IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 24 October 2024, accepted 11 November 2024, date of publication 14 November 2024,
date of current version 29 November 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3498601

RESEARCH ARTICLE

Enhancing Finger Vein Recognition With Image Preprocessing Techniques and Deep Learning Models

U. SUMALATHA[1], K. KRISHNA PRAKASHA[1], (Senior Member, IEEE), SRIKANTH PRABHU[2], (Senior Member, IEEE), AND VINOD C. NAYAK[3]

[1]Department of Information and Communication Technology, Manipal Institute of Technology (MIT), Manipal Academy of Higher Education (MAHE), Manipal, Karnataka 576104, India
[2]Department of Computer Science and Engineering, MIT, MAHE, Manipal, Karnataka 576104, India
[3]Department of Forensic Medicine, Kasturba Medical College (KMC), MAHE, Manipal, Karnataka 576104, India

Corresponding author: K. Krishna Prakasha (kkp.prakash@manipal.edu)

ABSTRACT The use of finger veins for biometric authentication is increasingly popular; however, low-quality images present significant challenges that necessitate innovative approaches for accurate identification. This study investigates the effectiveness of various image preprocessing techniques to enhance and analyze finger vein images for biometric recognition. We utilized a finger vein dataset from Kaggle, comprising diverse images captured under controlled conditions. Our preprocessing methods included sharpening images using convolution kernels to improve edge definition, employing thresholding techniques (simple binary, adaptive, and Otsu's) for effective image segmentation, and applying morphological operations (erosion, dilation, opening, and closing) to refine object shapes and reduce noise. We also implemented edge detection methods, including Sobel, Laplacian, and Canny, to identify significant boundaries within the images. Image resizing was performed using linear, cubic, and area interpolation to assess their effects on image quality. Additionally, various filtering techniques–such as Kalman, median, and Gaussian filters–were applied to reduce noise and enhance image clarity. The dataset comprised 3,816 images from 106 individuals, split into two configurations: 80-10-10 and 70-15-15. We assessed models such as VGG16, VGG19, ResNet101, AlexNet, MobileNet, DenseNet201, and EfficientNet based on metrics including accuracy, precision, recall, F1-score, and training time. VGG16, VGG19, and ResNet101 achieved accuracies of 99.9%, 99.8%, and 99.8%, respectively. Data augmentation techniques generated 76,320 augmented images, significantly improving model performance, especially for the 80-10-10 split. Visualizations through radar and bar charts indicated that VGG16, VGG19, and ResNet101 delivered the highest performance metrics, while DenseNet201 exhibited a slight decline in the 70-15-15 split due to increased test data. Overall, the findings demonstrate the models' efficacy for reliable finger vein biometric recognition, contributing to advancements in biometric authentication systems.

INDEX TERMS Finger vein, biometric recognition, deep learning, image preprocessing, feature extraction, biometric authentication.

I. INTRODUCTION

Biometric authentication has gained significant traction as a secure and efficient way to verify personal identity in modern security systems. Among the diverse range of biometric techniques, finger vein authentication stands out for its unique advantages in terms of security and reliability. This method involves capturing the vein patterns beneath the skin of a finger using near-infrared (NIR) light, making it particularly difficult to forge or counterfeit. Unlike

The associate editor coordinating the review of this manuscript and approving it for publication was Zahid Akhtar.

surface-based biometric traits, such as fingerprints or facial features, finger vein patterns are concealed within the body, offering a higher degree of protection against tampering and environmental degradation [1].

Finger vein authentication utilizes the distinct vascular structure inside the finger, which is highly individualized and remains stable over time. The uniqueness of vein patterns from person to person and across different fingers of the same individual provides an exact identification mechanism. NIR imaging technology is employed to detect these vein patterns, as the hemoglobin in the blood absorbs infrared light, creating a clear and distinguishable contrast between the veins and surrounding tissue. This method makes finger vein patterns difficult to replicate, enhancing the modality's security [2].

One of the critical benefits of finger vein technology is its built-in liveness detection [3]. Since the system relies on real-time blood flow within the veins, it can effectively distinguish between a live person and an artificial attempt to spoof the system using fake biometric samples [4]. This characteristic provides a robust defense against presentation attacks, where forged biometric data is used to deceive the system. Additionally, finger vein recognition systems are often designed to be non-contact or minimally invasive, ensuring hygiene and ease of use, particularly in environments requiring frequent and secure authentication, such as healthcare, banking, and high-security facilities [5]. When combined with multi-factor authentication–pairing vein with other biometric traits or security factors–finger vein systems provide an even stronger layer of defense against various cybersecurity threats, including identity theft and spoofing attacks [6].

Key contributions of this work include: This study evaluates the use of several popular pre-trained CNN architectures for finger vein authentication, including VGG16, VGG19, AlexNet, MobileNet, DenseNet201, EfficientNet, and ResNet101. These architectures, initially trained on large-scale datasets like ImageNet, have demonstrated remarkable performance in various image classification tasks and are now adapted to optimize their effectiveness for biometric recognition, specifically finger vein authentication. To enhance the performance of these models, we implement preprocessing techniques such as image resizing, grayscale conversion, binarization, noise reduction, and morphological processing. These preprocessing steps are crucial for improving the quality of the input images and ensuring that the CNNs can effectively learn the distinguishing features of finger vein patterns.

The remainder of this paper is organized as follows: Section II describes the related work, Section III delves into the image preprocessing techniques and the overview of CNN Architectures for Finger Vein Recognition and, Section IV describes the proposed methodology, Section V presents the experimental setup utilizing CNN Architectures and comparative results of each architecture, and Section VI concludes the study by highlighting the key findings and future research directions.

II. RELATED WORK

Hegde et al. proposed a novel approach for authentication using finger-vein images, addressing the growing security concerns of the current era. Recognizing that biometric methods are more reliable and accurate for individual authentication, the study utilized a basic CNN enhanced by transfer learning. The model was pre-trained on a diverse set of images from the ImageNet database using the ResNet-50 architecture. The methodology involved running the pre-trained model through the CNN framework, incorporating an appropriate number of hidden layers and activation functions, while employing suitable optimizers and loss functions for effective classification. The results demonstrated the model's effectiveness, achieving an impressive accuracy of 99.06% in classifying individuals based on their finger-vein images [7].

Madhusudhan et al. proposed an intelligent deep learning-based finger vein recognition (IDL-FVR) model to enhance biometric authentication systems. This model analyzes finger vein patterns for accurate identification by comparing captured data with a database. It includes stages such as acquisition using infrared imaging, preprocessing, feature extraction, and authentication. A region of interest extraction isolates the relevant finger portion, while a shark smell optimization algorithm tunes the hyperparameters of a bidirectional long-short-term memory (BiLSTM) network. The authentication process relies on Euclidean distance to assess similarity between images. The IDL-FVR model achieved an impressive accuracy of 99.93%, demonstrating effective authentication when the Euclidean distance is minimal [8].

Boucetta et al. proposed a finger-vein identification system that leverages vein patterns for biometric authentication, highlighting its advantages over traditional fingerprint methods. The methodology utilized a Squeezenet pretrained Deep-Convolutional Neural Network (Deep-CNN) as a feature extractor for left and right finger vein patterns. Features were combined using Discriminant Correlation Analysis (DCA) to reduce dimensions, and the resulting vectors served as input for a Support Vector Machine (SVM) classifier. Evaluated on the SDUMLA-HMT and FV-USM databases, the system achieved impressive accuracy rates, reaching 97.78% with the right index finger and 98.64% with the left middle finger. Notably, accuracy soared to 99.81% using five images for training on the SDUMLA database [9].

Sathishkumar et al. proposed a biometric authentication system utilizing finger vein recognition, which offered a secure and accurate method for user verification. This approach analyzed the unique vascular patterns beneath the skin. The methodology began with capturing finger vein images, which were preprocessed using Gaussian median filters to reduce noise. Segmentation was conducted through line tracking to improve image contrast. Features were extracted using CNNs, followed by deep learning classification to differentiate between genuine and impostor

TABLE 1. Summary of biometric authentication techniques - feature engineering.

Year	Feature Extraction Technique	Classification Technique	Dataset	Results	Advantages	Limitations
2024 [14]	Mask Condition, Fourier-space Attention	Diffusion Model	USM, THU-MVFV3V	USM: Dice: 84.35%, clDice: 93.30%, ACC: 88.18%, AUC: 89.33%, mIoU: 72.94%; THU-MVFV3V: Dice: 86.47%, clDice: 93.16%, ACC: 92.22%, AUC: 96.16%, mIoU: 76.17%	Concurrent handling of segmentation and authentication tasks, enhancing both processes.	Dataset specifics not mentioned; performance may vary across different datasets.
2024 [15]	Local Binary Pattern (LBP)	Adaptive Neuro-Fuzzy Inference System (ANFIS)	SDUMLA-HMT	Accuracy: 97.72%, FAR: 3.1%, FRR: 1.41%	Low risk of duplication, high accuracy, and minimal invasiveness	Traditional methods require accuracy improvements
2024 [16]	Ridge-based matching, Minutiae-based matching	Dempster-Shafer theory, Holistic fusion, Non-linear fusion, Sum rule-based fusion	NUPT-FPV	EER: 0% (Dempster-Shafer), 14% (other techniques); FAR: 0% for all techniques	0% FAR, 0% EER with Dempster-Shafer fusion, High security	Limited information on other performance metrics
2024 [17]	Multiscalar feature extraction	Bilinear fusion technique	Kaggle	Accuracy: 97.7%	Enhanced recognition accuracy and robustness	Limited discussion on preprocessing, scalability

patterns. Real-time scanning was managed by an Arduino board, with subsequent processing in MATLAB and results communicated via a GSM module, while data was also stored in an IoT framework. The system demonstrated an accuracy of 96%, making it suitable for various security applications [10].

Mathew et al. The author proposed a study to explore the effectiveness of deep learning techniques for finger vein biometric authentication, particularly in the context of low-quality images. Utilizing a dataset of diverse low-quality finger vein images, the research focused on two architectures: U-Net for image segmentation and feature extraction, and a modified Sequential Model incorporating VGG16 and LSTM layers for temporal context. The results revealed that U-Net achieved a precision of 0.9571, recall of 0.9702, and an F1-Score of 0.9750, along with a Kappa Score of 0.973 and a Matthews Correlation Coefficient of 0.664. In contrast, the Sequential Model outperformed with a precision of 0.9976, recall of 0.9742, and an F1-Score of 0.9976, as well as a Kappa Score of 0.986 and a Matthews Correlation Coefficient of 0.989 [11].

Wang et al. proposed a frequency-spatial coupling network (FVFSNet) for finger vein authentication, integrating features from both spatial and frequency domains to enhance accuracy. The model included three components: the frequency domain processing module (FDPM), the spatial domain processing module (SDPM), and the frequency-spatial coupling module (FSCM). Experiments on nine publicly available datasets, including USM-FV and SDUMLA-HMT, showed that FVFSNet achieved state-of-the-art performance with low computational cost. Notably, removing the FDPM increased the weighted average equal error rate (EER) by 1.20%, highlighting its essential role in effective feature extraction and confirming the benefits of combining spatial and frequency domain features for improved biometric authentication [12].

A. FINGER VEIN RECOGNITION BASED ON FEATURE ENGINEERING

In the initial phase of research on finger vein authentication, feature engineering methods were widely adopted. Numerous effective techniques emerged during this time, which can be classified into three main categories: vein pattern-based methods, local descriptor-based methods, and subspace learning-based methods. Vein pattern-based methods specifically emphasize the characteristics of finger veins, including their shape, position, and orientation. In a recent study by Sai Kishore and Magesh Kumar [13], a finger vein recognition method was evaluated using the PolyU Fingerprint Database, achieving an impressive finger vein accuracy of 98.56% compared to a Message Digest 5 (MD5) accuracy of 93.64%, with a statistically significant p-value of 0.0218. The results indicate a higher accuracy for the finger vein method, underscoring its effectiveness in biometric recognition applications. However, the study was limited in scope, involving only 170 individuals, divided equally into two groups of 85, which may affect the generalizability of the findings. This limitation highlights the need for further research with larger and more diverse populations to validate the robustness of the finger vein recognition method [13]. Summary of Biometric Authentication Techniques based on Feature Engineering is given in Table 1.

TABLE 2. Summary of biometric authentication techniques - deep learning (Part 1).

Year	Techniquew Used	Dataset	Results	Advantages	Limitations
2019 [18]	Finger vein pattern extraction via NIR imaging, CNN (AlexNet transfer learning)	Custom dataset (images from four classes)	Experiment 1: 91% accuracy, AUC: 0.942; Experiment 2: 100% accuracy, AUC: 1; Experiment 3: 100% accuracy, AUC: 1	High accuracy, secure against forgery, efficient execution	Limited dataset size; NIR imaging may require specific conditions
2022 [19]	Fingernail and Finger Knuckle Print extraction, CNN (AlexNet)	1090 hand dorsal images (10 from 109 persons)	Identification accuracy: 94.75%; GAR up to 88.63% at FAR 0.003%	Robustness against spoofing; low computational cost	Limited to a specific dataset; future work needed for larger datasets
2023 [10]	Finger vein image processing with Gaussian median filter; line tracking for segmentation, CNN	Custom dataset of finger vein images	96% accuracy	High security and accuracy, real-time communication via GSM, IoT integration for remote monitoring	Dependence on image quality; may require specific scanning conditions
2024 [20]	Non-Negative Matrix Factorization (NMF), Lightweight Deep Learning Model	NUPT-FPV, FVC2004, FV-USM	95.31% accuracy (NUPT-FPV), 92.71% accuracy (FVC2004 & FV-USM), Low EER	Improves privacy, enhances security, real-time processing, resistant to spoofing	Computationally complex, dependent on multimodal integration for full effectiveness
2024 [21]	Convolutional Multihead Attention Network (VeinAtnNet), Score-Level Fusion	FV-300, FV-USM, FV-PolyU	EER: 1.13%, FAR: 0.01%	Lightweight, fewer parameters, resists presentation attacks, better performance with enhanced images	Relies on specific image regions (minutiae points), needs multiple datasets for comprehensive testing
2024 [22]	Vision Transformer (ViT)	SDUMLA HMT (4000 images, 106 subjects)	EER: 0.025, Accuracy: 98.5%	Improved feature understanding, scalability, and generalization compared to CNNs	Focus on closed protocol scenarios; need for tailored attention mechanisms for small datasets
2024 [23]	CNN with ReLU activation function, Correlation-based Matching	SDUMLA-HMT	Max Accuracy: 99.83%, FAR: 0.11, FRR: 0.14, EER: 0.125	Improved accuracy and robustness over unimodal systems	Security risks in storing biometric templates
2024 [24]	CNN algorithm for finger vein detection, Embedded processing for recognition	customized database	Mean: 41.5751; Std. Dev.: 51.2256; MSE: 9.9691e-0.4; PSNR: 78.1443; Entropy: 5.0595	Remote monitoring via internet, Transaction verification with GSM, Real-time processing	Limited information on system scalability

B. FINGER VEIN RECOGNITION BASED ON DEEP LEARNING

The swift advancements in deep learning have highlighted the remarkable capabilities of CNNs in a wide range of image-processing applications. This has garnered significant interest among researchers focusing on finger vein authentication. Consequently, deep learning techniques have emerged as a vital resource for enhancing the accuracy and effectiveness of finger vein authentication systems. Through the application of these methods, researchers have introduced several innovative approaches to address various challenges within this field. A summary of Biometric Authentication Techniques based on Deep Learning is given in Tables 2, 3.

