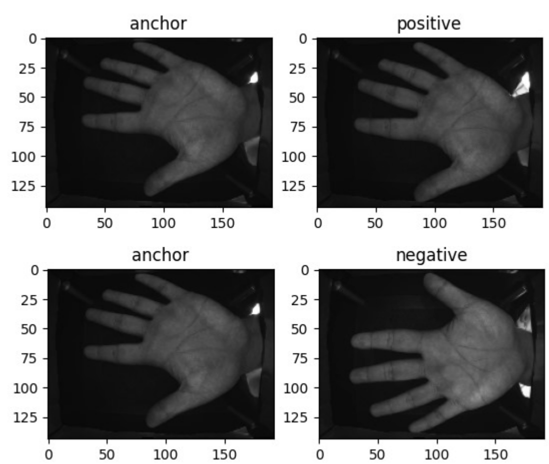

Fig. 1. Positive and negative pairs [11].

In the construction of positive pairs, all images from the same class were combined without repetition, so a class with *24* images has a total of *276* positive combinations. Considering the *200* classes, we have *55,200* pairs in total.

For negative pairs, a combination without repetition of all images from one class was performed against all images from other classes, thus for each class in the training model *576* pairs times *19,900* were produced, the *2* to *2* combination of all classes . Considering the *200* classes we have *576 x 19,900 = 11,462,400* negative pairs. Adding the negative and positive pairs we will have a total of *11,517,600* pairs of images for training.

Due to the large number of pairs generated, we chose to reduce the number of classes in order to reduce the time spent in the training process *12* classes were used, which created a number of *3,312* positive pairs and *76,032* negative pairs, adding up to a total of *79,344* pairs for the training base.

In the test base, following the same procedure described previously, we have a total of *792* pairs of positive images and *19,008* negative pairs, adding up to a total of *19,800* pairs for the test base.

Even so, the amount of data remained high to be placed in memory, therefore, the image pairs were grouped into batches of 64 units to facilitate processing and storage in memory.

C. Model

Siamese neural networks are an architecture designed to learn to measure similarity between pairs of inputs, using identical subnetworks (Hidden Layer) that share the same weights. To calculate the distance update, instead of using the traditional contrastive loss, it is proposed to train them with the *Binary Cross-Entropy* (BCE) loss function, due to the fact that the objective is to perform a binary classification, determining whether two images are similar (class 1) or dissimilar (class 0), the structure of the Siamese network can be observed in Fig.2.

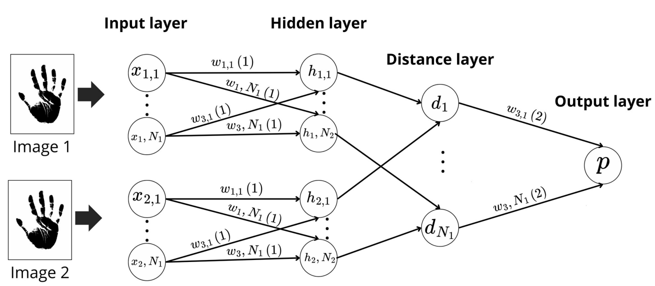

Fig. 2. Structure of a Siamese network.

HIDDEN LAYER				
	LAYER TYPE	NUMBER OF NEURONS	SIZE	ACTIVATION FUNCTION
	Input	144x192x1		
1° BLOCK	Conv2D	64	10x10	Relu
	MaxPooling2D	64	2x2	
2° BLOCK	Conv2D	64	7x7	Relu
	MaxPooling2D	64	2x2	
3° BLOCK	Conv2D	64	4x4	Relu
	MaxPooling2D	64	2x2	
4° BLOCK	Conv2D	256	4x4	Relu
	Flatten			
	Dense	256		Sigmoid

TABLE III
SUBNET ARCHITECTURE DEVELOPED FOR THIS PAPER.

The two identical subnets that process two inputs independently and share the same parameters were designed according to the structure presented in table.III. Each subnet generates a vector representation (embedding) with *256* dimensions from the input image.

The vector representations generated by the subnetworks are combined using a similarity layer, which in this work consists of the absolute difference between the embeddings, given by Eq. (1). Where, xi the image vector *1* and xii the vector of image *2*.

$$d = |x_i - x_i i| \tag{1}$$

The output of the similarity layer is passed through a final dense layer that produces a probability $\hat{y}$ that indicates the similarity between the inputs. This probability is interpreted as the prediction of how similar (1) or dissimilar (0) the two inputs are.

