Journal of Advanced Research in Applied Sciences and Engineering Technology

Journal homepage:
https://semarakilmu.com.my/journals/index.php/applied_sciences_eng_tech/index
ISSN: 2462-1943

Enhancing Finger Vein Authentication through Deep Learning: A Comparative Study of U-Net and Sequential Models

Amitha Mathew[1,2,*], P. Amudha[1]

[1] Department of Computer Science and Engineering, School of Engineering, Avinashilingam Institute for Home Science and Higher Education for Women, Tamil Nadu 641017, India
[2] Department of Computer Science and Engineering, Rajagiri School of Engineering and Technology, Kerala 682039, India

ARTICLE INFO

Article history:
Received 22 August 2023
Received in revised form 20 April 2024
Accepted 23 May 2024
Available online 20 June 2024

Keywords:
Biometric; Deep learning; Finger vein authentication; Sequential model; U-Net

ABSTRACT

The use of finger veins as biometric authentication is becoming increasingly popular. However, low-quality finger vein images pose challenges, necessitating innovative approaches for accurate authentication. This research investigates the potential of deep learning techniques in addressing this issue, focusing on two prominent architectures: U-Net and the proposed Sequential Model. The study conducts a comparative analysis of the performance of these models in low-quality finger vein image authentication scenarios. U-Net, known for its image segmentation capabilities, is explored for feature extraction, while the Sequential Model, incorporating a modified VGG16 architecture, brings temporal context through LSTM layers. The research presents an in-depth evaluation of both models based on accuracy, recall, precision, and other relevant metrics. The findings shed light on the suitability of each approach for enhancing the reliability of finger vein authentication in challenging data quality contexts.

1. Introduction

A finger vein authentication system uses patterns of veins beneath the skin to verify an individual's identity. A unique vein pattern on each finger makes it a secure biometric authentication method, even if they are identical twins. Since the veins are inherent to our bodies, they cannot be faked or stolen. In addition, the finger vein authentication system, being a contactless authentication system, reduces the spread of epidemic diseases compared to the widely used fingerprint system. It is not necessary to re-register finger vein patterns during the adult years because they remain relatively constant, and they are less affected by age or physical condition. Changes in the weather or an individual's physical condition have less effect on finger veins.

Due to hygiene-related issues, contactless biometrics like finger veins have been predicted to serve best in the future. Biometrics based on finger veins can be used in a variety of scenarios. The finger vein biometric authentication method allows access to various networks, web applications,

* *Corresponding author.*
E-mail address: amithamathew669@gmail.com

https://doi.org/10.37934/araset.47.1.230243

web portals etc. without the need to remember long passwords for each application. Furthermore, it can be used in the banking domain for credit card authentication, and online transactions to improve security and customer convenience. In addition to reducing the time it takes to log in, and eliminating the need to type usernames and passwords, it also reduces the financial burden of password resets. Physical access control can also be made possible by finger vein authentication including passenger verification at airports, attendance tracking, home security, school security, building access, automobile security, etc. This helps to avoid problems associated with lost, fraudulent, or stolen keys, cards, etc. [1].

The haemoglobin in veins changes colour when exposed to infrared light or visible light, which can be captured with an infrared scanner or a visible light camera. This can be utilised for acquiring finger vein images for enrolment and authentication [2].

In this study initially, a U-Net [3,4] architecture was employed for finger vein recognition. As authentication mechanisms are intended to grant access to legitimate individuals and prevent unauthorized imposters, achieving an accuracy close to 100% is crucial. To attain this goal a novel Sequential model is proposed in this study, which was able to achieve an accuracy close to 100. In the sequential model, a modified VGG16 architecture was used by removing the final fully connected layers and incorporating a time-distributed layer, an LSTM layer, and a dense layer instead. This innovative architecture significantly enhances the accuracy and reliability of the finger vein authentication system. This method works well with high-quality finger vein images as well since it was demonstrated with low-quality finger vein images.

The paper's structure is outlined as follows: Section 2 presents a discussion of related literature. Section 3 details the finger vein authentication system. The configuration of experiments and deep learning architectures is explained in Section 4. Section 5 evaluates the outcomes of comparative tests, and the paper concludes with Section 6.

2. Related Works

Deep learning approaches have shown promise in enhancing the effectiveness of finger vein recognition. However, they have limitations and face several challenges. While some systems propose robustness against noise and deformation [5], others claim to handle misalignment and shading [6], and some aim to address image quality variations [7]. However, none of these solutions are universally robust, and the performance may suffer in real-world scenarios with multiple challenges simultaneously.

Many of the proposed systems rely on deep learning models, such as CNNs [5,8], FCNNs [9], and capsule models [10]. However, obtaining a diverse and extensive dataset with image-level annotations is challenging, leading to limited training samples, and potentially affecting the system's performance. The absence of pixel-level texture labels in public finger vein databases further hinders the development of accurate recognition models. The multistage transfer learning used to address the lack of labelled training samples introduces dependencies on pre-trained models, potentially limiting the system's adaptability to new environments [11]. By combining deep learning, specifically semantic segmentation CNNs, with automatic label generation, Jalilian *et al.,* improved finger vein recognition performance compared to classical techniques. Their study demonstrates that certain FCN architectures, such as U-Net and RefineNet, outperform classical methods and highlight the significance of label quality [12].

