INJ INTERNATIONAL NEUROUROLOGY JOURNAL

Design and Development of Model Using Fingervein Authentication by Convolution Neural Network

Naitik ST, Dr. JV Gorabal
Department of CSE, Research Center- ATME College of Engineering, 13th Kilometer, Mysore – Kanakapura – Bangalore Road, Mysore – 570 028, and Visvesvaraya Technological University, Belagavi- 590018, and Department of CSE, Dayananda Sagar University, Devarakaggalahalli, Harohalli, Kanakapura Road,Ramanagara Dt.,Bengaluru – 562 112
Professor, Department of CSE, Research Center- ATME College of Engineering, 13th Kilometer, Mysore – Kanakapura – Bangalore Road, Mysore – 570 028, and Visvesvaraya Technological University, Belagavi- 590018

Abstract

In an era where safeguarding personal possessions and sensitive information is of paramount concern, conventional biometric identification systems based on physiological and behavioral attributes like faces, irises, or fingerprints have encountered various limitations. In recent times, biometric recognition techniques rooted in vascular patterns, particularly finger veins, have gained substantial popularity. However, the existing finger vein recognition systems face challenges related to intricate image pre-processing and feature vector representation, rendering their widespread adoption cumbersome. To solve this issue, the study proposes a novel approach for finger vein identification based on Convolutional Neural Networks (CNN). The research makes use of multiple CNNs, such as VGG, AlexNet, and an optimized CNN known as CNN-HPSGWO, which is fine-tuned utilizing Particle Swarm and Gray Wolf optimization techniques. Three benchmark datasets are collected and processed using advanced approaches to improve image quality and guarantee the effectiveness of the suggested approach. The accuracy, Equal Error Rate (EER), and matching time of the models are extensively investigated. Notably, the CNN-HPSGWO model performs excellent on the SDUMLA dataset, with an accuracy of 95.87%, 96.24% on the FV-USM dataset, and 97.07% on the HKPU dataset. These findings highlight the suggested CNN-HPSGWO model's effectiveness in improving the security and reliability of finger vein authentication systems.

Key Words: Finger Vein, Convolutional Neural Network, Bio-metric, Accuracy, Matching Time, Image Dataset.

Introduction

The importance of safeguarding personal belongings and sensitive information has grown significantly. Traditional biometric identification systems, which rely on physical and behavioral traits like faces, irises, or fingerprints, have their limitations [1, 2]. These traits can be easily falsified or altered over time, as is the case with fingerprints. Iris recognition, on the other hand, can be uncomfortable due to the bright lighting used during the process [3], and face recognition accuracy is affected by various factors such as poses, occlusions, facial expressions, and lighting conditions [4].

Nowadays, biometric recognition methods based on vascular patterns, particularly finger veins, have gained popularity [5-7]. Finger-vein biometric systems capture unique patterns inside the finger, making replication virtually impossible. Finger-vein features offer several advantages, including:

- Unique patterns for each individual, even identical twins, provide excellent distinction.
- Permanent patterns that don't change over time.
- Resistance to forgery, observation, damage, or obstruction, unlike fingerprint recognition.
- Wide acceptance due to small capturing devices and high recognition accuracy.
- User-friendly, with non-invasive and contactless image capture for convenience and cleanliness.
- Flexibility with ten fingers available for authentication, ensuring continued access in case of unexpected issues with one finger.

Despite these advantages, there are still challenges to address for real-world deployment of finger-vein biometric systems. Image quality is heavily influenced by the capturing device's characteristics, as the camera is very close to the finger, leading to optical blurring. Poor lighting or misalignment of the finger position can also hinder recognition. Variations in bone thickness and skin properties, as well as light scattering, introduce noise that requires complex image processing algorithms in conventional finger-vein recognition methods. Recent research indicates that Artificial Intelligence (AI) methods can effectively address these challenges, offering a promising solution to improve the efficiency of biometric authentication systems based on finger veins. In this research, we used DL techniques for reliable and automated authentication systems using finger vein.

Literature Survey

In finger vein recognition research, there has been a notable shift toward the adoption of DL techniques due to their remarkable performance. This section summarises the most recent research on finger vein biometrics, offering insight into the present state of the art. In the journal [8], the author introduced an enhanced deep network referred to as "Merge CNN." This network leverages the integration of several CNNs interconnected with shorter paths. The fundamental concept is to evaluate input images of varying quality using multiple identical CNNs and then integrate the outputs into a single layer. To accomplish this, they developed a large number of networks and trained them using finger vein images. The merged CNN was constructed using the most effective CNN design available, and it combines the original image with an improved version of the image generated via the CLAH method. The proposed methodology demonstrated superior performance compared to other techniques documented in the existing literature. Research [9] introduced a finger-vein identification system that leveraged a pre-trained Deep-CNN model, known as Squeezenet, to extract features from both left and right finger vein patterns. To minimize feature dimensions and pick the most essential features, feature-level Discriminant Correlation Analysis was applied. In the identification stage, the obtained composite feature vectors served as input for a Support Vector Machine classification. The system's performance was evaluated using two widely accessible finger vein databases, achieving impressively high mean accuracy rates. In the study [10], the focus was on enhancing the recognition performance of finger vein systems by addressing image quality issues. The researchers devised a novel method for evaluating finger vein image quality, which involved analysing the matching of samples from different fingers to automatically assess and classify image quality. A lightweight CNN was created to differentiate between high and low-quality images, recognizing common characteristics in low-quality images. The suggested technique has been evaluated on several public datasets with respect to existing identification methods, proving the quality evaluation standard. Utilizing the local binary pattern technique, the system improved its performance by removing low-quality images.

