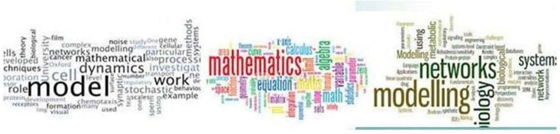


Journal of Advances in Mathematical & Computational Sciences
An International Pan-African Multidisciplinary Journal of the SMART Research Group
International Centre for IT & Development (ICITD) USA
© Creative Research Publishers
Available online at https://www.isteams.net/ mathematics-computationaljournal.info
CrossREF Member Listing - https://www.crossref.org/06members/50go-live.html

Development of an Optimized Lightweight CNN Model Using Chicken Swarm Optimization for Finger-Vein Biometric Recognition

[1]Adedara O. G., [2]Oke, A.O., [3]Oyinloye O.A., [4]Makinde, O., [5]Ipeayeda, F.W., & [6]Oladepo C.O.
[1]Department of Computer Science,Federal Polytechnic, Ado Ekiti, Ekiti State, Nigeria
[2]Dept of Computer Science & Engineering, Ladoke Akintola University of Tech. Ogbomoso, Nigeria
[3]Department of Computer Science, University of Ilesa, Ilesa, Nigeria
[4]Department of Data Sciences, Ajayi Crowther University, Oyo, Nigeria
[5]Department of Cyber Security, Ajayi Crowther University, Oyo, Nigeria
[6]Dept of Elect/Elect Engineering, Ladoke Akintola University of Tech. Ogbomoso, Nigeria
E-mails: adedara_og@fedpolyado.edu.ng; aooke@lautech.edu.ng; olufunke_oyinloye@unilesa.edu.ng
pe.makinde@acu.edu.ng; ipeayeda@acu.edu.ng; caleboladepo@gmail.com

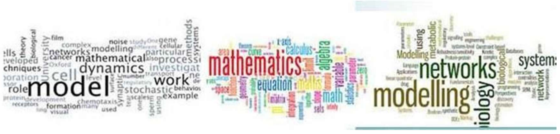
ABSTRACT

Finger-vein recognition has gained prominence in biometric security due to its internal vascular patterns, which offer high uniqueness and resistance to forgery. Despite progress, optimizing performance for the intricate, high-dimensional features of finger vein patterns remains challenging. This study introduces a finger vein verification and identification system that combines lightweight convolutional neural networks (CNNs) with Chicken Swarm Optimization (CSO) for hyperparameter tuning, referred to as the CSO-CNN technique. Evaluated on a dataset of 400 images (150 genuine, 250 impostor), images underwent preprocessing for enhancement and region of interest extraction. The CSO-CNN optimized hyperparameters to derive discriminative features, classified using a SoftMax layer. At an optimal threshold of 0.8, CSO-CNN yielded a False Acceptance Rate (FAR) of 7.20%, False Rejection Rate (FRR) of 14.00%, accuracy of 90.25%, and recognition time of 125.52 seconds. Comparatively, the standard CNN achieved FAR of 8.80%, FRR of 17.33%, accuracy of 88.00%, and recognition time of 175.53 seconds. The Equal Error Rate (EER) further validates CSO-CNN's superiority at 12.00% versus 15.45% for CNN. Confusion matrix analysis indicated CSO-CNN correctly identified 129 genuine and rejected 232 impostors, with fewer misclassifications. Paired t-tests confirmed significant enhancements, with p-values of 0.005 and 0.000. CSO's integration facilitated efficient hyperparameter tuning, boosting feature extraction and reducing recognition time. The finger vein's internal, stable vascular patterns contributed to robust accuracy. These outcomes establish CSO-CNN as an effective solution for finger-vein biometrics, enhancing security and efficiency for real-world applications.

Keywords: Finger vein recognition, vascular biometrics, lightweight convolutional neural networks, hyperparameter optimization, Chicken Swarm Optimization, deep learning

---

Adedara O. G., Oke, A.O., Oyinloye O.A., Makinde, O., Ipeayeda, F.W., & Oladepo C.O. (2026): Development of an Optimized Lightweight CNN Model Using Chicken Swarm Optimization for Finger-Vein Biometric Recognition. Journal of Advances in Mathematical & Computational Science. Vol. 14, No. 2. Pp 9-28. Available online at www.isteams.net/mathematics-computationaljournal.
dx.doi.org/10.22624/AIMS/MATHS/V14N2P2



1. INTRODUCTION

Advancements in vascular biometrics have positioned finger-vein recognition as a robust alternative for personal authentication, benefiting from its non-invasive capture and resistance to external influences (Qin and El-Yacoubi, 2017; Zhang and Wang, 2021). Unlike surface-based traits, finger-vein patterns rely on subcutaneous blood vessel structures illuminated by near-infrared (NIR) light, offering high uniqueness and stability across individuals (Yang et al., 2012; Yang et al., 2018). This modality has gained traction in applications requiring secure, contactless verification, such as financial services and access control, where traditional methods fall short against spoofing (Veldhuis et al., 2019; Uhl, 2020). The evolution of deep learning, particularly convolutional neural networks (CNNs), has revolutionized biometric systems by automating feature extraction from complex vascular images, often surpassing manual engineering approaches (Krizhevsky et al., 2012; LeCun et al., 1998).

In finger-vein contexts, CNNs have demonstrated resilience to variations in pose, lighting, and tissue conditions, enabling more reliable matching (Huang et al., 2017; Das et al., 2018). However, deploying these models in resource-limited environments, like mobile devices or embedded systems, necessitates lightweight architectures that minimize computational overhead without compromising discriminative power (Zhao et al., 2020; Kuzu et al., 2020). Despite these progressions, CNN efficacy hinges on hyperparameters such as learning rates, layer depths, and regularization terms, which govern training dynamics and model generalization (LeCun et al., 2012; Srivastava et al., 2014). Conventional tuning methods, like grid search, are computationally intensive and may overlook optimal configurations in expansive search spaces (Bergstra and Bengio, 2012). This gap has prompted the adoption of metaheuristic optimizers to streamline hyperparameter selection, enhancing convergence and performance in image-related tasks (Qolomany et al., 2017; Wang et al., 2019).

Chicken Swarm Optimization (CSO), a bio-inspired algorithm mimicking flock hierarchies and foraging behaviors, excels in balancing global exploration and local exploitation for multifaceted optimization challenges (Meng et al., 2014; Deb et al., 2019). Its variants have proven effective in parameter tuning across domains, addressing issues like premature convergence in high-dimensional problems (Lin et al., 2018; Amador-Angulo et al., 2021). For finger-vein biometrics, CSO can mitigate dataset limitations and variability, such as those from NIR capture artifacts or inter-finger differences (Kirchgasser et al., 2020; He et al., 2017).