The recognition issues caused by misalignment and illumination variations by introducing a biometric system. However, the integration of multiple sensors and the complexity of deep convolutional neural networks can increase the computational burden and raise practical

implementation challenges [13]. Tamilarasi *et al.,* used convolutional and recurrent neural networks in capturing temporal information and has shown promising results. But the sensitivity to hand movements and variations in behaviour can still affect the system's identification accuracy [14]. While the capsule model proposed by Yagoub, *et al.,* outperforms certain ANN models, its implementation is more complex and resource-intensive, making it less suitable for certain applications with limited computational resources [15].

Several proposed systems rely on preprocessing techniques, such as bias field correction, spatial attention mechanisms, and feature block fusion. These techniques may be sensitive to parameter tuning and might not be robust to unseen data [16]. The challenge of limited public finger vein datasets for training effective convolutional neural networks (CNNs) utilizing a pre-trained CNN model to extract vein features. Chiranjeevi, *et al.,* proposed a light weight deep-learning framework for real-time sentiment analysis, addressing the computational and dataset limitations associated with neural networks. They also mention the need for improved preprocessing to address rotation and displacement issues, addressing the insufficient number of training samples, and refining the topological structure for a more complete end-to-end framework [17].

While deep learning approaches hold great promise for enhancing finger vein recognition systems, the existing solutions face challenges in robustness, data availability, and integration complexity, reliance on preprocessing techniques and transfer learning, as well as performance issues. Addressing these disadvantages is essential for developing efficient and reliable authentication systems based on finger vein recognition. Deep learning-based authentication is influenced by a variety of factors. These factors include the learning architecture, the training strategies, and the feature extraction methods.

3. Architecture of Deep Learning-Based Finger Vein Recognition System

A finger vein recognition system using a deep model includes two phases, model generation and recognition:

i. Model Generation Phase: The steps involved in the model generation phase are data collection, preprocessing, data augmentation (if required), model training, and model evaluation. The first step is to collect a database containing finger vein images. Then pre-process the dataset by normalizing the images, crop vein part of the image, and resizing the images to improve the quality of the input data. The dataset size can be increased by performing data augmentation techniques such as translation, rotation, flipping etc. The augmented images are divided into training, validation, and testing sets. Design a deep learning model architecture suitable for finger vein recognition. Utilize an appropriate optimizer to optimize the model's weights based on the training data. Determine a loss function to train the model and monitor the training using metrics such as accuracy or loss. The performance of the model should be evaluated using a separate data set that was not used during the model's training or validation. Assess the accuracy of the model by examining metrics such as accuracy, precision, recall, and F1 score. Once the model has achieved satisfactory performance, deploy it in a production environment where it can be utilized for real-time finger vein recognition.

ii. Recognition Phase: During the recognition phase, the trained finger vein model takes the image of a person's finger vein as an input and outputs a prediction of whether the image belongs to the individual whose finger vein is being authenticated or not. This involves the following steps. The input image is pre-processed to normalize, crop, and resize the image

to match the input size of the trained model. The pre-processed image is fed into the model that has been trained. Based on the possible classes, a probability distribution is computed. If the probability score is above the threshold, the user is accepted as authenticated, and if it falls below the threshold, the user is rejected. The threshold for the probability score in finger vein authentication is typically set by the system administrator or the developer of the system based on the desired level of security, rate of false acceptances and rejections. Figure 1 shows the architecture of the finger vein authentication system employing deep learning techniques.

