



# Finger vein recognition based on PCA and fusion convolutional neural network

Mohammed S. H. Al-Tamimi<sup>a</sup>, Ruaa S. S. AL-Khafaji<sup>a,\*</sup>

<sup>a</sup>Computer Science Department, College of Science, University of Baghdad, Baghdad, Iraq

(Communicated by Madjid Eshaghi Gordji)

---

## Abstract

Finger vein recognition and user identification is a relatively recent biometric recognition technology with a broad variety of applications, and biometric authentication is extensively employed in the information age. As one of the most essential authentication technologies available today, finger vein recognition captures our attention owing to its high level of security, dependability, and track record of performance. Embedded convolutional neural networks are based on the early or intermediate fusing of input. In early fusion, pictures are categorized according to their location in the input space. In this study, we employ a highly optimized network and late fusion rather than early fusion to create a Fusion convolutional neural network that uses two convolutional neural networks (CNNs) in short ways. The technique is based on using two similar CNNs with varying input picture quality, integrating their outputs in a single layer, and employing an optimized CNN design on a proposed Sains University Malaysia (FV-USM) finger vein dataset 5904 images. The final pooling CNN, which is composed of the original picture, an image improved using the contrast limited adaptive histogram (CLAHE) approach and the Median filter, And, using Principal Component Analysis (PCA), we retrieved the features and got an acceptable performance from the FV-USM database, with a recognition rate of 98.53 percent. Our proposed strategy outperformed other strategies described in the literature.

*Keywords:* Finger vein, CNN, CLAHE, Median Filter, PCA, FV-USM.

---

\*Corresponding author

*Email addresses:* [m\\_altamimi75@yahoo.com](mailto:m_altamimi75@yahoo.com) (Mohammed S. H. Al-Tamimi),  
[ruaa.salman1201@sc.uobaghdad.edu.iq](mailto:ruaa.salman1201@sc.uobaghdad.edu.iq) (Ruaa S. S. AL-Khafaji)

*Received:* August 2021    *Accepted:* October 2021

## 1. INTRODUCTION

A biometrics system is essentially a pattern recognition system that identifies a person by determining the reliability of a specific physiological and/or behavioral characteristic that person possesses. Physical biometrics systems rely on the face, hand, iris, and fingerprint, whereas behavioral biometrics systems rely on walking, signature, and pattern recognition.

Speech and keystrokes [1] Vein-based biometric technology was developed for biometric recognition as a more appealing alternative to more traditional methods (such as fingerprints, palm prints, iris, and face) [2]. Whereas fingerprints have demonstrated significant limits with regard to the accuracy, such as degradation of the finger skin, finger surface particles, and so forth... Vein recognition systems have demonstrated excellent and accurate results that can overcome the limitations of fingerprints, due to their unique properties such as (i) preserving the vein is unaffected by human aging, (ii) hand and vein detection systems are not harmful to the user's overall health, and (iii) skin conditions such as uneven skin tone and/or skin burn do not affect vein detection results. (iv) Vein forms are difficult to fool and change, even with surgery, (v) Finger vein patterns are unique to each individual, even identical twins [3, 4]. Thus, it provides a strong contrast between individuals, and (vi) finger veins can only be picked up from a living body, which means that if a person dies, it is impossible to steal their identity; even if a person has a deformed finger, finger veins remain intact, or we can use an alternate finger for verification. These critical characteristics have inspired many to develop recognition technologies to provide secure authentication [3]. Several of these techniques involve preprocessing measures to enhance the image's quality. Three steps comprise the recognition system [5]. Figure 1 illustrates the technique for recognizing finger veins.

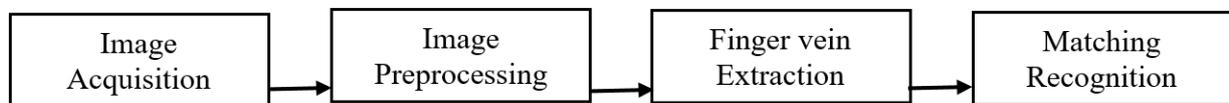


Figure 1: Block Diagram of Digit-Vein Recognition System [6].

Researchers have made significant strides in the detection of finger veins during the last few decades. Mukahar et al.[7] was proposed technique for finger vein detection based on the interval-valued fuzzy sets k-nearest neighbors (IVFKNN) approach, which utilizes fuzzy interval sets to compute instance membership values, allowing for the definition of membership values using a lower and upper bound over time. The training set introduces the notion of membership assignment periods for each condition, and membership is represented as a collection of intervals. The classification accuracy is based on the publicly accessible FV-USM picture database, which contains over 2,952 USM finger vein images. Classification precision is 78.1504. However, it is necessary to apply evolution search techniques to the model as a self-improvement procedure in order to establish the required parameters: the determinant used in the definition of the membership function, which is used to calculate the most significant nearest neighbor voting rule to increase the work's accuracy. S. SHAZEEDA et al. [8] suggested a method based on classification techniques using K-nearest centroid neighbors and sparse representations (k-NCN-SRC), the second step uses using the provided number of centroid nearest neighbors (k) to categorize the test sample. The original data defines a 300 x 100-pixel finger vein picture in the proposed system. Before using classification techniques, images are downsized to a resolution of 30 x 10 pixels. The proposed system is employed in a variety of ways (k) between 100 and 600, with the best result achieved with the suggested system 94.41 at k = 600. This requires more experiments than k = 600 to determine whether accuracy can be improved. B. Affendi et al. [9]. Was presented classification based on mutually sparse representations