Prior studies have explored metaheuristics for CNN optimization in biometrics, including PSO and GA, yielding gains in accuracy and efficiency (Nikbakht et al., 2021; Oguntoye et al., 2019). Yet, the synergy of CSO with lightweight CNNs specifically for finger-vein recognition is underexamined, presenting an opportunity to advance deployable systems (Meng et al., 2014; Wang et al., 2019b). This integration could tackle persistent challenges, including computational demands and robustness to environmental factors, while supporting privacy-focused designs (Hartung and Busch, 2009; Kirchgasser et al., 2019). This paper examines the hyperparameter optimization of lightweight CNNs via Chicken Swarm Optimization to elevate finger-vein biometric outcomes. Drawing on CNN's prowess in vascular feature representation (Huang et al., 2017; Das et al., 2018) and we present a CSO-guided framework that optimizes accuracy, compactness, and stability.

Evaluated on finger-vein datasets, it is benchmarked against literature methods, considering deployment factors like latency and template security (Ahmad et al., 2019; Veldhuis et al., 2019).
The proposed technique has the following contributions:

1) The introduction and application of a cultural-based Chicken Swarm Optimization (CSO) technique tailored for hyperparameter optimization of lightweight CNN architectures in finger-vein biometric recognition.
2) Demonstration of the effectiveness of the cultural-based CSO approach in enhancing finger-vein biometric access control by improving recognition accuracy and reducing error rates.
3) Comprehensive validation of the proposed cultural-based CSO method against the standard CSO technique specifically for finger-vein biometric recognition.
4) Performance evaluation of the proposed approach using rigorous metrics including accuracy, precision, False Acceptance Rate (FAR), False Rejection Rate (FRR), Equal Error Rate (EER), and computational time, with comparison to state-of-the-art hyperparameter tuning and biometric recognition techniques.

The structure of this research is as follows: Section 2 discusses a review of related works. Section 3 provides the detailed research methodology utilized. Section 4 presents the results and comprehensive discussion, while Section 5 concludes the study and discusses potential future research directions to further enhance finger-vein biometric recognition systems.

2. LITERATURE REVIEW

Finger-vein recognition stands out among vascular biometrics for its reliance on internal blood vessel patterns captured under near-infrared (NIR) light, which inherently resists spoofing and offers stable, unique features across individuals (Yang et al., 2012; Uhl, 2020). Early explorations established the modality's feasibility through simple NIR imaging setups and basic processing, underscoring its potential for low-cost, contactless authentication in security applications (Hashimoto, 2006; Khan et al., 2009; Veldhuis et al., 2019). Building on this foundation, researchers refined core stages such as image enhancement, ROI extraction, and alignment correction to handle real-world variations like finger pose shifts, lighting inconsistencies, or tissue thickness differences (Podgantwar and Raut, 2013; Matsuda et al., 2016; Kirchgasser et al., 2020).

The shift toward convolutional neural networks (CNNs) marked a pivotal advancement, as these models automatically learn hierarchical features from raw vascular images, often outperforming traditional handcrafted methods in capturing subtle vein textures and overcoming noise or low contrast (Krizhevsky et al., 2012; LeCun et al., 2012). In finger-vein contexts, CNN architectures have excelled at addressing intra-class variability arising from factors such as blood flow changes or environmental conditions while delivering robust verification and identification performance (Huang et al., 2017; Das et al., 2018; Boucherit et al., 2020). More recent efforts have incorporated advanced elements like multi-scale fusion, attention mechanisms, and residual connections to better model both local vein details and global structural relationships, pushing accuracy higher on standard datasets (Assim and Alkababji, 2021; Kuzu et al., 2020).

Practical constraints, particularly for embedded or mobile deployment, have driven the development of lightweight CNN variants that maintain strong discriminative power with far fewer parameters and lower computational demands (Zhao et al., 2020; Kuzu et al., 2020). Designs featuring efficient blocks have demonstrated competitive results often exceeding 98 - 99% accuracy while enabling real-time inference on resource-limited hardware. Such compact models address key deployment challenges, including energy efficiency, latency, and memory footprint, making finger-vein systems more viable for everyday use. CNN effectiveness, however, depends critically on hyperparameter configuration, encompassing learning rates, filter sizes, dropout levels, regularization strengths, and network depth which governs training stability, generalization, and avoidance of overfitting (LeCun et al., 2012; Srivastava et al., 2014).

Manual or grid-search tuning becomes prohibitively expensive in these high-dimensional spaces, motivating the use of metaheuristic algorithms to automate and intelligently navigate the search process (Bergstra and Bengio, 2012; Qolomany et al., 2017). Chicken Swarm Optimization (CSO), modeled on chicken flock social dynamics and foraging hierarchies, stands out for its effective exploration–exploitation balance in multimodal, constrained problems (Meng et al., 2014; Deb et al., 2019; Lin et al., 2018; Qu et al., 2017). Variants and improvements of CSO have shown particular promise in parameter tuning for classification and deep models, helping mitigate premature convergence and enhance solution quality (Bharanidharan and Rajaguru, 2020; Wang et al., 2019b; Wu et al., 2016). CSO and related swarm-inspired techniques have already contributed to biometric optimization, including classifier enhancement and network fine-tuning, where they improve accuracy while controlling complexity (Oguntoye et al., 2019; Nikbakht et al., 2021).

In the context of vascular biometrics, metaheuristic-optimized CNNs have shown promising results in improving recognition rates and hybrid swarm-deep approaches have enhanced feature discriminability and reduced error rates in related tasks (Gao et al., 2020; Li et al., 2019; Assim and Alkababji, 2021). For finger vein systems, optimization addresses challenges like dataset variability, acquisition noise, and the need for lightweight deployment (Kirchgasser et al., 2019; Nikbakht et al., 2021). Swarm intelligence algorithms, inspired by natural collective behaviors, have gained prominence for hyperparameter optimization in deep learning due to their global search capabilities and adaptability to non-convex problems. Chicken Swarm Optimization (CSO), modeled after chicken flock hierarchies and foraging dynamics, balances exploration through diverse roles (roosters, hens, chicks) and demonstrates fast convergence on benchmark tasks (Meng et al., 2014; Deb et al., 2019; Lin et al., 2018). Variants addressing premature convergence, such as those incorporating mutation operators or chaotic mappings, further strengthen CSO's performance in high-dimensional optimization (Wu et al., 2016; Qu et al., 2017; Bharanidharan and Rajaguru, 2020).

Overall, the progression from traditional feature-based methods to advanced deep learning, combined with intelligent optimization strategies, continues to elevate finger vein recognition toward practical, high-security applications. While CNNs provide powerful automatic feature learning, metaheuristics like CSO offer essential support for achieving optimal configurations efficiently (Serizawa and Fujita, 2020). Hybrid approaches combining metaheuristics with deep architectures hold clear potential for vascular biometrics, yet the application of CSO specifically to hyperparameter optimization in lightweight CNNs for finger-vein recognition remains an underexplored area.

This gap highlights an opportunity to harness CSO's strengths robust global search and adaptive diversity mechanisms to develop more efficient, accurate, and deployable finger-vein systems that overcome computational limitations and variability in real-world data (Gao et al., 2020; Li et al., 2019).

3. MATERIALS AND METHODS

The implementation stages for the finger-vein biometric recognition system using the enhanced CSO-based CNN technique are as follows:

1) Finger-vein image acquisition using a near-infrared sensor.
2) Preprocessing of finger-vein images.
3) Feature extraction using the enhanced CSO-CNN technique.
4) Hyperparameter normalization.
5) Classification into identity classes using SoftMax.
6) Identification of the subject as genuine user or impostor.