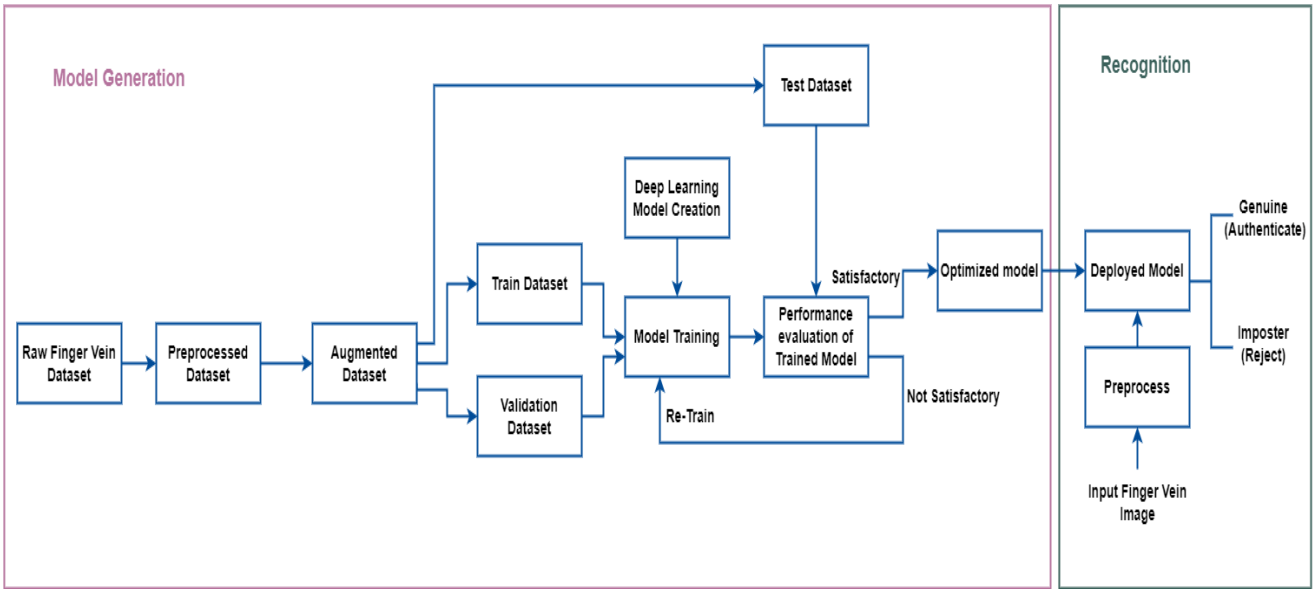


Fig. 1. Finger vein authentication system

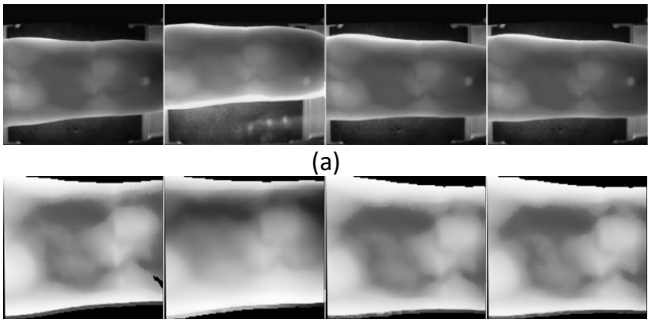
4. Experimental Setup

4.1 Database

This study uses the SDUMLA-HMT finger vein database captured by Shandong University which contains finger vein images of 106 persons. The ring, middle and index fingers of the right and left hand of all these persons are captured repeatedly 6 times which adds up to 3816 images. The images are of size 320*240 and are available in bmp format. The images are captured without using any guide bar. Since the images are prone to misalignments and shading, we can consider this to be a low-quality finger vein database.

4.2 Pre-Processing

The vein region is extracted from the captured image using a set of pre-processing operations. First the contrast of the original image was enhanced through histogram equalization. Then, the intensity values are blurred using bilateral filtering, which maintains sharp intensity changes while blurring the intensity values similar to the central pixel. Then, the image is divided horizontally, and separate masks are created for each half. The “canny edge detection method” is used to separate the region of interest (ROI), which contains the actual finger veins. Then dilation and erosion operations are applied to ROI to refine it [20]. ROI is transformed into an image with dimensions of 200x200. The original images from the “SDUMLA-HMT” database and the pre-processed and ROI extracted images are shown in Figure 2.



(a)

(b)

Fig. 2. (a) Original images from the "SDUMLA-HMT" finger vein database (b) Images after preprocessing and ROI extraction

4.3 Data Augmentation

In image augmentation, different transformations are applied to original images, resulting in multiple transformed copies. It helps in increasing the diversity of the dataset and can lead to better generalization and improved performance of machine learning models [18,19]. Depending on the augmentation technique, each copy differs in certain aspects. The dataset comprises only four images for each individual, which is insufficient for training a deep-learning model. To address this limitation, we employ various transformations such as shifting, rotation, brightness adjustments, and zooming [20].

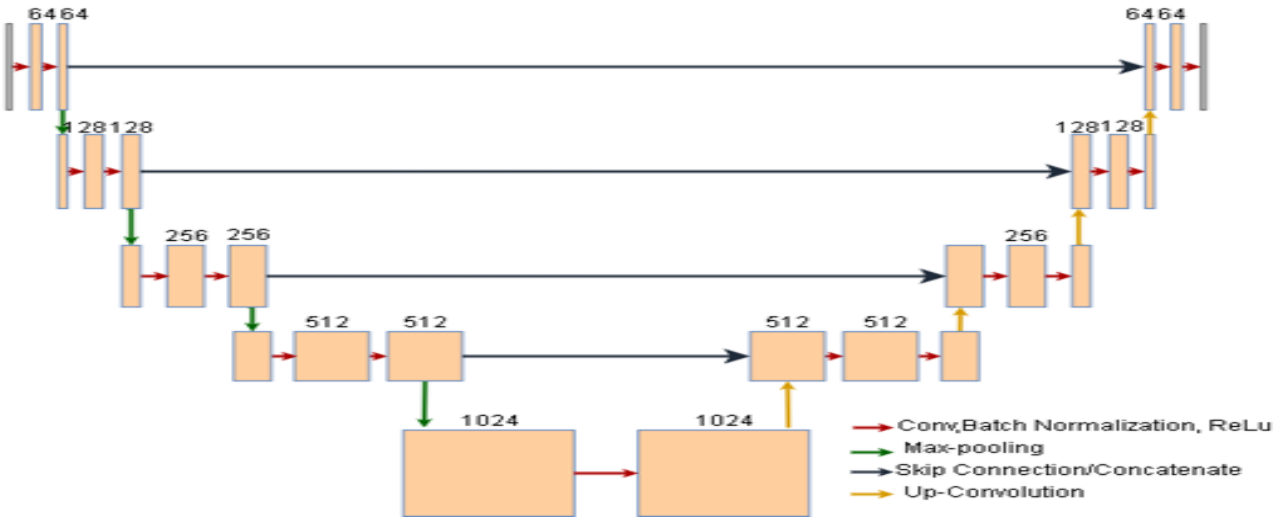


Fig. 3. Architecture of U-Net

These augmentation techniques can be mathematically denoted as follows:

Shifting or Translation: $x=x+dx$ (1)

$y' = y+dy$ (2)

In Eq. (1), x, x' are the original and shifted horizontal pixel coordinates, respectively. In Eq. (2) y, y' are the original and shifted vertical pixel coordinates, respectively. dx, dy represents the shift in x, y directions, respectively. The value of dx and dy may range from [-10,10] pixels.