III. IMAGE PREPROCESSING TECHNIQUES AND CNN ARCHITECTURES USED FOR FINGER VEIN RECOGNITION

A. DATASET

For this experiment, we employed a finger vein image dataset from Kaggle [33], a widely recognized benchmark in finger vein image classification. The dataset consists of 3816 images collected from 106 individuals during a single session. The images are organized into 106 distinct folders, with each folder containing subfolders for both the left and right hands. Each subfolder includes 18 images from the index, middle, and ring fingers of both hands. The images are of high quality, ensuring detailed analysis for classification tasks. This dataset is particularly suitable for our study as it provides a comprehensive representation of finger vein patterns, enabling effective model training and evaluation. Prior to analysis, we performed necessary preprocessing steps, including normalization and data augmentation, to enhance the robustness of our model.

B. IMAGE PROCESSING TECHNIQUES

In this study, we employed various image processing techniques to enhance and analyze images from our dataset. The following sections detail the methods used, accompanied by relevant figures illustrating the results.

1) IMAGE SHARPENING

We first sharpened an image from the dataset using a convolution kernel designed for sharpening. The kernel applied is defined as follows:

$$\text{Kernel} = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix} \tag{1}$$

This kernel emphasizes edges by enhancing high-frequency components, resulting in a clearer image. Figure 1 shows the original and sharpened images side by side.

2) THRESHOLDING TECHNIQUES

Subsequent to sharpening, we applied various thresholding techniques to binarize the image. The first method employed

TABLE 3. Summary of biometric authentication techniques - deep learning (Part 2).

Year	Techniques Used	Dataset	Results	Advantages	Limitations
2022 [25]	Transfer Learning (VGG16, VGG19, AlexNet, Inception V3), Deep Learning (CNN)	SDUMLA Vein Image Dataset	Highest accuracy of 99.99%	High security and difficult to spoof; effective with varying image quality	Performance may be affected by low-quality images; requires a larger dataset for improved generalization
2024 [26]	Discrete Wavelet Transform (DWT), ResNet-18	SDUMLA, FVUSM	ASR: 95.36%, Rad: 0.37% (FVUSM)	High attack performance; capable of bypassing backdoor defenses	Vulnerable to various types of backdoor attacks
2024 [27]	Vein Pattern Attention Module (VPAM), VPCFormer (Transformer-based model)	THUMVFV-3V	Accuracy: 99.80	Addresses 3D feature loss; robust to finger rotation and translation	Limited studies on multi-view recognition before this work
2024 [28]	Long Short-Term Memory (LSTM), Deep Stacked Auto Encoder (DSAE)	Finger vein dataset (proposed)	FAR: 0.0290 - 0.0749, FRR: 3.8302 - 5.717, Accuracy: 98.91% - 98.95%, EER 1.0445 - 1.0890	High accuracy, robust to variations, effective image preprocessing	Relatively limited comparative performance against existing models (VGG-16, CNN, etc.)
2024 [29]	Large kernel and attention mechanism, Let-Net	FV_USM dataset	EER: 0.04%, Accuracy: 99.77%	Low parameter count (0.89M), low FLOPs (0.25G), effective at capturing local and global information	Potential challenges in extremely diverse lighting or skin conditions
2024 [12]	BWR-ROIAlign (Localization, Compression, Transformation), CFVNet	Four public datasets	Accuracy: 99.82%, EER: 0.01%, Dsys 0.025	End-to-end cancelable biometric system, integrated preprocessing, and template protection	Limited studies on fully integrated secure systems
2024 [30]	Local Binary Pattern (LBP), Gabor Filter, Adaptive Neuro-Fuzzy Inference System (ANFIS), Support Vector Machine (SVM)	SDUMLA-HMT	Accuracy: 97.72%, FAR: 3.1%, FRR: 1.41%	Low risk of duplication, High accuracy, Non-invasive	Traditional methods are unreliable
2024 [31]	Pyramidal convolution with scale-aware attention, SA attention module	FV-USM	Accuracy: 99.92%, EER: 1.02%	Improved performance over PA attention	Security aspects not covered
2024 [32]	GABOR and LPQ filters for ROI extraction, KNN with various distance metrics	Synthetic dataset	Recognition rate: 100%	High accuracy and forgery resistance	Large image set not tested

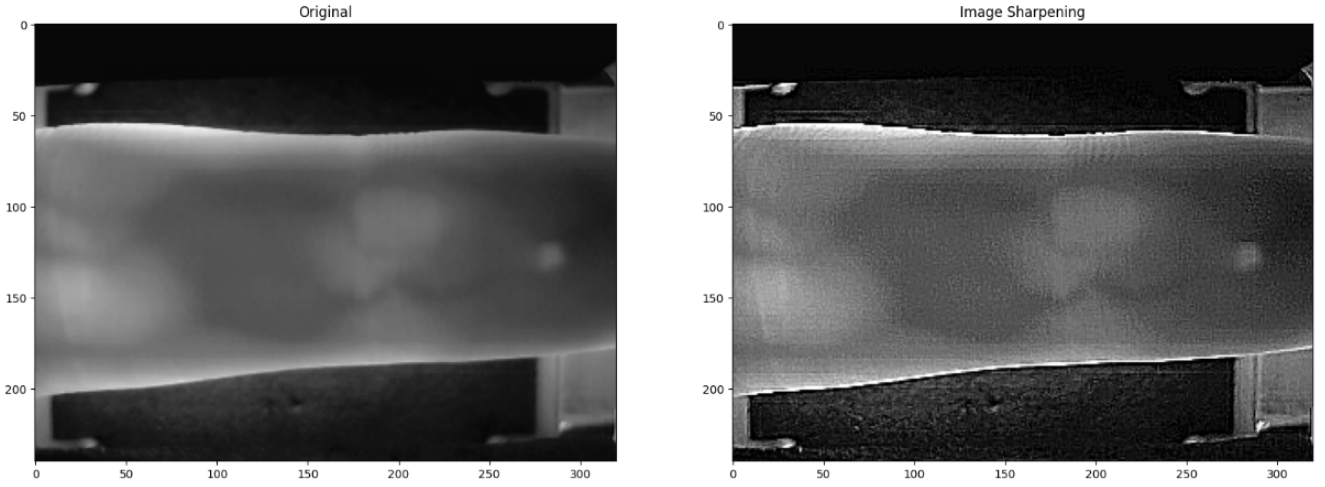


FIGURE 1. Original finger vein image (left) and sharpened finger vein image using convolutional filtering (right).

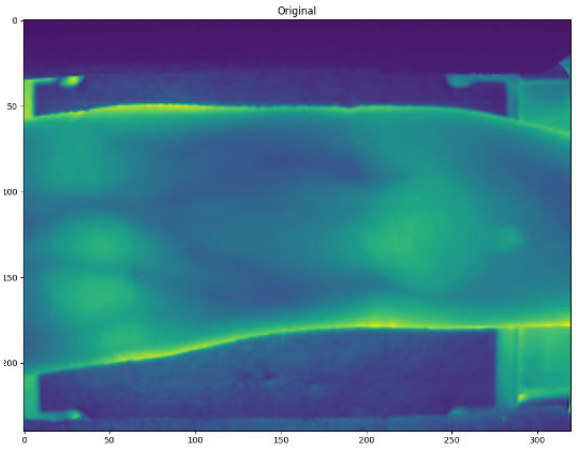
was simple binary thresholding, setting a threshold value of 127. Any pixel value above this threshold was set to 255 (white), while those below were set to 0 (black). The result is depicted in Figure 2.

The next technique was adaptive thresholding, specifically using the mean of the pixel values in a neighborhood around each pixel. This method accounts for varying lighting conditions in the image, as shown in Figure 3.

Additionally, we implemented Otsu's thresholding method, which automatically calculates the optimal threshold value to minimize intra-class variance. The results are presented in Figure 4.

3) MORPHOLOGICAL OPERATIONS

Morphological operations, including erosion and dilation, were utilized to refine the shapes of objects within the image.



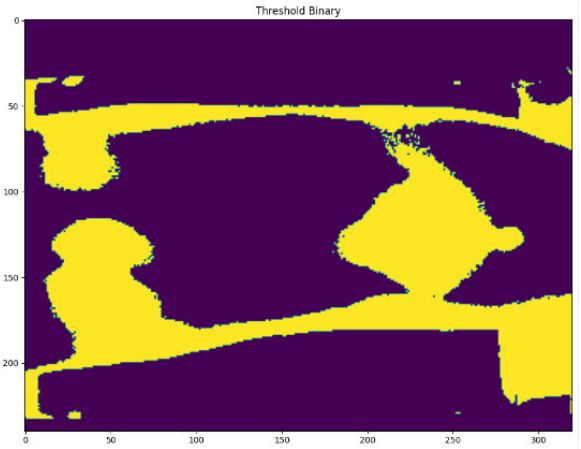
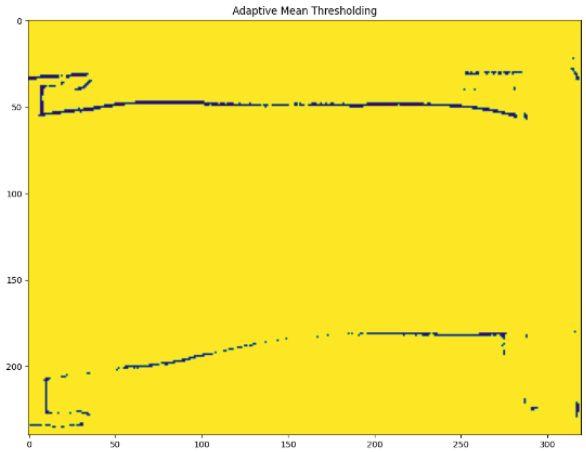


FIGURE 2. Threshold binary image obtained through simple thresholding.



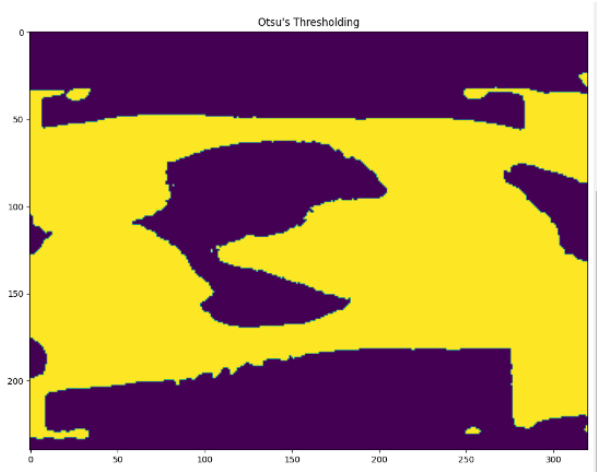


FIGURE 3. Adaptive mean thresholding result, demonstrating improved handling of varying lighting conditions.

FIGURE 4. Otsu's thresholding result, showing optimal binarization by automatically selecting the threshold.

Erosion removes pixels on object boundaries, while dilation adds pixels, helping to eliminate small noise and fill gaps. The results of these operations are illustrated in Figure 6.

Further, opening and closing operations were applied, which are useful for removing small objects from the foreground (opening) and closing small holes within foreground objects (closing). These results are shown in Figures 7 and 8, respectively.

4) EDGE DETECTION

Edge detection was performed using the Sobel operator, which computes gradients in the x and y directions, as depicted in Figures 9 and 10. The combined result of these gradients is presented in Figure 11. The Laplacian operator, which provides a second-order derivative measure, was also employed to enhance edge detection, shown in Figure 12. Finally, the Canny edge detection method was utilized for its effectiveness in detecting a wide range of edges, depicted in Figure 13.

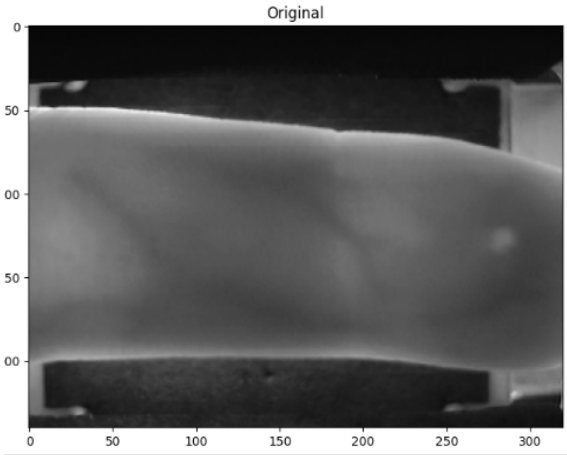
5) IMAGE RESIZING

We performed scaling operations on a selected image to demonstrate the effects of linear, cubic, and area interpolation methods. The images were resized to different scales, as shown in Figures 15 to 16, illustrating the impact of interpolation methods on image quality during resizing. Figure 17 represents image scaled using skewed Size.

6) FILTERING TECHNIQUES

Finally, various filtering techniques were employed to reduce noise and smooth the images. A Kalman filter, median filter, and Gaussian filter were applied, demonstrating their effects on image quality as illustrated in Figures 18 to 19.

The series of image-processing techniques applied in this study highlights the importance of pre-processing steps in enhancing image quality and preparing images for further analysis. The results showcase the effectiveness of these techniques in addressing common challenges encountered in image processing tasks [34].



