ISSN: 2393-8048
International Advance Journal of Engineering, Science and Management (IAJESM)
Multidisciplinary, Multilingual, Indexed, Double Blind, Open Access, Peer-Reviewed, Refereed-International Journal.
SJIF Impact Factor =8.152, January-June 2025, Submitted in April 2025

Deep Learning Based Finger Vein Recognition with Effective Optimization and Residual Feature Exaction Strategy

Mr. S. V. Deshmukh, Department of Computer Science, College of Computer Science and Information Technology (COCSIT), Latur, Maharashtra, India deshmukhsv3069@gmail.com
Dr. N. S. Zulpe, Department of Computer Science, College of Computer Science and Information Technology (COCSIT), Latur, Maharashtra, India nitishzulpe@gmail.com

Abstract

Finger vein recognition is an advanced biometric modality which enhance the reliable, emerging and security. Several finger vein recognition system were developed to provide more security. But, the efficiency of finger vein recognition have some issues such as poor performance, recognition interference and so on. Hence, the proposed methodology improved to overcomes these issues and enhance the performance. Initially, per-processing the data which is gathered from finger vein dataset. The Gaussian Bilateral filter (Bi-GLa) utilized for noise removal and Clipping Intensity Quadrant Histogram equalization (CliqHE) used to enhance the contrast of image. Then, the Integrated dense residual network (InDeRn) model employed to extract the data. It done by extracting the feature from two models and then it fused together to provide efficient feature. Then, the Integrated Gated convolutional bidirectional LSTM (InGC-BiLstm) model is developed to recognise the finger-vein as Genuine user or imposter. Then, the hyperparameter utilized in the classification model are tuned by Hybrid levy dwarf mangoose optimizer (LeDMO). The efficiency of proposed technique determined by comparing it with other related techniques. The accuracy of proposed technique is obtained to be 99.60% which is higher than other technique. The False Rejection rate (FRR), False Acceptance rate (FAR) and Genuine Acceptance Rate (GAR) are obtained to be 0.0039, 0.107 and 0.26 7. These analysed determines the efficiency of proposed technique is improved its performance than other existing technique.

Keywords: Finger Vein Recognition, Bilateral Filter, Histogram Equalization, Dense Network, Residual Network, Bi-LSTM.

1. Introduction

Automated personal identification using vascular biometrics such as finger vein images, which is widely used in many applications like e-governance and e0busioness applications. Finger vein recognition is an important mechanism to identify the personal identity based on exact patterns of finger veins in each person fingers [1]. In real-time, physical traits or behaviours of the person can be identified by bio-metric authentication system. Bio-metric authentication system has two models of operations such as identification and verification [2]. The hand vein recognition can be performed by collecting different vein images from the dataset and enhanced by different filtering techniques for better performances. The major advantages of finger vein images identification is high accuracy, avoid spoofing attack and provide more security [3, 4]. There are several bio metric signatures such as fingerprint, face, iris and hand geometry are used to control access and user authentication in security systems [5].

The finger veins models has been divided into two main categories such as learning and non-learning models. In learning models, the features are extracted with the help of Gabor filter and non-learning models has some issues like finger-vein pictures quality and texture deletion [6]. Biometric recognition is an important technology to provide security, which is more reliable and secure than other existing models. Bio-metrics technology based on finger vein images are very popular in both industrial and education system due to its robustness of the models [7, 8]. Several Deep learning models (DL) are developed such as convolutional neural network (CNN), recurrent neural network (RNN), artificial neural network (ANN) and so on. In existing research works, the finger vein recognition can be performed by the combination of convolutional RNN model with finger vein image dataset

[9].

The recognition of finger veins has been performed by deep generalized label finger vein (DGLFV) to gather feature maps, which helps to enhance the accuracy of the model [10]. Finger vein recognition is performed with enhanced images, thus the image enhancement can be performed by Contrast limited adaptive histogram equalization (CLAHE) and median filter are applied to the images. CNN model with transfer learning is used to recognise finger veins effectively [11, 12]. CNN classifier model used loss function like triplet loss function with hard triplet online selection for finger vein recognition [13]. Finger vein recognition can be performed by following stages like pre-processing for image enhancement, feature extraction and selection of features to enhance the classifiers model performances and finally, the classifier model is used to recognize the finger vein accurately [14].

In previous research work, the features are extracted for finger knuckle print (FKP) and finger vein (FV) using AlexNet, VGG 16 and ResNet 50 models for effective enhancement in classification phase [15]. Based on shape and texture features in finger vein images are classified by using densely connected CNN model was performed by two different datasets such as Shandong university homologous multi-modal traits finger vein datasets and hong-kong university finger vein database [16]. Vascular biometrics recognition (VBR) can be designed on a smartphone with DL model. Here, the features can be extracted by CNN model for classification of finger veins [17]. The vascular biometrics are used for automatic person identification and a robust finger vein identification model has been developed with CNN model and the finger vein images can be captured with the help of near infrared (NIR) light [18]. The vein texture features in the input are gathered by Fully CNN model with conditional random field (CRF)and the hybrid RNN and Residual network k is used to recognise the finger veins based on the collected features [19, 20]. Finger vein recognition is an important mechanism for bio-metric technology, which is used in both industrial and education areas. In existing research analysis, several classification models were developed such as CNN, SVM and other transformer based models. These models has some issues like less classification accuracy, the robustness of the model is very less, the hyperparameters are not tuned properly which leads to reduce the efficiency of the modes. In addition to that, the existing models use less number of datasets and parametric analysis and the complexity of the model was very high, which results in very less amount of performances and consumption of energy is very high. In order to suppress these issues, an efficient DL based Finger vein recognition using effective optimization and feature extraction strategy. The major contribution of proposed technique described as follows.

- To pre-process the image to improve its quality, the Bi-GLa and CliqHE utilized for noise removal and contrast enhancement.
- To extract finger vein features from the input images using Integrated dense residual network (InDeRn).
- To recognize finger vein based on optimal features using Integrated Convolutional Gated bidirectional LSTM (InGC-BiLstm) and the hyperparameters are tuned by Hybrid levy dwarf mongoose optimizer (LeDMO), which improves classification accuracy.
- To verify the efficient performance of proposed technique, comparison would be conducted with several related techniques.

The remaining content of this research work has been organized in the following manner: Section 2 includes the survey of various related technique with its limitation and performance. In Section 3, the details and function of proposed methodology described briefly with Figures and equation. Then, the results are obtained and discussed in Section 4 and overall conclusion of the paper given in Section 5.

2. Related works

Survey of some related techniques over finger vein recognition described briefly as follows.

Zhang et al. [21] suggested a lightweight convolutional attention model (LCAModel) was used for finger vein recognition to attain better accuracy. The finger vein features in the images were extracted by attention and convolutional models. These attention and convolutional models based features were fused in to a single features and passed in to classification phase. Here, Support vector machine (SVM) models was used to classify the finger vein images effectively. The performances such as accuracy, loss curve and time consumption. Single dataset was used in this model, thus the model had less accuracy and less robustness were the major limitations.
Safie et al. [22] developed a multiple clip contrast limited adaptive histogram equalization (MC-CLAHE) to enhance the finger images before passed in to classifier models. The classification of finger vein images were classified by pre-trained CNN model called AlexNet model for both feature extraction and classification. The dataset was collected from open source namely FV-USM dataset for finger vein recognition. The performances such as false rejection rate (FRR), false acceptance rate (FAR) and ROC curve were analysed for this model. The limitations of the models was less accuracy has attain in classification phase due to the traditional classifier model.
Htet et al. [23] suggested a segmentation and recognition model for palm finger vein to enhance identification and authentication performances. Here, the U-Net model was used for segmentation of images and recognition finger vein images using attention based residual networks. Here, the attention models helps to remove irrelevant features from the input images. The performances such as accuracy, precision, recall, intersection over union (IoU), dice similarity coefficient (DSC) and f1-score were analysed for this model. The overfitting problems were occurred due to class imbalance problem was one of the limitations in this model.
Lian et al. [24] suggested a federated learning based finger vein authentication framework (FedFV) to reduce the issues like small sample sizes. The non-independent identically distribution problems was solved by federated weighted proportion reduction (FedWPR), which helps to achieve better performances. The performances such as FRR and FAR for this model by using the dataset namely finger vein dataset. Very less number of parameters were analysed in this models was one of the limitations.
Li et al. [25] suggested a vision transformer model for finger vein recognition (ViT-FV) model to enhance the performances than other models. Here the dataset was collected from two different source namely, FV-USM and SDUMLA-HMT for finger vein recognition. ViT-FV model contains three major parts such as patch and position embedding, transformer encoder and regularization multi-layer perceptron (regMLP). This model didn't able to focus on open protocols, which was used for real life. It was one of the limitations in this model. The performance analysis such as techniques, dataset, demerits and parameter analysis for existing research works is described in Table 1.