This study employs a single-modality finger-vein biometric system to provide secure and reliable authentication. Figure 1 illustrates the schematic layout of the proposed finger-vein recognition system.

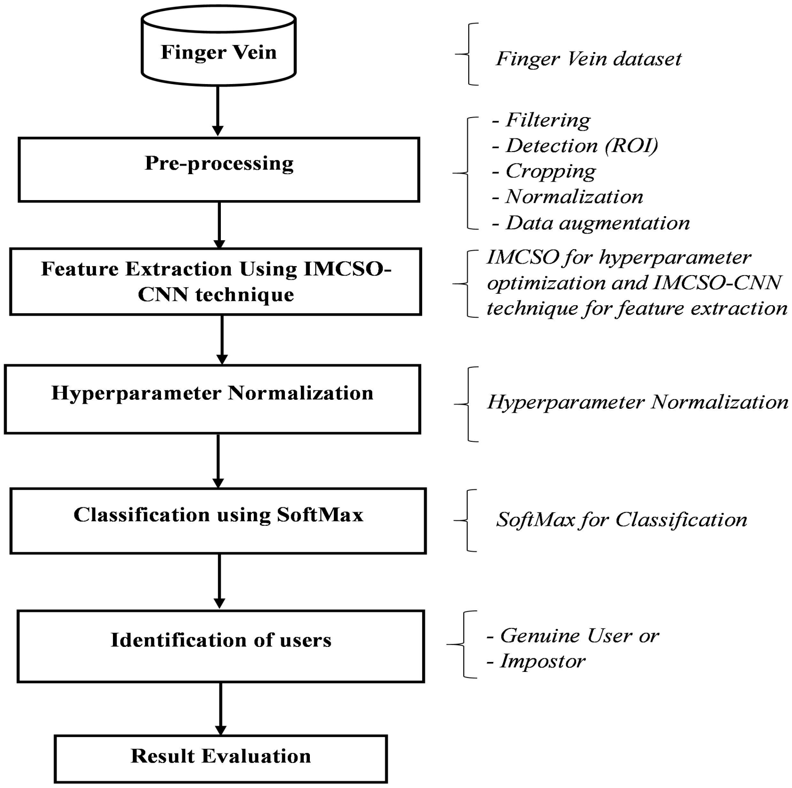


Figure 1: Block diagram of proposed finger-vein biometric recognition system

3.1 Acquisition of Finger Vein Biometric Trait

This phase centers on the acquisition of the finger-vein biometric dataset, which serves as the cornerstone of the proposed recognition system. A total of 1,000 finger-vein images were collected from 200 subjects at Federal Polytechnic Ado-Ekiti. The images were captured using an M2SYSTH Charge Coupled Device (CCD) camera sensitive to near-infrared (NIR) light. For each subject, images of the index and middle fingers (both left and right hands) were acquired five times each, yielding a robust bi-instance dataset. The images were taken in multiple orientations to replicate real-world conditions and to enhance system resilience across varying finger positions. Of the complete dataset, 600 images were designated for model training, while the remaining 400 were set aside for testing and validation, maintaining a balanced and representative distribution. Sample finger-vein images are presented in Figure 2.

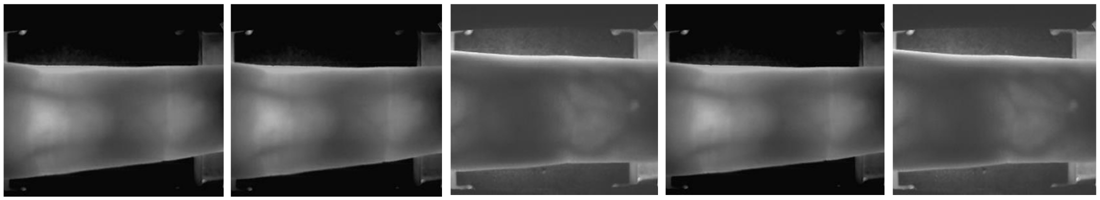


Figure 2: Some acquired finger-vein images

3.2 Pre-processing of Finger-Vein Images

Preprocessing of finger-vein images constitutes a vital stage in elevating image quality and readying the data for accurate feature extraction and recognition. This stage encompasses multiple operations, including image cropping, resizing, contrast enhancement, normalization, and data augmentation, all aimed at strengthening the distinctiveness of the vascular patterns. The central element of preprocessing is the identification of the Region of Interest (ROI), which isolates the portion of the finger-vein image that contains the most informative vein structures. Performing feature extraction on this ROI markedly lowers computational demands and boosts recognition precision by mitigating distortions arising from finger movements or rotations during acquisition.

The ROI is located through the following steps:

1. Image binarization to distinguish vein patterns from the surrounding background.
2. Boundary detection to identify gaps and edges in the finger-vein image.
3. Tangent line calculation by linking two prominent gap points (x_1, y_1) and (x_2, y_2), thereby establishing the Y-axis of the finger coordinate system.
4. Defining the X-axis as a perpendicular line to the tangent, intersecting at the midpoint between the two gap points.
5. Specifying a fixed-size square ROI, placed at a uniform distance from the finger-coordinate origin according to the defined axes.
6. Extracting the sub-image contained within this ROI for subsequent analysis.

To further improve image quality prior to feature extraction, histogram equalization is employed. This method enhances overall contrast by redistributing and expanding the gray-level histogram of the image, thereby rendering vein patterns more prominent.

The mathematical formulation is expressed in Eq. (1):

$$f(x, y) = \frac{I_{min}-I_{max}}{h_{max}-h_{min}} \left(h(x, y) - h_{min}\right) + I_{max} \tag{1}$$

where h (x, y) denotes the original histogram value at pixel (x, y), f (x, y) represents the enhanced histogram value, h_{min} and h_{max} are the minimum and maximum gray levels in the input image, while I_{min} and I_{max} define the target output intensity range.

To align with the requirements of the lightweight CNN architecture, the extracted ROI images are resized to 128 × 256 pixels. Moreover, several data augmentation strategies, including rotation, shearing, zooming, and horizontal/vertical shifting are applied to artificially increase the training dataset size and mitigate the likelihood of overfitting. This thorough preprocessing sequence guarantees that the finger-vein images are optimally conditioned for effective feature learning and subsequent classification. Examples of preprocessed images are illustrated in Figure 3.

Collected Finger-vein Cropped Image Normalized Image

Figure 3: Some Preprocessed Images

3.3 The Lightweight CNN architecture

The lightweight convolutional neural network is intentionally designed with a reduced number of convolutional and fully connected layers to efficiently capture the essential features of finger-vein patterns in minimal processing time. The resulting low-dimensional feature representations effectively encapsulate the core characteristics of the original finger-vein images, thereby facilitating faster and more accurate classification. In the proposed architecture, the hidden layers consist of convolutional layers, pooling layers, and fully connected layers. The second layer primarily emphasizes low-level details such as edges and corners. Layer 3 captures more intricate information, including complex invariances and similar texture patterns. Layer 4 highlights pronounced variations and becomes increasingly class-specific. Layer 5 encodes higher-level abstractions, accounting for notable pose and orientation variations inherent in finger-vein images. A comprehensive description of the network structure is provided in Table 1. The input size for this CNN architecture is fixed at 128 × 256 pixels for finger-vein images.