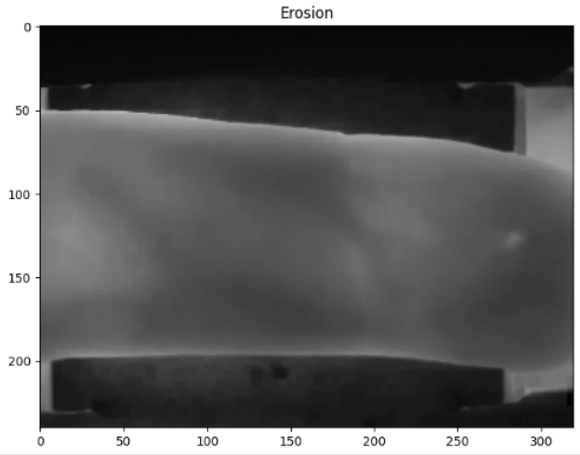


FIGURE 5. Erosion operation applied to the original image.

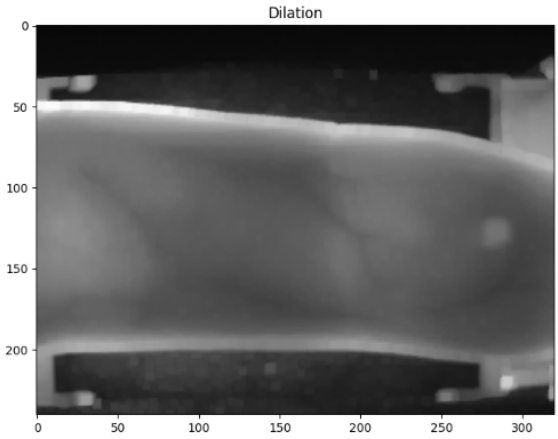


FIGURE 6. Dilation operation applied to the original image.

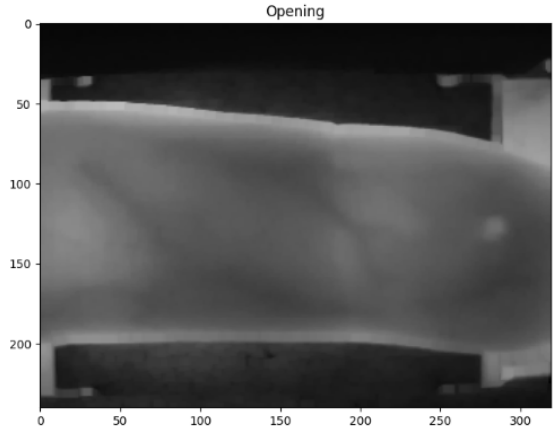


FIGURE 7. Opening operation result, combining erosion followed by dilation.

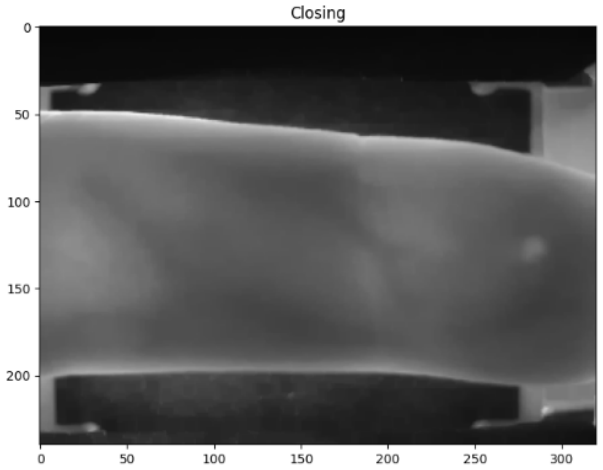


FIGURE 8. Closing operation result, combining dilation followed by erosion.

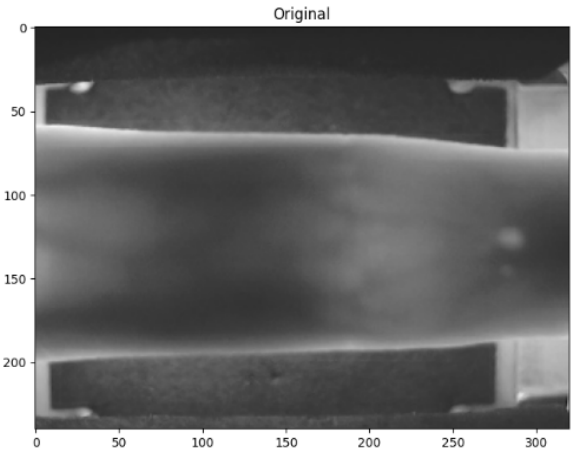
C. CNN ARCHITECTURES FOR FINGER VEIN RECOGNITION

In biometric authentication, pre-trained models offer a significant advantage through transfer learning by applying learned weights from general image datasets to the specific domain of finger vein patterns. This approach reduces training time and improves model generalization, especially when dealing with smaller biometric datasets, allowing for enhanced pattern recognition even with few labeled examples.

For this study, we evaluate several pretrained architectures on the finger vein biometric dataset: VGG16, VGG19, AlexNet, MobileNet, DenseNet121, EfficientNet, and ResNet101. Each architecture offers unique advantages:

VGG16 & VGG19: Both architectures utilize 3×3 convolutional filters, with VGG16 comprising 16 layers and VGG19 featuring 19 layers. They are favored for their simplicity and effectiveness in various image classification tasks [25], [35].

AlexNet: A pioneering CNN with eight layers (five convolutional and three fully connected), it utilizes large filter sizes (11×11 and 5×5), max pooling, Local Response Normalization (LRN), and dropout for



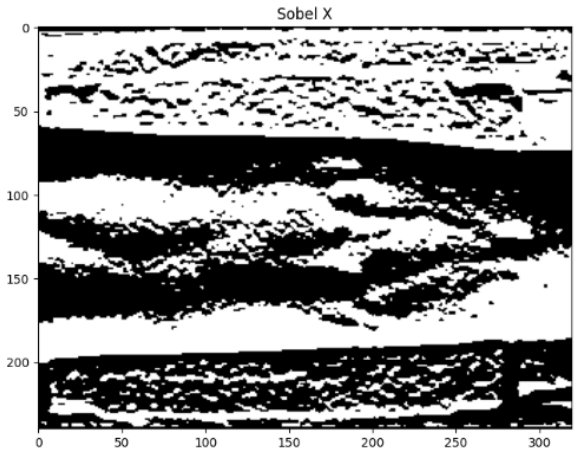


FIGURE 9. Sobel X gradient result, highlighting horizontal edge detection.

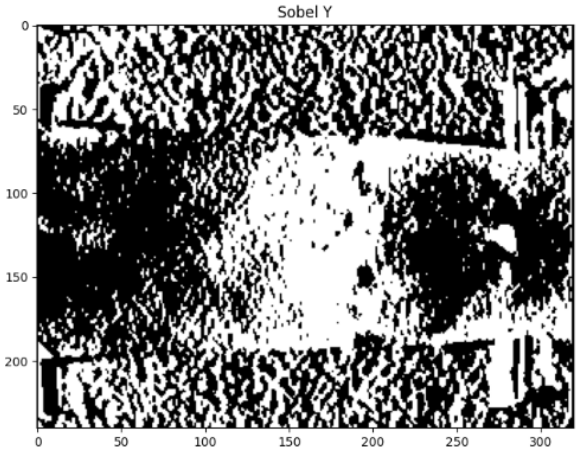


FIGURE 10. Sobel Y gradient result, highlighting vertical edge detection.

FIGURE 11. Combined Sobel gradient image, highlighting both horizontal and vertical edges.

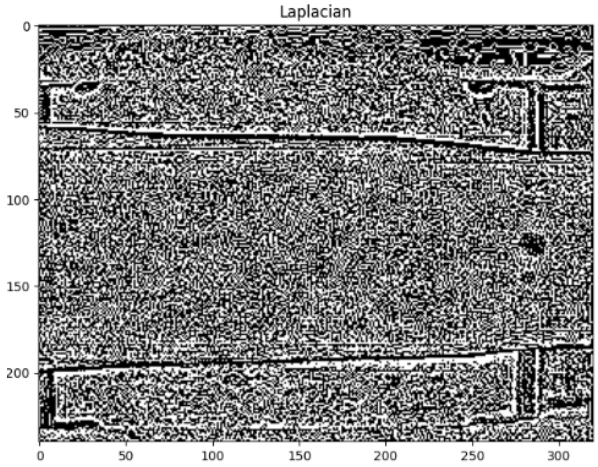


FIGURE 12. Laplacian edge detection result, highlighting areas of rapid intensity change.

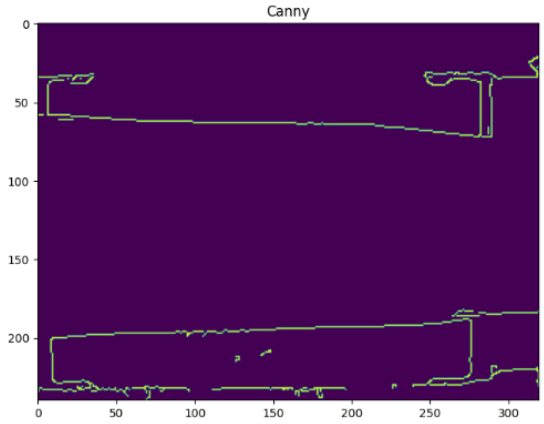


FIGURE 13. Canny edge detection result, showcasing precise edge localization with reduced noise.

regularization. AlexNet marked a turning point in deep learning for computer vision [18].

MobileNet: Designed for efficiency, this architecture employs depthwise separable convolutions to reduce

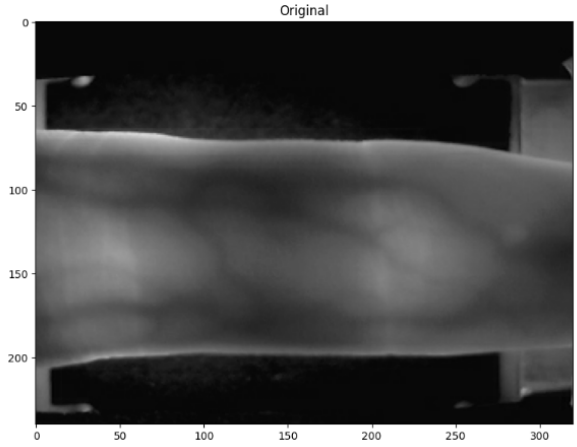


FIGURE 14. Original finger vein image.

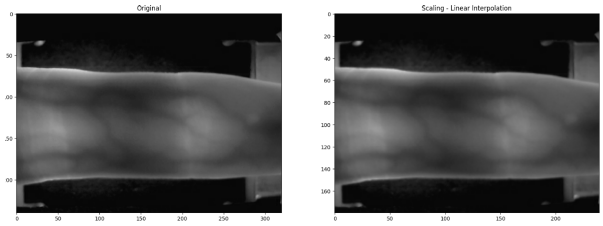


FIGURE 15. Image scaled using linear interpolation, ensuring smooth transitions between pixel values.

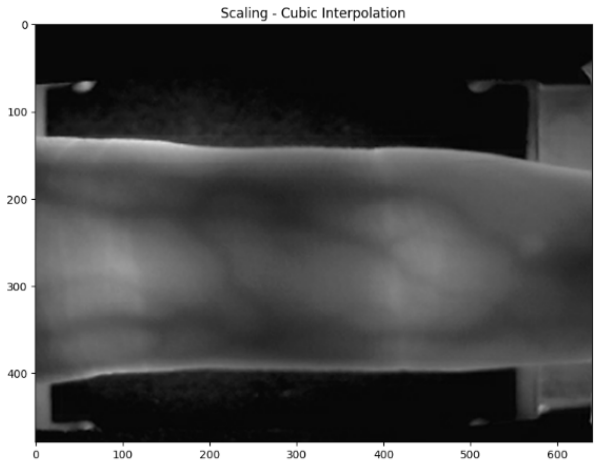


FIGURE 16. Image scaled using cubic interpolation, providing higher quality and smoother transitions compared to linear interpolation.

computational load while maintaining performance. It is particularly suited for mobile and embedded applications [36].

DenseNet121: Known for its dense connectivity, each layer receives inputs from all previous layers, promoting feature reuse and mitigating the vanishing gradient problem [37].

EfficientNet: This architecture scales efficiently, balancing depth, width, and resolution to optimize performance with fewer parameters [38].

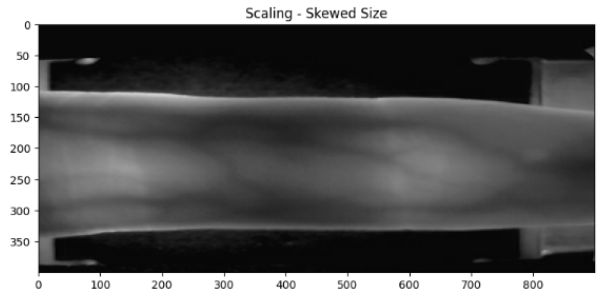


FIGURE 17. Image scaled using skewed size, resulting in a non-uniform distortion of the original proportions.

TABLE 4. Model hyperparameters of the proposed system.

Hyperparameter	Value
Learning Rate	0.0001
Batch Size	32
Epochs	500
Optimizer	Adam
Dropout Rate	0.5
Input Image Size	224x224 pixels

ResNet101: A deep architecture utilizing residual learning through skip connections, allowing for improved gradient flow and enabling the training of very deep networks [39].

These models have demonstrated substantial effectiveness in biometric recognition tasks, and our evaluation will identify the best-performing architecture for finger vein authentication.

IV. METHODOLOGY

The classification performance of the CNN architectures is evaluated specifically for finger vein biometric recognition. This includes an analysis of how these models perform on the imbalanced dataset and the effect of data augmentation techniques. To enhance feature extraction, the layers of the original architectures are frozen, and each model is trained using weights pretrained on the ImageNet dataset. Adam optimizer and batch size is set to 32 are employed to facilitate training based on trial and error method. The categorical cross-entropy loss function is used to quantify the differences between predicted class probabilities and actual labels, guiding the model in adjusting its parameters for improved accuracy. All the models are trained for 500 epochs. The finger vein dataset is divided into 2 splits. Training, validation, and test sets with proportions of 80%, 10%, and 10% and 70%, 15%, 15% respectively.