Table 1: Performance analysis

Author name and references	Techniques	Dataset	Demerits	Performances
Zhang et al. [21]	LCAModel used for finger vein recognition	Real-time	Less accuracy and Less robustness	Accuracy Curve Loss curve Time consumption
Safie et al. [22]	MC-CLAHE to enhance the finger images and AlexNet model for both feature extraction and classification	FV-USM	Less accuracy has attain in classification phase	FRR FAR ROC
Htet et al. [23]	Recognition finger vein images using	Real-time	Overfitting problems were	Accuracy Precision

	attention based residual networks		occurred due to class imbalance problem	Recall IoU DSC F1-score
Lian et al. [24]	FedFV used for finger vein recognition	Finger vein dataset	Very less number of parameters were analysed	FRR FAR
Li et al. [25]	ViT-FV	FV-USM and SDUMLA-HMT	Didn't able to focus on open protocols	Accuracy

Based on the survey of related techniques, some of issues experienced with these models include low classification accuracy, very low model robustness, and poorly tuned hyperparameters that lower mode efficiency. Furthermore, the current models consume a great deal of energy and produce very few performances since they employ fewer datasets and parametric analysis, and their complexity is very high. Some of model consumes high computational time to training the data, low quality of data also reduce the prediction. The performance of few techniques were reduced, due to the imbalance class issues in dataset.

3. Proposed methodology

In this research work, a novel finger vein recognition for bio-metric technology using effective deep learning model with optimization strategy. Initially, the input images are collected from dataset and passed in to pre-processing stage for noise removal and contrast enhancement. Bi-GLa is used to remove noise and blur in the input images and the contrast of the images can be enhanced by using CliqHE. The relevant finger vein features are exacted from pre-processed data using InDeRn. Figure 1 represents the workflow of proposed technique.

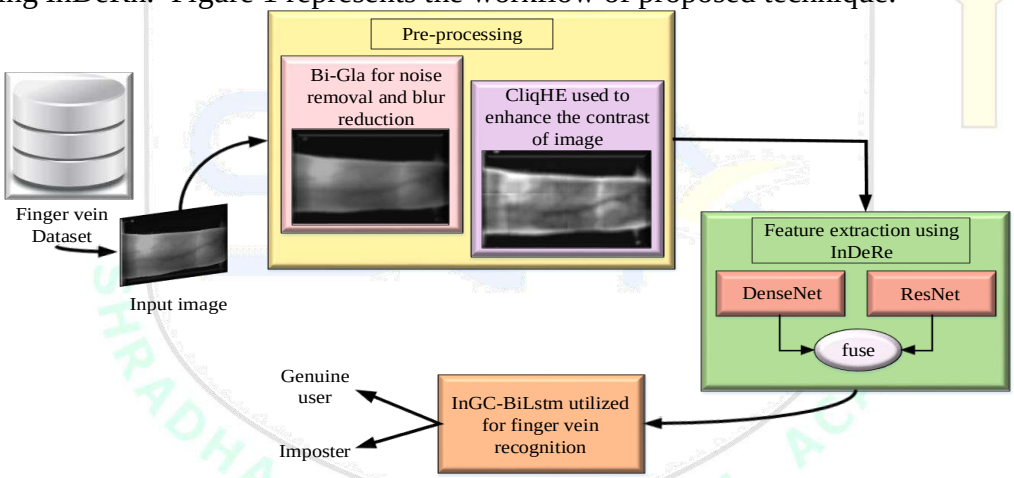

Figure 1: Workflow of proposed technique

The features from these two models are fused together and passed in to the classification stage. Based on the selected features, the finger vein recognition can be performed by InGC-BiLstm. The classifier model has some hyperparameters, these hyperparameter are tuned by LeDMO. This hyper parameter tuning mechanism help to enhance the performances and efficiency of the model.

3.1 Pre-processing

The pre-processing process is performed to improve the quality of image by removing the noise, reducing blur and contrast enhancement. The Bi-GLa filter utilized to remove the noise and blur, then CliqHE employed to enhance the contrast of image.

3.1.1 Noise removal and blur reduction using Gaussian Bilateral Filter

Assuming the noise filtering input X and guidance b are identical, the bilateral filter dramatically declines. Due to additive noise distortion of the range kernel, the bilateral kernel offers a limited amount of smoothing. Therefore, an essential characteristic of Bi-GLa occurs when b and X are non-identical, indicating that the image contours are properly retained immediately when the low-pass filtering procedure is carried out to provide a guide for the range kernel.

The Gaussian blur process [26] is corresponds to low-pass filtering for a given image X. Initially, the low-pass guidance image is obtained by applying the Gaussian filter. A weighted average of the pixels in the nearby locations with a weight decreasing from the centre position x. It used to obtain low-pass guidance image which is described in the below equation.

$$g(x) = \sum_{y} G^{b}_{x,y}(B) X_{y}, \quad (1)$$

Here, the filter kernel is described in the below equation.

$$G^{b}_{x,y}(B) = \frac{1}{H_x} \exp\left(-\frac{\|x-y\|^2}{\sigma_v^2} \right) \quad (2)$$

Here, the output the Gaussian blur process reduce the content and sharp edges of images. A sharp gap occurs when σ_v values increase, indicating the requirement to perform further averaging. Most edges in the obtained image are blurred consequently, when B and X are identical and the filtering of input X is over smooth, the BF blurs to a standard Gaussian filter. For the noise filtering inputs, when B and X are non-identical. It means that the Gaussian range kernel defines the impact of pixels from low-pass guiding $\bar{B}_y$, which is derived from the Gaussian blur process on X_x, while the Gaussian spatial kernel is utilised for filtering input X. The Gaussian Bilateral kernel is described in the below equation.

$$G^{bijd}_{x,y}(\mathrm{X}, \bar{B}) = \frac{1}{H_x} \exp\left(-\frac{\|x-y\|^e}{\sigma_v^2} \right) \exp\left(-\frac{\|\mathrm{X}_x - \bar{B}_y\|^2}{\sigma_u^2} \right), \quad (3)$$

Here. The low-pass guidance is represented as $\bar{B}$ which could be obtained by equation (1). Although the filtering input I is affected by varying noise composition, the Gaussian spatial kernel on X combined with the Gaussian range kernel from $\bar{B}$ provides an accurate preservation of image edges and outlines while smoothing the surfaces of objects within the filtering output $g(i)$. The filtering output $g(i)$ of the Bi-GLa is described in the below equation.

$$g(x) = \sum_{y} G^{bijd}_{x,y}(\mathrm{X}, \bar{B}) \mathrm{X}_{y} \quad (4)$$

3.1.2 Contrast enhancement using CliqHE

The CliqHE utilized for enhance the contrast of image which includes four process named histogram partitioning, clipping, allocation of gray level and histogram equalization.

3.1.2.1 Histogram Partitioning

CliqHE utilized to overcome the falsely detection due to noisy input histogram even the filter applied for earlier histogram partition [27]. The histogram partition of CliqHE is performed using median intensity value of the image. First, the histogram of image divided into two sub-histograms. Similarly, in order to divide the two sub-histograms into two smaller sub-histograms each, the medians from the two partition sub-histograms are utilised as separating points. As a result, four sub-histograms in total were obtained. Then the input histogram's minimum and maximum intensity values have been selected as the separating points.

The partition approach of CliqHE is similar to Recursive sub-image histogram equalization (RSIHE). The median-based partition technique has an ability to divide the total

number of pixels into equal segments for every sub-histogram. Therefore, mathematical expressions to evaluate each separating point is described in the below equation.

$$a_1 = 0.25 \times \{X_{wid} \times X_{hei}\} \quad (5)$$

$$a_2 = 0.50 \times \{X_{wid} \times X_{hei}\} \quad (6)$$

$$a_3 = 0.75 \times \{X_{wid} \times X_{hei}\} \quad (7)$$

Here, the intensities are represented as a_1, a_2 and a_3 which set to be 0.25, 0.50 and 0.75 for total number of pixels in the histogram of image. The width and height of image are represented as X_{wid} and X_{hei}.

3.1.2.2 Clipping Process

The enhancement rate of histogram equalization is controlled using this clipping process which utilized to overcome the occurrence of unnatural and over-enhanced image. This clipping process is inspired by the self-adaptive plateau histogram equalization (SAPHE) to enhance the contrast of infrared image. Due to the unsuccessful local peak detection, this technique may fail in the natural image. Hence, the median value of non-empty bins are located as clipping threshold, C_T using modified SAPHE. Then, the average number of intensity is replaces clipping threshold C_T which is utilized for the clipping process.