(MSRC). MSRC classifies the test sample using a decision procedure that is dependent on not just the dispersed nearest neighbor (NN), but also on the determination of the training sample from which the test sample is drawn NN. If this condition is not fulfilled, the switch operation with the following classes is repeated. The photos are downsized to a dimension of 30 by 10 pixels using this procedure. A total of 2,952 training images and 2,952 test images was analyzed. Although the experiment demonstrated a classification accuracy of 95.73, the overall experimental findings proved that there is no absolute winner capable of providing the highest performance for all data sets in terms of classification accuracy and computing efficiency. Z, Dongdong et al. [10] employed a proposed technique convolutional neural networks (CNNs) with a center loss function to recognize finger veins. CNN is intended to be a competitor to Alex-Net. Experiments demonstrate that the proposed approach is up to 97.95 percent accurate. In FV USM (using just the first session of the experiment), more persuasive assessment criteria must be used since the parameters must be optimized due to their initialization to a random value. H, Ng Tze et al. [11] proposed Adaptive k-Nearest Centroid Neighbor Classifier (a k-NCN) proposes the introduction of rules for adaptively determining the neighborhood size of the test sample. If the nearest neighbor's center point distance is greater than the limit, the test sample size is increased by k. The Image Database Classification incorporates the experimental outcomes (FV-USM). And the proposed system was used to collect a total of 2,952 samples in this experiment. The proposal demonstrates that by comparing a K-NCN to the original k-NCN, it was determined that  $k = 19$  achieves the highest accuracy rate of 85.33 and therefore the neighborhood size, k, was determined for the second experiment akNCN.v2, but the first experiment akNCN.v1 was not reliant on k. In my experience with k-NCN.v1 and akNCN v2. The accuracy was 85.64 and there was no gain in accuracy, however, the time difference was 5,153 s for v2 versus 6,321 s for v1. While the proposed classifier achieves lower classification accuracy than the original k-NCN, it leaves out a great deal of information. This strategy minimizes the quantity of the training data and eliminates templates.

In this work, we present a new approach based on a deep learning model to achieve personalized recognition; The thesis aims to develop a method to support finger vein recognition by developing two important factors, namely time and accuracy, this was done by working in parallel for two different processing processes, one working on noise and one working on contrast optimization of the same image to improve it and speed up the system architecture and implement the process with selection features Best in (PCA) and step (normalizing) in order to maintain system stability. Thus, the results of the models will be combined through the integrated CNN algorithm, A short, multi-layered CNN that and optimized with various parameters, such as activation functions and kernel size. Thus, the basic elements of the architecture are: a convolutional layer, a maximum pooling layer, a dropout layer, a flat layer, and two dense layers. Where by the CNN provides complementary information that contributes to a more accurate determination. Another contribution is a new classification method that improves the accuracy rate with a new decision rule, where the class prediction is based on the file name, and also finds the proximity of each training sample to the test sample based on patterns.

## MATERIALS AND METHODS

### *Proposed system*

The proposed system, the details it contains, and the steps to be followed in the work was shown in Figure 2.