1) MODEL HYPERPARAMETERS OF THE PROPOSED SYSTEM

The hyperparameters utilized during training are summarized in Table 4.

2) MODEL ARCHITECTURE SUMMARY

The architecture of each model employed in the experiments is summarized in Table 5.

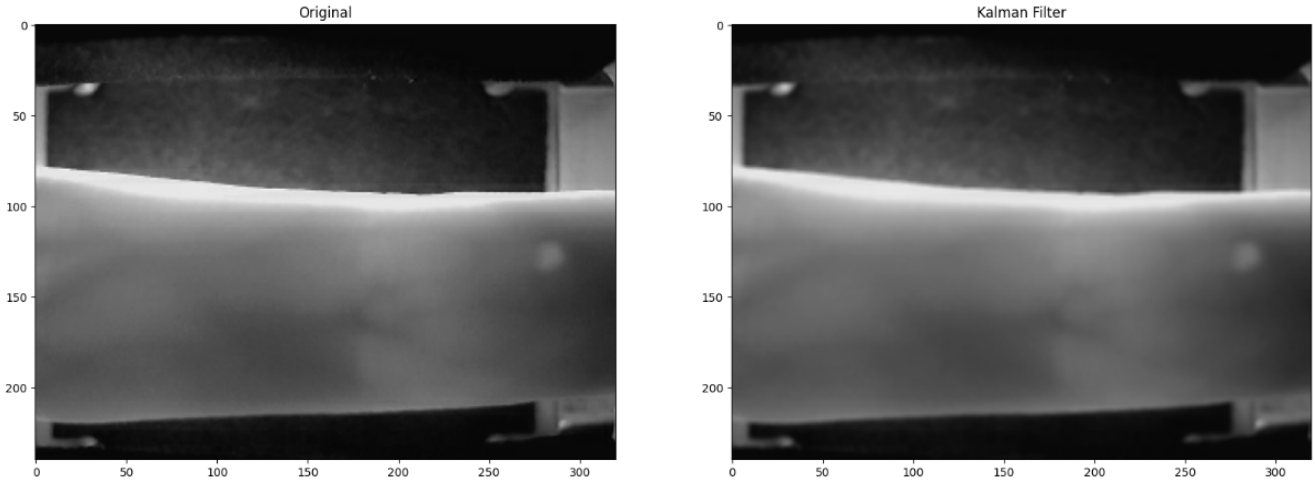


FIGURE 18. Image after Kalman filtering, demonstrating enhanced noise reduction and improved clarity.

TABLE 5. Model architecture summary.

Model	Number of Layers	Parameters	Key Features
VGG16	16	138 million	Convolutional layers, pooling layers, ReLU activation
VGG19	19	143.67 million	Convolutional layers, pooling layers, ReLU activation
AlexNet	8	60 million	ReLU activation, dropout
MobileNet	88	4.2 million	Depthwise separable convolutions
DenseNet201	201	20 million	Dense connections, feature reuse
EfficientNet	290	5.3 million	Compound scaling, mobile-friendly architecture
ResNet101	101	11 million	Residual connections, deep learning

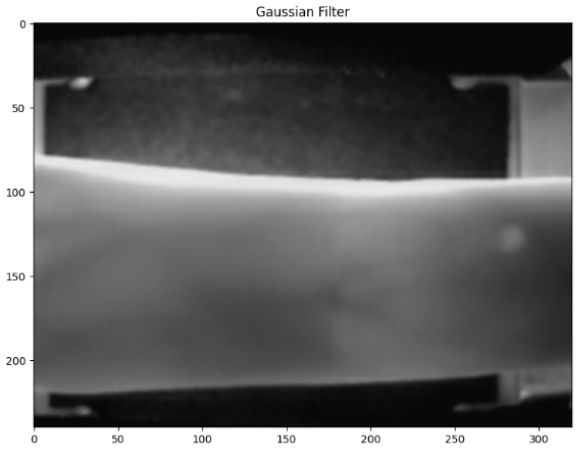


FIGURE 19. Image after median and Gaussian filtering, showing effective noise reduction and improved image quality.

A. FINGER VEIN RECOGNITION ALGORITHM

1) Data Acquisition

a) *Dataset Loading:* Access and load the finger vein images from the designated Kaggle database, which consists of 3816 images. These images represent multiple categories relevant to our classification task. The dataset was divided into training, validation, and test sets as follows:

- **Training Set:** 70% or 80% of the images
- **Validation Set:** 15% or 10%of the images
- **Test Set:** 15% or 10% of the images

b) *Image Resizing:* Standardize the dimensions of all images (e.g., resize to 224×224 pixels) to ensure uniformity during processing.

2) Preprocessing of Images

a) *Input Image Retrieval:* Extract each finger vein image from the dataset for processing.

b) *Grayscale Transformation:* Convert each image to grayscale to simplify subsequent analysis by reducing color complexity.

c) *Binarization:* Implement binary thresholding to emphasize the finger vein patterns. Choose a threshold value (e.g., 127) to distinguish between vein and background.

d) *Noise Reduction:* Apply a Gaussian filter to mitigate noise in the images, enhancing the clarity of features.

e) *Morphological Processing:* Utilize morphological operations, such as dilation and erosion, to refine the visibility of vein structures.

f) *Region of Interest (ROI) Masking:* Employ masks to isolate and focus on the central area of the finger, where vein patterns are most prominent.

3) **Convolutional Neural Network (CNN) Development**
   a) *Library Importation:* Import necessary libraries, such as TensorFlow or Keras, for CNN construction.
   b) *Model Architecture Design:* Construct the CNN model by sequentially adding layers, including Conv2D, MaxPooling, and Dense layers.
   c) *Model Compilation:* Define the model's loss function and optimization algorithm to prepare for training.
   d) *Model Training:* Utilize the preprocessed dataset to train the CNN for effective extraction of vein features.
   e) *Evaluation and Optimization:* Assess model performance using accuracy metrics and make necessary adjustments to improve results.
4) **Implementation of Pre-Trained Model**
   a) *Loading Pre-Trained CNN:* Access pre-trained models (e.g., VGG16, VGG19) for the purpose of transfer learning.
   b) *Layer Freezing:* Freeze specific layers within the pre-trained model to retain existing learned features during training.
   c) *Addition of Custom Layers:* Integrate additional layers tailored for finger vein feature extraction.
   d) *Training the Enhanced Model:* Train the composite model with the finger vein dataset, keeping the base layers fixed.
5) **Model Assessment**
   a) *Performance Comparison:* Evaluate and compare the performance metrics of the CNN and the pre-trained model, focusing on accuracy, precision, and recall.
   b) *Final Model Training:* Conduct final training iterations on the selected model to optimize performance.
   c) *Testing and Deployment:* Validate the model's effectiveness with unseen data, ensuring its reliability for real-world authentication applications before deployment.

V. EXPERIMENTAL SETUP AND RESULT ANALYSIS

The experiments were conducted on a cloud-based GPU platform running an Ubuntu Linux environment, equipped with an NVIDIA A100 GPU. The computational setup included Cuda version 12.2 and cuDNN version 11.5, along with TensorFlow 2.11.0, to optimize deep learning workflows and ensure efficient model training and evaluation. System architecture of the proposed model is represented using Figure 20.

A. PERFORMANCE METRICS FOR FINGER VEIN RECOGNITION

Assessing the performance of a finger vein recognition model is essential following the training phase. The loss function is primarily used as a metric during training to optimize the model. Once training is concluded, various metrics are utilized to thoroughly evaluate the model's capability to accurately classify finger vein patterns. In particular, accuracy (as defined in Equation 2) serves as a comprehensive indicator of the model's correct predictions [40]:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

Precision (shown in Equation 3) measures the reliability of the model's positive predictions:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

Recall (presented in Equation 4) assesses the model's effectiveness in identifying true positives:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

The F1-score (illustrated in Equation 5) provides a balanced assessment of both precision and recall, presenting a more holistic view of the model's performance:

$$\text{F1 Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

B. EXPERIMENT DETAILS

The performance of various deep learning models was evaluated based on their ability to accurately classify data using different metrics, including accuracy, precision, recall, and F1 score. The models analyzed include VGG16, VGG19, AlexNet, MobileNet, DenseNet, EfficientNet, and ResNet101. Two experiments were conducted, each utilizing a different data split configuration: **Experiment 1** with an 80-10-10 split and **Experiment 2** with a 70-15-15 split.

In **Experiment 1** (80-10-10 Split), the dataset consisted of a total of **3816 images**, divided into **3053 training images**, **382 validation images**, and **381 test images**. For **Experiment 2** (70-15-15 Split), the dataset also contained **3816 images**, but was split into **2672 training images**, **572 validation images**, and **572 test images**. The results are presented through radar charts, bar charts, grouped bar charts, and accuracy/loss curves, providing a comprehensive comparison of the models' effectiveness.

As illustrated in Table 6 and Table 7, a comparison of deep learning models, including VGG16, VGG19, AlexNet, MobileNet, DenseNet, EfficientNet, and ResNet101, has been carried out using key performance metrics. These metrics include training, testing, and validation accuracy, along with precision, recall, F1-score, training time, and loss. The results highlight the models' capability to achieve high accuracy and precision while maintaining minimal training loss, showcasing their efficiency in biometric recognition tasks.

1) DATA AUGMENTATION IMPACT

Data augmentation techniques were employed to enhance model robustness and prevent overfitting. Techniques such as

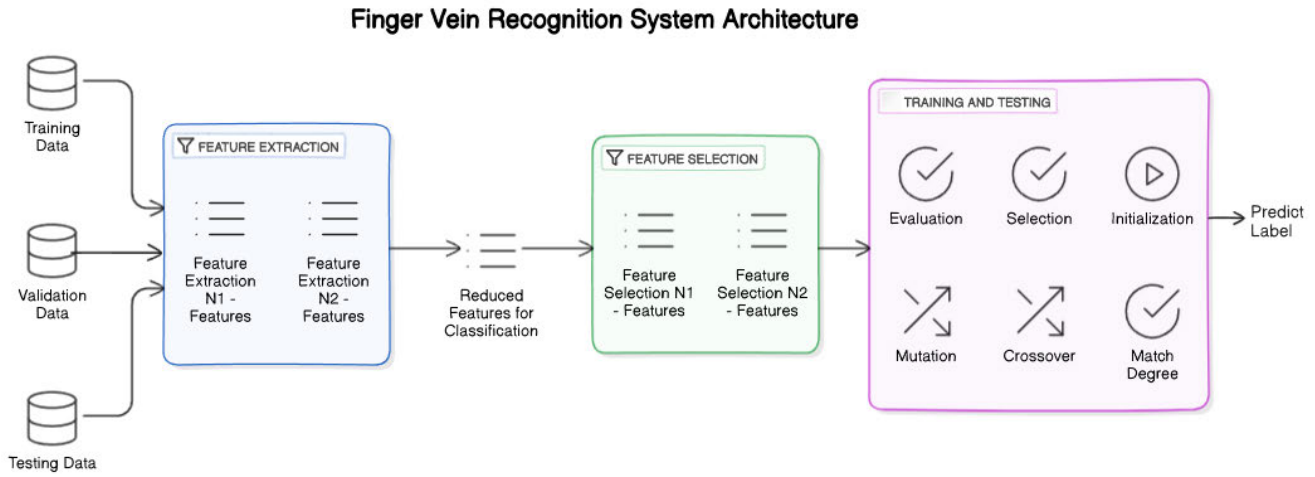


FIGURE 20. System architecture of the proposed model, illustrating the components and flow of data in the fingerprint recognition system.

TABLE 6. Performance metrics of various deep learning models (70-15-15 split).

Model	Train Acc.	Test Acc.	Val Acc.	Precision	Recall	F1 Score	Time (s)	Train Loss
VGG16	0.997	0.995	0.996	0.997	0.998	0.997	534	0.0013
VGG19	0.996	0.994	0.995	0.995	0.996	0.995	524	0.0018
AlexNet	0.993	0.975	0.980	0.994	0.993	0.993	376	0.0169
MobileNet	0.995	0.994	0.995	0.996	0.996	0.996	615.42	0.0003
DenseNet	0.960	0.940	0.950	0.970	0.960	0.960	590.9	0.0130
EfficientNet	0.994	0.990	0.992	0.993	0.995	0.994	612	0.0014
ResNet101	0.995	0.992	0.993	0.996	0.995	0.995	753	0.0012

TABLE 7. Performance metrics of various deep learning models (80-10-10 split.

Model	Train Acc.	Test Acc.	Val Acc.	Precision	Recall	F1 Score	Time (s)	Train Loss
VGG16	0.999	0.997	0.998	0.999	0.999	0.999	590	0.0013
VGG19	0.998	0.996	0.997	0.998	0.998	0.998	580	0.0020
AlexNet	0.997	0.988	0.990	0.997	0.997	0.997	364	0.0011
MobileNet	0.998	0.995	0.996	0.998	0.998	0.998	541	0.0011
DenseNet	0.995	0.970	0.980	0.996	0.995	0.995	560	0.0210
EfficientNet	0.997	0.994	0.995	0.997	0.997	0.997	600.5	0.0014
ResNet101	0.998	0.995	0.996	0.998	0.998	0.998	781	0.0013

rotation, width shift, height shift, rescale, shear, brightness adjustment, horizontal flip, and vertical flip were applied to the training dataset, resulting in a total of 76,320 augmented images generated from 3,816 original images.

The inclusion of these augmentation methods positively influenced model performance, particularly in the 80-10-10 split configuration. The radar charts and grouped bar charts indicate that models trained with augmented data achieved higher accuracy and recall scores compared to those without augmentation. This suggests that augmentation effectively diversified the training dataset, allowing models to generalize better to unseen data.

C. MODEL PERFORMANCE ANALYSIS

The trained models undergo comprehensive testing to evaluate their accuracy and generalization capabilities on a separate test dataset specifically designed for finger vein biometric recognition. The performance of each model is assessed using a range of metrics, including accuracy, precision, recall, and F1 score after the application of data augmentation.

By employing various assessment measures, we gain insights into the models' capabilities and their overall suitability for the biometric authentication task, ensuring that different performance criteria are considered in the evaluation process. This rigorous testing phase not only highlights the strengths and weaknesses of each model but also helps identify the most effective approach for enhancing biometric recognition accuracy and reliability.