3.1.2.3 Gray Level Range Allocation

The CliqHE allocates a new gray level dynamic range based on total number of pixel for every sub-histogram and ration the gray level spans. The mathematical expression of a new gray level range allocation described in the below equation.

$$S_x = a_{x+1} = a_x \quad (8)$$

$$F_x = S_x \times (\log_{10} A_x)^{\gamma} \quad (9)$$

$$R_x = \frac{(Q-1) \times F_x}{\sum_{h=1}^{4} F_h} \quad (10)$$

Here, the dynamic gray level of x^{th} sub-histogram in image is represented as S_x and the x^{th} separating point is denoted as a_x. Total number of pixel in x^{th} sub-histogram is denoted as A_x and the dynamic range for x^{th} sub-histogram in output image is represented as R_x. The amount of emphasis given to A_x is represented as γ which is adjusted to determine the each sub-histograms span in output histogram. The equation does not affect new dynamic range significantly, when CliqHE have equal total number of pixels in every sub-histogram. For the purpose of complexity reduction and remove the parameter γ, the equation (9) is re-written to be below described equation.

$$R_x = \frac{(Q-1) \times S_x}{\sum_{h=1}^{4} S_h} \quad (11)$$

The new dynamic range is allocated from $[x_{start}, x_{end}]$ in the i^{th} sub-histogram which is described in the below equations.

$$x_{start} = (x-1)_{end} + 1 \quad (12)$$

$$x_{end} = x_{start} + R_x \quad (13)$$

The minimum intensity value of new dynamic range is initialized by the first x_{start}.

3.1.2.4 Histogram Equalization

The CliqHE equalize every sub-histogram independently, after the determination of new dynamic ranges for every quadrant sub-histogram. The outcome of histogram equalization, when i^{th} histogram is allocated at the gray level from $[x_{start}, x_{end}]$ which $b(e)$ of partition determined using transfer mapping function. The mathematical expression of $b(e)$ in the below described equation.

$$b(e) = (x_{start} - x_{end}) \times CDF(E_h) + x_{start} \quad (14)$$

Here, the cumulative density function (CDF) in sub-histogram is denoted as $CDF(E_h)$. The general histogram equalization equation is utilized in this CliqHE, the minimum and maximum intensities used by x_{start} and x_{end} in output dynamic range.

3.2 Feature extraction using InDeRn model

The InDeRn model utilized to extract the features which is the combination of Dense Network and Residual network. Initially, the pre-processed image send to both Dense Network and Residual network individually. Then, the feature from these two modes are fused together to provide efficient performance.

3.2.1 Dense Network

Every layers in traditional CNN are connected gradually described in equation (15) which utilized to make the network hard to go wider and deeper. Assume the input is a_{i-1} and the output obtained after two convolutional layers is $T_i(a_{i-1})$, these are added with shortcut to input layer a_{i-1}. Hence, the output of i^{th} layer is the summation which is described in equation (5).

Then, DenseNet [28] made further modifications to the model, substituting the sum of the output feature maps of every prior layer which is described in below equation with a combination of all the feature maps sequentially.

$$a_i = T_i(a_{i-1}) \quad (15)$$

$$a_i = T_i(a_{i-1}) + a_{i-1} \quad (16)$$

$$a_i = T_i([a_0, a_1, a_2, ..., a_{i-1}]) \quad (17)$$

Here, the layer index is denoted as i, non-linear operation is represented as T and the feature from i^{th} layer is denoted as a_i.

Every feature maps propagates to prior layers and connected to new feature map using the equation (15). Some of beneficial of DenseNet such as feature reuse, reduce either exploding issue or gradient vanishing issue. Down-sampling the feature map for concatenation in order to make the DenseNet feasible. The concatenation operation does not perform when the size of feature map keep changing. There are three transition layers which include the operation such as convolution, pooling and batch normalization operation. The Dense block include the number of layers=5 and the set the value of growth rate as h. The feature maps of every layer received form every prior layers.

The h feature map are generated for every operation T_i and there are five layer utilized in DenseNet which the feature maps to be obtained as $h_0 + 4h$. Here, the number of feature maps from prior layer which is represented as h and its default value is 32. According to the network's multiple inputs, a bottleneck layer was included in DenseNet. This layer is carried out by a 1×1 convolution preceding the 3×3 convolution layer to save computing costs by reducing the number of feature maps. The transition layer utilized to reduce feature may by considering the compactness of model. When n feature maps generated by the dense block and consider compression factor as $\theta \in (0,1]$, then feature maps is reduced to $[\theta n]$. Here, the number of feature map will be same when $\theta = 1$.

ISSN: 2393-8048
International Advance Journal of Engineering, Science and Management (IAJESM)
Multidisciplinary, Multilingual, Indexed, Double Blind, Open Access, Peer-Reviewed, Refereed-International Journal.
SJIF Impact Factor =8.152, January-June 2025, Submitted in April 2025

The structure of DenseNet includes three Dense Blocks such as input layer, transition layer and global average pooling layer (GAP) layer. Here, the transition layer includes a 1×1 convolutional layer, 2×2 average pooling layer and batch normalization layer. The GAP layer is similar as the traditional pooling methods which reduce the whole slice into a single digit. Figure 2 represents the Architecture of InDeRn model to extract the features.

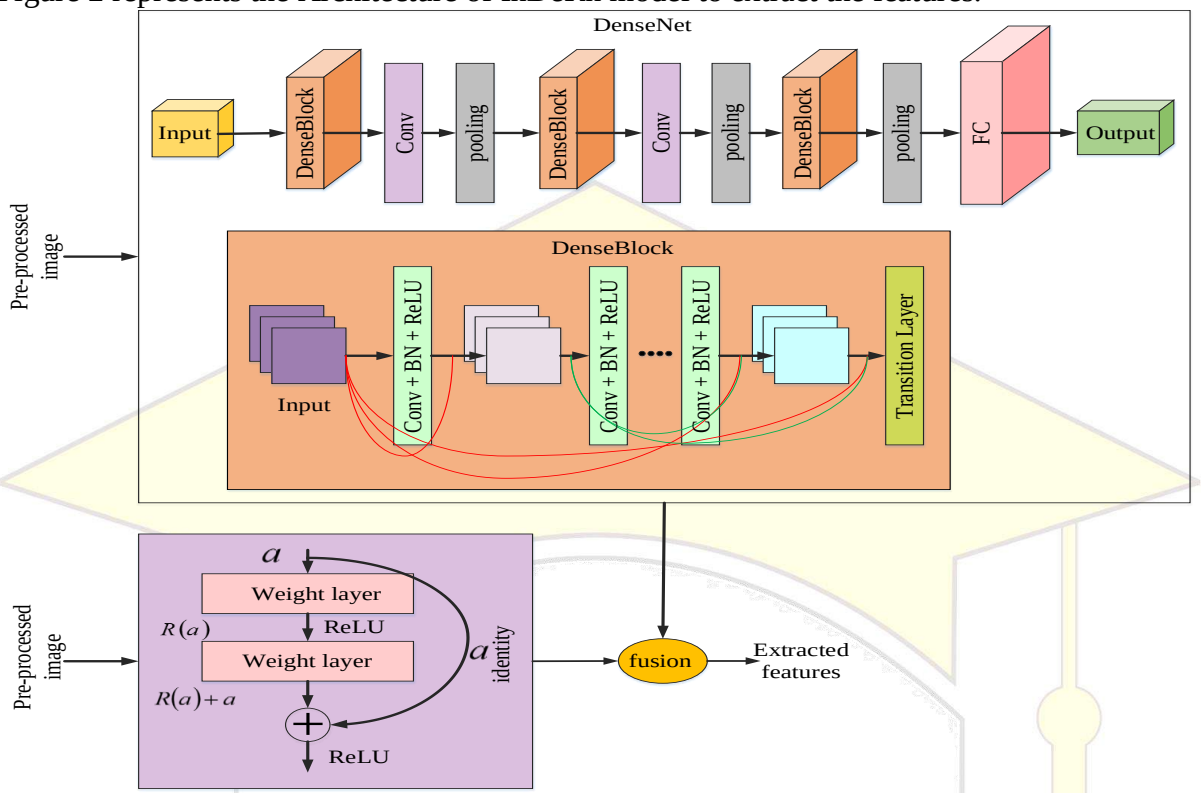

Figure 2: Architecture of InDeRn model

3.2.2 Residual Network

The residual network (ResNet) utilized to overcome the vanishing gradient issue during the training process. As training continues with the gradient norm of prior layers becomes lower and less extensive until it is zero. The yield of every residual layer in ResNet is combined with its input to become the input of next layer. The residual mapping is utilized to build residual learning bloc which is denoted as $R(a)$. The ResNet block approximately evaluates $R(a)+a$ which is identified by the feedforward neural system with shortcut connection. The input and output of stacked layer combined together is known as shortcut connections through the identity mapping operation without using any additional parameters. Gradients can rapidly flow back which provides substantially more layers and faster training. The ResNet include more identity connection to enhance the training. The ResNet includes two main blocks named identity and convolutional blocks.

3.2.2.1 Identity Block

The identity block is the initial block utilized in ResNet which is described in the below equation.

$$b = R(a,\{Gi\})+a \qquad (18)$$

Here, the input and output vectors of layer are represented as a and b, then the trained residual mapping denoted by the function $R(a,\{Gi\})$. This identity block have equal dimension of $\times$ and its structure includes three components which is described as follows.