The research [11] introduced a novel approach to recognizing finger vein-related data. The proposed approach based on Long-Short-Term-Memory and DL techniques, focused on the classification of finger vein images. To simplify the authentication structure, a Deep Stacked Auto Encoder-based decision-making was employed to classify various vein features found in the extensive data. The model encompassed various stages, including finger-vein data gathering, pre-processing, Feature Extraction (FE), database matching, and evaluating accuracy. Extensive experimental testing showcased the superior performance and advanced quality of the proposed method across various metrics. The research [12] suggested a strong multimodal biometric identification solution that included finger vein and electrocardiogram (ECG) biometrics. The system was divided into three parts: pre-processing, FE, and authentication. Normalization and filtering methods were used for every biometric throughout the pre-processing stage. FE was accomplished using a deep CNN model, and the authentication process utilized well-known Machine Learning (ML) classifiers. Multi-Canonical Correlation Analysis (MCCA) was implemented to reduce the dimensionality of feature representation and speed up authentication. The integration of ECG and finger vein systems through feature and score fusion significantly enhanced authentication performance, reducing EER by feature and score fusion. Notably, the suggested multimodal system, incorporating MCCA feature fusion with a KNN model, achieved greater accuracy compared to other ML algorithms.

The study [13] investigated the application of triplet loss function and DL techniques for finger vein authentication. To train and validate the model, it was applied to various databases. The study evaluated accuracy and other performance metrics for various training-validation set combinations, generating ROC and AUC values. The best result achieved excellent accuracy, ROC-AUC values, indicating the model's effective identification of finger veins. Importantly, cross-validation between diverse datasets demonstrated the model's adaptability and applicability, suggesting that models trained on more challenging datasets exhibited greater robustness. The paper [14] introduced the Xception network, a pre-trained CNN structure. This network, which is built on depth-wise separable CNNs having residual connections, is well-known for its ability to acquire robust features. The work encompassed a three-stage process: data pre-processing to standardize input samples, data augmentation to address the shortage of training samples, and FE and categorization done by the pre-trained Xception network. The network's performance was evaluated using the various datasets. The results underscore the excellent performance of the proposed method in comparison to existing approaches. In the article [15], a lightweight Siamese architecture with a self-attention system was developed to improve

DOI: 10.5123/inj.2023.4.in77
ISSN:2093-4777 | E-ISSN:2093-6931
Vol. 27 Iss. 4 (2023)

INJ INTERNATIONAL NEUROUROLOGY JOURNAL

authentication effectiveness for low-quality finger vein images. The researchers used a three-layer CNN to obtain finger vein characteristics. It included a global context network for capturing information across various layers, a multi-scale feature fusion technique for increasing features, and a self-attention convolution mechanism for weighing and vectorizing the fused features. Comprehensive research conducted with various datasets showed that the suggested strategy beats state-of-the-art techniques.

Materials and Methods

The methodology detailed in this section follows a well-structured path, consisting of three essential stages: gathering data, processing data, and building a deep learning (DL) model.

Data Collection

This research employs three benchmark datasets, namely SDUMLA, FV–USM, and HKPU. Each of these datasets is comprehensively described, outlining their specific characteristics below. Figure 1 gives the sample images from each dataset.

SDUMLA Dataset: The SDUMLA database [16], developed by Shandong University, is a multimodal biometric database. It comprises a finger vein database that contains finger vein images of 6 fingers from 106 individuals. These fingers encompass the index, middle, and ring fingers on both hands, resulting in a total of 636 (106 × 6) distinct categories, each with six instances. The images in this database are captured at a resolution of 320 × 240 pixels and may exhibit variations, including finger posture tilts and background noise introduced by the acquisition device.

FV–USM Dataset: The Finger Vein Universiti Sains Malaysia (FV-USM) database [17] was compiled from a sample of 123 individuals, made of 83 men and 40 women, whose ages ranged from 20 to 52 years. The process of capturing images took place during two distinct sessions, separated by a period of over two weeks. Within each session, four samples were gathered from every individual. Each sample underwent six separate image captures. As a result, the database comprises a total of 5,904 images (which arises from 2 sessions, each involving 123 subjects, further multiplied by 4 fingers, each with 6 images), each possessing dimensions of 640*480 pixels. For the purpose of our research, we exclusively focused on one session, which encompassed a total of 2,952 images. Within this subset of the FV-USM database, 146 out of 492 subjects were classified as having images of high quality, while the remaining subjects had images of low quality.

HKPU Dataset: The Hong Kong Polytechnic University (HKPU) vein database [18] was established through the utilization of a non-contact imaging device on the university campus, covering the period from April 2009 to March 2010. This comprehensive database encompasses 3,132 images, with each image measuring 513x256 pixels, all sourced from 156 subjects. Among these subjects, the initial 105 were captured during two separate sessions. The time for these sessions ranged from a month to more than six months, having an average gap of around 66.8 days. The 105 participants submitted six samples of their middle and index fingers. The other 51 individuals, received their images recorded in a session.

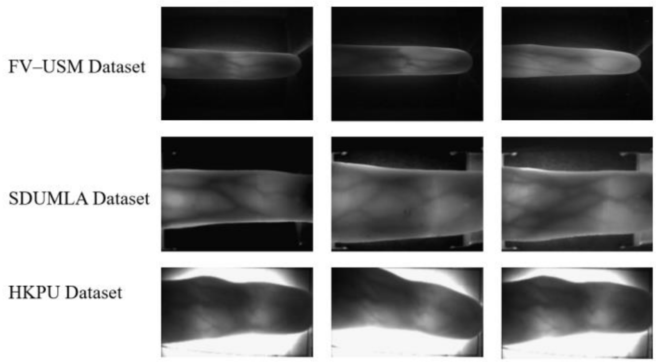

Fig. 1. Sample finger vein images from each dataset

Data Pre-processing

The effectiveness of finger vein authentication systems frequently hinges on the quality of the captured vein images. Consequently, it is essential to pre-process the raw vein images obtained from infrared sensors before proceeding with FE.