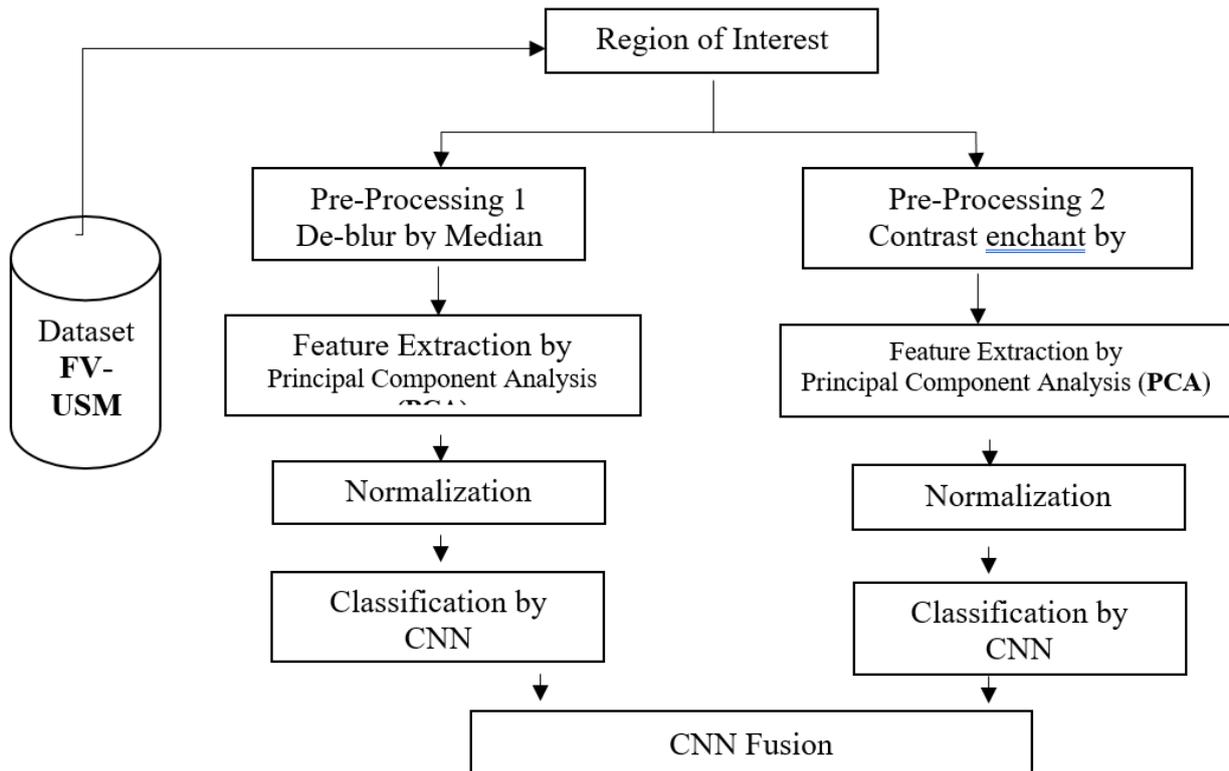


Figure 2: Architecture of the proposed system.

## Dataset

FV-USM database[12]. The FV-USM database was compiled by the University Sains Malaysia. It contains photos of 123 patients' left and right (index and middle) digit veins. There are 83 males and 40 females among them, ranging in age from 20 to 52 years. Photographs were obtained in two separate sessions, each session containing six images of each finger. At a resolution of 640 x 480 pixels, 5904 pictures were obtained from 492 finger categories. The vein image's region of interest (ROI) is 300 by 100 pixels. Two sessions were employed in the proposed system, with photographs from the first session being used for Model 1 and images from the second session being used for Model 2. Several representative photos are included in (Figure 3, a sample of the data set).

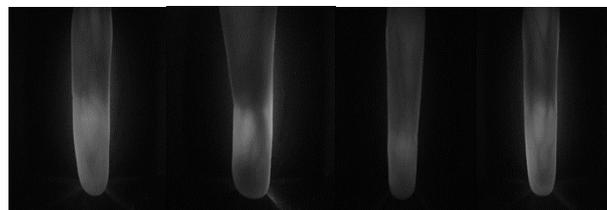


Figure 3: Sample images in the finger vein dataset.

## RESULTS AND DISCUSSION

### ROI

The first and most important stage is to extract the return on investment. In finger vein photographs, there are undesirable parts (the image's background) and a valuable portion (the fingertip

region). The value region is referred to as the ROI [13], & ROI extraction is the process of extracting the fingertip region from a taken image, removing the image's background, and removing any undesired regions. The suggested model utilized ROI photos to enhance our data set by removing noise and gray levels in order to obtain more accurate findings, Figure 4 illustrates the sample of ROI employed in the system.

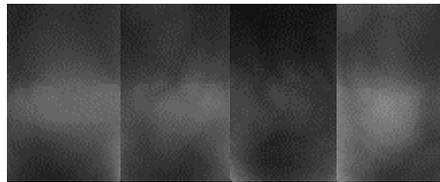


Figure 4: Some ROI images of FV-USM finger vein database.

## Preprocessing

In proposed system used Image information is the main objective of image processing. Because the image resolution varies with different acquisition systems, the images captured include a large background. A finger vein image usually consists of noise, some irregular shadows, areas of high saturation, and low contrast. This is due to the fluctuation of light and the performance of the capture device, it is necessary to address it[14]. In general, we used the optimization method to get good performance up front by processing images from contrast and blur in each model in parallel. In our work, we loaded images obtained from image processing for the region of interest. Then the image was processed in two ways: the first method is CLAHE [15], and the second method is the median filter[16]., parallel to the 2 models, and the flowing points displaying the two step of pre-prosing system:

- **Contrast Enhancement:** The most common method for improving image contrast and for adjusting image brightness to resolve differences in lighting and to solve the problem of over-optimization in the normal histogram equation is the CLAHE, which was originally applied to improve low-contrast medical images in proposed system the CLAHE works on small areas of the image called squares rather than the entire image. CLAHE has two main parameters: block size ( $BS = (8 * 8)$ ) and segment limit ( $CL = 4.0$ ). These two parameters mainly control the improved image quality, and have stander steps in CLAHE as follows:
  - I. Divided into small regions
  - II. Histogram accounts for those regions
  - III. Determine the clip limit and compare it with the histogram.
  - IV. Redistribute indirect hyper values to graph bins

Figure 5. Which shows the shape of the image before and after contrast enhancement.