- The first components have 2D convolutional layer with 1×1 filter size and a stride of (1, 1). The ReLU function is utilised for the nonlinear activation function, and BN performs the channel axis normalisation.
- The second component is similar to the first component but with $s\times s$ filter size.
- The third component is also similar to first component without activation function ReLU.

Before applying the ReLU activation function, the shortcut and input are combined together.

3.2.2.1 Convolutional Block

The dimension of input and output are not matched in this block, hence the shortcut connection performs linear projection G_p. It utilized to resize the dimensions between a and R which is described in the below equation.

$$b = R(a,\{Gi\}) + G_p a \quad (19)$$

Here, the output of stacked layer is denoted as R, the input and combined output vector of convolutional block are represented as a and b. In this phase, the structure of identity block followed by the structure of convolutional block with an additive 2D convolutional layer in shortcut way. Depends on output dimension, the input a resized to be matched up with the main path and the 2D convolutional includes 1×1 filter size and a stride of (r,r). Finally, it combined together with the outcome of main path which utilized to control the vanishing gradient issues.

3.3 Figure vein recognition using InGC-BiLstm model

The InGC-BiLstm model utilized to recognize the finger vein which is compressed with some DL technique to make the classification model efficiently. The classification model combined with Bidirectional Long Short Term Memory (Bi-LSTM) and Convolutional Gated Recurrent unit (C-GRU).

3.3.1 Bidirectional LSTM (Bi-LSTM)

Long Short Term Memory (LSTM) is an advanced version of RNN which developed to overcome the issue of RNN layer by memory cells. The memory cell has self-connection which utilized to store the network temporal state. It is controlled by three gates such as input gate, forget gate and output gate. The input and output get utilized to control the flow of input of memory cell and output into the rest of network. The forget gate utilized to pass the output with weights from the prior to the next neurons. Based on the high activation result, the information resides in the memory. The information stored in memory when input unit have high activation. The information passed to next neuron when output unit has high activation, else the information with high weights reside in the memory cell.

The mapping between input sequence $I = (I_1, I_2, ..., I_n)$ and output sequence $j = (j_1, j_2, ..., j_n)$ are evaluated by the LSTM which is described in the below equation.

$$F = \sigma(w_F I_t + w_{kF} k_{t-1} + s_F) \quad (20)$$

$$P = \sigma(w_P I_t + w_{kP} k_{t-1} + s_P) \quad (21)$$

$$U = \sigma(w_U I_t + w_{kU} k_{t-1} + s_U) \quad (22)$$

$$(L)_t = (L)_{t-1} \otimes (F)_t + (P)_t \otimes (\tan k(w_L I_t + w_{kL} k_{t-1} + s_L)) \quad (23)$$

$$k_t = U \otimes \tan k((L)_{t-1}) \quad (24)$$

Here, the weights and bias variables of three gate and memory cell are represented as w_F, w_U, w_{kL}, w_P and s_F, s_P, s_U, s_L. The prior hidden layers units symbolized as k_{t-1} which add the weights of three gates based on element-wise. Then, the current memory cell unit converted from $(L)_t$, after processing equation (23). The prior hidden unit outputs and previous memory cell unit are performed by element wise multiplication which is described in equation (24). Then, the non-linearity on top of three gates as $\tan k$ form and sigmoid activation function

described in equation (20-24). Here, the previous and current time steps are represented as $t-1$ and t.

The Bi-LSTM [29] is developed based on the bidirectional RNN (BRNN) to overcome the issues of LSTM cell. Usually, the LSTM can able to process prior data but it can't able to access the future content. Two individual LSTM hidden layer with similar result in different direction is known as Bi-LSTM. The prior and future information are performed in output layer of BiLSTM. In Bi-LSTM, the input sequence $I=(I_1,I_2,....,I_n)$ is evaluated in forward direction $\overrightarrow{k_t}=(\overrightarrow{k_1},\overrightarrow{k_2},....,\overrightarrow{k_n})$ and the backward directions as $\overleftarrow{k_t}=(\overleftarrow{k_1},\overleftarrow{k_2},....,\overleftarrow{k_n})$. The outcome of Bi-LSTM is denoted as j_t which obtained by evaluating both $\overrightarrow{k_t}$ and $\overleftarrow{k_t}$. The final output sequence of Bi-LSTM is represented as $j=(j_1,j_2,....j_t,....,j_n)$.

3.3.2 Convolutional Gated Recurrent Unit (C-GRU)

The C-GRU is performed for finger vine recognition which utilized CNN's convolutional feature extractor to generate the dynamic features. Then, the feature selection phase utilized to select optimal feature. Then, the GRU utilized for finger vein recognition.

3.3.2.1 Gated Recurrent Unit (GRU)

There are two gate utilized in the general GRU named update gate and forget gate. It trained to obtain the data form past without losing the information via time. It also utilized to eliminate the irrelevant data which were not essential for recognition.

- **Update gate:** It utilized to determine the amount of pass data form prior block which have to be forwarded to the subsequent block. It utilized to copy every past information and eliminate the issue of vanishing gradient. The update gate is denoted as u_t for t time step which is evaluated using the equation (25). Here, the input at step t is denoted as i_t, the hidden state which includes the information for previous unit $t-1$ is denoted as k_{t-1}. The weights of i_t and k_{t-1} are represented as w_u and h_u, then the sigmoid activation function kept the value of u_t between 0 to 1. The mathematical expression of u_t is described in the below equation.

$$u_t=\sigma(w_u i_t+h_u k_{t-1}) \quad (25)$$

- **Reset gate:** The reset gate is represented as s_t which utilized to determine the amount of past information to be forget which is described in the below equation.

$$g_t=\sigma(w_u g_t+h_u k_{t-1}) \quad (26)$$

- **Current memory content:** It is represented as k_t' which utilized to save the relevant data from past by reset gate which is evaluated using the below described equation.

$$k_t'=\tan k(wi_t+g_t\otimes hk_{t-1}) \quad (27)$$

Here, the Hadamard element-wise product is represented as $\otimes$.

- **Final memory:** It represented as k_t at current time t which is determined as vector that includes the block's information and it passed to the next block. The mathematical equation of k_t is described in the below equation.

$$k_t=u_t\otimes k_{t-1}+(1-u_t)\otimes k_t' \quad (28)$$

Here, the amount of data retained from current h_t' and previous memory content h_{t-1}' are determined by the update gate.

Two layer of GRU network utilized to model the C-GRU which maximize the conditional probability $(p(T|C))$ of convolutional timestep T over the input C at next timestep. The current prediction are optimized by this model based on the input P. During the training phase, the frames and timestep of input are modified and the number of input vector become less for

ISSN: 2393-8048
International Advance Journal of Engineering, Science and Management (IAJESM)
Multidisciplinary, Multilingual, Indexed, Double Blind, Open Access, Peer-Reviewed, Refereed-International Journal.
SJIF Impact Factor =8.152, January-June 2025, Submitted in April 2025

this experiment. For every timestep, it moved forward to the input vectors with a single stride. The mathematical expression of conditional probability described in the below equation.

$$p(T \mid C) = \prod_{b=1}^{e} p\left(i_{(a+b)_b}, C\right) \quad (29)$$

The convolutional feature of input at timestep t is the input of first GRU cell. The output of initial GRU cell is the input of next GRU cell which passed through SoftMax layer to perform classification.

$$p\left(d_t \mid (d_{t+1})\right) = soft\max\left(w_l k_t\right) \quad (30)$$

Here, the learnable parameter is denoted as w_l and the output of current GRU cell at time t is represented as d_t The hidden states are represented as k_t, the two adjacent input vectors of model are represented as d_t and d_{t+1}. The mathematical expression of output hidden state at current time is evaluated using the below described equation.

$$k_t = c\_gru\left(d_{t-1}, k_{t-1}\right) \quad (31)$$

The Architecture of InGC-BiLstm model to recognize the finger vein as genuine user or imposter is represented in Figure 3