Table 1: The Lightweight CNN architecture.

Type	Filter Size/Stride	Output Size
Conv1	3×3/1	32×60×128
Pool1	2×2/2	32×30×64
Conv2	3×3/1	64×30×64
Pool2	2×2/2	64×15×32
Conv3	3×3/1	128×15×32
Pool3	2×2/2	128×7×16
FC1	-	256
FC2	-	200

3.4 Improved CSO Based on Mutation Operator

The conventional Chicken Swarm Optimization (CSO) algorithm is prone to becoming trapped in local optima when tackling high-dimensional optimization problems, largely because of diminishing population diversity over iterations. To overcome this limitation, an enhanced mutation mechanism has been incorporated into the position update process. Specifically, after chicks update their positions, a mutation operation is immediately applied to promote greater diversity and facilitate escape from local minima. The mutation produces a perturbed solution x_s^* as described in Eq. (2) using bounds and a mutation factor δ_s as shown in Eq. (3). Instead of purely random parameters, the mutation uses chaotic sequences

The chaotic map used in this study is gauss map, and in the ith iteration $u_s = u_i$, then in the $(i + 1)th$ iteration is given as Eq. (4). This map generates chaotic sequences in (0, 1).

$$x_s^* = x_s + \left(x_s^u - x_s^l\right) \times \delta_s \qquad (2)$$

where x_s^u and x_s^l are the upper and lower bounds of x_s.

$$\delta_s = \begin{cases} (2u_s)^{1/(\eta_m+1)} - 1 & if\ u_s \leq 0.5, \\ 1 - (2 \times (1 - u_s))^{1/(\eta_m+1)}, & others \end{cases} \qquad (3)$$

where u_s is a random number ranging from 0 to 1. η_m is the distribution index for the mutation operator. The phase for mutation is that $u_s$is calculated by chaotic maps in iterations.

$$u_s = u_{i+1} = \begin{cases} 0, & u_i = 0, \\ \frac{1}{u_i} \quad mod(1), & otherwise \end{cases} \qquad (4)$$

The mutation operator in the improved CSO algorithm helps chickens escape local optima by updating their positions, increasing the likelihood that new offspring have better fitness than existing roosters. Through the hierarchical order mechanism, roosters trapped in local optima are replaced by fitter hens, enhancing both search efficiency and speed.

3.5 Improved CSO for Hyperparameter Optimization

The improved CSO technique explained in section 3.4 will be used to optimize the hyperparameter of the CNN technique. The improved CSO technique based on mutation operator is described in detail.

Step 1: Initialization of the chicken swarm x_i and the related parameter $(N, Ir, G, R_N, H_N, C_N, M_N)$ where N is the total number of chickens, Ir is the maximum iterations, G is the frequency of change of hierarchical order in the swarm, R_N is the numbers of roosters, H_N is the numbers of hens, C_N is the numbers of chicks, and M_N is the numbers of mothers respectively.

Step 2: Evaluate the fitness values of the chicken swarm x_i and initialize the personal best position p_{best} and the global best position g_{best}. Set *t* = 1.

Step 3: If *t* mod *G* is 1, sort the fitness values of the individuals within the chicken swarm, and build the hierarchal order of the chicken swarm; divide the whole chicken swarm into several subgroups and ensure the relationship between the hens and the chicks.

Step 4: Update the positions of roosters, hens, and chicks and recalculate the fitness values, respectively: roosters, hens, and chicks update their positions according to Eq. (5), Eq. (7) and Eq. (8), respectively.

$$x_{i,j}(t+1) = x_{i,j}(t) \times \left(1 + Randn(0, \sigma^2)\right) \qquad (5)$$

$$\sigma^2 = \begin{cases} 1 & if\ f_i < f_k \\ exp\left(\frac{f_k - f_i}{|f_i| + \in}\right) & otherwise, k \neq i\ , \end{cases} \qquad (6)$$

where $x_{i,j}(t+1)$ and $x_{i,j}(t)$ denote the new and old position of ith rooster in jth dimension, respectively; t refers to the iteration number; $Randn$ is the normal distribution with zero mean and σ^2 SD, f_i and f_k are the fitness value of rooster *i* and rooster *k.*

$$x_{i,j}(t+1) = x_{i,j}(t) + s1 \times Rand1 \times \left(x_{r1,j}(t) - x_{i,j}(t)\right) + s2 \times Rand2 \times \left(x_{r2,j}(t) - x_{i,j}(t)\right) \qquad (7)$$

$$s1 = exp\left(\frac{f_i - f_{r1}}{|f_i| + \in}\right), \qquad s2 = \exp(f_{r2} - f_i)$$

where $x_{i,j}(t+1)$ and $x_{i,j}(t)$ denote the new and old position of *ith* hen in *jth* dimension, respectively; r1 is the rooster of the same group, and r2 is the randomly chosen rooster or hen from the entire chicken swarm such that $r1 \neq r2$; $Rand1$ and $Rand2$ is the random number in the range [0,1].

$$x_{i,j}(t+1) = x_{i,j}(t) + FL \times \left(x_{m,j}(t) - x_{i,j}(t)\right) \qquad (8)$$

$x_{i,j}(t+1)$ and $x_{i,j}(t)$ denote the new and old position of *ith* chick in *jth* dimension, respectively; $x_{m,j}(t)$ is the position of the mother; FL refers to the random number in the range of [0,2].

Step 5: Update individual's best fitness value (p_{best}) and the global best (g_{best}) solution: if the chicken's fitness value is better than the previous best one, the chicken's best fitness value or the global best solution will be updated using Eq. (9).

$$x_i(t) = \begin{cases} x_i(t-1), & f\big(x_i(t)\big) > f\big(x_i(t-1)\big), \\ x_i(t), & f\big(x_i(t)\big) \leq f\big(x_i(t-1)\big). \end{cases} \qquad (9)$$

Step 5: Calculate the variance of the current iteration using Equation 3.10

$$V_{ci}(t) = v_{min}^{r} + \frac{(v_{max}^{r} - v_{min}^{r})}{Ir * C\_iter} \qquad (10)$$

where v_{min}^{r} represents the minimum variation rate set in advance, v_{max}^{r} represents the maximum variability set in advance, Ir is the maximum number of iterations, C_iter is the current number of iterations. In this study v_{min}^{r} is set to 0.3 and v_{max}^{r} is set to 0.6. The idea of variability design of the CSO is to use the smaller mutation rate, and gradually increase the diversity of the algorithm with the iteration.

Step 6: Use Eq. (2) on the chicken swarm position for update base on mutation operation.

Step 7: If the number of iteration times t is less than Ir, return to **Step (3)**. Otherwise, terminate the algorithm and output the global best solution.

Step 8: Select Optimal Parameter for the optimum solution
Select the best global position $x_i(t)$ of the Chicken.