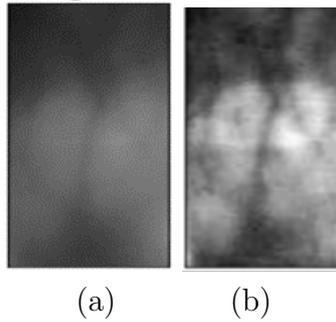


Figure 5: Finger Vein image enhanced and its related histogram. Original Images, (B) The image after applying CLAHE.

And the following algorithm is CLAHE in the proposed system:

**Algorithm 1: Contrast-Limited Adaptive Histogram Equalization (CLAHE)**

**Input:** original image

**Output:** contrast-enhancement image and keep the edgeVariable: I, J: integer

- Begin
- For I =1 to N
- For J=1 to M
- Divided into small region.
- Compute the histogram using equation  $S_k = T(r_k) = \sum_{j=1}^k pr(r_j) \sum_{j=1}^k \frac{r_j}{n}$
- Calculate the excess using equation  $Clp = \lfloor \frac{w}{256} \rfloor + \lfloor \beta \cdot (w - \lfloor w/256 \rfloor) \rfloor$
- Reallocate pixel values, evenly distribute the cut-off pixel values using equation
- $cdf(X_k) = \sum_{j=0}^k p(x_j)$  for  $k = 0, 1, 2, 3, \dots, L - 1$
- End.

### Median Filtering

The median filter is the most well-known nonlinear filter due to its ability to remove salt or pepper noise (pulse noise) with significantly less blur than linear smoothing filters, while simultaneously keeping edges and visual features [16]. A strategy for improving the quality of vein images through opacity has been offered, and this work seeks to identify an appropriate solution for image noise. To successfully deal with a light scattering in finger vein imaging, the fuzzy image is processed using a median filter. The filter used in this study is the filter size (3 x 3) (using equation (6), this method is effective when the average filter window size is not large because there is a critical problem if choosing a large filter size is to get rid of a small structure, it is part of the image because it preserves better details such as corners and fine lines, and the purpose of the filter is to use the average value of neighboring pixels, including itself, as a substitute for the pixel in the input image Figure 6.

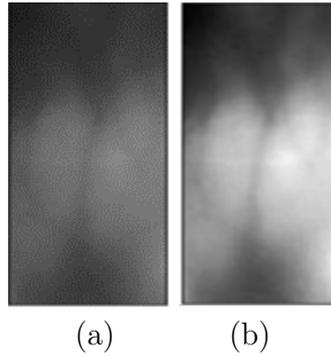


Figure 6: De-blur operation from image. (a) Original Images, (b) The image after applying Median Filter.

And the next algorithm is De-blur by median filter:

<b>Algorithm 2: Median filter</b>
<b>Input: Image (ROI) Session 2</b>
<b>Output: New image</b>
<b>Variable: x, y, I: integer, fx,fy</b>
<pre> Begin   allocate output Pixel Value  //(image width)(image height)//   allocate window              //(window width × window height)//   edge x := (window width / 2) rounded down   edge y := (window height / 2) rounded down   for x from edge x to image width - edge x do     for y from edge y to image height - edge y do       I = 0       for fx from 0 to window width do         for fy from 0 to window height do           window[i] := input Pixel Value[x + fx - edge x][y + fy - edge y]           I := I + 1         sort entries in window[]         output Pixel Value[x][y] := window[window width * window height]       End. </pre>

### Feature Extraction using PCA reeducation

PCA is an unsupervised statistical technique that utilizes data sets to extract useful features and information. As a result, it is one of the most widely used approaches in multivariate statistics, particularly when big data sets are analyzed. Additionally, it is capable of reducing the feature dimension without sacrificing much information [17]. A principal component analysis is an extremely strong technique for analyzing data, discovering patterns, and displaying data in ways that emphasize distinctions. The variance of the data set is described. PCA prioritizes the features that contribute to the value and accuracy of our training process. We obtain eigen veins in the proposed system

by using PCA. According to experience, the optimal number of feature values for veins while using PCA is seven, and the following algorithm demonstrates how PCA works in the system:

<b>Algorithm 3: PCA</b>
<b>Input : Image</b>
<b>Output: PCA set of most significant features</b>
<b>Variable: C<sub>x</sub>, AA ,I, i</b>
<ul style="list-style-type: none"> <li>• Begin</li> <li>• Input image</li> <li>• Compute the mean of data according to equation: <math>M_x = E\{X_i, \dots, x_n\}</math></li> <li>• Subtract the mean from each data to calculate the matrix of covariance according to Equation: <math>C_x = E\{(X - M_x)(X - M_x)^T\}</math></li> <li>• Calculate the eigenvectors of <math>C_x</math> and sort the eigenvectors in Descending order according to the equation: <math>y = A(X - M_x)</math></li> <li>• Compress feature vector for each image according to equation: <math>x = A^T \cdot y</math></li> <li>• End</li> </ul>