The Binary Cross-Entropy (BCE) loss function is used to measure the error between the prediction $\hat{y}$ and the true label y. The BCE is defined according to Eq.(2). Where, n is the number of samples, yi is the real value (0 or 1) of the ith sample and $\hat{y}_i$ is the probability predicted by the model for the ith sample.

$$BCE = -\frac{1}{n}\sum_{i=1}^{n}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)] \tag{2}$$

D. Training

In training neural networks, the loss function (or loss function) plays a fundamental role in quantifying the discrepancy between the model's predictions and the actual values of the training data. This function provides a metric that the optimization algorithm can use to adjust the network weights, minimizing error during the learning process. A widely used approach to optimization is the backpropagation algorithm, which benefits from using batches of data. In this section, we will discuss how the loss function operates in conjunction with the backpropagation algorithm using batches of size *64*.

The loss function is a measure that evaluates the performance of a model. For a training data set (x_i, y_i), where x_i are the inputs and y_i are the desired outputs, the loss function L is defined to quantify the error between model predictions $\hat{y}_i$ and the real values y_i. The function to calculate the distance between vectors is given by the absolute difference calculated by the similarity layer.

Backpropagation is an efficient algorithm for calculating the gradients necessary to adjust the neural network weights, minimizing the loss function. It operates in three main steps:

- **Forward Pass:** Inputs xi are passed through the network to obtain predictions $\hat{y}_i$;
- **Loss Calculation:** The loss function is calculated using the predictions $\hat{y}_i$ and the actual values yi;

- **Backward Pass:** The gradients of the loss function with respect to the network weights are computed using the chain rule, allowing the weights to be updated via optimization (e.g. gradient descent).

Training a neural network on the entire dataset can be computationally expensive. To mitigate this cost, training can be performed on batches of data. A batch is a subset of the training data set. Using batches, the training process is carried out in stages, where each step involves updating the weights based on a batch of data.

The use of batches offers a balance between the accuracy of weight updates and computational efficiency. Smaller batches can lead to noisier and more volatile updates, while larger batches can be computationally expensive and less memory efficient.

Due to computational power limitations, we use batches of size *64* units. This means that in each training iteration, the backpropagation algorithm is applied to a batch of *64* data samples.

IV. Results

To evaluate the model's performance, it performed the similarity calculation using all pairs of each class individually, thus making it possible to separately measure the performance of the neural network. Each class has *66* positive pairs and *1584* negative pairs, adding up to *1650* pairs per class for a total of *19,800* test pairs.

Rank-1 and Rank-5 metrics are commonly used to evaluate the performance of recognition and retrieval systems, such as facial recognition, biometric verification, and information retrieval. They measure the system's accuracy in correctly identifying an entry from a database.

The Rank-1 metric indicates the proportion of times that the correct entry is the first option returned by the system. The Rank-5 metric indicates the proportion of times that the correct entry is among the first five options returned by the system.

Equal error rate, also known as false positive rate, is an important concept in statistics and machine learning. It represents the proportion of incorrect results, false acceptance (FAR) and false rejection (FRR) in relation to the total positive predictions made by a model or test. FAR is the proportion of non-legitimate recognition or authentication attempts that are accepted by the system. FRR is the proportion of legitimate recognition or authentication attempts that are rejected by the system. The results of the metrics can be seen in Table IV and the confusion matrix in Table V.

V. Conclusions

In this work, a low computational cost biometric system was proposed, created from a Siamese neural network model capable of calculating the similarity between two images. The model's performance was evaluated using some common metrics for biometric systems: *Equal Error Rate* (EER) and accuracy for *Rank-1* and *Rank-5*.

METRICS	
FAR	0,97%
FRR	1,97%
ERR	1,47%
RANK 1	99,99%
RANK 5	86,67%

TABLE IV
THE METRICS OBTAINED IN THIS PAPER.

CONFUSION MATRIX			
	ACTUAL		
PREDICT		POS	NEG
	POS	600	390
	NEG	192	18618

TABLE V
CONFUSION MATRIX OF THE MODEL.

The EER rates were *1.5%* for false rejection and *0.97%* for false acceptance, demonstrating that the model is highly effective in protecting data.

The recognition accuracy was *99.99%* using the rank-1 metric and *86.67%* in rank-5. These results are indicators that the model has a robust recognition capacity even when the task becomes more challenging, such as in situations of large databases with many identities.