Rotation: $x' = x*\cos(\theta) - y*\sin(\theta)$ (3)

$y = x*\sin(\theta) + y*\cos(\theta)$ (4)

In Eq. (3), 'x', 'x'' corresponds to the original and rotated horizontal pixel coordinates, respectively. Similarly in Eq. (4) 'y', 'y'' refer to the original and rotated vertical pixel coordinates, respectively. θ represents the angle of rotation. The value of θ may range from [-15, 15].

Brightness adjustment: $np = op + bf$ (5)

In Eq. (5) np is the new pixel value, op is the old pixel value and bf is the brightness factor which may vary between [-0.2, 0.2].

Zooming or Scaling: $x' = x * sf$ (6)

$y' = y * sf$ (7)

In Eq. (6), x, x' are the old and new horizontal pixel coordinates respectively. In Eq. (7) y, y' are the old and new vertical pixel coordinates respectively, and sf is the scaling factor which lies between [0.8,1.2].

By using the above augmentation techniques 40 augmented images per finger per person are generated. Hence a total of 240 images were available for each person and this can be used as the sequence of images which are to be input to the model. The augmented images are shown in Figure 4.

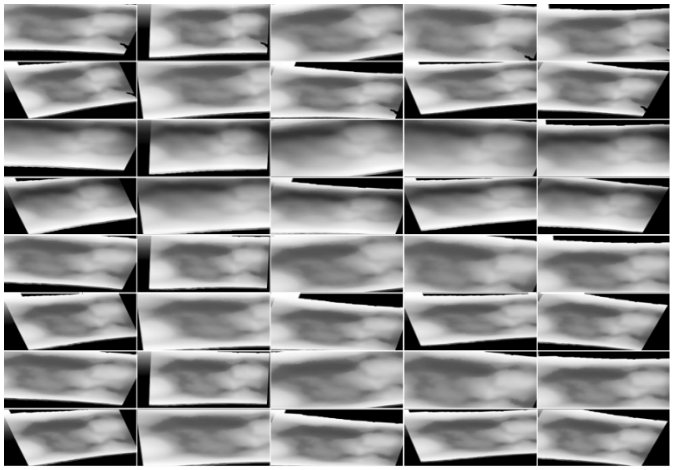


Fig. 4. Images after applying augmentation techniques

4.4 U-Net

U-Net can be used for computer vision applications, including biometric recognition. Feature extraction and matching can be performed with U-Net. The network's ability to learn to identify and extract unique patterns and features present in finger vein images can be used for matching and authentication [21].

U-Net architecture consists of two paths: a contracting one and an expanding one. Information from the input image is obtained through the contracting path, which involves extracting both

context and features, while the expanding path up samples the features and produces a segmentation mask. In finger vein recognition, the segmentation mask can be used to identify and extract vein patterns and features.

The structure of the model is described below. Several layers undergo encoding and decoding via a pair of convolutional layers, each accompanied by Batch Normalization and activated by ReLU. After a convolutional block, a max-pooling block follows within the encoder segment. Through max pooling, the spatial dimensions can be reduced, and while augmenting the count of feature channels. The decoder block performs up-sampling using Conv2DTranspose to increase the spatial dimensions and concatenates it with the corresponding skip features from the encoder block. The concatenated features are then processed by a convolution block. The encoder blocks capture hierarchical features at different scales, while the decoder blocks use skip connections to help recover spatial information lost during max pooling.

The optimizer used is Adam optimizer which uses regularization technique to prevent overfitting. The employed loss function is "binary cross entropy" which can be defined as follows.

$$\text{Binary Cross Entropy} = -\frac{1}{N}\sum_{i-1}^{N}(\mathrm{Y} * \log(\mathrm{p(y)}) \ + 1 - \mathrm{Y} * \log(1 - \mathrm{p(y)})) \quad (8)$$

In Eq. (8), N corresponds to total count of instances (persons), Y represents class 1, P(Y) stands for the probability associated with class 1, while 1-Y refers to class 0, 1-P(Y) signifies the probability for class 0. Figure 4 shows the details of U-Net architecture. The implementation details of different layers of the U-Net model with the number of parameters are shown in Figure 5.

It is necessary to train the U-Net to learn to extract unique vein patterns and features, to use it for finger vein recognition. During testing, an input finger vein image can be passed through the trained U-Net, which would extract the vein patterns and features and produce a segmentation mask. This mask can then be used for matching and identification [22].

batch_normalization	(none, 128,128,64)	256	conv2d[0][0]
activation	(none, 128,128,64)	0	batch_normalization[0][0]
conv2d_1	(none, 128,128,64)	36928	activation [0][0]
batch_normalization_1	(none, 128,128,64)	256	conv2d_1[0][0]
activation_1	(none, 128,128,64)	0	batch_normalization_1[0][0]
max_pooling2d	(none, 64,64,64)	0	activation_1[0][0]
conv2d_2	(none, 64,64,128)	73856	max_pooling2d[0][0]
batch_normalization_2	(none, 64,64,128)	512	conv2d_2[0][0]
activation_2	(none, 64,64,128)	0	batch_normalization_2[0][0]
conv2d_3	(none, 64,64,128)	147584	activation_2[0][0]
batch_normalization_3	(none, 64,64,128)	512	conv2d_3[0][0]
activation_3	(none, 64,64,128)	0	batch_normalization_3[0][0]
conv2d_12	(none, 128,128,64)	73792	activation_11[0][0]
batch_normalization_12	(none, 128,128,64)	256	conv2d_12[0][0]
activation_12	(none, 128,128,64)	0	batch_normalization_12[0][0]
conv2d_transpose_3	(none, 256,256,32)	8224	activation_12[0][0]
concatenate_3	(none, 256,256,64)	0	conv2d_transpose_3[0][0]
conv2d_13	(none, 256,256,32)	18464	concatenate_3[0][0]
batch_normalization_13	(none, 256,256,32)	128	conv2d_13[0][0]
activation_13	(none, 256,256,32)	0	batch_normalization_13[0][0]
conv2d_14	(none, 256,256,32)	9248	activation_13[0][0]
batch_normalization_14	(none, 256,256,32)	128	conv2d_14[0][0]
activation_14	(none, 256,256,32)	0	batch_normalization_14[0][0]
conv2d_15	(none, 256,256,1)	33	activation_14[0][0]