### Data Normalization

The majority of recognition systems employ some form of normalization. It is the process of adjusting the range of pixel density values in an image to a more uniform value. The image of the finger vein is normalized to account for geometrical variations and to maintain a constant image size. A data set's property is normalized by rescaling its values to fit inside a given narrow range, such as (0 to 1). Numerous normalization strategies are available, including the z-score and the max-min method [18]. Maximum-minimum was employed in the proposed system because (The min-max normalization method ensures that all features will have exactly the same scale, normalization for min-max maintains relationships between original data values) When features are utilized without normalization, larger features can take care of the classifier's cost function and prevent the other features from being "learned" (which are smaller in scale). The following equation is used to normalize features, and the flowing procedure illustrates the normalization process.

<b>Algorithm 4: .Normalization</b>
<b>Input : The most significant features</b>
<b>Output: Normalize smooth feature</b>
<b>Variable : I, integer</b>
<ul style="list-style-type: none"> <li>• Begin <ul style="list-style-type: none"> <li>For I =1 to N</li> <li>Compute max-min normalization using the equation <math display="block">X = \frac{X - \text{Min}(x)}{\text{Max}(x) - \text{Min}(x)}</math> </li> </ul> </li> <li>• End.</li> </ul>

### Classification using CNN

By utilizing an end-to-end method, systems based on deep learning have supplanted conventional systems. Additionally, CNN is distinguished by its precision and exceptional speed when compared to conventional algorithms [19]. CNN is a multi-layer private network (MLP). It is composed primarily of a series of filters that are applied to various areas within the input data to generate the output map. CNN networks are typically composed of three major layers: convolutional layers (input layer), pooling layers (hidden layer), and fully connected layers (output layer) [20]. Fusion CNNs are constructed using information fusion techniques such as early fusion, middle fusion, or late fusion. This architecture has been demonstrated to improve recognition rates when compared to conventional CNNs. Concatenation of images at the input space is used in early fusion. In contrast, in late fusion, each picture is analyzed independently by a CNN and then concatenated into a single vector at the outputs. Numerous investigations have established that late fusion is superior to early fusion [21]. The task is divided into two categories in the suggested system ("instead of modifying the CNN, we modified the input images and kept the same CNN"). As a result, the output vector contained information that was different from the information obtained from the same finger vein. In accordance with the tenfold principle, samples were drawn from the database in an 8:2 ratio for training and testing purposes. And, in order to comprehend the operation of CNN in both models, the following is the CNN organizational structure:

- ✓ For calculating the output, a convolution layer using the Relu activation function is employed.
- ✓ Max-pooling layer,
- ✓ Dropout layer,
- ✓ Flatten layer,
- ✓ Dense layers, with a Relu activation function for computing the output, and a Relu activation function
- ✓ # Fused Layer# Dense layers, with softmax activation function for calculating the output

We've constructed a condensed, multi-layered CNN. The suggested CNN diagram for the proposed vein identification system for the selected data set is depicted in Figure 7.