1) PERFORMANCE METRICS (80-10-10 SPLIT) AND (70-15-15 SPLIT)

The radar chart in Figure 21 illustrates the performance of each model using metrics normalized between 0 and 1 for

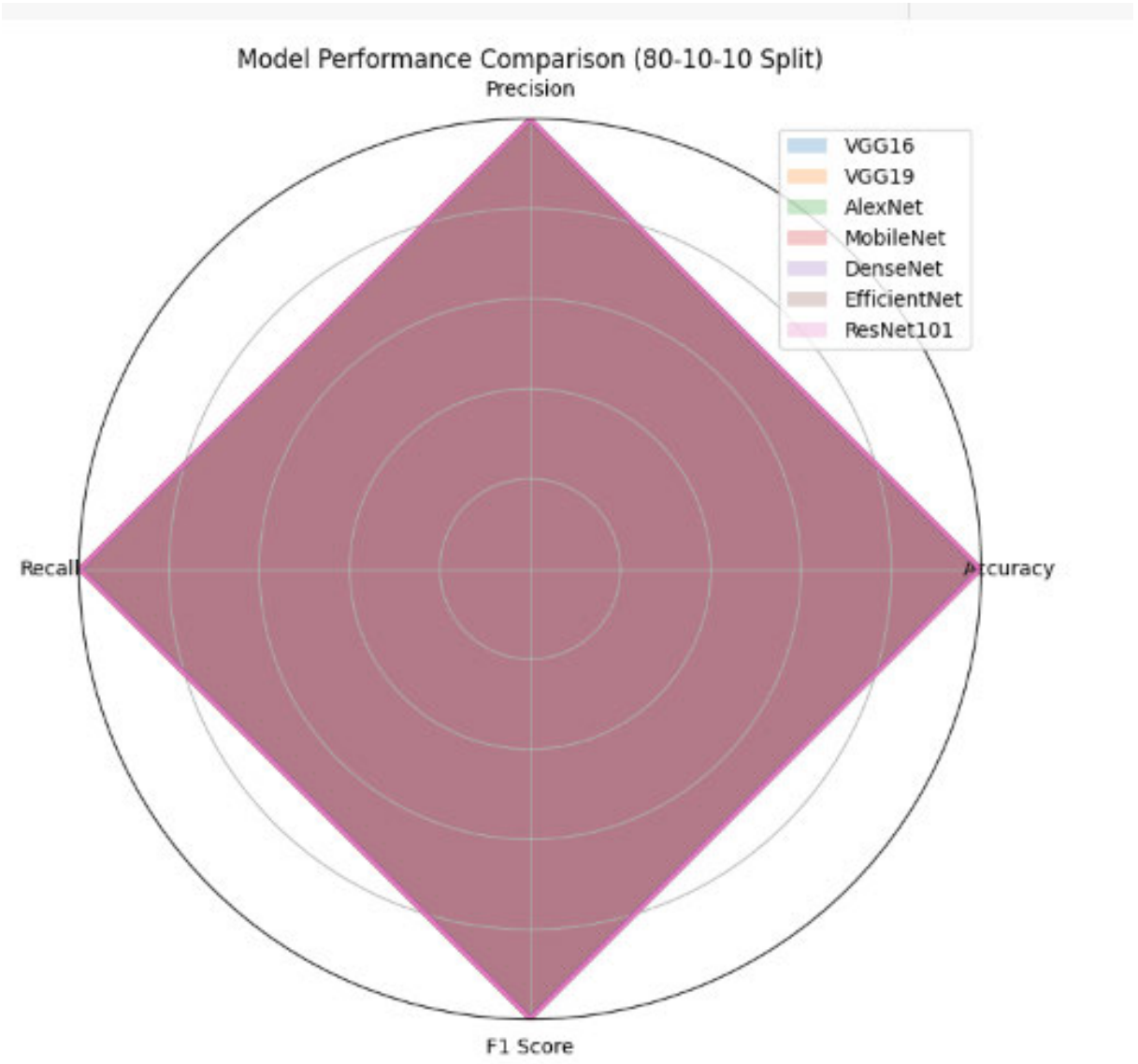


FIGURE 21. Performance of each model for (80-10-10 Split).

the 80-10-10 split. VGG16,VGG16 and ResNet101 demonstrated exceptional performance, achieving high accuracy, precision, recall, and F1 scores nearing 1.0.

Similarly, Figure 22 presents the performance metrics for the 70-15-15 split. VGG16 and VGG19 maintained strong performance, with high scores in accuracy and precision. However, a slight decrease in metrics was observed compared to the 80-10-10 split, especially for DenseNet201, suggesting that the increased validation and test data may impact model performance.

2) ACCURACY AND LOSS CURVES

The accuracy and loss curves for VGG16, VGG19, and ResNet101 are depicted in Figures 23, 24, and 25. Each figure illustrates the training and validation accuracy and loss for 500 epochs, based on an 80-10-10 split of the dataset.

For **VGG16**, the training accuracy achieved approximately 99.9%, with the training loss stabilizing near 0.0013. **VGG19** exhibited a training accuracy close to 99.8% and a training loss nearing 0.002. **ResNet101** reached a training accuracy of about 99.8%, with the training loss stabilizing at around 0.0013. These curves indicate effective learning across all three models, with minimal overfitting apparent from the close alignment of training and validation metrics. The models demonstrated strong performance, particularly under the 70-15-15 data split configuration, reinforcing the positive impact of data augmentation techniques utilized in the training process.

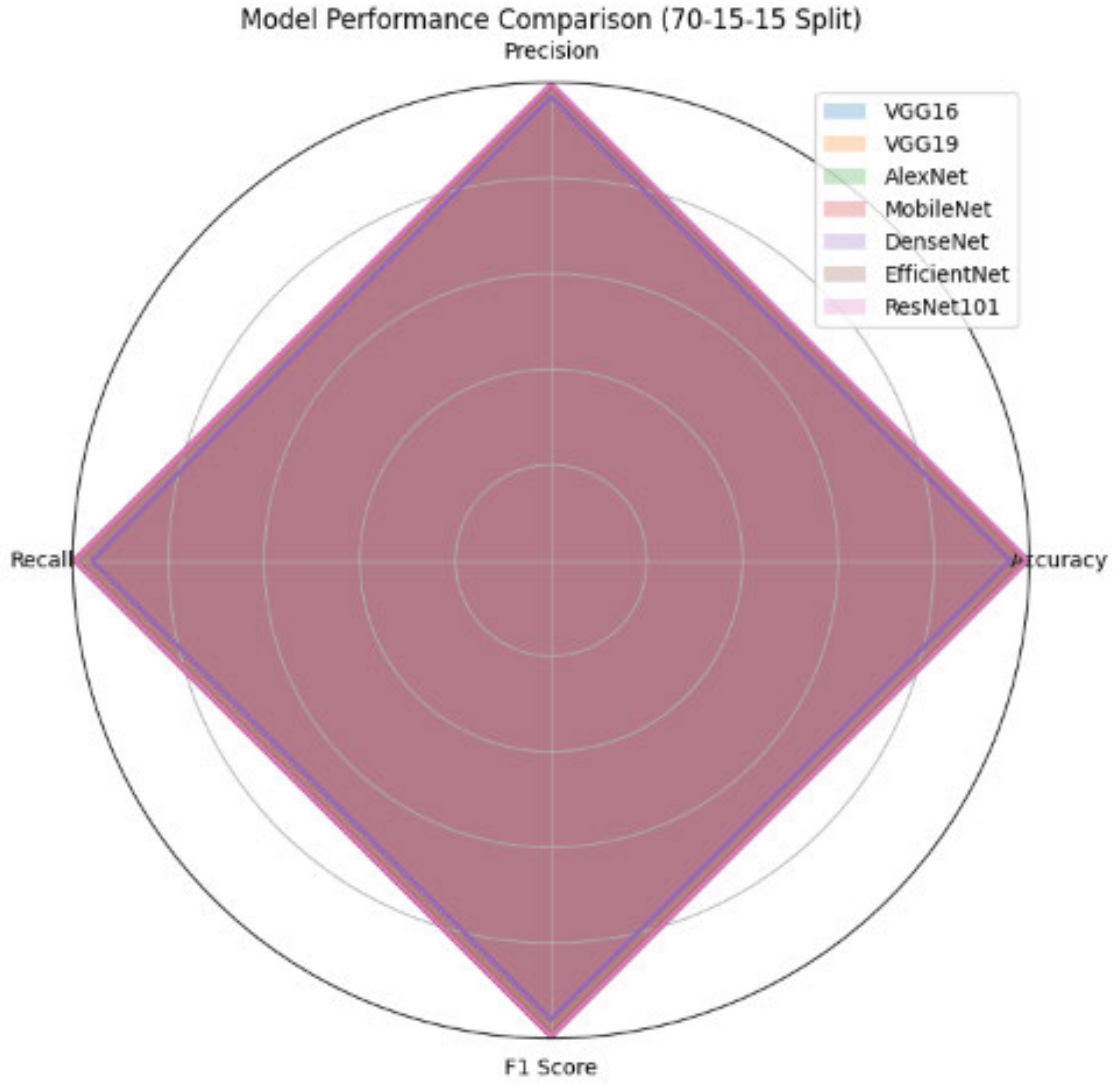


FIGURE 22. Performance of each model for (70-15-15 split).

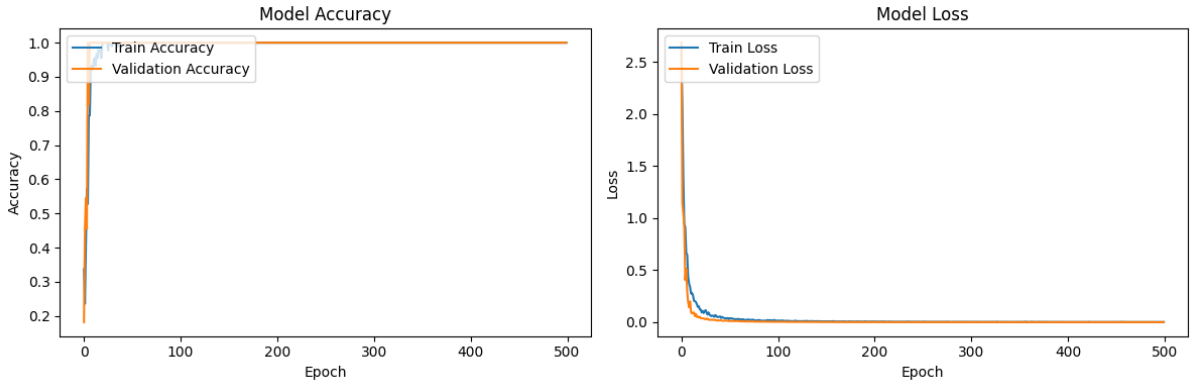


FIGURE 23. The accuracy and loss curves for VGG16 (80-10-10 Split).

The accuracy and loss curves for VGG16, VGG19, and ResNet101 are presented in Figures 26, 27, and 28. Each figure illustrates the training and validation accuracy and loss for 500 epochs, based on a 70-15-15 split of the dataset.

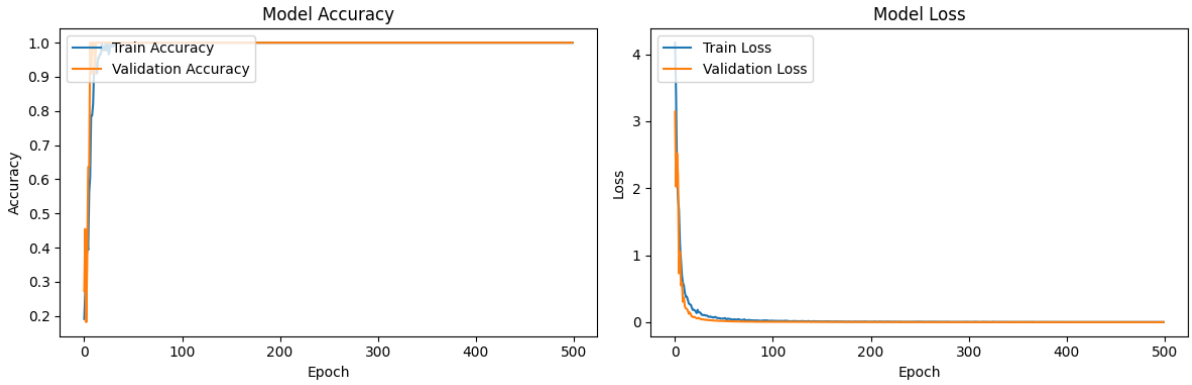


FIGURE 24. The accuracy and loss curves for VGG19 (80-10-10 Split).

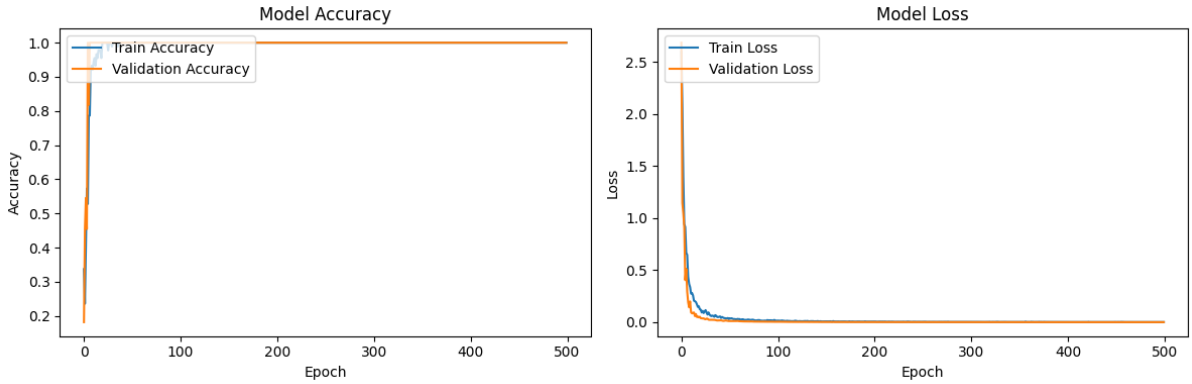


FIGURE 25. The accuracy and loss curves for Resnet (80-10-10 Split).