Fig. 5. Details of different layers of U-Net Model

4.5 The Proposed -Sequential Model

The proposed sequential model uses VGG16 as a feature extractor, followed by a Time Distributed layer, LSTM layer, and dense layer for classification. The images in the finger vein dataset are organized into sequences, where each sequence contains a series of finger vein images for an individual.

VGG16 architecture is used due to its strong feature extraction capabilities and the ability to learn intricate patterns from images. With VGG16 architecture, the finger veins can be extracted more effectively for proper finger vein biometric authentication because of its strong feature extraction capabilities. In this architecture, 13 convolutional layers are followed by ReLU activation for each of them, using 3x3 filters and stride 1 with zero padding. Max pooling layers with a 2x2 window and stride 2 are applied after every two convolutional layers, reducing spatial dimensions while preserving key features. The fully connected layers from VGG16 architecture are removed. For

each input image, a fixed-size feature vector is generated using the convolutional layers of VGG16 [23].

A Time Distributed layer with a flatten layer inside it is used to treat each frame of the finger vein sequence separately, while the flatten layer reshapes the data for the subsequent LSTM layer. This layer will create a sequence of feature vectors that represents the temporal information for the LSTM [24]. Figure 6 illustrates the structure of the sequential model put forth in this study.

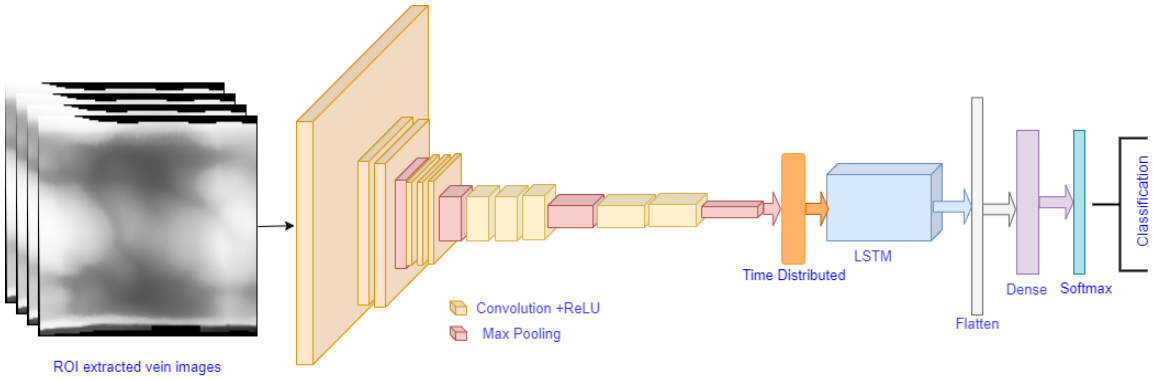


Fig. 6. The proposed Sequential model for finger vein authentication

The LSTM model will take in this sequence of feature vectors as input and learn to capture temporal dependencies between vectors. An LSTM layer with the 32 hidden unit neurons and a flatten layer is then added. A Dense layer with 106 neurons representing the 106 classes (person) and a SoftMax activation function is added for classification. The model is trained for 25 epochs, and it could produce an accuracy of 99.76%. Figure 7 displays distinct layers within the Sequential model along with their corresponding parameter counts.

Layer	Output Shape	Param#
VGG16 (Functional)	(None, 7, 7, 512)	14714688
Time Distributed	(None, 7, 7, 512)	0
LSTM	(None, 7, 100)	245200
Flatten	(None,700)	0
Dense	(None,106)	74206

Fig. 7. Details of different layers of Sequential Model

5. Results and Discussion

Table 1 presents a comparison of various parameters utilized in both models. The loss function used in both cases is binary cross entropy. The optimizer used in U-Net is Adam with an initial learning rate set as 0.001 and the Sequential model utilizes the optimizer RMS prop with the initial learning rate set as 0.0001. Adam combines the features of both RMSprop and momentum-based approaches whereas RMS prop uses second order momentum only for adjusting the learning rates. RMSprop has fewer hyperparameters which makes it easier to tune and use.

It is important to note that the U-Net model underwent a lengthy training period of 40 epochs, during which time it consistently achieved an accuracy of 95.7% from the 20th epoch to the 40th epoch. The sequential model, in contrast, provided a significantly higher level of performance by achieving an accuracy of 99.76% across 25 epochs, outperforming the U-Net model by a considerable margin by training through 25 epochs. Being a biometric finger vein authentication technique, it is crucial to achieve this high level of accuracy to authenticate genuine and imposter in the real time scenarios.