The Selected best global position $x_i(t)$ of the improved CSO becomes the optimal solution for optimization of hyperparameter of CNN. The flowchart of the improved CSO is shown in Figure 4.

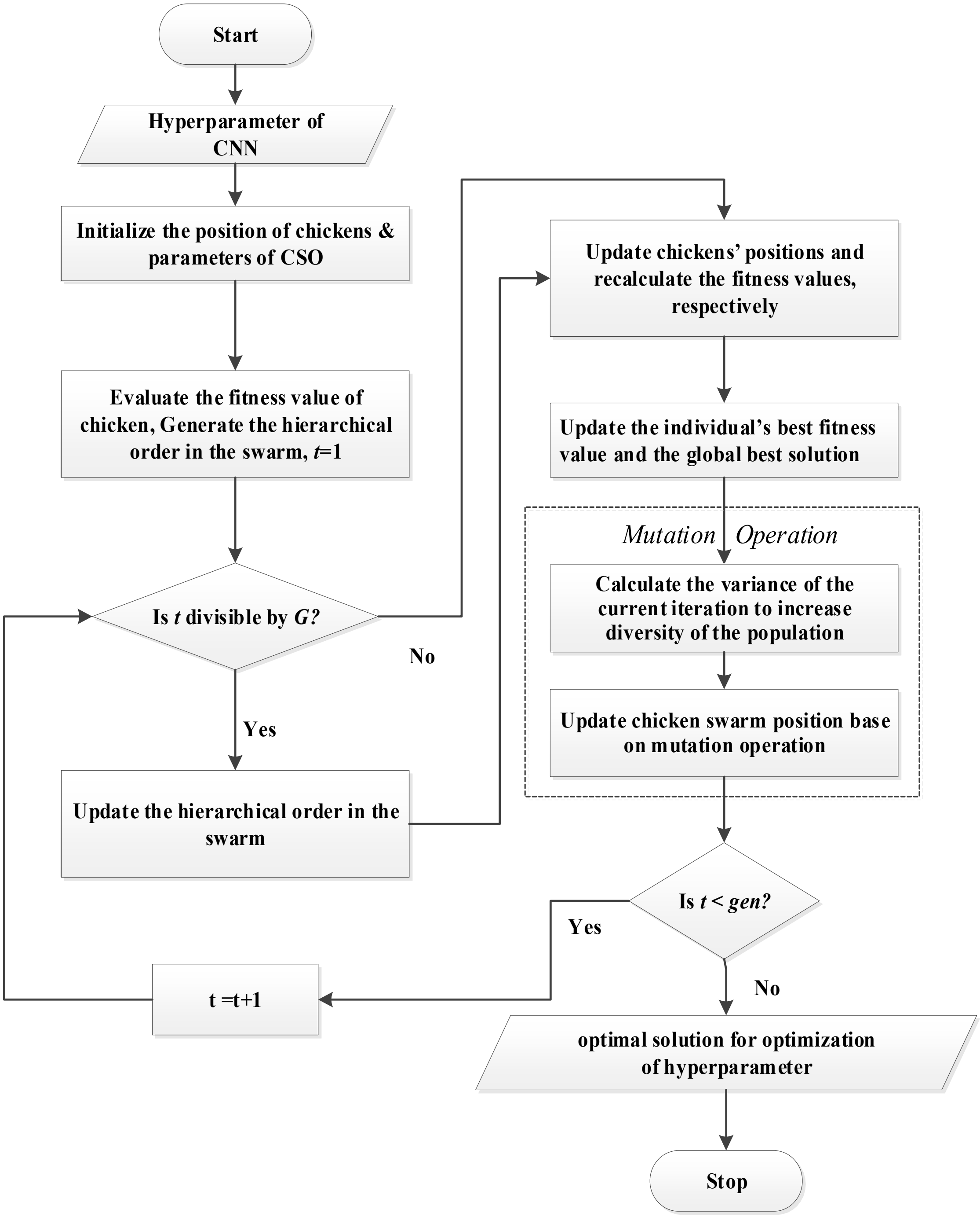


Figure 4: The Flowchart of the Improved CSO

3.6 Hyperparameter Optimization

Selecting appropriate hyperparameters presents a fundamental challenge when training convolutional neural networks (CNNs). Hyperparameter tuning entails determining the most effective values for the parameters that govern both the model's architecture and its learning dynamics. In the context of finger-vein recognition, key hyperparameters of the CNN include the learning rate, total number of training epochs, dropout probabilities, L_1 and L_2 regularization strengths, inclusion and configuration of batch normalization layers, and batch size. In the present work, the Improved Mutation-based Chicken Swarm Optimization (IMCSO) method is utilized to systematically optimize these hyperparameters of the lightweight CNN architecture specifically designed for finger-vein biometric data.

3.7 SoftMax Classifier

SoftMax classifier is a multi-class classification function. It takes a vector of n real numbers The SoftMax classifier serves as a multi-class classification function within the neural network. It receives a vector of n real-valued scores, where n corresponds to the total number of identity classes, and transforms these raw scores into a probability distribution that sums to 1 across all classes. Each resulting value lies in the range [0, 1], enabling the model to assign normalized confidence scores to every possible class. The SoftMax function computes the probability of each class relative to the others, and the class with the highest probability is selected as the predicted identity. For the finger-vein CNN, given the output vector, $F_{finger-vein}^{\varphi'}(t)$ from the final fully connected layer of the finger-vein CNN, the SoftMax score for class *j* is computed as shown in Eq. (11).

$$S_{finger-ve}\quad\left(F_{finger-vein}^{\varphi'}(t)\right)_j = \frac{e^{F_{finger-vei}^{\varphi'}\ (t)_j}}{\sum_{n=1}^{K} e^{F_{finger-vei}^{\varphi'}\ (t)\times n}} \qquad (11)$$

where: *K* is the number of classes, $F_{finger-vein}^{\varphi'}(t)_j$ is the output for class *j*, and $S_{(finger-vein)_j}$ is the probability that the input belongs to class *j*. The class with the highest probability is selected as the predicted identity.

3.8 Decision Module

The decision module uses the SoftMax classification score to verify user identity. The predicted class probability $S_{finger-vein}$ is compared against a predefined threshold *th*. If the probability exceeds the threshold, the user is accepted as genuine; otherwise, the user is rejected as an impostor.

Formally:

$$Decision(S_{finger-vein}) = \begin{cases} Accept\ (Genuine), & if\ \ S_{finger-vein} > th \\ Reject(Impostor), & otherwise \end{cases} \qquad (12)$$

3.9 Implementation of the Biometric Recognition System

The unimodal finger-vein biometric recognition system was designed, implemented, and rigorously evaluated to assess the performance of the proposed Improved Mutation-based Chicken Swarm Optimization–Convolutional Neural Network (IMCSO-CNN) technique in terms of effective feature extraction and accurate classification. The entire framework was developed and executed using MATLAB version 9.4 (R2018a), running on a Hewlett-Packard G56 laptop powered by an Intel® Core™ i5 Duo processor clocked at 2.7 GHz, with 6 GB of RAM and a 1 TB hard disk drive, under the Windows 10 Professional 64-bit operating system. This hardware configuration, while modest by contemporary standards, was deliberately selected to simulate realistic deployment scenarios on resource-constrained devices typical in many practical biometric applications. The system database was composed solely of finger-vein images acquired from multiple subjects, ensuring a diverse and representative sample set suitable for thorough validation of the recognition pipeline under varied conditions. A custom graphical user interface (GUI) was integrated to facilitate intuitive user interaction, supporting seamless enrollment of new finger-vein templates, automated feature extraction via the optimized IMCSO-CNN model, and real-time authentication/verification of finger-vein biometric samples.