DOI: 10.5123/inj.2023.4.in77
ISSN:2093-4777 | E-ISSN:2093-6931
Vol. 27 Iss. 4 (2023)

Given that input images from publicly available datasets come in various sizes and orientations, the initial step involves Region of Interest (ROI) selection, ensuring that the intended portion of the raw image is uniformly represented in terms of dimensions. This step guarantees consistency in all images entering the CNN for FE. Subsequently, all images undergo normalization, a process in which they are subsampled to the desired dimensions by cropping them to m*n pixels. Following normalization, the cropped images are subjected to image enhancement through Contrast Limited Adaptive Histogram Equalization (CLAHE) [19]. This enhancement adjusts the image intensities to achieve a more balanced distribution on the histogram. As a result, global contrast in the image is enhanced, with frequently occurring intensity values being spread effectively, while lower contrast areas are elevated. Further refinement of vein features is achieved through the implementation of a Gabor filtering technique. A Gabor filter [20] is a specialized variant of a bandpass filter specifically designed for texture segmentation. It has optimal image localization capabilities in both the spatial and frequency domains. This filter substantially reduces the effects of noise, allowing for more efficient FE as well as better identification accuracy.

For clarity enhancement, the vein images undergo morphological operations [21]. These operations are intended to remove extraneous details while preserving critical shape characteristics. Grayscale erosion is initially applied, next to grayscale dilation. The dilation and erosion transformations prove to be particularly valuable for obtaining a clear representation of the vein patterns. Finally, a binarization technique is utilized to roughly identify the vein patterns. This entails applying a threshold of 110, with pixel values exceeding 110 representing the vein pattern in white, and the remaining values denoting background by black color. Once this pre-processing stage is completed, the images are promptly passed to the DL model for further analysis.

Deep Learning

The architecture of four CNN models such as VGG16, AlexNet, CNN, and CNN-HPSGWO is detailed below.

VGG-16: The VGG16 CNN model extends the AlexNet architecture through a replacement of big kernel-sized filters with a sequence of 3x3 kernel-sized filters [22]. The VGG-16 model comprises four primary components: the SoftMax classifier, FCL, convolution module, and attention module [23]. The attention module's role is to capture relationships among features in finger vein images. It conducts average pooling and max pooling on the input tensor, utilizing the 4th pooling layer (PL) of the VGG-16 model. For a convolution with a 7x7 filter size, the max-pooled 2D tensor is fused with every other layer using a sigmoid function. The convolution module corresponds to the 4th PL of the VGG-16 network and serves to retrieve features from finger vein images in a scale-invariant manner. It acquires the features required for real-time processing of finger vein images from midlevel layers. Features from other layers, whether low-level or high-level, are deemed unsuitable for real-time finger vein images. Therefore, the attention module is paired with the 4th PL, and the result of this module is combined with the 4th PL. The merged features from the convolution and attention blocks are transformed into one-dimensional (1D) features through fully connected layer (FCL), consisting of dense, dropout, and flattened layers. The dropout rate is 0.5, and the Dense Layer (DL) is restricted to 24 units. The FCL extracts features for classification, and the softmax layer is employed for the final recognition of finger vein images. The total neurons in the softmax layer correspond to the categories in the classification task, serving as the final DL. The softmax layer generates probability scores based on the classification.

AlexNet: In 2012, a team led by Alex Krizhevsky introduced a CNN model, AlexNet, which surpassed LeNet in terms of depth and width [24]. This model had a pivotal breakthrough in ML and computer vision by winning the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2012, a contest focused on visual object recognition and classification. The architecture of AlexNet included several key components. The initial convolutional layer (CL) included Local Response Normalisation (LRN) as well as convolution and max pooling processes [25]. This layer is made of 96 different 11x11 receptive filters. Max pooling was carried out using 3x3 filters and a stride length of 2. The second layer used 5x5 filters to do similar processes. The three subsequent CLs used 3x3 filters. The structure also had two FCLs with dropouts, which were accompanied by a SoftMax layer. AlexNet introduced two innovative concepts: Local Response Normalization (LRN) and dropout. AlexNet featured three CLs and two FCLs.

CNN: The advantages of CNN, including automated feature selection and complete training, have led to its widespread use in image categorization [26]. Different layers within the CNN, including pooling, dropout, DLs, and the CL, perform distinct roles. The CNN effectively processes images through the functionalities of its layers: the CL automatically selects features, the PL reduces features, and the DL handles classification [27]. Each layer in the CNN plays a significant role, with the number of features extracted determined by factors like kernel size and the number of kernels. Kernel weights are initialized with random values and adapted during the model's training. The output of the CL is passed through the ReLu (Rectified Linear Unit), which utilizes a

DOI: 10.5123/inj.2023.4.in77
ISSN:2093-4777 | E-ISSN:2093-6931
Vol. 27 Iss. 4 (2023)

nonlinear activation function to keep values within a defined range. To combat overfitting, the dropout layer is applied after ReLu to assist with regularization. Subsequently, the PL downsamples feature maps, enhancing their invariance to minor input variations. The PL executes individual feature maps autonomously, utilizing methods like max pooling for obtaining low-level features. Max pooling chooses the highest possible value from each region of feature maps. The higher-level layers of the CNN are frequently made up of FCLs, which use the PL information to make categorization decisions. The CNN model's last layer employs a SoftMax function to provide a probability distribution for multiclass categorization. In the CNN, regularisation is used to prevent overfitting. Regularisation reduces overfitting by including a penalty term in the loss function. Dropout can be employed for addressing overfitting by restricting the utilization of each neuron. Following the definition of CNN's framework, backpropagation is employed to alter the core weights to fit with the objectives of the particular challenge.