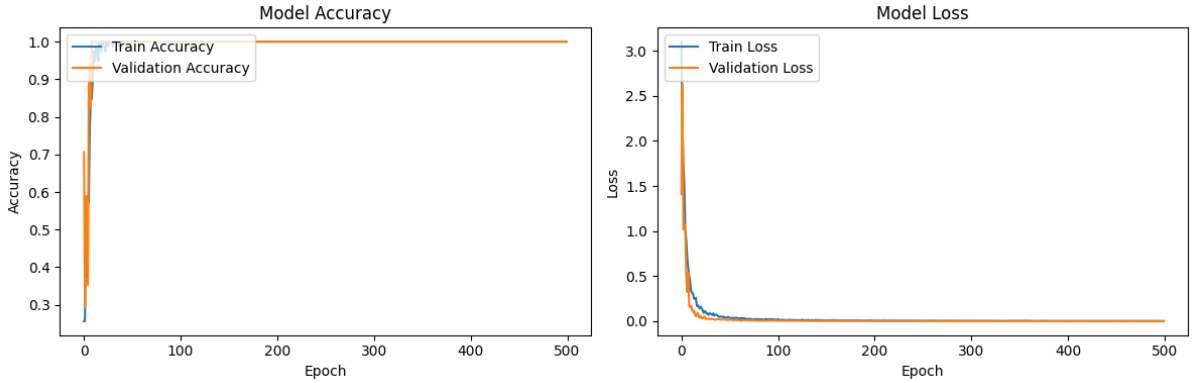


FIGURE 26. The accuracy and loss curves for VGG16 (70-15-15 Split).

For **VGG16**, the training accuracy reached approximately 99.7%, with training loss stabilizing near 0.0013. The validation metrics closely aligned with the training performance, indicating effective generalization. **VGG19** exhibited a training accuracy close to 99.6%, and the training loss approached 0.0018. The validation results demonstrated similar trends, reflecting the model's ability to learn from the training data. **ResNet101** achieved a training accuracy of about 99.5%, with training loss stabilizing at around 0.0012. The validation performance further confirmed the model's robust learning capacity. These curves indicate effective learning across all three models, with minimal overfitting

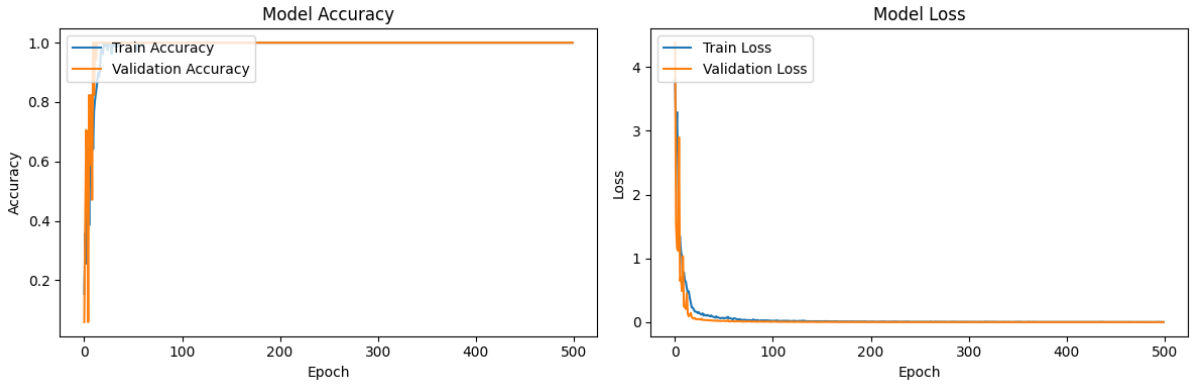


FIGURE 27. The accuracy and loss curves for VGG19 (70-15-15 Split).

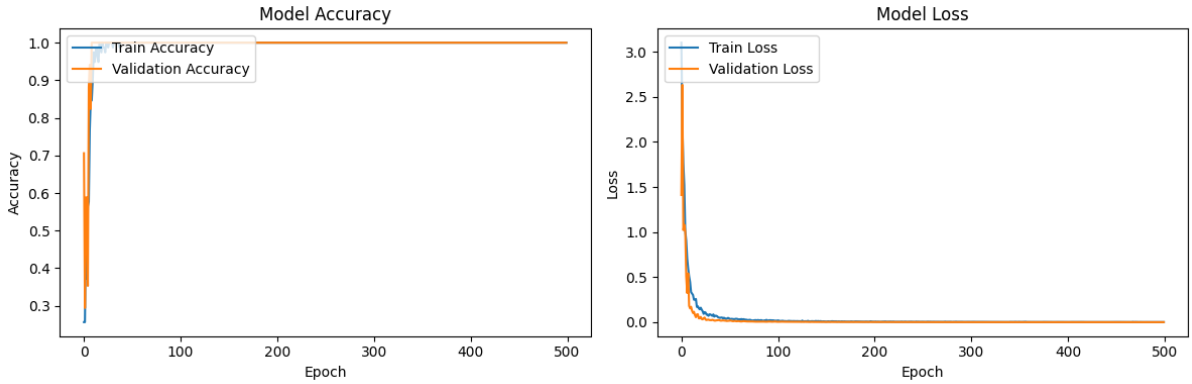


FIGURE 28. The accuracy and loss curves for Resnet (70-15-15 Split).

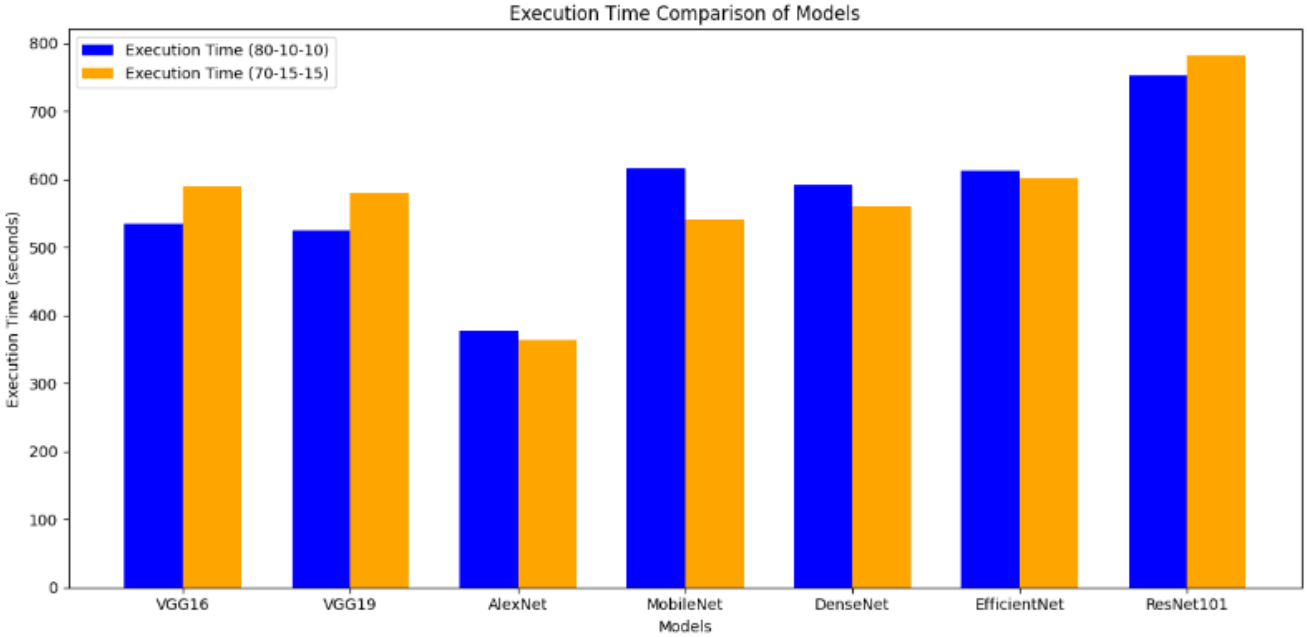


FIGURE 29. Execution time comparison graph for both data split.

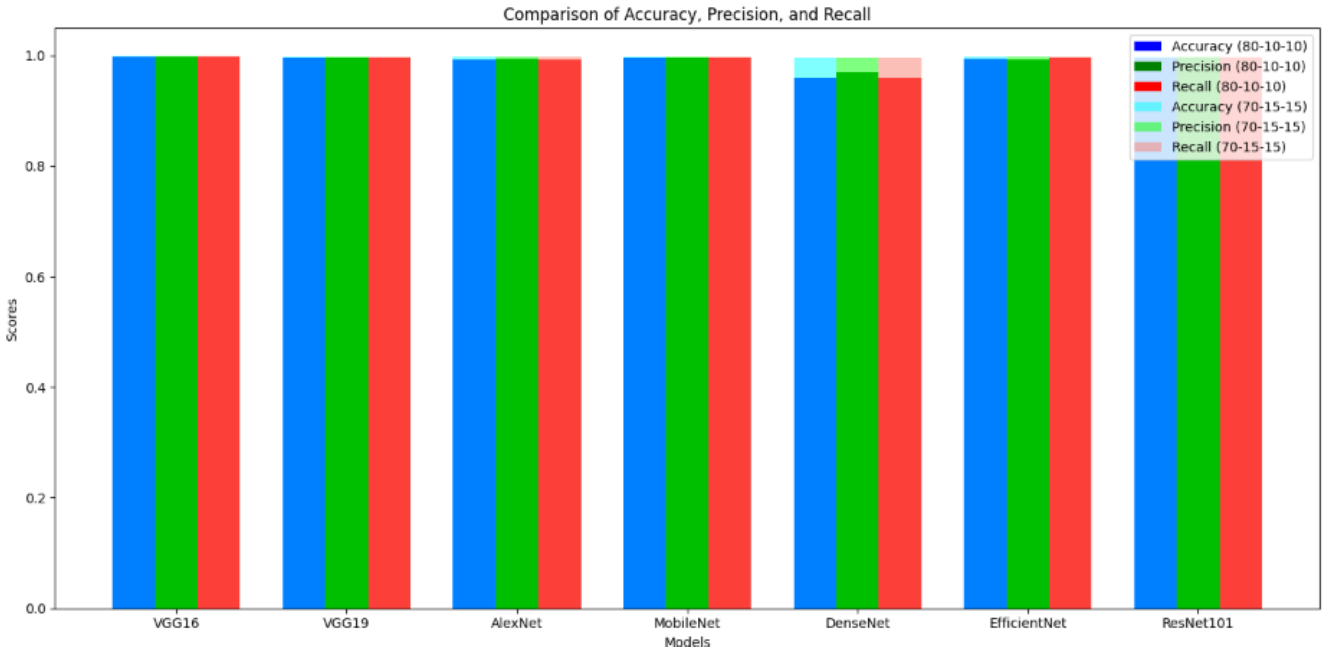


FIGURE 30. Grouped bar chart for metrics for both data split.

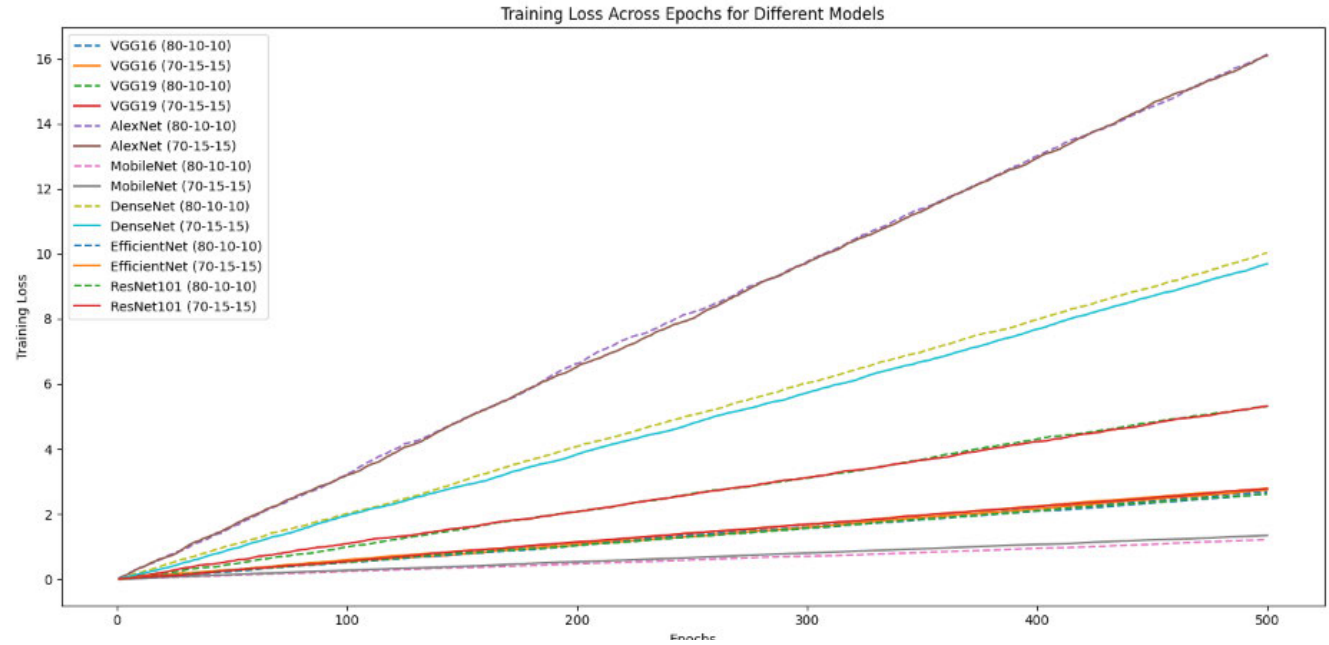


FIGURE 31. Training loss curves for both data split.

evident from the close alignment of training and validation metrics, reinforcing the efficacy of the 70-15-15 data split configuration.

3) EXECUTION TIME COMPARISON FOR BOTH DATA SPLIT

Execution time for each model was assessed and visualized in Figure 29. AlexNet exhibited the shortest execution time, while models like ResNet101 and DenseNet required significantly more time due to their complexity. This trend was consistent across both split configurations, with MobileNet performing favorably in terms of speed, indicating its efficiency for real-time applications.