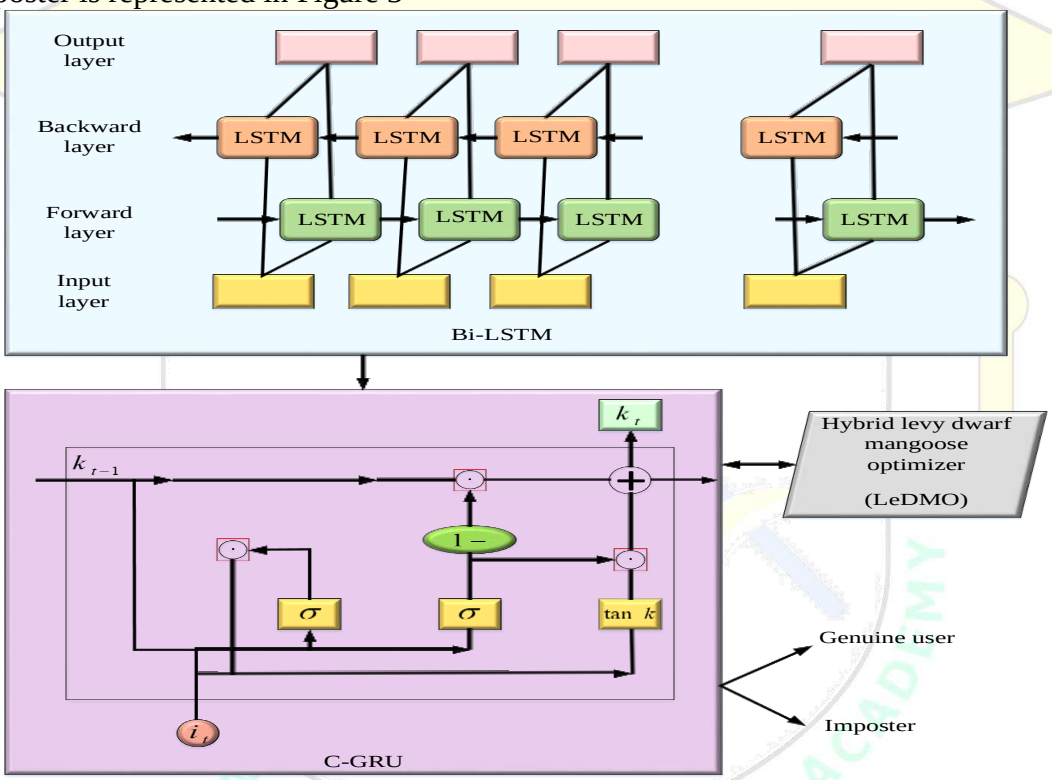

Figure 3: Architecture of InGC-BiLstm model

3.3.3 Hyper parameter tuned by LeDMO

The Dwarf Mongoose is a member of stochastic population based metaheuristic algorithm [30]. The behaviour of dwarf mongoose's social and foraging mimicked by this algorithm. The animals forage in groups, but individual dwarf mongoose does a thorough food search as feeding is not a collective exercise. Due to their seminomadic attribute, they build their sleeping mound close to an abundant food source and search for the next abundant food source. The DMO initialises the mongoose candidate population before begins its updating which is described in equation (32). Generating the population stochastically between the lower bound (LB) and upper bound (UB) of particular issues.

$$D = \begin{bmatrix} d_{1,1} & d_{1,2} & \Lambda & d_{1,m-1} & d_{1,m} \\ d_{2,1} & d_{2,2} & \Lambda & d_{2,m-1} & d_{2,m} \\ \text{M} & \text{M} & \Lambda & \text{M} & \text{M} \\ d_{n,1} & d_{n,2} & \Lambda & d_{n,m-1} & d_{n,m} \end{bmatrix} \quad (32)$$

Here, the set of candidate's current population is denoted as D which generate randomly using the equation (33). The position of j^{th} dimension of i^{th} population is represented as $d_{i,j}$. The population size is denoted n and the problem dimension is denoted as m.

$$d_{i,j} = R(Var\min, Var\max, Var\_size) \quad (33)$$

Here, the uniformly distributed random number is denoted as R. The upper and lower bound are represented as $Var\max$ and $Var\min$. The problem's dimension is represented as Var_size. Hence, the best solution is obtained as the best solution at each iteration.

The DMO algorithm include two phases exploitation and exploration, the activities were obtained by three social structure of DMO named the alpha group, the scout group and the babysitters. The rest of family unit controlled by the alpha female (α) which is selected based on below described equation.

$$\alpha = \frac{Fitness_i}{\sum_{i=1}^{n} Fitness_i} \quad (34)$$

Here, the number of mongooses in alpha group is denoted as $n-B$. Number of babysitters is denoted as B while the sound of female alpha which indicates $peep$ to confirm that the family is in right path. The position of sleeping mound determined by the abundant food source which is described in the below equation.

$$D_{i+1} = D_i + L * peep \quad (35)$$

Here, the random uniformly distributed number [-1, 1] is denoted as L. The sleeping mound is evaluated after each iterations which described in the below equation.

$$J_i = \frac{Fitness_{i+!} - Fitness_i}{\max\left\{\left|Fitness_{i+!}, Fitness\right|\right\}} \quad (36)$$

The average value when the sleeping mound is identified which is described in the below equation.

$$\varphi = \frac{\sum_{i=1}^{n} J_i}{n} \quad (37)$$

After the completion of the babysitter exchange criterion, the next sleeping mound is scouted with the goal to assess its suitability based on the availability of food. Since mongooses are known to avoid return to an already visited sleeping mound, the scout group searches for another location to ensure exploration. In DMOA, foraging and scouting are carried out simultaneously on the theory that the further the family forages. The position of next sleeping mound is evaluated which is described in the below equation.

$$D_{i+1} = \begin{cases} D_i - P * L * r * \left[D_i - \overrightarrow{H}\right] & if\ \varphi_{i+1} > \varphi_i \\ D_i + P * L * r * \left[D_i - \overrightarrow{H}\right] & else \end{cases} \quad (38)$$

Here, the random number between the interval [0, 1] is represented as r. The parameter utilized to direct the collective-volatile movement which indicated as $L = \left(1 - \frac{T}{\max_T}\right)^{\left(2\frac{T}{\max_T}\right)}$. The

vector utilized for movement of mongoose to another sleeping mound is denoted as $\vec{H} = \sum_{i-1}^{n} \frac{D_i \times J_i}{D_i}$.

When the group that goes scouting and foraging looks for a sleeping mound and food source, the babysitter's group stays with the young animals. Since, it absence from scouting or foraging, number of members of the group is subtracted from the total amount of candidates. The babysitters exchange with scouting for foraging group when a certain parameter were achieved to search for food.

The levy flights is a class of non-Gaussian random process which evaluate the random process based on levy stable distribution. The simple power law of distribution is $F(u) \sim |u|^{-f-\beta}$, here the index is $0 < \beta < 2$. The mathematical expression of Levy distribution is described in the below equation.

$$F(u,\gamma,\mu) = \begin{cases} \sqrt{\frac{\gamma}{2\pi}} \exp\left[-\frac{\gamma}{2(u-\mu)}\right] \frac{1}{(u-\mu)^{3/2}} & if \quad 0 < \mu < u < \infty, \\ 0 & if \quad u \le 0 \end{cases} \tag{39}$$

Here, the location or shift parameter is denoted as μ and the scale parameter is denoted as $\gamma > 0$. The mathematical expression of Levy distribution based on Fourier transform is described in the below equation.

$$FT(h) = \exp\left[-\lambda[h]^{\beta}\right], \qquad 0 < \beta \le 2, \tag{40}$$

The parameter in the interval [-1, 1] is denoted as λ which known as the scale factor or skewness. The Levy index referred as an index of o stability $\beta \in (0,2]$ and the integral's analytic form is not known for β except some special cases.

Case 1: The integral could be carried out analytically which is Cauchy probability distribution for $\beta = 1$.

$$FT(h) = \exp\left[-\lambda|h|\right], \tag{41}$$

Case 2: The distribution corresponds to the Gaussian distribution when $\beta = 2$.

$$FT(h) = \exp\left[-\lambda h^2\right], \tag{42}$$

Here, the parameter β and λ are utilized as major part of the determination of distribution, in which γ and μ act as minor part. The shape of probability distribution controlled by β which obtain various shapes of the probability distribution in tail region which depends on β. The Algorithm of LeDMO is described in the Table 2.

Table 2: Algorithm of LeDMO algorithm

Initialize *peep*, n and B
Set $n = n - B$
Set the babysitter exchange parameter E
for $T = 1 : \max\ T$
Evaluate the fitness function
Set the time counter t
Find alpha using equation (34)
Obtain position of candidate food using equation (35)
Evaluate new fitness function of D_{i+1}
Calculate the sleeping mound using equation (36)
Evaluate average value of sleeping mound determined using the equation (37)

Evaluate movement vector using $\rho H = \sum_{i-1}^{n} \frac{D_i \times J_i}{D_i}$.

Babysitter are exchange if $t \geq E$ and set $Fitness_i = 0$

Updated position of scout mongooses using the equation (38) and Levy distribution method

4. Result

The efficiency of proposed technique analysed based on some performance metric such as accuracy, precision, recall and so on. Then, comparison is conducted with various related technique to determine the superiority of proposed technique.

4.1 Dataset Description

The finger vein dataset is developed by Kaggle which include 106 classes and each classed contained 2 classes such as right and left. In each classes, the image of index, middle and ring finger are included individually. The dataset can be downloaded from the link as https://www.kaggle.com/datasets/ryeltsin/finger-vein.