CNN-HPSGW: The CNN model includes various hyperparameters, such as kernel size, kernel count, kernel stride, activation function in the CL, max-pooling kernel size, dropout rate, DL size, learning rate, number of convolutions, max-PLs, and more [28]. These hyperparameters significantly influence the CNN model's output. To do this we suggest HPSGW optimization for CNN network. The theoretical and mathematical foundation of HPSGW is explored in this part, along with its application in the development of an automated CNN using the metaheuristic method. The hyperparameter tuning, such as the number of layers and epochs, kernel, and batch size, is automated throughout network construction. To get the best possible results, the final CNN model is built with the most optimal hyperparameters.

PSO: The author [29] developed the PSO model, which mimics the swarming behaviour of species. The birds will spread out and then converge on a food source before finally deciding where to eat. During this quest for food, one bird consistently detects the scent of food more clearly than the others. This bird communicates the location of the prey effectively, prompting the flock to simultaneously converge on the food source. The PSO algorithm harnesses this animal behaviour to address global optimization problems, with each member of the swarm known as the particle [30]. This algorithm utilizes the below equations to update the positions. The first equation governs the calculation of each particle's next velocity v_i^{t+1}:

$$v_i^{t+1} = w * v_i^t + c1r1(xBest_i^t) + c2r2(gBest_i^t - x_i^t) \quad [1]$$

In this equation, w represents the initial inertia, v_i^{t+1} is the particle's previous velocity at time t, and the terms r and c are random numbers and acceleration coefficients. The term $xBest_i^t$ and $gBest_i^t$ represents the local and global best fitness value. x_i^t is the particle's previous position at time t. The subsequent position of the particle, x_i^{t+1}, is determined by the second equation:

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad [2]$$

These equations play a crucial role in the real-time updating of the local and global best values within the swarm (as expressed in Eq. 3 and Eq. 4). The best values are evaluated at a given time t, ensuring that the PSO algorithm consistently strives for optimization. The local best value ($xBest_i^{(t)}$) is updated according to the following conditions:

$$xBest_i^{(t+1)} = \begin{cases} xBest_i(t) iff (xBest_i(t) \leq f(x_i(t+1)) \\ x_i(t+1) iff (xBest_i(t) \geq f(x_i(t+1)) \end{cases} \quad [3]$$

Additionally, the global best value ($gBest$) is updated based on the maximum fitness value (f) achieved during the previous iterations. The specific selection is guided by the following equation:

$$gBest(t+1) = \max\{f(y), f(gBest(t))\} \quad [4]$$
$$y \in pBest_0(t), pBest_1(t), \ldots\ldots, pBest_2(t)(t) \quad [5]$$

Here, y represents various $pBest$ values (local best values) at different times t.

GWO: The GWO is a metaheuristic optimization algorithm that was introduced in a study [31]. It is based on the social structure and hunting techniques of grey wolves in the wild. Within a GWO, wolves are categorized into distinct ranks, including Beta(β), Alpha(α), Delta(δ), and Omega(γ), which simulate a leadership structure [32]. Alpha wolves hold the key leadership roles and make decisions for the group, overseeing all group

DOI: 10.5123/inj.2023.4.in77
ISSN:2093-4777 | E-ISSN:2093-6931
Vol. 27 Iss. 4 (2023)

activities, including hunting. Beta wolves play a supportive role under the alpha wolves. Omega wolves follow the hierarchical structure and maintain dominance in the group, while the subordinate wolves are referred to as Delta. The GWO algorithm classifies solutions into three levels based on their fitness and optimality. Typically, the decision made by the alpha wolf is the most favourable for optimization problems. However, when there is no centralized leader to guide the group throughout the optimization process, swarm intelligence methods are employed. GWO tackles this challenge by allowing individual grey wolves to take on leadership roles, ensuring self-organization within the group. GWO, inspired by the behaviour of grey wolves, is especially adept at handling tasks like image recognition and classification, making it a valuable tool in swarm intelligence applications [33]. The primary processes in the GWO optimization involve the behaviour of each swarm agent, as described by mathematical equations (Eqs. 6 and 7):

$$\vec{D} = \left|\vec{C}.\vec{X}p(t) - \vec{X}(t)\right| \quad [6]$$
$$\vec{X}(t+1) = \left|\vec{X}p(t) - \vec{A}.\vec{D}\right| \quad [7]$$

$'D'$ reflects each agent's behavior, $'t'$ indicates the present iteration, $'Xp'$ is the position of the prey, and $'X'$ is the location of the Grey Wolf. Equations (8) and (9) can be used to determine the values of $'A'$ and $'C'$:

$$\vec{A} = 2 * \vec{a} * \overrightarrow{r1} - \vec{a} \quad [8]$$
$$\vec{C} = 2 * \overrightarrow{r2} \quad [9]$$

Over multiple cycles, the variable $'a'$ drops gradually from 2 to 0. Random vectors having values between [0, 1] are selected, including $'r1'$ and $'r2'$. Equations 10 to 12 are used to revise the Grey Wolf's position in relation to the location of the prey. The method allows the exploration of multiple areas encircling the optimal searching agents by altering the values of $'A'$ and $'C'$. Various distance vectors and positions are employed in the mathematical formulas required to determine the Grey Wolf's hunting method:

$$\vec{D}_\alpha=\left|C1.\vec{X}_\alpha - \vec{X}(t)\right| \quad [10]$$
$$\vec{D}_\beta = \left|C2.\vec{X}_\beta - \vec{X}(t)\right| \quad [11]$$
$$\vec{D}_\delta=\left|C3.\vec{X}_\delta - \vec{X}(t)\right| \quad [12]$$
$$\vec{X}_1 = \left|\vec{X}_\alpha - A_1\vec{D}_\alpha\right| \quad [13]$$
$$\vec{X}_2 = \left|\vec{X}_\beta - A_2\vec{D}_\beta\right| \quad [14]$$
$$\vec{X}_3 = \left|\vec{X}_\delta - A_3\vec{D}_\delta\right| \quad [15]$$

Here,$\vec{D}, \vec{D}_\beta, \vec{D}_\delta$ represent distance vectors, while $\vec{X}, \vec{X}_\beta, \vec{X}_\delta$ denote position vectors for wolves β and δ. $'A1,' 'A2,' 'A3,' 'C1,' 'C2,'$ and 'C3' are coefficient vectors.