The result show that the developed Siamese neural network model is highly effective for the biometric recognition process. The low error rate and high accuracy in rank-1 and rank-5 metrics demonstrate that the system can be used with great confidence in practical applications. Furthermore, the methodology can be adapted to different types of biometric data, expanding its potential for use in different contexts.

This work contributes to the area of biometric recognition, showing that Siamese neural networks are a viable and efficient approach to measuring similarity between images and identifying individuals with high accuracy. The success of the model suggests future applications and improvements, such as adaptation to larger and more varied databases, as well as integration with other types of biometric systems to further increase security and reliability.

References

[1] Zhang, D. et al. **Online Palmprint Identification.** IEEE transactions on pattern analysis and machine intelligence 25.9 (2003): 1041–1050.

[2] Zhenan Sun et al. **Ordinal Palmprint Represention for Personal Identification [Represention Read Representation].** 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) 1 (2005): 279–284 Vol. 1.

[3] Zhao, Shuping, and Bob Zhang. **Deep Discriminative Representation for Generic Palmprint Recognition.** Pattern recognition 98 (2020): 107071.

[4] Munir, Rumaisah, and Rizwan Ahmed Khan. **An Extensive Review on Spectral Imaging in Biometric Systems: Challenges and Advancements.** Journal of visual communication and image representation 65 (2019): 102660.

[5] Guo, Zhenhua et al. **Palmprint Verification Using Binary Orientation Co-Occurrence Vector.** Pattern recognition letters 30.13 (2009): 1219–1227.

[6] Zhao, Shuping, Lunke Fei, and Jie Wen. **Multiview-Learning-Based Generic Palmprint Recognition: A Literature Review.** Mathematics (Basel) 11.5 (2023): 1261.

[7] Zhao, Shuping, and Bob Zhang. **Joint Constrained Least-Square Regression With Deep Convolutional Feature for Palmprint Recognition.** IEEE transactions on systems, man, and cybernetics. Systems 52.1 (2022): 511–522.

[8] Zhao, Shuping, and Bob Zhang. **Learning Complete and Discriminative Direction Pattern for Robust Palmprint Recognition.** IEEE transactions on image processing 30 (2021): 1001–1014.

[9] Zhao, Shuping et al. **Double-Cohesion Learning Based Multiview and Discriminant Palmprint Recognition.** Information fusion 83–84 (2022): 96–109.

[10] Koch, Zemel and Ruslan Salakhutdinov. **Siamese Neural Networks for One-Shot Image Recognition.** Proceedings of the 32nd International Conference on Machine Learning 37 (2015).

[11] Tan, T. **CASIA Palmprint Database.** Em Proceedings of the International Conference on Biometrics (2006): pp. 123-135.

[12] Jaswal, G., Kaul, A., Nath, R. **Multimodal Biometric Authentication System Using Hand Shape, Palm Print, and Hand Geometry.** In: Verma, N., Ghosh, A. (eds) Computational Intelligence (2019): Theories, Applications and Future Directions - Volume II. Advances in Intelligent Systems and Computing, vol 799. Springer, Singapore.

[13] Khan, Zohaib et al. **Multispectral Palmprint Encoding and Recognition.** arXiv.org (2014): arXiv.org, 2014-02.

[14] P.M. da S. Pereira, et al. **Biometric verification system.** Paraná Microelectronics Seminar. Curitiba Brazil. October (2018).

[15] Ying Hao et al. **Multispectral Palm Image Fusion for Accurate Contact-Free Palmprint Recognition.** 2008 15th IEEE International Conference on Image Processing (2008): 281–284.

[16] Kisku, Dakshina Ranjan et al. **Multispectral Palm Image Fusion for Person Authentication Using Ant Colony Optimization.** 2010 International Workshop on Emerging Techniques and Challenges for Hand-Based Biometrics (2010): 1–7.

[17] Khan, Z., A. Mian, and Yiqun Hu. **Contour Code: Robust and Efficient Multispectral Palmprint Encoding for Human Recognition.** 2011 International Conference on Computer Vision (2011): 1935–1942.

[18] Canestraro, A. **Multimodal biometric system for identity verification based on hand geometry, palm print and palm veins.** XXIV Brazilian Congress of Biomedical Engineering, v. 1, p. 2556–2559, 2015.

[19] Gupta, Puneet, Saurabh Srivastava, and Phalguni Gupta. **An Accurate Infrared Hand Geometry and Vein Pattern Based Authentication System.** Knowledge-based systems 103 (2016): 143–155.

[20] Srinivas, B., Gupta, P. **Palm print based verification system using SURF features.** Contemp. Comput. 250–262 (2009)