Table 1
Comparison of parameters in U-Net and Sequential Model

	U-Net	Sequential Model
Loss function	"Binary Cross Entropy"	"Binary Cross Entropy"
Optimizer	"Adam Optimizer"	"RMSprop optimizer"
Initial Learning Rate	1e-3	1e-4
No of Epochs Trained	40	25
Evaluation Metric	Accuracy 95.7%	Accuracy 99.7%

A comparison of the count of parameters in both models are displayed in Table 2. Total Parameters refer to both trainable and non-trainable parameters, including weights and biases. Trainable Parameters are a subset of total parameters that undergo learning and updates during the training process to optimize the model. Non-trainable parameters encompass the remaining parameters that remain unaltered during the training process.

The U-Net model has a relatively large number of total parameters, indicating that it is a complex model with a significant number of learnable weights and biases. Around 31,793,480 of these parameters are trainable, which means they can be updated during training to enhance the model's efficiency. The remaining 5,440 parameters are not trainable, whose values are fixed throughout training.

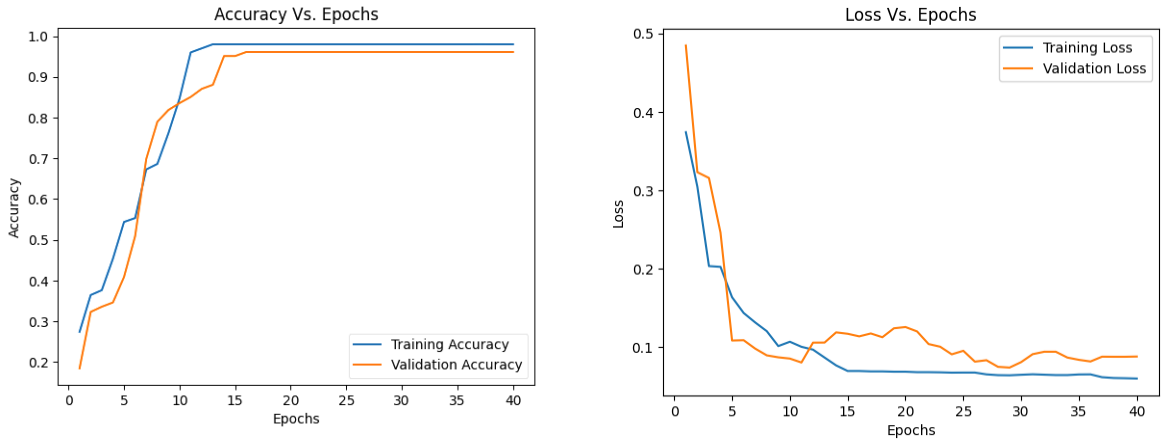
The Sequential model has a total parameter count of 15,034,094, which is smaller than the U-Net model. The trainable parameters amount to 319,406, while the non-trainable parameters are much larger, at 14,714,688.

Table 2
Comparison of parameters in U-Net and Sequential model

	Total Parameters	Trainable Parameters	Non-Trainable Parameters
U-Net	31,798,920	31,793,480	5,440
Sequential Model	15,034,094	319,406	14,714,688

The large count of parameters to be trained in the U-Net model indicates a more complex model. The smaller count of non-trainable parameters in the Sequential model suggests faster training and resource efficient operation.

Figure 8 shows the accuracy and loss graph for the U-Net model. The plot indicates that the model has been converged with a validation accuracy of 95.7% as the accuracy remains stable from 18th epoch to 40th epoch which indicates that further optimization may not lead to significant improvement in the accuracy.



(a) (b)

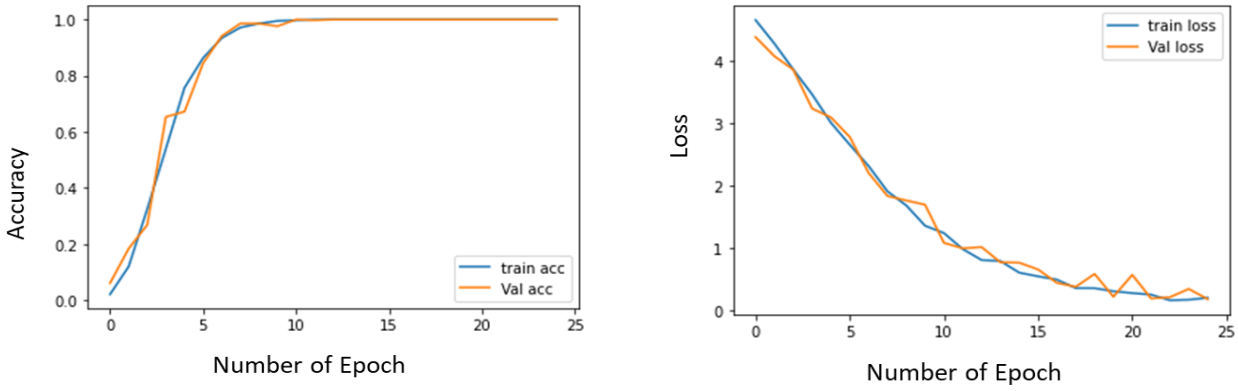
Fig. 8. U-Net Model: (a) Accuracy Vs Number of Epochs plot (b) Loss Vs Number of Epochs plot

The accuracy and loss graph for the Sequential model is plotted in Figure 9. The plot indicates that the model has been converged with a validation accuracy of 99.7%.

An authentication system's effectiveness and reliability can be evaluated by considering the evaluations metrics like accuracy, Precision, Recall, F1-Score, Kappa Score and Matthews Correlation Coefficient as defined below.