The Grey Wolf's position is updated as follows:

$$\vec{X}_{(t+1)} = \frac{\vec{X}_1+\vec{X}_2+\vec{X}_3}{3} \quad [16]$$

Random values between [-2a, 2a] are determined at random, and the selected value is compared to a threshold. If |A|< 1, it signifies that the wolf should direct its attacks toward the prey. The algorithm investigates the prey's searching and attacking capabilities, and the results guide the movements towards the prey. The population members are required to avoid straying from the prey's location.

The HPSGW technique was created while adhering to the core concepts of PSO and GWO techniques. PSO is highly effective at addressing real-world problems. However, there is a risk of PSO getting trapped in local minima. In our proposed approach, GWO is introduced to support PSO and mitigate the chances of falling into local minima. To prevent such issues, PSO with a low success rate, relocate certain particles to random positions. To address this, the GWO's exploring capacity is used to direct particles to GWO-influenced regions instead of completely random positions. This integration with GWO enhances the efficiency of the algorithm, albeit it may extend the running time. The HPSGW Algorithm, as proposed, amalgamates the functionalities of PSO and GWO. This is achieved through equations (20) and (21), which integrate the PSO and GWO variants to update velocities:

DOI: 10.5123/inj.2023.4.in77
ISSN:2093-4777 | E-ISSN:2093-6931
Vol. 27 Iss. 4 (2023)

$\vec{D}_{\alpha} = |C1.\vec{X}_{\alpha} - w * \vec{X}_{(t)}|$ [17]
$\vec{D}_{\beta} = |C2.\vec{X}_{\beta} - w * \vec{X}_{(t)}|$ [18]
$\vec{D}_{\delta} = |C3.\vec{X}_{\delta} - w * \vec{X}_{(t)}|$ [19]

These equations provide a mechanism to integrate both PSO and GWO, enabling the updating of velocities through equation (20):

$$v_i(t+1) = w * (v_i t + c1r1(x_1 - x_i t) + c2r2(x_2 - x_i t) + c3r3(x_3 - x_i t)) \quad [20]$$
$$x_i(t+1) = x_i t + v_i(t+1) \quad [21]$$

A fitness function is used to identify the optimal values. Using the suggested hybrid optimization approach, the fitness function makes use of the Rosenbrock and the objective function. As indicated in equation (22), the Rosenbrock function is optimized by adapting an appropriate system of coordinates without depending on gradient data or developing localized approximating models.

$$f(x) = \sum_{i=1}^{N-1}[100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2] \quad [22]$$

The main objective of this research is to create a CNN that excels in performance by leveraging hybrid optimization.

Result And Discussion

In this study, we designed and developed a Finger Vein Authentication model using CNNs. To assess the performance of the model, we utilized various benchmark datasets, including SDUMLA, FV–USM, and HKPU, and evaluated key metrics such as Accuracy (%), Equal Error Rate (EER) (%), and Matching Time (s).

SDUMLA Dataset:

We initiated our analysis using the SDUMLA dataset, and the corresponding results are summarized in Table 1. The CNN model achieved an accuracy of 92.54%, with an EER of 4.36% and a matching time of 3.07 seconds. The VGG-16 model demonstrated a slightly higher accuracy at 92.99% but with an EER of 3.89% and a matching time of 4.32 seconds. AlexNet outperformed both, boasting an accuracy of 94.12%, an EER of 3.52%, and a matching time of 4.869 seconds. Notably, the CNN-HPSGW model stood out, attaining an accuracy of 95.87%, the lowest EER of 1.98%, and a matching time of 5.365 seconds. These results highlight the suitability of CNN-HPSGW for the SDUMLA dataset, as it balances high accuracy and relatively low EER.

Table 1. DL model performance on SDUMLA dataset

Model	Accuracy (%)	EER (%)	Matching time (s)
CNN	92.54	4.36	3.07
VGG-16	92.99	3.89	4.32
AlexNet	94.12	3.52	4.869
CNN-HPSGW	95.87	1.98	5.365

FV–USM Dataset:

Secondly, we proceeded with the FV–USM dataset, and the outcomes of our model are provided in Table 2. The CNN model achieved an accuracy of 91.21%, with an EER of 3.878% and a matching time of 3.56 seconds. VGG-16 showed improved accuracy at 93.75% but with an EER of 3.34% and a matching time of 4.65 seconds. AlexNet demonstrated excellent results with an accuracy of 94.58%, an EER of 2.24%, and a matching time of 5.21 seconds. Nevertheless, the CNN-HPSGW model once again shone, with an impressive accuracy of 96.24%, a remarkably low EER of 1.254%, and a matching time of 5.78 seconds. These results reinforce the effectiveness of CNN-HPSGW for the FV–USM dataset, as it excels in both accuracy and EER.