3.10 Evaluation Measures

The performance of the evaluated biometric systems was assessed by computing the False Acceptance Rate (FAR), Equal Error Rate (EER), False Rejection Rate (FRR), and Recognition Accuracy. The FAR, FRR, EER, and accuracy values were derived by producing all possible genuine and impostor matching scores, followed by applying a decision threshold to determine acceptance or rejection of a match. A confusion matrix was employed to calculate these performance metrics. It included True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN) counts. The system’s effectiveness was quantified using the following measures:

$$\mathrm{FAR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} \qquad (13)$$

$$\mathrm{FRR} = \frac{\mathrm{FN}}{\mathrm{FN} + \mathrm{TP}} \qquad (14)$$

$$\mathrm{Accuracy} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{FP} + \mathrm{FN} + \mathrm{TP} + \mathrm{TN}} \qquad (15)$$

$$\mathrm{ERR} = \frac{\mathrm{FAR}(\eta) + \mathrm{FRR}(\eta)}{2} \qquad (16)$$

4. RESULT AND DISCUSSION

Tables 2, 3, and 4 summarize the performance of the CNN and CSO-CNN techniques on the finger-vein biometric dataset, which comprises 400 test images; 150 genuine and 250 impostors. From the confusion matrices in Tables 2 and 3, the CNN model correctly classified 124 genuine finger-vein images and misclassified 26 as impostors. It also correctly identified 228 impostor images while misclassifying 22 as genuine. In comparison, the CSO-CNN model improved these results, correctly identifying 129 genuine images and misclassifying only 21, while correctly rejecting 232 impostors and misclassifying 18.

Table 2: Contingency Table for Genuine/Impostor Classification Stage Using CNN Technique

		Predicted Class	
		Genuine	Impostor
Actual	Genuine (150)	124 (TP)	26 (FN)
Class	Impostor (250)	22 (FP)	228 (TN)

Table 3: Genuine/Impostor Classification Stage Using CSO-CNN Technique

		Predicted Class	
		Genuine	Impostor
Actual	Genuine (150)	129 (TP)	21 (FN)
Class	Impostor (250)	18 (FP)	232 (TN)

Table 4 provides the detailed metrics of False Acceptance Rate (FAR), False Rejection Rate (FRR), accuracy, and recognition time for both models at the optimal threshold of 0.8. The CNN achieved FAR of 8.80%, FRR of 17.33%, accuracy of 88.00%, and recognition time of 175.53 seconds. The CSO-CNN model outperformed the CNN with a False Acceptance Rate (FAR) of 7.20%, False Rejection Rate (FRR) of 14.00%, accuracy of 90.25%, and recognition time of 125.52 seconds.

Table 4: Results Based on Evaluation Measures

Threshold value	Technique	FAR (%)	FRR (%)	Accuracy (%)	Recognition Time (seconds)
0.80	CNN	8.80	17.33	88.00	175.53
	CSO-CNN	7.20	14.00	90.25	125.52

Figure 5 depicts the graphs of FAR against FRR for the finger-vein biometrics. The Equal Error Rate (EER) is the intersection between the FAR and FRR of each of the CSO-CNN and CNN techniques. It was revealed that in Figure 5, it was discovered that the CNN technique has an EER of 13.33% while the CSO-CNN technique has an EER of 10.67% for finger-vein biometric. From the EER results gotten from the graphs, it can be inferred that the CSO-CNN technique has a lower EER. Hence, the CSO-CNN outperformed the CNN technique for finger-vein biometric.

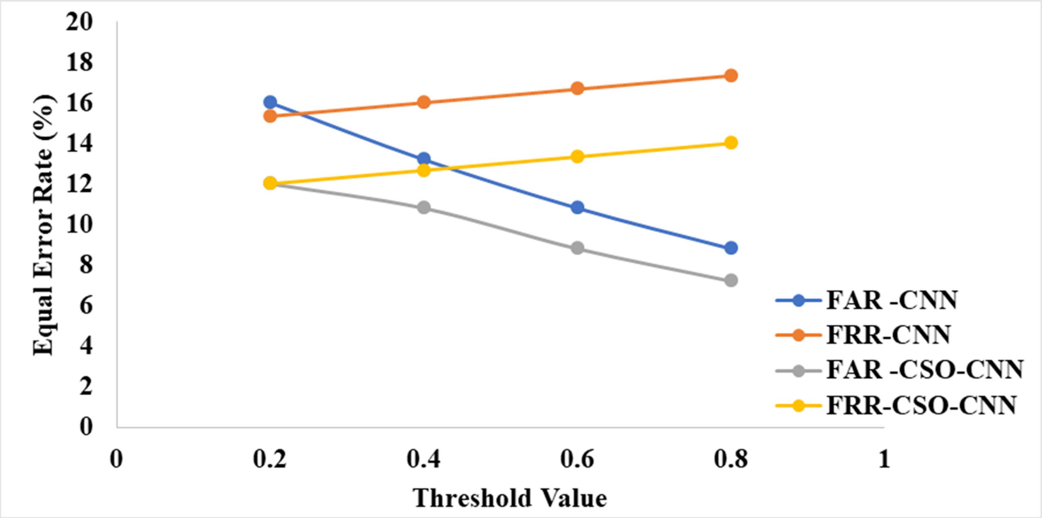


Figure 5: Equal Error Rate for Finger-Vein Biometric

Experimental evaluation of the finger-vein biometric system revealed that both CNN and CSO-CNN techniques achieved peak performance at a threshold of 0.8, consistent with literature showing better results near threshold. CSO-based hyperparameter optimization substantially reduced recognition time, despite the high-dimensional and complex nature of finger-vein data (see Figure 6). In accuracy, FAR, FRR, and EER, the CSO-CNN outperformed the standard CNN, delivering lower EER and greater robustness. Hyperparameter tuning proved essential for enhancing both accuracy and efficiency in the lightweight CNN model. The superior performance of finger-vein biometrics stems from its rich vascular features and effective capture of discriminative patterns.