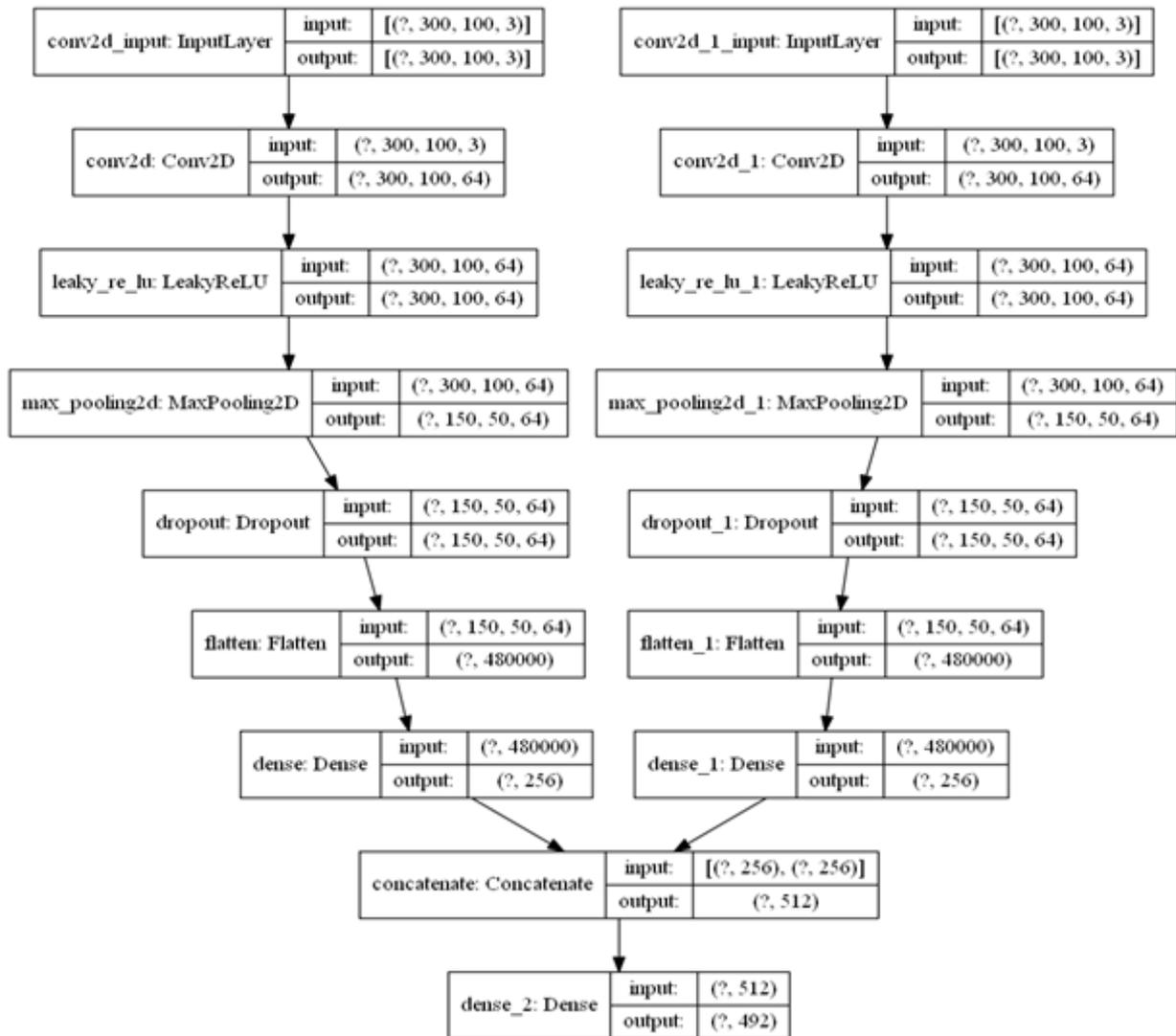


Figure 7: The architecture of the proposed CNN.

### Optimum CNN parameters

The input picture in this part was  $300 * 100$  pixels in size. We employed a CNN with a convolutional (Conv2D) layer in the experiment. It's similar to a series of programmable filters. For the two first conv2D layers, I chose to set 64 filters. Each filter applies the kernel filter= on a subset of the picture (specified by the kernel size) (3, 3). On the entire picture, the kernel filter matrix is applied. Filters may be thought of as picture transformations. The CNN can extract important information from these modified pictures (feature maps). The pooling (MaxPool2D) layer is the second critical layer in CNN. This layer is nothing more than a downsampling filter. It examines the two adjacent pixels and chooses the maximum value. These are used to lower the computing cost of the algorithm and, to a lesser extent, over fitting. We must pick a pooling size of (2, 2); the larger the pooling dimension, the more critical down sampling is. By combining convolutional and pooling layers, CNNs can integrate local features and learn about the image's global properties. Dropout is a regularization technique

in which a subset of nodes in the layer is randomly disregarded (their weights are set to 0.5) for each training sample. This randomly loses a subset of the network's members and causes the network to learn features in a scattered fashion. Additionally, this strategy enhances generalization and minimizes over fitting. 'relu' denotes the rectifier (maximum activation function)  $(0, x)$ . The rectifier activation function is utilized to give the network nonlinearity. The Flatten layer is used to combine all of the final feature mappings into a single one-dimensional vector. This flattening phase is required to utilize fully linked layers after some convolutional/max-pool layers. It incorporates all of the previously discovered local characteristics from the convolutional layers. Finally, we combined the characteristics into a single dense (completely connected) layer, which is just an artificial neural network (ANN) classifier. The last layers are smooth, flat, thick, leaky, and flat. And in the suggested system, we will demonstrate in detail how the system will merge the two CNN and why we chose this direction:

Step1: load image from the dataset

Step: ROI of two session

Step3: preprocessing by using CLAHE in model one and Median filter on model two in a parallel way.

Step4: feature extraction by using PCA in two models.

Step5: normalize using the max-min algorithm in two models.

Step 6: classification by using CNN in two models.

Step 7: merge the result of the two models.