Table 2. DL model performance on FV–USM dataset

Model	Accuracy (%)	EER (%)	Matching time (s)
CNN	91.21	3.878	3.56
VGG-16	93.75	3.34	4.65
AlexNet	94.58	2.24	5.21

DOI: 10.5123/inj.2023.4.in77
ISSN:2093-4777 | E-ISSN:2093-6931
Vol. 27 Iss. 4 (2023)

CNN-HPSGW	96.24	1.254	5.78

HKPU Dataset:
Thirdly, we employed the HKPU dataset, and the results of our model's performance are presented in Table 3. The CNN model achieved an accuracy of 92.83%, with an EER of 4.05% and a matching time of 4.52 seconds. VGG-16 exhibited higher accuracy at 94.05%, but with an EER of 3.45% and a matching time of 4.98 seconds. AlexNet outperformed with an accuracy of 95.83%, a lower EER of 2.02%, and a matching time of 5.68 seconds. The CNN-HPSGW model once again led the pack, boasting the highest accuracy at 97.07%, the lowest EER of 0.84%, and a matching time of 6.12 seconds.

Table 3. DL model performance on HKPU dataset

Model	Accuracy (%)	EER (%)	Matching time (s)
CNN	92.83	4.05	4.52
VGG-16	94.05	3.45	4.98
AlexNet	95.83	2.02	5.68
CNN-HPSGW	97.07	0.84	6.12

Figure 2 illustrates the accuracy comparison of DL models across various datasets. For all datasets, the suggested CNN-HPSGW model consistently delivers the highest accuracy, demonstrating its superior performance in finger vein authentication. Figure 3 depicts the EER comparison of DL models across different datasets. In every case, the CNN-HPSGW model stands out by yielding the lowest EER, emphasizing its reliability in maintaining high authentication accuracy. Finally, Figure 4 provides a time comparison in seconds of DL models on various datasets. While the CNN model excels in minimizing matching time, the recommended CNN-HPSGW model consistently exhibits longer matching times.