4.2 Performance metrics and formulation

The performance metric utilized to evaluate the performance of proposed technique such as Accuracy, Precision, Recall, F1-scoare, False Acceptance Rate (FAR), False Rejection Rate (FRR) and Genuine Acceptance rate (GAR). These performance metric are described with its formulation to define the superiority of proposed technique.

4.2.1 Accuracy

The effectiveness and accurate detection of proposed technique is measured by the performance metric named accuracy which evaluated using the below described equation.

$$Accuracy = \frac{T_{pos} + T_{neg}}{T_{pos} + T_{neg} + F_{pos} + F_{neg}} \quad (43)$$

Here, T_{pos}, T_{neg}, F_{pos} and F_{neg} are denoted as True positive, True negative, False positive and False negative.

4.2.2 Precision

The precision is average ratio of all positive sample's accuracy over predictions that are positive which is described in the below equation.

$$\Pr ecision = \frac{T_{pos}}{T_{pos} + F_{pos}} \quad (44)$$

4.2.3 Recall

The ratio of the total number of positive data obtained is known as the recall rate which is described in the below equation.

$$\mathrm{Re}\, call = \frac{T_{pos}}{T_{pos} + F_{neg}} \quad (45)$$

4.2.4 F1-score

The F1-score is one of performance metric which utilized to measure the harmonic mean of precision and recall which is described in the below equation.

$$F1 = \frac{2 \times \Pr ecision \times \mathrm{Re}\, call}{\Pr ecision + \mathrm{Re}\, call} \quad (46)$$

4.2.5 FAR

The FAR utilized to measure the probability of invalid input which is incorrectly accepted and it described in the below equation.

$$FAR = \frac{no.of\ \ accepted\ imposter}{Total amount of\ imposter} \quad (47)$$

4.2.6 FRR

The FRR is a performance metric which evaluated the ratio of number of false rejection rate with total amount of transaction and it described in the below equation.

$$FAR = \frac{no.of \quad false\ rejection}{Total\,amount of\ recognition\ attempts} \tag{48}$$

4.2.7 GAR

The ratio of number of input sample are correctly recognized from the total amount of positive input samples is known as GAR. The mathematical expression of GAR is described in the below equation.

$$GAR = 100 - FRR \tag{49}$$

4.3 Performance analysis of proposed technique

In this section, the overall performance of proposed technique are analysed based on Accuracy, precision, recall and so on. Then, the comparison is conducted with other related technique such as CNN, RNN, ResNet, VGG16, EfficientNet and Vision Transformer. It describes the efficiency of proposed technique. Figure 4 describes the performance of (a) Accuracy, (b) Precision, (c) Recall and (d) F1-score.

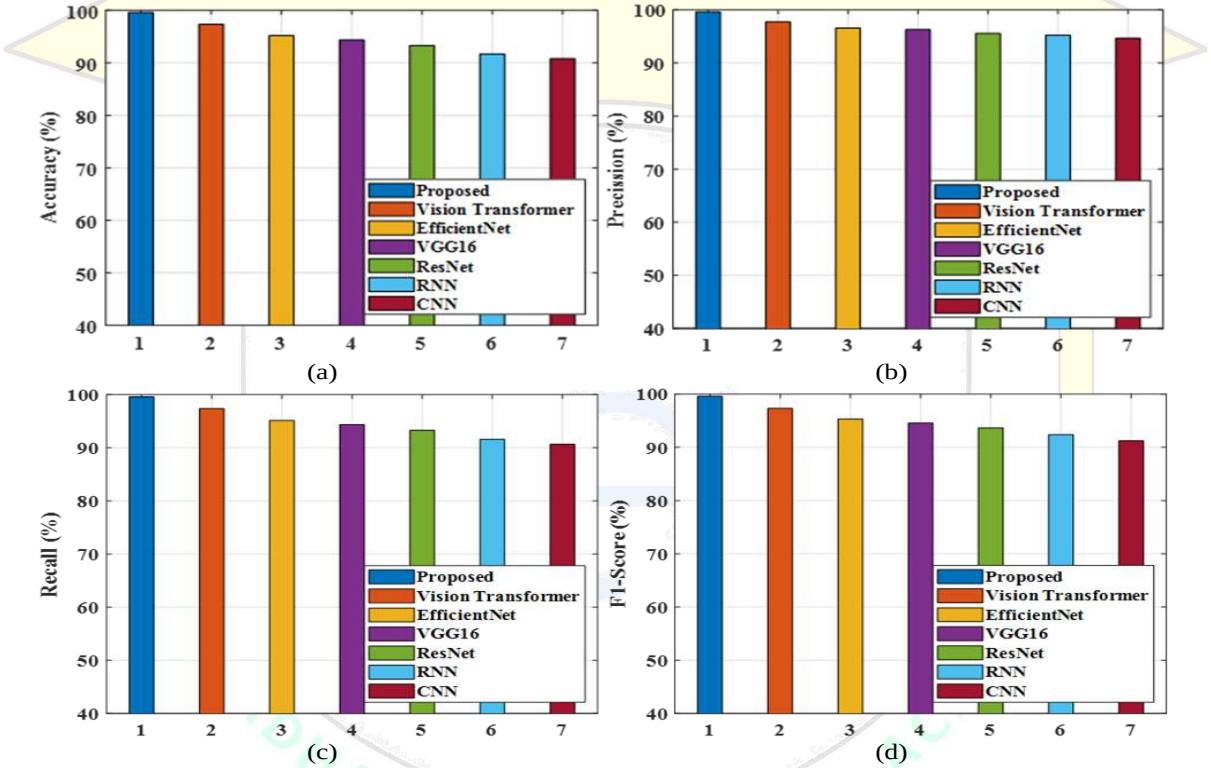

Figure 4: Performance of (a) Accuracy, (b) Precision, (c) Recall and (d) F1-score

The performance of (a) Accuracy, (b) Precision, (c) Recall and (d) F1-socre of proposed model represented in Figure 4 clearly describes the efficiency of proposed technique. The Accuracy of the proposed method is obtained to be 99.60% which is higher than CNN (90.79%), RNN (91.71%), ResNet (93.31%), VGG16 (94.36%), EfficientNet (95.18%) and Vision Transformer (97.34%). Similarly, the other performance metric such as precision, recall and F1-score analysed and compared with several related techniques which is illustrated in Figure 5 (b), (c) and (d). The Precision, Recall and F1-score of proposed methodology is obtained to be 99.60%, 99.60% and 99.59%. These comparison clearly determines the efficiency of proposed methodology for finger vein recognition. Figure 5 represents the performance of (a) Train Accuracy, (b) Train Loss, (c) Test Accuracy and (d) Test Loss.

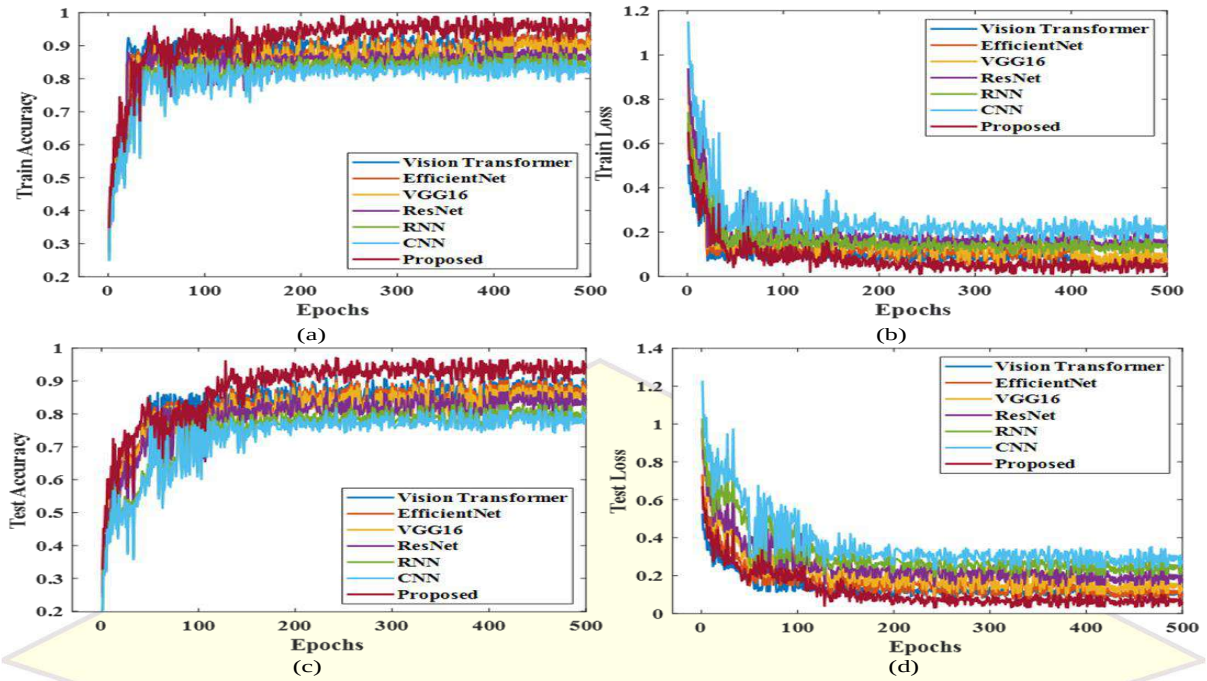

Figure 5: Performance of (a) Train Accuracy, (b) Train Loss, (c) Test Accuracy and (d) Test Loss