Here is a scenario of how the step works with the algorithm to show how the proposed system works: First, an image is called from the database and ROI is taken from it, as shown above, then we make two different currencies, CLAHE, it is intended to improve image contrast and a field filter and it works On removing image blurring. As a result of these two steps working in parallel with the factor of time and accuracy, the result will be a clear picture. The opposite of the origin whose meanings are ambiguous, then feature extraction works through the PCA that was explained earlier, its purpose is to extract the feature and rehabilitate based on which the process of matching, selection, or classification will be taken and then the normalization process was used and the purpose of which was to avoid mutations and train CNN. In a controlled manner, then a short and optimized 2CNN was used with different standards and they have the same structure but the difference was in the inputs and therefore the outputs will be different and as a result of this difference we combine them trained on FV-USM database to get the best ratios and correct identification. In the end, we will get a structure with a new suggestion, and as a result of what was mentioned earlier, matching or non-conforming images will be selected. And the following algorithm shows CNN working in the system:

**Algorithm 5: Proposed CNN****Input: 3D Data (( Normal and Smooth Data ))****Output: Vein Class Variable: I, J: integer**

- Begin
- For I = 1 to N \\ Building left Model\\
- Read 3D image to 2D
  - //Calculate the convolutional //
  - model1.add(Conv2D(64, kernel\_size=(3, 3),padding = 'same',
  - activation= activ, kernel\_initializer=init))
- \\\*Calculate the max pool\*\\
  - model1.add(MaxPool2D(pool\_size = (2, 2)))
- \\\*Calculate the Dropout at dense layer\*\\
  - model1.add(Dropout(0.5))
- For J= 1 to N \\ Building right Model\\
- Read 3D image
- \\\*Calculate the convolutional \*\\
  - model1.add(Conv2D(64, kernel\_size=(3, 3),padding = 'same',
  - activation= activ, kernel\_initializer=init))
- \\\*Calculate the max pool\*\\
  - model1.add(MaxPool2D(pool\_size = (2, 2)))
- \\\*Calculate the Dropout at dense layer\*\\
  - model1.add(Dropout(0.5))
- \\\*Calculate the Fused Layer\*\\
  - Fused = Concatenate()([left, ) right)
  - Fused = concatenate([model1.output, model2.output], axis=-1)
  - Fused = Dense(1024, activation='softmax')( fused)
  - Fused = Dense(num\_classes, activation='softmax')( fused)
- \\\*Calculate the Build Model\*\\
  - model = Model(inputs = [model1.input, model2.input], outputs = fused)
  - model = Sequential()
  - model.add(fused)
  - model.add(Dense(1024, activation='relu', kernel\_initializer=init))
  - model.add(Dense(num\_classes 492, activation='softmax'))
- End

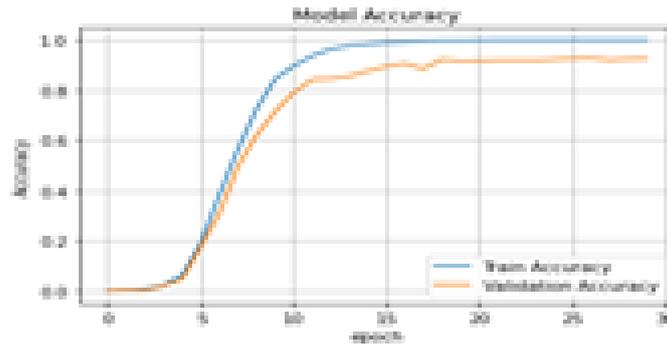


Figure 8: Trend of accuracy

## Evaluation of the Result

The assessment recommended a method through the use of a matrix of confusion[22]. This summarizes the number of examples that a classification model is correctly actually plagiarized or incorrectly predicted. The following terminology is frequently used to refer to the counts reported in a confusion matrix:

- TP and TN denote the True Positive and True Negative states, respectively, as well as the fraction of positive and negative states that were properly categorized.
- False Positive refers to all negative statuses that were wrongly classed as positive, and False Negative refers to all positive statuses that were incorrectly classified as negative.

## Accuracy (AC)

Accuracy measures the classifier's capability to produce the level of accurate diagnosis[23]. Equation (1.1) shows the accuracy formula.

$$\text{Accuracy} = \frac{\text{no. of correctly classification image}}{\text{Total no. of image}} * 100 \quad (1.1)$$

And bay impalement the above the equation of accuracy is:

$$\text{Accuracy} = 98.53$$

And the fowling table1. Show the result above the Actual = 5815 images and the predict = 89 images. And the following figure 7 shows the accuracy of the model, which is 98.53.

## Results with previous studies

Refers	Dataset	Methods	Result
(proposed system)	<b>FV-USM 5904</b>	<b>Fusion CNN</b>	<b>98.53</b>
2017[7]	FV-USM 5904	IVFKNN	78.15
2018[8]	FV-USM 5904	KNCN-SRC	94.41
	FV-USM 5904	MSRC	95.7
2020 (core paper ) [10]	FV-USM 2952	CNN	97.95
2019[9] 2021[11]	FV-USM 2952	KNCN	85.64

## CONCLUSION

A method based on a deep learning model for achieving individualized recognition of finger vein patterns was developed in this study. The primary goal of this suggested model is to boost the recognition rate by training the entire network with a tiny CNN. The superiority of CNN is due to its ability to discriminate between training and test data. The databases for finger vein detection are limited, and the vein structures are basic witches that impact the training process. Embedded CNN is a relatively new design that employs several networks to execute distinct jobs. Rather than employing a single picture in a multi-network model, we employed the same architecture to integrate the outputs of many finger vein images in this investigation. There are no hard and fast guidelines for developing a CNN model. We have included apparently irreversible choices such as activation function, filter size, and layer count. We examined our architecture's generalization performance and discovered considerable improvements. The model was validated for use with a public database. In comparison to a single CNN, the combined CNN gives supplementary information that aids in the identification process. Our strategy is cost effective and effective.