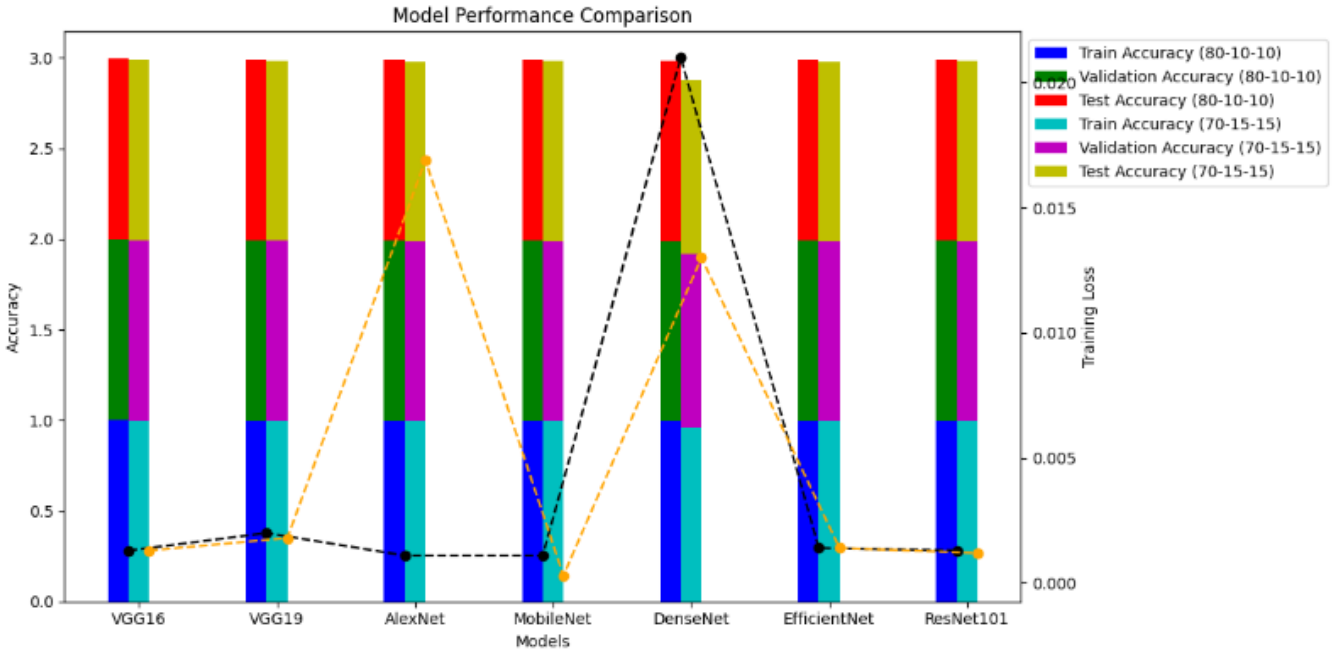


FIGURE 32. Model performance comparison for both data split.

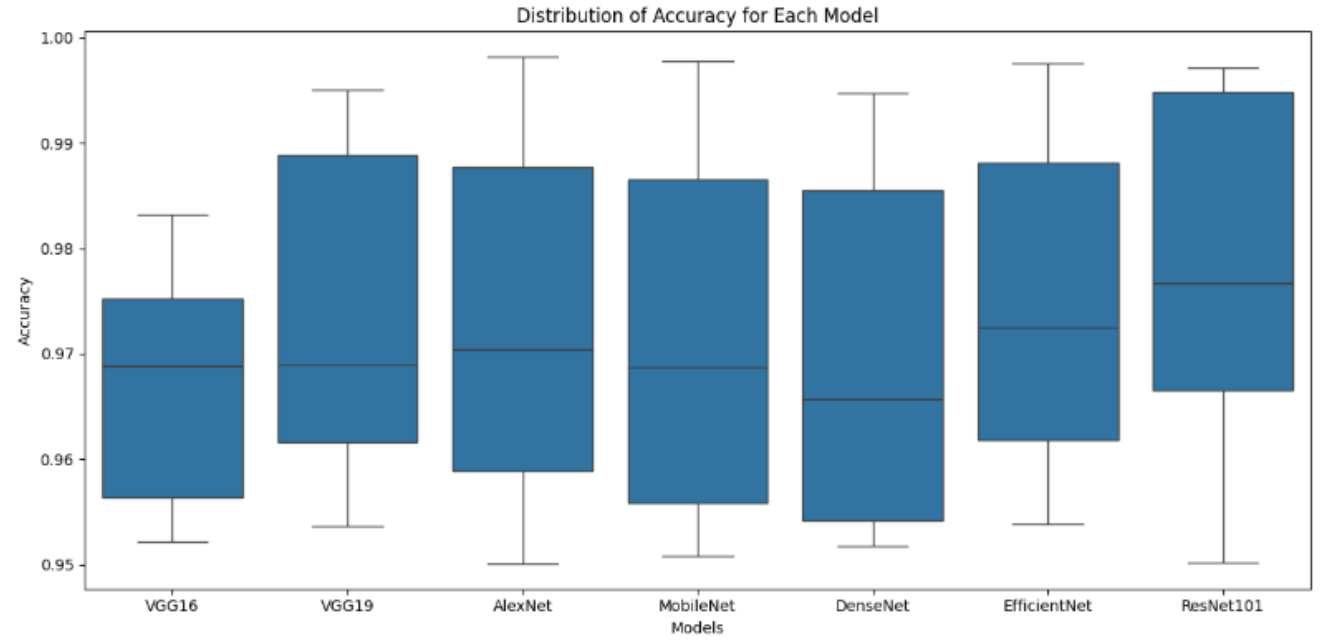


FIGURE 33. Box plot of model accuracy for both data split.

4) GROUPED BAR CHART FOR METRICS FOR BOTH DATA SPLIT

The grouped bar chart in Figure 30 provides a detailed comparison of accuracy, precision, and recall across both splits for each model. The metrics for the 80-10-10 split were generally higher than those for the 70-15-15 split, underscoring the impact of data distribution on model performance. Notably, the models displayed high sensitivity in detecting true positives, as indicated by strong recall scores.

FIGURE 34. Heatmap of model performance metrics for both data split.

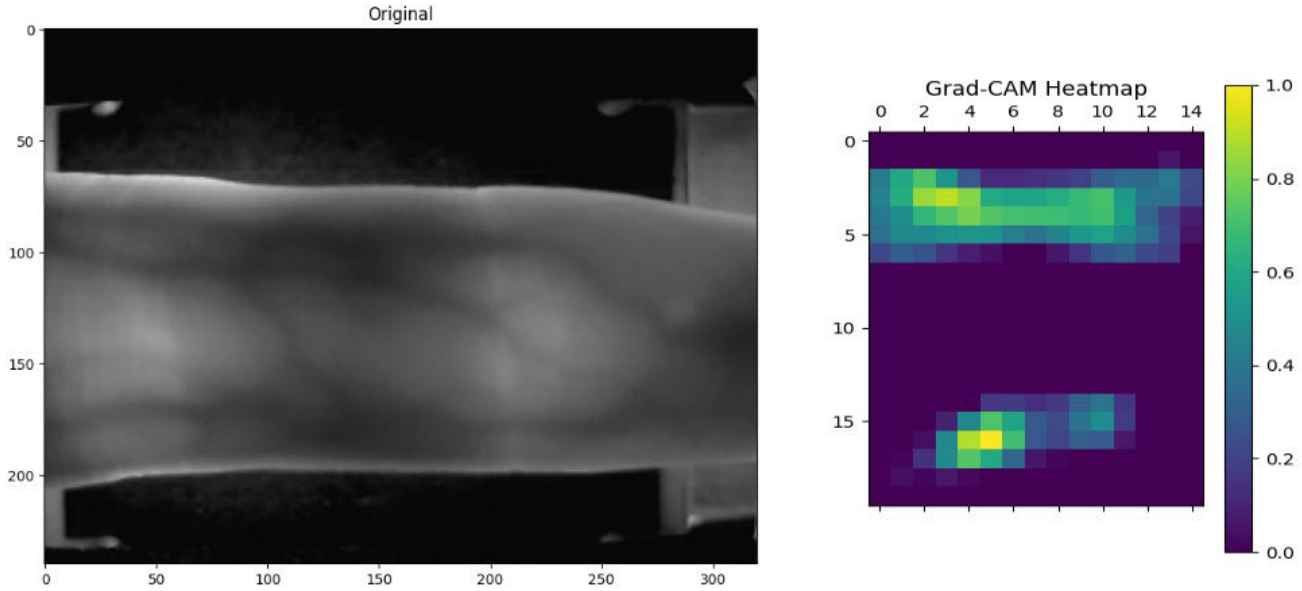


FIGURE 35. Grad-CAM Visualization for VGG16 70-15-15 split.

5) TRAINING LOSS CURVES FOR BOTH DATA SPLIT

Figure 31 illustrates the training loss of various models (VGG16, VGG19, AlexNet, MobileNet, DenseNet201, EfficientNet, and ResNet101 over 500 epochs. The dashed lines represent the loss for the 80-10-10 data split, while the solid lines represent the loss for the 70-15-15 split. Overall, the models demonstrate a consistent decrease in training loss, indicating effective learning and generalization across different data splits.

6) MODEL PERFORMANCE COMPARISON FOR BOTH DATA SPLIT

This bar chart 32 compares the training, validation, and test accuracies for various models under the two data splits:

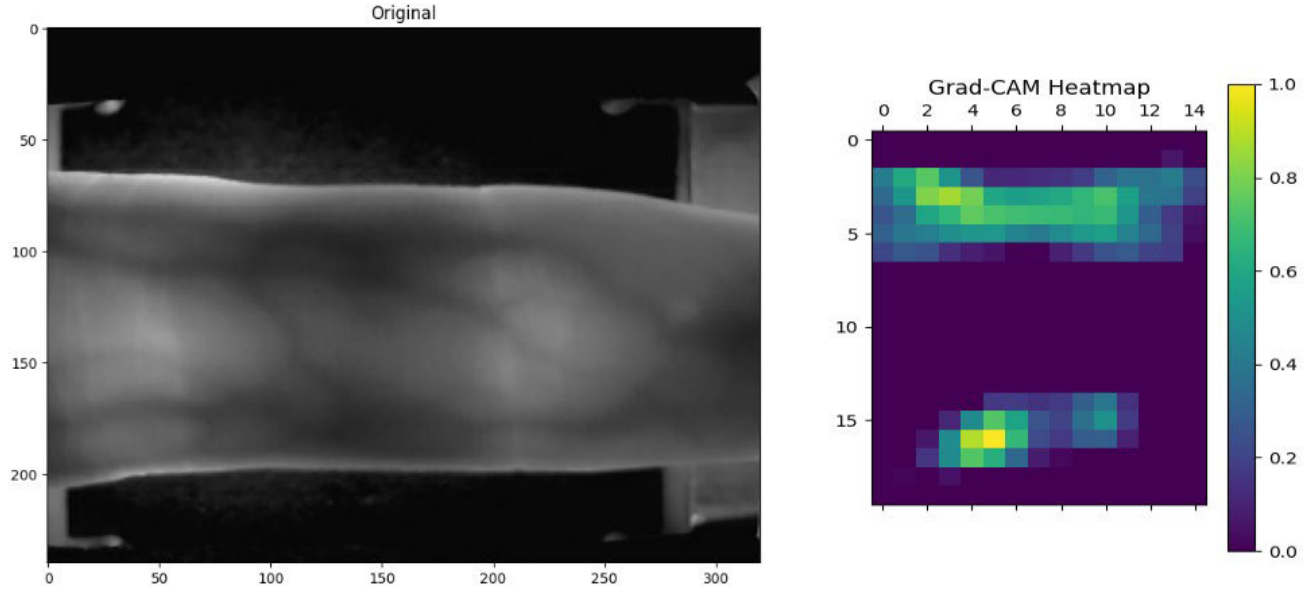


FIGURE 36. Grad-CAM Visualization for VGG16 80-10-10 split.

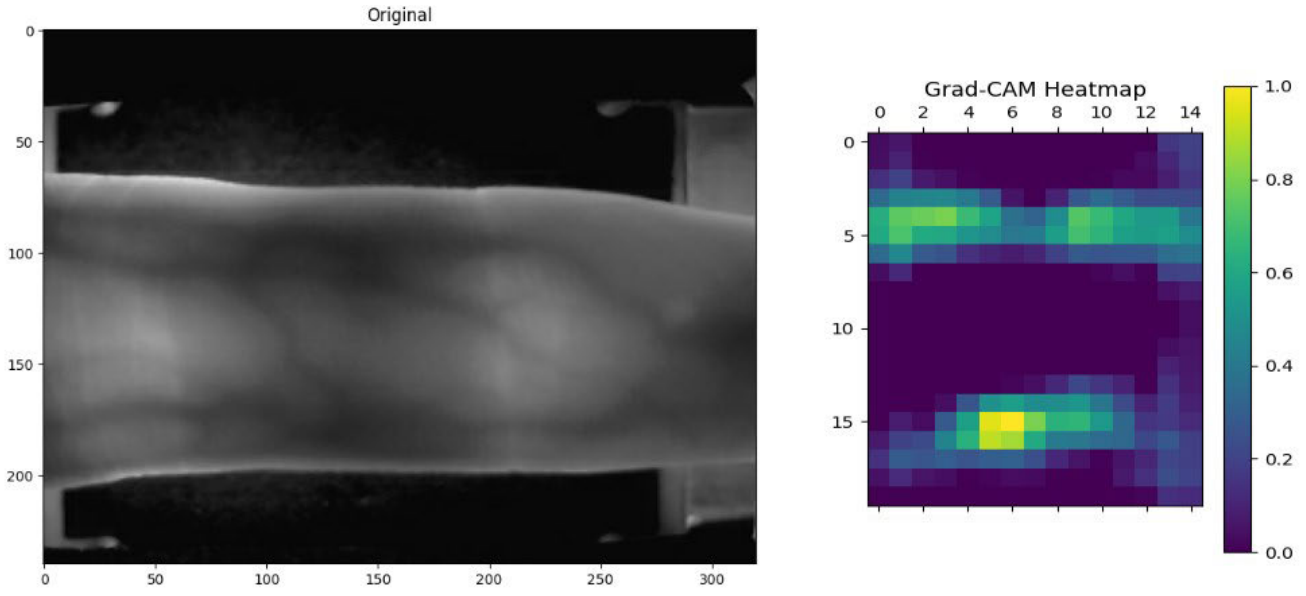


FIGURE 37. Grad-CAM Visualization for VGG19 70-15-15 split.

80-10-10 and 70-15-15. The training loss is represented as a line plot overlay. The results indicate that all models achieve high training accuracy, with VGG16 performing the best at 99.9% for the 70-15-15 split. The minimal training losses further reinforce the effectiveness of the training process.