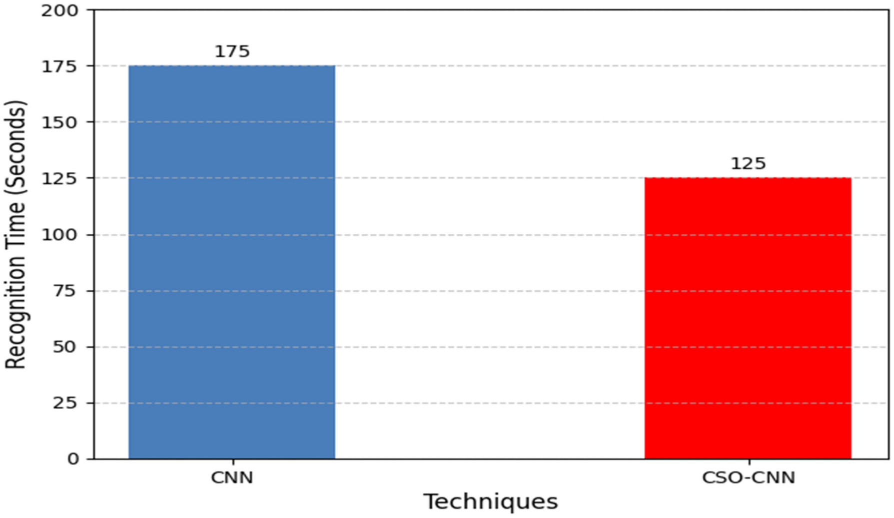


Figure 6: Recognition Time of Finger-Vein Biometric

The study confirmed the classic inverse relationship between FAR and FRR, underscoring the inherent trade-off between security (low FAR) and usability (low FRR), with the Equal Error Rate (EER) acting as the primary balanced performance indicator. The notably lower EER achieved by the CSO-CNN approach reflects stronger overall security through reduced false acceptances and false rejections alike. Hyperparameter optimization within the lightweight CNN proved essential for boosting both recognition accuracy and computational efficiency in finger-vein biometrics. The finger-vein modality itself exhibited superior accuracy compared to many other biometric traits, attributable to its extensive vascular surface area and dense, discriminative feature patterns that enable more reliable capture and differentiation. Paired-sample t-tests were conducted to statistically compare accuracy between the CSO-CNN and standard CNN techniques. The SPSS-derived results, presented in Tables 5 and 6, provide inferential evidence of the observed improvements.

The hypothesis is defined as:
H_0: There is no significant difference between CSO-CNN and CNN technique.
H_1: There is a significant difference between CSO-CNN and CNN technique.

Table 5: Summary of Result of T-test of Accuracy for CSO-CNN and CNN technique

Parameter	T	Degree of Freedom (df)	p-value	Comment
Finger-vein	8.555	3	0.003	Significant

Table 6: Summary of Result of T-test of Accuracy for Finger-vein

Parameter	T	Degree of Freedom (df)	p-value	Comment
Accuracy	18.520	7	0.000	Significant

5. CONCLUSION AND RECOMMENDATIONS

This study demonstrates that Chicken Swarm Optimization (CSO) constitutes a highly effective approach for hyperparameter optimization within lightweight convolutional neural network (CNN) architectures, substantially elevating the overall performance of finger-vein biometric recognition systems. Through precise tuning of essential parameters, CSO effectively overcomes the challenges associated with training compact CNN models on high-dimensional finger-vein datasets, delivering superior classification accuracy alongside markedly reduced recognition latency. These findings underscore the practical value of CSO in constructing reliable and efficient biometric authentication systems, where achieving both high precision and low computational overhead is essential.

The key contributions of this research are summarized as follows:

- Integration of Chicken Swarm Optimization for tuning hyperparameters in lightweight CNN architectures specifically designed for finger-vein biometrics
- Demonstration of improved recognition accuracy, reduced False Acceptance Rate (FAR), False Rejection Rate (FRR), and Equal Error Rate (EER) through CSO optimization.
- Confirmation of CSO's superiority over conventional CNN training approaches in terms of computational efficiency and recognition performance on finger-vein datasets.
- Collectively, these contributions fill important gaps in efficient hyperparameter optimization for finger-vein biometrics, offering an integrated framework that effectively reconciles high

accuracy with reduced computational demands. The investigation affirms CSO as a robust and practical optimization technique for biometric applications involving the intricate, high-dimensional characteristics of finger-vein patterns

6. LIMITATIONS AND FUTURE WORKS

It is expedient to note, in spite of the improved features of this algorithm, Chicken Swarm Optimization (CSO) algorithm exhibits notable sensitivity to its internal hyperparameters including swarm population size, maximum number of iterations, and frequency of hierarchy updates requiring meticulous manual adjustment to reliably reach optimal outcomes. Poorly selected values may result in delayed convergence or premature entrapment in local optima. Furthermore, the IMCSO-CNN framework could face scalability constraints when processing extremely large finger-vein datasets which may elevate computational requirements beyond the capabilities of resource-limited deployment settings.

These limitations highlight promising directions for future research and refinement. Future studies should evaluate the robustness of CSO-optimized finger-vein recognition systems against adversarial attacks and spoofing attempts, while also assessing energy consumption and real-world user experience in practical deployment settings (e.g., mobile devices, access control kiosks, or wearable authentication). Emerging metaheuristic algorithms, such as the Farmland Fertility Algorithm and African Vultures Optimization Algorithm, could serve as effective alternatives or complementary methods to CSO, potentially offering faster convergence, better handling of multimodal search spaces, or improved performance on high-dimensional finger-vein datasets.

Based on the outcomes of this investigation, the following recommendations are proposed:

1. Further exploration of advanced variants and hybrid forms of the Chicken Swarm Optimization (CSO) technique applied to lightweight deep CNN architectures for finger-vein recognition should be pursued.
2. Fusion of vein patterns from multiple fingers (e.g., right and left index/middle fingers) to create a multi-instance finger-vein biometric system is recommended, as this approach could significantly enhance recognition accuracy and robustness to intra-class variations.

REFERENCES

1. Ahmad, F., Cheng, L. M. and Khan, A. (2019). Lightweight and privacy-preserving template generation for palm-vein-based human recognition. IEEE Transactions on Information Forensics and Security, 15: 184-194.
2. Amador-Angulo, L., Castillo, O., Peraza, C., and Ochoa, P. (2021). An Efficient Chicken Search Optimization Algorithm for the Optimal Design of Fuzzy Controllers. Axioms, 10(1), 30.
3. Assim, O. M., and Alkababji, A. M. (2021). CNN and Genetic Algorithm for Finger Vein Recognition. In 2021 14th International Conference on Developments in eSystems Engineering (DeSE). 503-508.