The Accuracy and Loss of proposed methodology analysed while train and testing the dataset. The performance are analysed and compared with some related techniques to determine the efficiency of proposed technique. Figure 5 (a) and (c) illustrates the Accuracy while train and testing the data which is higher than other comparative techniques. Similarly, Figure 5 (b) and (d) presents the loss rate while testing the data with other related techniques describes the efficiency of proposed technique. The comparison of FAR and FRR represented in Figure 6 (a) and (b)

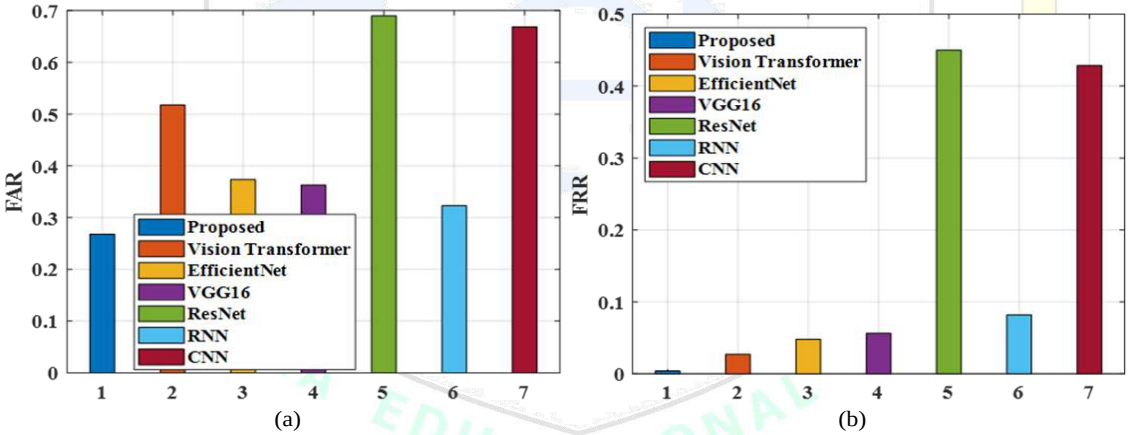

Figure 6: Performance of (a) FAR and (b) FRR

The performance of FAR and FRR illustrated in Figure 6 (a) and (b) clearly describes the efficiency of proposed model. The obtained FAR rate of proposed methodology is 0.107 which is less other related technique such as CNN (0.668%), RNN (0.322%), ResNet (0.669%), VGG16 (0.363%), EfficientNet (0.373%) and Vision Transformer (0.517%). The FRR rate of proposed methodology is obtained to be 0.003 which is lower than other related technique. The related technique obtains CNN (0.428%), RNN (0.081%), ResNet (0.449%), VGG16 (0.056%), EfficientNet (0.047%) and Vision Transformer (0.026%) which is illustrated in Figure 6 (b). The comparison of GAR with some existing technique represented in Figure 7.

ISSN: 2393-8048
International Advance Journal of Engineering, Science and Management (IAJESM)
Multidisciplinary, Multilingual, Indexed, Double Blind, Open Access, Peer-Reviewed, Refereed-International Journal.
SJIF Impact Factor =8.152, January-June 2025, Submitted in April 2025

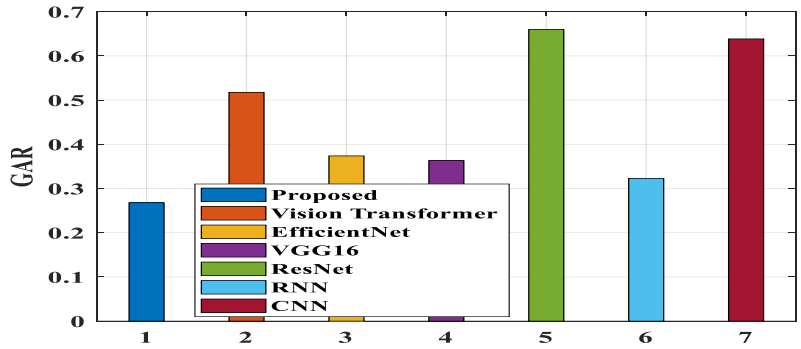

Figure 7: Performance of GAR rate

The comparison of GAR rate analysed for both proposed and related technique which describes the recognition rate of finger vein. The Proposed technique achieves the GAR as 0.26% which is higher than other related techniques. The GAR of the CNN as 0.63%, RNN as 0.32%, ResNet as 0.65%,, VGG16 as 0.36%, EfficientNet as 0.37% and Vision Transformer as 0.52%. It clearly describes the efficiency of proposed technique which reduced the GAR rate than other related technique. The comparison of ROC curve analysed and compared with the existing technique represented in Figure 8.

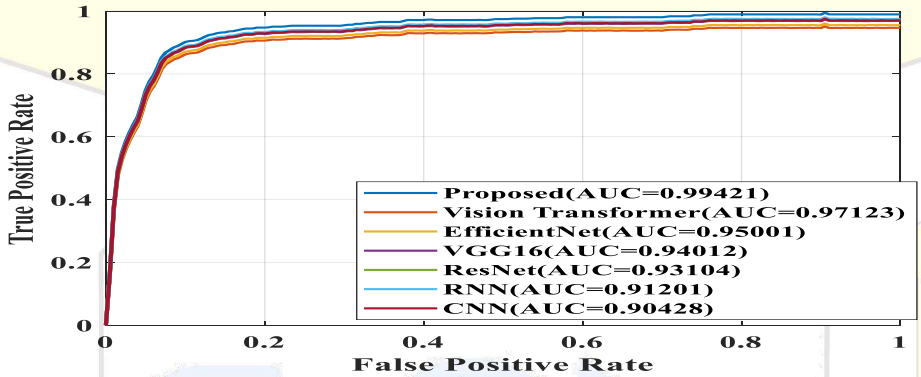

Figure 8: Comparison of ROC curve

The ROC curve of proposed model analysed its performance with other related technique such as CNN, RNN, ResNet, VGG16, EfficientNet and Vision Transformer. It analysed its performance based on the value of Area under the Curve (AUC). Figure 8 clearly illustrate the AUC rate of proposed technique is higher than other related technique. Figure 9 represents the confusion matrix for Finger vein dataset.

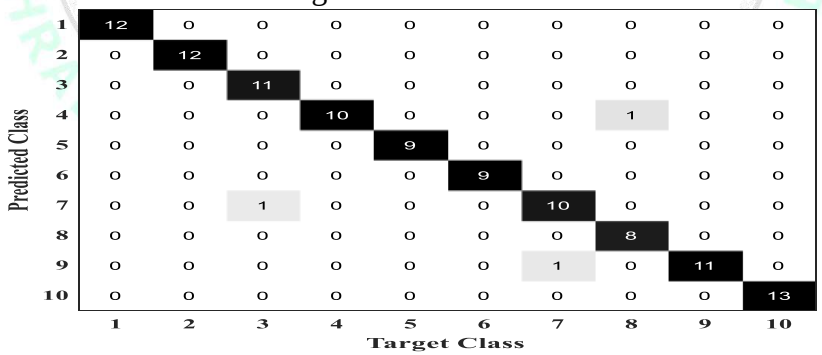

Figure 9: Confusion matrix

The confusion matrix utilized to describe the efficiency proposed technique for Finger vein dataset. It majorly utilized to determine the how many data were wrongly will be prediction which analysed through the confusion matric. The prediction rate of proposed method analysed based on the Predicted and Target class. It describes that the proposed method achieves efficient finger vein recognition. Nearly 10 classes were analysed which majorly achieves 100% recognition, one or two data would be wrongly predicted data.

4.4 Discussion

The performance of proposed method analysed with various related technique which determines the efficiency of proposed technique. Comparison of the related technique with proposed model analysed based on the Accuracy of Finger vein recognition. Table 3 describes the comparison of proposed technique with related technique based on their performance. Zhang et al. [21] developed LCAModel used for finger vein recognition which achieves 99.52% accuracy. Safie et al. [22] developed MC-CLAHE to enhance the finger images and AlexNet model for both feature extraction and classification which achieves 96.4% accuracy.