Accuracy is computed by dividing the count of accurate predictions by the total number of predictions made.

$$\text{Accuracy} = \frac{True\ Positives + True\ Negatives}{Total\ Predictions} \quad (9)$$



(a) (b)

Fig. 9. U-Net Model: (a) Accuracy Vs Number of Epochs plot (b) Loss Vs Number of Epochs plot

Precision measures the correctly predicted true positives to the total positive predictions.

$$\text{Precision} = \frac{True\ Positives}{Total\ Positive\ \ Predictions} \quad (10)$$

The proportion of correctly predicted positive instances relative to the total number of actual positive instances is referred to as recall, sensitivity, or true positive rate.

$$\text{Recall} = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (11)$$

In the F1-Score, precision and recall are combined to form a harmonic mean.

$$\text{F1-Score} = \frac{2(Precision * Recall)}{Precision + Recall} \quad (12)$$

The Kappa Score, also referred to as Cohen's Kappa Coefficient, quantifies the agreement between predicted and actual labels, considering the potential for random agreement. It provides insight into the quality and reliability of the model.

$$\text{Kappa} = \frac{(Po - Pe)}{(1 - Pe)} \quad (13)$$

"Po" denotes the fraction of observed agreement and "Pe" denotes the fraction of expected agreement by chance.

The Matthews Correlation Coefficient evaluates the effectiveness of binary classifications, considering true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

$$\text{MCC} = \frac{(TP * TN - FP * FN)}{\sqrt{((TN + FN)(TP + FP)(TP + FN)(TN + FP))}} \quad (14)$$

Table 3 shows various metric values for the different deep learning models. The sequential model exhibits better performance compared to other models, suggesting its suitability for finger vein authentication.

Table 3
Comparison of evaluation metrics for U-Net and Sequential Model

	Accuracy	Precision	Recall	F1-Score	Kappa Score	Matthews Corrcoef
AlexNet [25]	0.7020	0.7010	0.9030	-	-	-
ResNet18 [24]	0.9713	0.9837	0.9587	-	-	-
VGGNet16 [24]	0.9710	0.9861	0.9554	-	-	-
VGG19 [25]	0.8757	0.88	0.78	-	-	-
U-Net	0.9571	0.9702	0.9750	0.973	0.664	[illegible]
Sequential	0.9976	0.9742	0.9976	0.986	[illegible]	0.990

The graphical representation of the basic evaluation metrics has been provided in Figure 10 for better comparison.

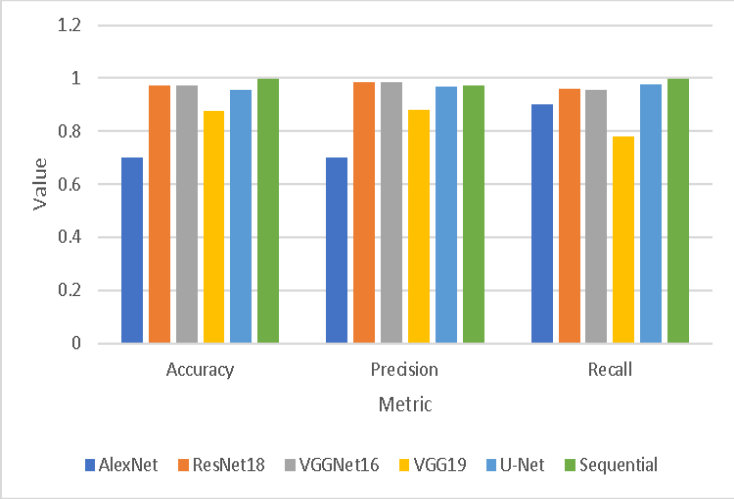


Fig. 10. Comparison of evaluation metric for different models

5. Conclusion

As the finger vein biometric authentication gains more importance in the scenario of contactless authentication system, the significance for accurate and reliable authentication is evident. This study addresses the challenges posed by low quality finger vein images through a comparative analysis of two deep learning architectures: U-Net and the proposed sequential model. The investigation brings out the capability of deep learning models to overcome the limitations of low-quality images. U-Net demonstrated its capacity to extract relevant features from noisy inputs, whereas the sequential model with modified VGG16 architecture and integration of LSTM layers introduced temporal context further enhancing accuracy and reliability. This research sheds light on the effectiveness of finger vein authentication using low quality images. The primary challenge faced in this work is the training time required to find an optimal architecture for achieving this accuracy, which could be improved by future researchers.

Acknowledgement
This research was not funded by any grant.

References
[1] Zayed, Hossam L., Heba M. Abdel Hamid, Yasser M. Kamal, and Abdel Halim A. Zekry. "A comprehensive survey on finger vein biometric." *Journal of Advances in Information Technology* 14, no. 2 (2023). https://doi.org/10.12720/jait.14.2.212-223

[2] Uhl, Andreas. "State of the art in vascular biometrics." *Handbook of Vascular Biometrics* (2020): 3-61. https://doi.org/10.1007/978-3-030-27731-4_1

[3] Wang, Peng, and Huafeng Qin. "Palm-vein verification based on U-Net." In *IOP Conference Series: Materials Science and Engineering*, vol. 806, no. 1, p. 012043. IOP Publishing, 2020. https://doi.org/10.1088/1757-899X/806/1/012043

[4] 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. https://doi.org/10.3390/app13116363