4. Bergstra, J., and Bengio, Y. (2012). Random search for hyper-parameter optimization. Journal of machine learning research, 13(2): 281–305.
5. Bharanidharan, N., and Rajaguru, H. (2020). Improved chicken swarm optimization to classify dementia MRI images using a novel controlled randomness optimization algorithm. International Journal of Imaging Systems and Technology, 30(3), 605-620.
6. Boucherit, I., Zmirli, M. O., Hentabli, H., and Rosdi, B. A. (2020). Finger vein identification using deeply-fused Convolutional Neural Network. Journal of King Saud University-Computer and Information Sciences. 1-11.
7. Das, R., Piciucco, E., Maiorana, E., and Campisi, P. (2018). Convolutional neural network for finger-vein-based biometric identification. IEEE Transactions on Information Forensics and Security, 14(2), 360-373.
8. Deb, S., Gao, X. Z., Tammi, K., Kalita, K., and Mahanta, P. (2019). Recent studies on chicken swarm optimization algorithm: a review (2014–2018). Artificial Intelligence Review, 1-29.
9. Gao, Z., Li, Y., Yang, Y., Wang, X., Dong, N., and Chiang, H. D. (2020). A GPSO-optimized convolutional neural networks for EEG-based emotion recognition. Neurocomputing, 380, 225-235.
10. Hartung, D. and Busch, C. (2009). Why vein recognition needs privacy protection. In 2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing. 1090-1095.
11. Hashimoto, J. (2006). Finger vein authentication technology and its future. In 2006 Symposium on VLSI Circuits, 2006. Digest of Technical Papers. (pp. 5-8). IEEE.
12. He, Z., Xu, Q., Ye, Y., and Li, W. (2017). Effects of meteorological factors on finger vein recognition. In 2017 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA) (pp. 1-8). IEEE.
13. Huang, H., Liu, S., Zheng, H., Ni, L., Zhang, Y., and Li, W. (2017). DeepVein: Novel finger vein verification methods based on deep convolutional neural networks. In 2017 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA). 1-8.
14. Khan, M. H. M., Subramanian, R. K., and Khan, N. A. M. (2009). Representation of hand dorsal vein features using a low dimensional representation integrating cholesky decomposition. In 2009 2nd International Congress on Image and Signal Processing. 1-6.
15. Kirchgasser, S., Kauba, C., Uhl, A., Andreas, S. M. R. V., and Busch, U. C. (2019). Cancellable Biometrics for Finger Vein Recognition—Application in the Feature Domain. In Handbook of Vascular Biometrics. 16, pp. 507-525). Springer Nature.
16. Kirchgasser, S., Kauba, C., and Uhl, A. (2020). Towards Understanding Acquisition Conditions Influencing Finger Vein Recognition. In Handbook of Vascular Biometrics (pp. 179-199). Springer, Cham.
17. Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25, 1097-1105.
18. Kuzu, R. S., Piciucco, E., Maiorana, E. and Campisi, P. (2020). On-the-fly finger-vein-based biometric recognition using deep neural networks. IEEE Transactions on information Forensics and Security, 15, 2641-2654.
19. LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11): 2278-2324.

20. LeCun, Y. A., Bottou, L., Orr, G. B., and Müller, K. R. (2012). Efficient backprop. In Neural networks: Tricks of the trade (pp. 9-48). Springer, Berlin, Heidelberg.
21. Li et al., 2019 Li, Y., Xiao, J., Chen, Y., and Jiao, L. (2019). Evolving deep convolutional neural networks by quantum behaved particle swarm optimization with binary encoding for image classification. Neurocomputing, 362: 156-165.
22. Lin et al., 2018 Lin, M., Zhong, Y., Lin, J., and Lin, X. (2018). Enhanced chicken swarm optimization for function optimization problem. International Journal of Wireless and Mobile Computing, 15(3), 258-269.
23. Matsuda et al., 2016 Matsuda, Y., Miura, N., Nagasaka, A., Kiyomizu, H., and Miyatake, T. (2016). Finger-vein authentication based on deformation-tolerant feature-point matching. Machine Vision and Applications, 27(2), 237-250.
24. Meng et al., 2014 Meng, X., Liu, Y., Gao, X., and Zhang, H. (2014). A new bio-inspired algorithm: chicken swarm optimization. In International conference in swarm intelligence. 86-94.
25. Nikbakht et al., 2021 Nikbakht, S., Anitescu, C., and Rabczuk, T. (2021). Optimizing the neural network hyperparameters utilizing genetic algorithm. Journal of Zhejiang University-Science A, 22(6): 407-426.
26. Oguntoye et al., 2019 Oguntoye, J. P, Ola, B. O. and Awodoye, O. O. (2019). Development of an Improved Palm Vein Recognition System Using a Swarm Intelligent Based Support Vector Machine. Proceedings of the 22nd SMART iSTEAMS SPRING Multidisciplinary Conference. 171-182.
27. Podgantwar and Raut, 2013 Podgantwar, U. D., and Raut, U. K. (2013). Extraction of finger vein patterns using gabor filter in finger vein image profiles. International Journal of Engineering Research and Technology, 2(6), 3294-3298.
28. Qin and El-Yacoubi, 2017 Qin, H. and El-Yacoubi, M. A. (2017). Deep representation-based feature extraction and recovering for finger-vein verification. IEEE Transactions on Information Forensics and Security, 12(8): 1816-1829.
29. Qolomany et al., 2017 Qolomany, B., Maabreh, M., Al-Fuqaha, A., Gupta, A., and Benhaddou, D. (2017). Parameter optimization of deep learning models using Particle swarm optimization. In 2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC). 1285-1290.
30. Qu et al., 2017 Qu, C., Zhao, S. A., Fu, Y., and He, W. (2017). Chicken swarm optimization based on elite opposition-based learning. Mathematical Problems in Engineering, 2017.
31. Serizawa and Fujita, 2020 Serizawa, T., and Fujita, H. (2020). Optimization of convolutional neural network using the linearly decreasing weight particle swarm optimization. arXiv preprint arXiv:2001.05670.
32. Srivastava et al., 2014 Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1): 1929-1958.
33. Uhl, 2020 Uhl, A. (2020). State of the art in vascular biometrics. In Handbook of Vascular Biometrics. 3-61.
34. Veldhuis et al., 2019 Veldhuis, R., Spreeuwers, L., Ton, B., and Rozendal, S. (2019). A high quality finger vein dataset collected using a custom designed capture device. In Handbook of Vascular Biometrics. 13-15.

35. Wang et al., 2019 Wang, Y., Zhang, H., and Zhang, G. (2019). cPSO-CNN: An efficient PSO-based algorithm for fine-tuning hyper-parameters of convolutional neural networks. Swarm and Evolutionary Computation, 49: 114-123.
36. Wang et al., 2019b Wang, J., Cheng, Z., Ersoy, O. K., Zhang, M., Sun, K., and Bi, Y. (2019b). Improvement and application of chicken swarm optimization for constrained optimization. IEEE Access. 7: 58053-58072.
37. Wu et al., 2016 Wu, D., Xu, S., and Kong, F. (2016). Convergence analysis and improvement of the chicken swarm optimization algorithm. IEEE Access, 4: 9400-9412.
38. Yang et al., 2012 Yang, G., Xi, X., and Yin, Y. (2012). Finger vein recognition based on (2D) 2 PCA and metric learning. Journal of Biomedicine and Biotechnology, 1-9.
39. Yang et al., 2018 Yang, L., Yang, G., Xi, X., Su, K., Chen, Q. and Yin, Y. (2018). Finger vein code: From indexing to matching. IEEE Transactions on Information Forensics and Security, 14(5): 1210-1223.
40. Zhang and Wang, 2021 Zhang, Z. and Wang, M. (2021). Finger Vein Recognition by Generating Code. arXiv preprint arXiv:2101.08415.
41. Zhao et al., 2020 Zhao, D., Ma, H., Yang, Z., Li, J. and Tian, W. (2020). Finger vein recognition based on lightweight CNN combining center loss and dynamic regularization. Infrared Physics & Technology. 105(103221): 1-10.