Table 3: Comparison Analysis with proposed technique

Author name and reference	Technique used	Dataset used	performance
Zhang et al. [21]	LCAModel used for finger vein recognition	Real-time	Accuracy-99.52%
Safie et al. [22]	MC-CLAHE to enhance the finger images and AlexNet model for both feature extraction and classification	FV-USM	Accuracy-96.4%
Htet et al. [23]	Recognition finger vein images using attention based residual networks	Real-time	(Equal Error Rate) EER – 0.018%
Lian et al. [24]	FedFV used for finger vein recognition	Finger vein dataset	EER- 0.95%
Li et al. [25]	ViT-FV	FV-USM and SDUMLA-HMT	EER-0.068%
Proposed	**InGC-BiLstm model**	**Finger-vein dataset**	**Accuracy-99.60%**

Htet et al. [23] developed Recognition finger vein images using attention based residual networks which achieves 0.018% EER. Lian et al. [24] developed FedFV used for finger vein recognition which achieves 0.95% EER. Similarly the other related technique analysed its performance with proposed techniques and compared which describes the efficient performance of proposed technique.

5. Conclusion

The proposed approach was developed to address the problems and increase performance. Initially per-processing the collected data from the finger vein dataset. CliqHE is used to improve image contrast, and Bi-GLa is used to reduce noise. Then, the data was extracted using the InDeRn model. To develop an effective feature, the features from two models are extracted and then fused together. Next, the InGC-BiLstm model is created to identify whether a finger vein belongs to a genuine user or an impostor. Next, LeDMO adjusts the hyperparameters used in the classification model. By contrasting the proposed method with other relevant procedures, its efficiency is established. The proposed method yields a higher accuracy of 99.60% than other techniques. The FRR, FPR and GAR rate of proposed technique is obtained to be 0.0039, 0.1071 and 0.2678. In future, the performance of proposed model improved will be improved by developing more efficient DL-based feature extraction and classification technique. Then, more dataset will be explore to analyse the efficiency of proposed method.

References

1. Al-ogaili, Hussein, and Ameen Majid Shadhar. "The Finger Vein Recognition Using Deep Learning Technique." *Wasit Journal of Computer and Mathematics Sciences* 1, no. 2 (2022): 1-11.

2. Liu, Wenjie, Weijun Li, Linjun Sun, Liping Zhang, and Peng Chen. "Finger vein recognition based on deep learning." In *2017 12th IEEE conference on industrial electronics and applications (ICIEA)*, pp. 205-210. IEEE, 2017.
3. Xie, Cihui, and Ajay Kumar. "Finger vein identification using convolutional neural network and supervised discrete hashing." *Pattern Recognition Letters* 119 (2019): 148-156.
4. NAWWAR, NADIA MOSTAFA, Hany Kasban, and May Salama. "Improvement of confusion matrix for Hand Vein Recognition Based On Deep-Learning multi-classifier Decisions." *Arab Journal of Nuclear Sciences and Applications* 54, no. 4 (2021): 133-146.
5. 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.
6. Tao, Zhiyong, Xinru Zhou, Zhixue Xu, Sen Lin, Yalei Hu, and Tong Wei. "Finger-vein recognition using bidirectional feature extraction and transfer learning." *Mathematical Problems in Engineering* 2021 (2021): 1-11.
7. Yang, Weili, Wei Luo, Wenxiong Kang, Zhixing Huang, and Qiuxia Wu. "Fvras-net: An embedded finger-vein recognition and antispoofing system using a unified cnn." *IEEE Transactions on Instrumentation and Measurement* 69, no. 11 (2020): 8690-8701.
8. Ali, Ashraf Tahseen, Hasanen S. Abdullah, and Mohammad N. Fadhil. "Finger veins recognition using machine learning techniques." *Mater. Today Proc* (2021).
9. Kuzu, Ridvan Salih, Emanuela Piciucco, Emanuele Maiorana, and Patrizio Campisi. "On-the-fly finger-vein-based biometric recognition using deep neural networks." *IEEE Transactions on information Forensics and Security* 15 (2020): 2641-2654.
10. Tao, Zhiyong, Haotong Wang, Yalei Hu, Yueming Han, Sen Lin, and Ying Liu. "DGLFV: Deep generalized label algorithm for finger-vein recognition." *IEEE Access* 9 (2021): 78594-78606.
11. Bilal, Anas, Guangmin Sun, and Sarah Mazhar. "Finger-vein recognition using a novel enhancement method with convolutional neural network." *Journal of the Chinese Institute of Engineers* 44, no. 5 (2021): 407-417.
12. Alay, Nada, and Heyam H. Al-Baity. "Deep learning approach for multimodal biometric recognition system based on fusion of iris, face, and finger vein traits." *Sensors* 20, no. 19 (2020): 5523.
13. Wimmer, Georg, Bernhard Prommegger, and Andreas Uhl. "Finger vein recognition and intra-subject similarity evaluation of finger veins using the cnn triplet loss." In *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 400-406. IEEE, 2021.
14. Sidiropoulos, George K., Polixeni Kiratsa, Petros Chatzipetrou, and George A. Papakostas. "Feature extraction for finger-vein-based identity recognition." *Journal of Imaging* 7, no. 5 (2021): 89.
15. Daas, Sara, Amira Yahi, Toufik Bakir, Mouna Sedhane, Mohamed Boughazi, and El-Bay Bourennane. "Multimodal biometric recognition systems using deep learning based on the finger vein and finger knuckle print fusion." *IET Image Processing* 14, no. 15 (2020): 3859-3868.
16. Noh, Kyoung Jun, Jiho Choi, Jin Seong Hong, and Kang Ryoung Park. "Finger-vein recognition based on densely connected convolutional network using score-level fusion with shape and texture images." *Ieee Access* 8 (2020): 96748-96766.
17. Garcia-Martin, Raul, and Raul Sanchez-Reillo. "Deep learning for vein biometric recognition on a smartphone." *Ieee Access* 9 (2021): 98812-98832.
18. Jasim, Zinah Khalid Jasim, Alaa Hamid Mohammed, Lina Elwiya, Baraa Dhafer Al-Jabbari, and Haitham Alhaji. "Human Identification with Finger Vien Image Using Deep

Learning." In *2021 5th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, pp. 672-677. IEEE, 2021.

19. Yang, Kenan, Peiyu Fang, and Jin Wu. "Deep learning-based region of interest extraction for finger vein images." In *IOP Conference Series: Materials Science and Engineering*, vol. 782, no. 3, p. 032056. IOP Publishing, 2020.
20. Zeng, Junying, Fan Wang, Jianxiang Deng, Chuanbo Qin, Yikui Zhai, Junying Gan, and Vincenzo Piuri. "Finger vein verification algorithm based on fully convolutional neural network and conditional random field." *IEEE access* 8 (2020): 65402-65419.
21. Zhang, Zhongxia, and Mingwen Wang. "Finger vein recognition based on lightweight convolutional attention model." *IET Image Processing* (2023).
22. Safie, Sairul Izwan, and Puteri Zarina Megat Khalid. "Practical Consideration in using Pre-Trained Convolutional Neural Network (CNN) for Finger Vein Biometric." *iJOE* 19, no. 02 (2023): 163.
23. Htet, Aung Si Min, and Hyo Jong Lee. "Contactless Palm Vein Recognition Based on Attention-Gated Residual U-Net and ECA-ResNet." *Applied Sciences* 13, no. 11 (2023): 6363.
24. Lian, Feng-Zhao, Jun-Duan Huang, Ji-Xin Liu, Guang Chen, Jun-Hong Zhao, and Wen-Xiong Kang. "FedFV: A Personalized Federated Learning Framework for Finger Vein Authentication." *Machine Intelligence Research* (2023): 1-14.
25. Li, Xiaoye, and Bin-Bin Zhang. "FV-ViT: Vision Transformer for Finger Vein Recognition." *IEEE Access* (2023).
26. Singh, Karamjeet, Bhisham Sharma, Jaiteg Singh, Gautam Srivastava, Suchita Sharma, Ashutosh Aggarwal, and Xiaochun Cheng. "Local statistics-based speckle reducing bilateral filter for medical ultrasound images." *Mobile Networks and Applications* 25 (2020): 2367-2389.
27. Bulut, Faruk. "Low dynamic range histogram equalization (LDR-HE) via quantized Haar wavelet transform." *The Visual Computer* 38, no. 6 (2022): 2239-2255.
28. Hemalatha, Jeyaprakash, S. Abijah Roseline, Subbiah Geetha, Seifedine Kadry, and Robertas Damaševičius. "An efficient densenet-based deep learning model for malware detection." *Entropy* 23, no. 3 (2021): 344.
29. Hameed, Zabit, and Begonya Garcia-Zapirain. "Sentiment classification using a single-layered BiLSTM model." *Ieee Access* 8 (2020): 73992-74001.
30. Mehmood, Khizer, Naveed Ishtiaq Chaudhary, Zeshan Aslam Khan, Khalid Mehmood Cheema, Muhammad Asif Zahoor Raja, Ahmad H. Milyani, and Abdullah Ahmed Azhari. "Dwarf Mongoose optimization metaheuristics for autoregressive exogenous model identification." *Mathematics* 10, no. 20 (2022): 3821.