[5] Liu, Zhi, Yilong Yin, Hongjun Wang, Shangling Song, and Qingli Li. "Finger vein recognition with manifold learning." *Journal of Network and Computer Applications* 33, no. 3 (2010): 275-282. https://doi.org/10.1016/j.jnca.2009.12.006

[6] Hong, Hyung Gil, Min Beom Lee, and Kang Ryoung Park. "Convolutional neural network-based finger-vein recognition using NIR image sensors." *Sensors* 17, no. 6 (2017): 1297. https://doi.org/10.3390/s17061297

[7] Chawla, Bhavya, Shikhar Tyagi, Rupav Jain, Archit Talegaonkar, and Smriti Srivastava. "Finger vein recognition using deep learning." In *Proceedings of International Conference on Artificial Intelligence and Applications: ICAIA 2020*, pp. 69-78. Springer Singapore, 2021. https://doi.org/10.1007/978-981-15-4992-2_7

[8] Kharbas, Pratiksha, Arati Deshmukh, Siddhi Khalate, and Sumit Shinde. "Finger Vein Detection Based On Deep Learning."

[9] Jalilian, Ehsaneddin, and Andreas Uhl. "Finger-vein recognition using deep fully convolutional neural semantic segmentation networks: The impact of training data." In *2018 IEEE international workshop on information forensics and security (WIFS)*, pp. 1-8. IEEE, 2018. https://doi.org/10.1109/WIFS.2018.8630794

[10] Gumusbas, Dilara, Tulay Yildirim, Mustafa Kocakulak, and Nurettin Acir. "Capsule network for finger-vein-based biometric identification." In *2019 IEEE symposium series on computational intelligence (SSCI)*, pp. 437-441. Ieee, 2019. https://doi.org/10.1109/SSCI44817.2019.9003019

[11] Huang, Zhe, and Chengan Guo. "Robust finger vein recognition based on deep CNN with spatial attention and bias field correction." *International Journal on Artificial Intelligence Tools* 30, no. 01 (2021): 2140005. https://doi.org/10.1142/S0218213021400054

[12] Jalilian, Ehsaneddin, and Andreas Uhl. "Enhanced segmentation-CNN based finger-vein recognition by joint training with automatically generated and manual labels." In *2019 IEEE 5th international conference on identity, security, and behavior analysis (ISBA)*, pp. 1-8. IEEE, 2019. https://doi.org/10.1109/ISBA.2019.8778522

[13] Kalaivani, K., Pravin R. Kshirsagarr, J. Sirisha Devi, Surekha Reddy Bandela, Ilhami Colak, J. Nageswara Rao, and A. Rajaram. "Prediction of biomedical signals using deep learning techniques." *Journal of Intelligent & Fuzzy Systems* Preprint (2023): 1-14. https://doi.org/10.3233/JIFS-230399

[14] Tamilarasi, K., K. Maheswari, S. Ramesh, Samson Isaac, and A. Rajaram. "A Decentralized Smart Healthcare Monitoring System using Deep Federated Learning Technique for IoMT." (2023).

[15] Yagoub, Sami Abdelrahman Musa, Gregorius Eldwin Pradipta, and Ebrahim Mohammed Yahya. "Prediction of bubble point pressure for Sudan crude oil using Artificial Neural Network (ANN) technique." *Progress in Energy and Environment* (2021): 31-39.

[16] Karim, Abdul Razif Abdul, and Roslina Mohammad. "Meta-study of sensitivity analysis in solar renewable energy application." *Progress in Energy and Environment* (2023): 14-25. https://doi.org/10.37934/progee.23.1.1425

[17] Chiranjeevi, Phaneendra, and A. Rajaram. "A lightweight deep learning model based recommender system by sentiment analysis." *Journal of Intelligent & Fuzzy Systems* Preprint (2023): 1-14. https://doi.org/10.3233/JIFS-223871

[18] Harikrishnan, G., and A. Rajaram. "Improved throughput based recognition connection denies for aggressive node in wireless sensor network." *Journal of Computational and Theoretical Nanoscience* 14, no. 12 (2017): 5748-5755. https://doi.org/10.1166/jctn.2017.7008

[19] Rosario, B. BERIL DANIEL, and A. Rajaram. "CDMA based secure cross layer framework for authentication and scheduling in MANET." *Journal of Theoretical and Applied Information Technology* 69, no. 3 (2014).

[20] Mukundan, V., A. Rajaram, and S. Gopinath. "Securing Mobile Ad Hoc Network using Double Hash Authentication technique."

[21] Palaniswami, S., and Ayyasamy Rajaram. "An enhanced distributed certificate authority scheme for authentication in mobile ad hoc networks." *The International Arab Journal of Information Technology (IAJIT)* 9, no. 3 (2012): 291-298.

[22] Rajaram, A., and S. Palaniswami. "A high certificate authority scheme for authentication in mobile ad hoc networks." *International Journal of Computer Science Issues (IJCSI)* 7, no. 4 (2010): 37.

[23] Indira, D. N. V. S. L. S., Rajendra Kumar Ganiya, P. Ashok Babu, A. Xavier, L. Kavisankar, S. Hemalatha, V. Senthilkumar *et al.,* "Improved artificial neural network with state order dataset estimation for brain cancer cell diagnosis." *BioMed Research International* 2022 (2022). https://doi.org/10.1155/2022/7799812

[24] Sathiyaraj, K., and A. Rajaram. "An optimized design modelling of neural network based green house management system using solar and rectenna." *DYNA–Ingen. Ind* 97, no. 1 (2022): 85-91. https://doi.org/10.6036/10089