## References

- [1] D. Maltoni, D. Maio, A.K. Jain and S. Prabhakar, *Handbook of Fingerprint Recognition*, Springer Science & Business Media, 2009.
- [2] A. K. Jain, A.A. Ross and K. Nandakumar, *Introduction to Biometrics*, Springer Science & Business Media, 2011.
- [3] K. Syazana-Itqan, A.R. Syafeeza, N.M. Saad, N.A. Hamid and W. Saad, *A review of finger-vein biometrics identification approaches*, Indian J. Sci. Technol. 9(32) (2016) 1–9.
- [4] S. Varastehpour, H. Sharifzadeh, I. Ardekani and H. Sarrafzadeh, *Human Biometric Traits: A Systematic Review Focusing on Vascular Patterns*, 2020.
- [5] A.N. Hoshyar and R. Sulaiman, *Review on finger vein authentication system by applying neural network*, Int. Symp. Inf. Technol. 2010, pp. 1020–1023.
- [6] E. Ting and M.Z. Ibrahim, *A review of finger vein recognition system*, J. Telecommun. Electron. Comput. Eng. 10(1–9) (2018) 167–171.
- [7] N. Mukahar and B. A. Rosdi, *Interval valued fuzzy sets k-nearest neighbors classifier for finger vein recognition*, J. Phys. Conf. Ser. 890(1) (2017) 12069.
- [8] S. Shazeeda and B.A. Rosdi, *Nearest centroid neighbor based sparse representation classification for finger vein recognition*, IEEE Access 7 (2018) 5874–5885.
- [9] S. Shazeeda and B.A. Rosdi, *Finger vein recognition using mutual sparse representation classification*, IET Biometrics 8(1) (2019) 49–58.
- [10] D. Zhao, H. Ma, Z. Yang, J. Li and W. Tian, *Finger vein recognition based on lightweight CNN combining center loss and dynamic regularization*, Infrared Phys. Technol. 105 (2020) 103221.

- [11] B.A. Rosdi, N. Mukahar and N.T. Han, *Finger Vein Recognition Using Principle Component Analysis and Adaptive k-Nearest Centroid Neighbor Classifier*, Int. J. Integr. Eng. 13(1) (2021) 177–187.
- [12] M.S.M. Asaari, S.A. Suandi and B.A. Rosdi, *Fusion of band limited phase only correlation and width centroid contour distance for finger based biometrics*, Expert Syst. Appl. 41(7) (2014) 3367–3382.
- [13] I. Takawale, T. Garud, S. Ingale, N. Udgaonkar, N. Date and S. Jadhav, *Finger vein authentication system using convolutional neural network*, Int. J. Res. Appl. Sci. Eng. Technol. 8(1) 676–682.
- [14] K.A. Akintoye, M.R.M. Shafry and H. Abdullah, *A novel approach for finger vein pattern enhancement using Gabor and Canny edge detector*, Int. J. Comput. Appl. 157(2) (2017).
- [15] J. Ma, X. Fan, S.X. Yang, X. Zhang and X. Zhu, *Contrast limited adaptive histogram equalization based fusion for underwater image enhancement*, Preprints, (2017) 1-27.
- [16] N. Marturi, *Vision and Visual Servoing for Nanomanipulation and Nanocharacterization in Scanning Electron Microscope*, Université de Franche-Comté, 2013.
- [17] T.S. Beng and B.A. Rosdi, *Finger-vein identification using pattern map and principal component analysis*, IEEE Int. Conf. Signal Image Process. Appl. 2011, pp. 530–534.
- [18] M. Kaufmann, J. Han and J. Pei, *Data Mining Concepts and Techniques*, Cyber Secur. Technol. 6(2) (2012).
- [19] W. Liu, W. Li, L. Sun, L. Zhang and P. Chen, *Finger vein recognition based on deep learning*, 12th IEEE Conf. Indust. Electron. Appl. 2017, pp. 205–210.
- [20] Y. LeCun, K. Kavukcuoglu and C. Farabet, *Convolutional networks and applications in vision*, Proc. 2010 IEEE Int. Symp. Circuits Syst. 2010, pp. 253–256.
- [21] N. Ketkar and E. Santana, *Deep Learning with Python*, vol. 1. Springer, 2017.
- [22] A.K. Santra and C.J. Christy, *Genetic algorithm and confusion matrix for document clustering*, Int. J. Comput. Sci. 9(1) (2012) 322.
- [23] A.E. Hassaniien, R. Bhatnagar and A. Darwish, *Artificial Intelligence for Sustainable Development: Theory, Practice and Future Applications*, vol. 912. Springer Nature, 2020.