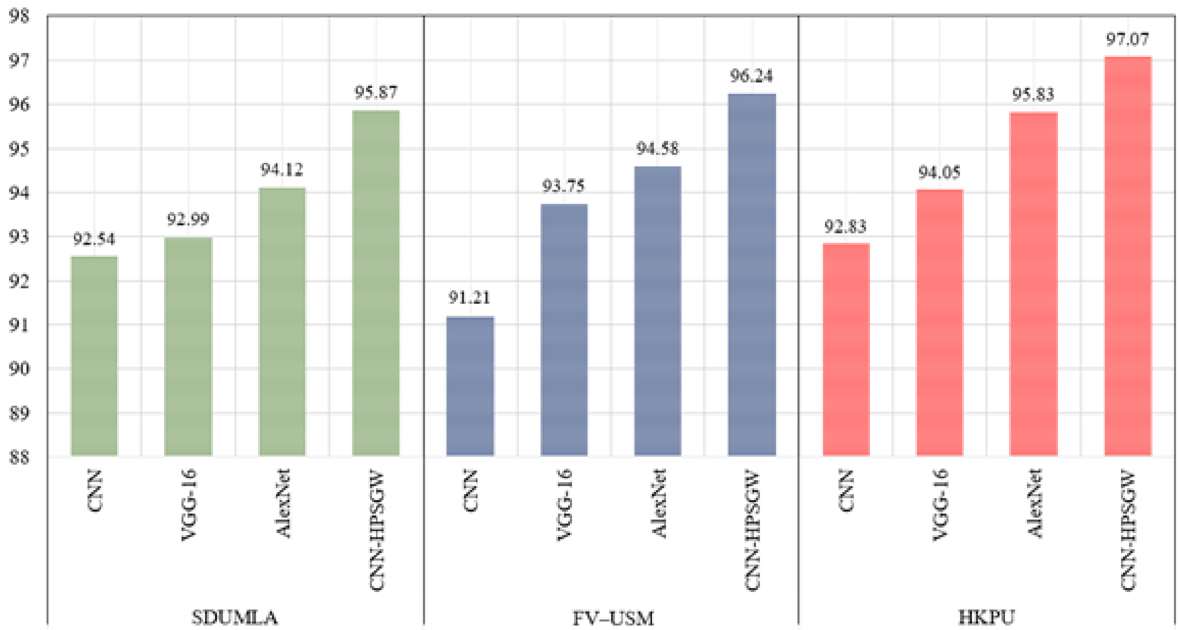

Fig. 2. Accuracy comparison of DL model on various dataset

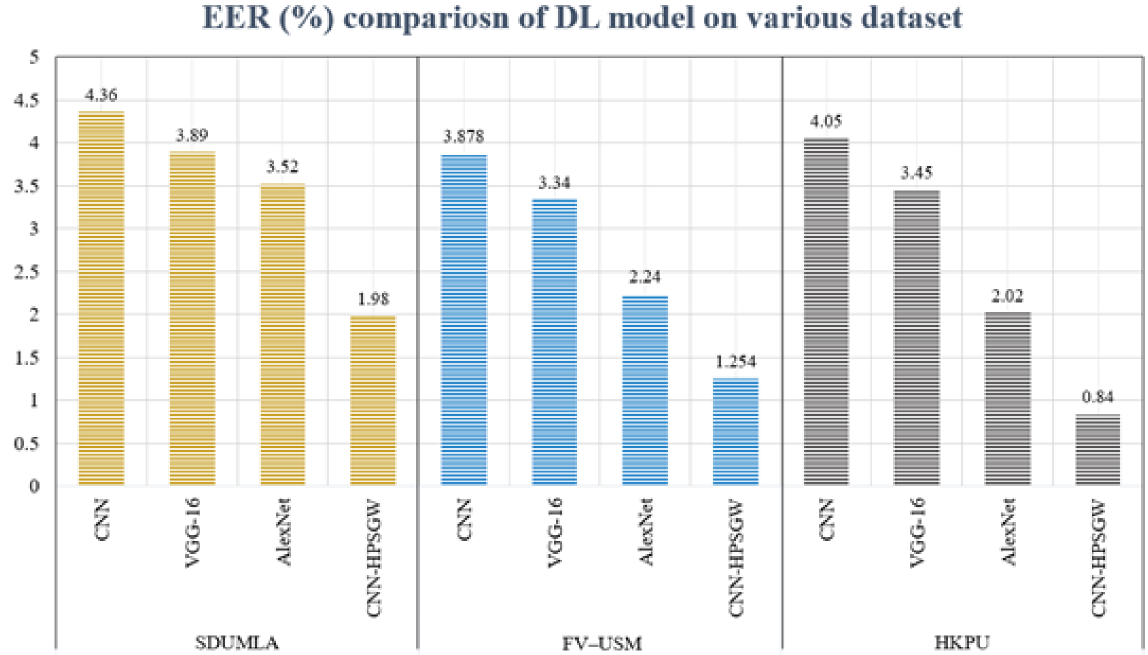

Fig. 3. EER comparison of DL model on various dataset

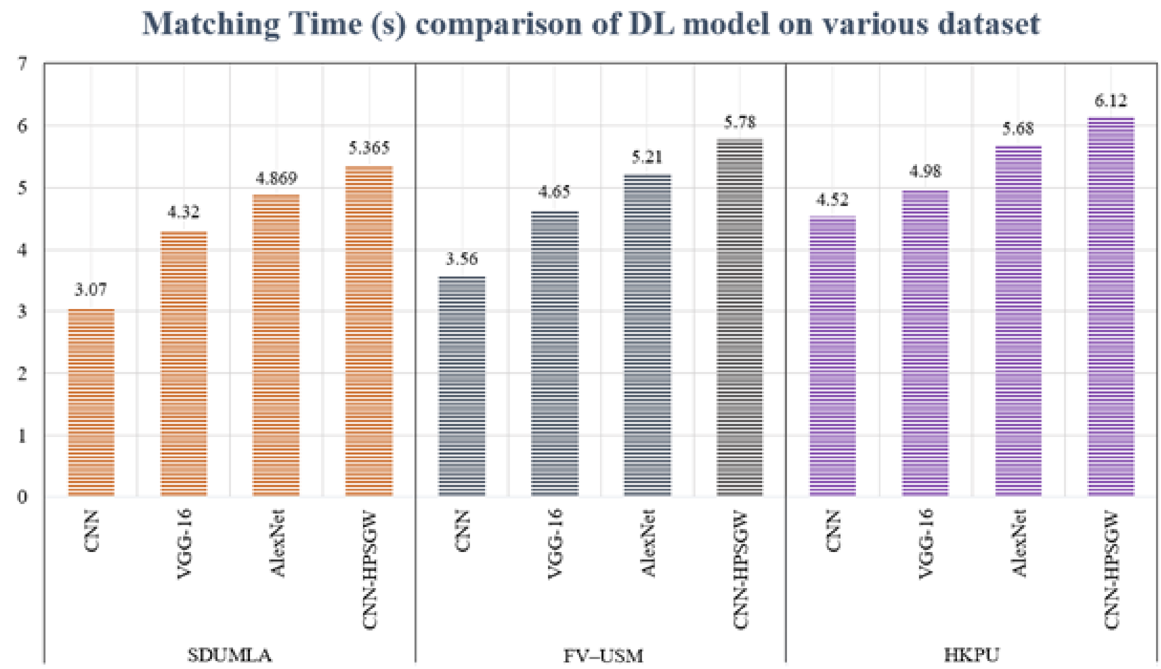

Fig. 4. Matching time comparison of DL model on various dataset

Conclusion

Introducing a novel approach, an optimized CNN (CNN-HPSGW) to significantly enhance the efficiency of existing finger-vein recognition. This model is carefully built by combining three essential parts: finger-vein pre-processing, a deep learning classifier, and comprehensive evaluation. Empirical findings derived from our experiments unequivocally establish the supremacy of the CNN-HPSGW model outlined in this paper. Across all three datasets, this model consistently outperformed the traditional CNN models by delivering the highest levels of accuracy and the most minimal EER. Although there was a marginal increase in the matching time, this was a deliberate trade-off, one that yielded exceptional accuracy and minimized error rates. These findings underscore the applicability and suitability of the CNN-HPSGW model in the realm of finger vein recognition

DOI: 10.5123/inj.2023.4.in77
ISSN:2093-4777 | E-ISSN:2093-6931
Vol. 27 Iss. 4 (2023)

applications, where accuracy and security are very important. In our future research, we are preparing to extend and diversify our recognition system by integrating various biometric parameters alongside finger vein data. This expansion will involve incorporating additional biometric features or characteristics, such as facial recognition, iris scans, or fingerprint data, to create a more comprehensive and robust biometric identification system. The goal is to improve the overall accuracy, security, and reliability of our recognition system by harnessing the power of multiple biometric data sources, making it more versatile and effective in various applications.

References

1. Dargan, Shaveta, and Munish Kumar. "A comprehensive survey on the biometric recognition systems based on physiological and behavioral modalities." Expert Systems with Applications 143 (2020): 113114.
2. Jain, Anil, Lin Hong, and Sharath Pankanti. "Biometric identification." Communications of the ACM 43, no. 2 (2000): 90-98.
3. Phang, Shiau Shing. "Investigating and developing a model for iris changes under varied lighting conditions." PhD diss., Queensland University of Technology, 2007.
4. Anwarul, Shahina, and Susheela Dahiya. "A comprehensive review on face recognition methods and factors affecting facial recognition accuracy." Proceedings of ICRIC 2019: Recent Innovations in Computing (2020): 495-514.
5. Kolivand, Hoshang, Shiva Asadianfam, Kayode Akinlekan Akintoye, and Mohd Shafry Rahim. "Finger vein recognition techniques: a comprehensive review." Multimedia Tools and Applications (2023): 1-35.
6. Nayar, Gayathri R., and Tony Thomas. "Partial palm vein based biometric authentication." Journal of Information Security and Applications 72 (2023): 103390.
7. Zayed, Hossam L., Heba M. Abdel Hamid, Yasser M. Kamal, and Abdel Halim A. Zekry. "A Comprehensive Survey on Finger Vein Biometric." Journal of Advances in Information Technology 14, no. 2 (2023).
8. Boucherit, Ismail, Mohamed Ould Zmirli, Hamza Hentabli, and Bakhtiar Affendi Rosdi. "Finger vein identification using deeply-fused Convolutional Neural Network." Journal of King Saud University-Computer and Information Sciences 34, no. 3 (2022): 646-656.
9. Boucetta, Aldjia, and Leila Boussaad. "Biometric authentication using finger-vein patterns with deep-learning and discriminant correlation analysis." International Journal of Image and Graphics 22, no. 01 (2022): 2250013.
10. Ren, Hengyi, Lijuan Sun, Jian Guo, Chong Han, and Ying Cao. "A high compatibility finger vein image quality assessment system based on deep learning." Expert Systems with Applications 196 (2022): 116603.
11. Madhusudhan, M. V., V. Udayarani, and Chetana Hegde. "An intelligent deep learning LSTM-DM tool for finger vein recognition model USING DSAE classifier." International Journal of System Assurance Engineering and Management (2022): 1-9.
12. El-Rahiem, Basma Abd, Fathi E. Abd El-Samie, and Mohamed Amin. "Multimodal biometric authentication based on deep fusion of electrocardiogram (ECG) and finger vein." Multimedia Systems 28, no. 4 (2022): 1325-1337.
13. Li, Jun, Luokun Yang, Mingquan Ye, Yang Su, and Juntong Liu. "Finger Vein Verification on Different Datasets Based on Deep Learning with Triplet Loss." Computational and Mathematical Methods in Medicine 2022 (2022).
14. Shaheed, Kashif, Aihua Mao, Imran Qureshi, Munish Kumar, Sumaira Hussain, Inam Ullah, and Xingming Zhang. "DS-CNN: A pre-trained Xception model based on depth-wise separable convolutional neural network for finger vein recognition." Expert Systems with Applications 191 (2022): 116288.
15. Fang, Chunxin, Hui Ma, and Jianian Li. "A finger vein authentication method based on the lightweight Siamese network with the self-attention mechanism." Infrared Physics & Technology 128 (2023): 104483.
16. Yin, Y., Liu, L., Sun, X. (2011). SDUMLA-HMT: A Multimodal Biometric Database. In: Sun, Z., Lai, J., Chen, X., Tan, T. (eds) Biometric Recognition. CCBR 2011. Lecture Notes in Computer Science, vol 7098. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-25449-9_33
17. Mohd Asaari, Mohd Shahrimie & Suandi, Shahrel Azmin & Rosdi, Bakhtiar. (2014). Fusion of Band Limited Phase Only Correlation and Width Centroid Contour Distance for finger based biometrics. Expert Systems with Applications. 41. 3367–3382. 10.1016/j.eswa.2013.11.033.

DOI: 10.5123/inj.2023.4.in77

ISSN:2093-4777 | E-ISSN:2093-6931

Vol. 27 Iss. 4 (2023)

18. Ajay Kumar and Yingbo Zhou, “Human Identification using Finger Images”, IEEE Trans. Image Processing, vol. 21, pp. 2228-2244, April 2012.
19. Reza, Ali M. "Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement." Journal of VLSI signal processing systems for signal, image and video technology 38 (2004): 35-44.
20. Movellan, Javier R. "Tutorial on Gabor filters." Open source document 40 (2002): 1-23.
21. Chanda, Bhabatosh. "Morphological algorithms for image processing." IETE Technical Review 25, no. 1 (2008): 9-18.
22. Tammina, Srikanth. "Transfer learning using vgg-16 with deep convolutional neural network for classifying images." International Journal of Scientific and Research Publications (IJSRP) 9, no. 10 (2019): 143-150.
23. Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).
24. Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems 25 (2012).
25. Shaha, Manali, and Meenakshi Pawar. "Transfer learning for image classification." In 2018 second international conference on electronics, communication and aerospace technology (ICECA), pp. 656-660. IEEE, 2018.
26. Deepak, S., and P. M. Ameer. "Automated categorization of brain tumor from mri using cnn features and svm." Journal of Ambient Intelligence and Humanized Computing 12 (2021): 8357-8369.
27. Chauhan, Rahul, Kamal Kumar Ghanshala, and R. C. Joshi. "Convolutional neural network (CNN) for image detection and recognition." In 2018 first international conference on secure cyber computing and communication (ICSCCC), pp. 278-282. IEEE, 2018.
28. Tuba, Eva, Nebojša Bačanin, Ivana Strumberger, and Milan Tuba. "Convolutional neural networks hyperparameters tuning." In Artificial intelligence: theory and applications, pp. 65-84. Cham: Springer International Publishing, 2021.
29. Kennedy, James, and Russell Eberhart. "Particle swarm optimization." In Proceedings of ICNN'95-international conference on neural networks, vol. 4, pp. 1942-1948. IEEE, 1995.
30. Wang, Dongshu, Dapei Tan, and Lei Liu. "Particle swarm optimization algorithm: an overview." Soft computing 22 (2018): 387-408.
31. Mirjalili, S. M. S. M., Seyed Mohammad Mirjalili, and Andrew Lewis. "Grey Wolf Optimizer Adv Eng Softw 69: 46–61." (2014).
32. Mittal, Nitin, Urvinder Singh, and Balwinder Singh Sohi. "Modified grey wolf optimizer for global engineering optimization." Applied Computational Intelligence and Soft Computing 2016 (2016).
33. Teng, Zhi-jun, Jin-ling Lv, and Li-wen Guo. "An improved hybrid grey wolf optimization algorithm." Soft computing 23 (2019): 6617-6631.