7) BOX PLOT OF MODEL ACCURACY FOR BOTH DATA SPLIT

This box plot 33 displays the distribution of accuracy for each model across multiple training runs. It highlights that VGG16, VGG19, Densenet201 maintain high median accuracy values, indicating strong and consistent performance. The interquartile ranges are narrow, suggesting reliability in the accuracy metrics across different iterations of model training.

8) HEATMAP OF MODEL PERFORMANCE METRICS FOR BOTH DATA SPLIT

This heatmap 34 presents a comparative overview of accuracy, precision, and recall metrics for the models under two dataset splits: 70-15-15 and 80-10-10. The higher

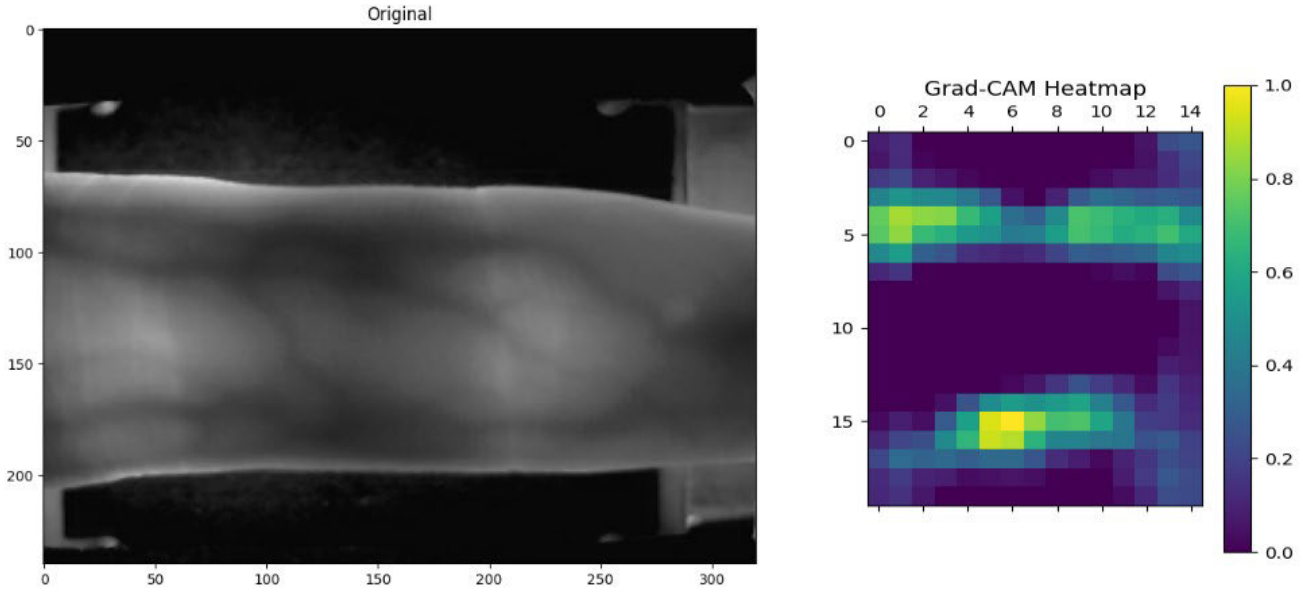


FIGURE 38. Grad-CAM Visualization for VGG19 80-10-10 split.

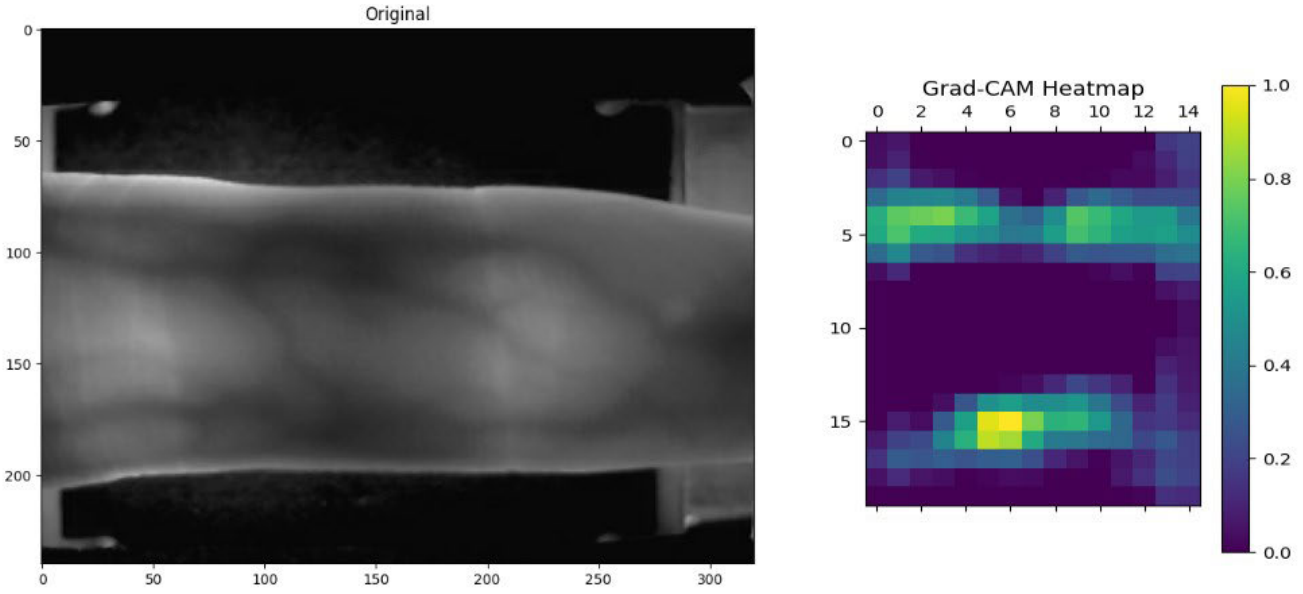


FIGURE 39. Grad-CAM Visualization for Resnet101 70-15-15 split.

values observed for the 80-10-10 split suggest that the additional validation data enhances model performance and generalization capabilities, providing a clear indication of each model's strengths and weaknesses.

9) GRAD-CAM VISUALIZATION FOR BOTH DATA SPLIT

To further analyze model predictions, Grad-CAM (Gradient-weighted Class Activation Mapping) was applied to VGG16, VGG19, and ResNet101, generating visual explanations for their classifications. The Grad-CAM results, illustrated in Figures 35, 37, and 39, highlight the regions of input images that significantly influenced model predictions.

For **VGG16**, the activation maps showed clear focus on critical features of the classified objects. **VGG19** also produced meaningful visualizations, emphasizing important areas that contributed to its predictions. **ResNet101's** Grad-CAM results displayed robust localization of features, demonstrating its effectiveness in identifying significant characteristics within the input data.

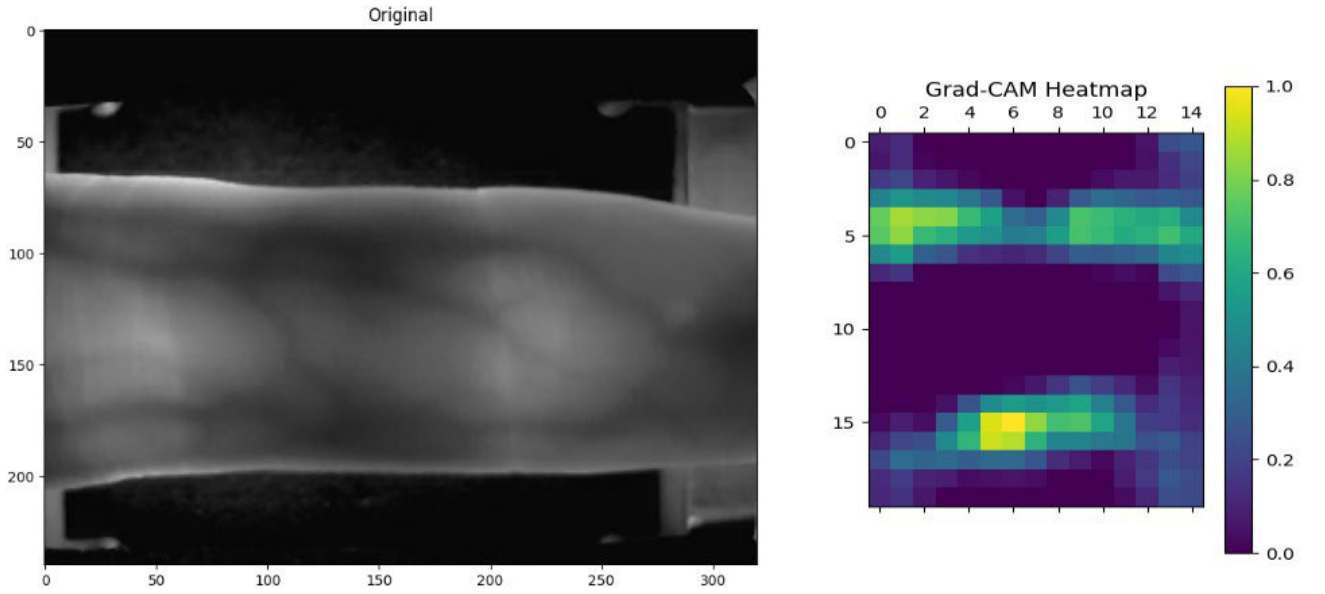


FIGURE 40. Grad-CAM Visualization for Resnet101 80-10-10 split.

These visualizations provide valuable insights into the decision-making processes of the models, enhancing interpretability and confidence in their classifications.

VI. CONCLUSION

Our study demonstrates the effectiveness of several deep learning models, particularly VGG16, VGG19, and ResNet101, for finger vein recognition. Performance metrics reveal high accuracy, with VGG16 achieving 99.9% accuracy, closely followed by VGG19 and ResNet101 at 99.8% each, underscoring both the accuracy and robustness of these models in classification tasks. Using a Kaggle dataset of 3,816 images with an 80-10-10 training-validation-test split, we observed significantly better performance than with the 70-15-15 split. Additionally, data augmentation proved essential in enhancing model generalization.

Analyses using accuracy/loss curves and Grad-CAM visualizations further affirm the suitability of these models for practical finger vein recognition applications. However, there are limitations: the high computational demands of VGG16, VGG19, and ResNet101 may limit their feasibility in real-time or low-resource contexts, where more efficient models could be preferable. Furthermore, our dataset's controlled conditions may not fully reflect real-world variability, suggesting a need for additional testing on diverse datasets. The resaerach work establishes a solid foundation for future developments in multimodal biometric integration and real-time identification, highlighting the potential of deep learning to enhance the accuracy and reliability of biometric systems.

REFERENCES

[1] M. Kaur, A. Verma, and P. J. Kaur, "Deep learning-based finger vein recognition and security: A review," in *Applied Data Science and Smart Systems*. Boca Raton, FL, USA: CRC Press, 2025, pp. 34–42.

[2] U. Sumalatha, K. K. Prakasha, S. Prabhu, and V. C. Nayak, "A comprehensive review of unimodal and multimodal fingerprint biometric authentication systems: Fusion, attacks, and template protection," *IEEE Access*, vol. 12, pp. 64300–64334, 2024.

[3] U. Sumalatha, K. Krishna Prakasha, S. Prabhu, and V. C. Nayak, "Deep learning applications in ECG analysis and disease detection: An investigation study of recent advances," *IEEE Access*, vol. 12, pp. 126258–126284, 2024.

[4] R. Alrawili, A. A. S. AlQahtani, and M. K. Khan, "Comprehensive survey: Biometric user authentication application, evaluation, and discussion," *Comput. Electr. Eng.*, vol. 119, Oct. 2024, Art. no. 109485.

[5] K. Shaheed and I. Qureshi, "A hybrid proposed image quality assessment and enhancement framework for finger vein recognition," *Multimedia Tools Appl.*, vol. 83, no. 5, pp. 15363–15388, Feb. 2022.

[6] N. R, S. S, and P. Selvi Rajendran, "Finger vein extraction and authentication for security purpose based on CNN," in *Proc. Int. Conf. Adv. Data Eng. Intell. Comput. Syst. (ADICS)*, Apr. 2024, pp. 1–6.

[7] M. V. Madhusudhan, V. Udayarani, and C. Hegde, "Finger vein based authentication using deep learning techniques," *Int. J. Recent Technol. Eng. (IJRTE)*, vol. 8, no. 5, pp. 5403–5408, Jan. 2020.

[8] M. V. Madhusudhan, V. U. Rani, and C. Hegde, "Finger vein recognition model for biometric authentication using intelligent deep learning," *Int. J. Image Graph.*, vol. 23, no. 3, May 2023, Art. no. 2240004.

[9] A. Boucetta and L. Boussaad, "Biometric authentication using finger-vein patterns with deep-learning and discriminant correlation analysis," *Int. J. Image Graph.*, vol. 22, no. 1, Jan. 2022, Art. no. 2250013.

[10] S. Sathishkumar, P. Padmapriya, P. K. Thabjul, and S. Raghavan, "A biometric-finger vein authentication system for security purpose using deep learning technique," in *Proc. 14th Int. Conf. Comput. Commun. Netw. Technol. (ICCCNT)*, Jul. 2023, pp. 1–10.

[11] A. Mathew and P. Amudha, "Enhancing finger vein authentication through deep learning: A comparative study of U-Net and sequential models," *J. Adv. Res. Appl. Sci. Eng. Technol.*, vol. 47, no. 1, pp. 230–243, Jun. 2024.

[12] Y. Wang, J. Gui, Y. Y. Tang, and J. Tin-Yau Kwok, "CFVNet: An end-to-end cancelable finger vein network for recognition," *IEEE Trans. Inf. Forensics Security*, vol. 19, pp. 7810–7823, 2024.

[13] T. S. Kishore and S. M. Kumar, "A framework for protected digitized registry generation using novel finger vein problem," in *Proc. AIP Conf.*, vol. 2853, 2024, Art. no. 020110.

[14] Y. Liu, W. Yang, and Q. Liao, "DiffVein: A unified diffusion network for finger vein segmentation and authentication," 2024, *arXiv:2402.02060*.

[15] G. P. Babu, U. S. Aswal, M. Sindhu, V. J. Upadhye, L. Natrayan, and H. Patil, "Enhancing security with machine learning-based finger-vein biometric authentication system," in *Proc. 5th Int. Conf. Mobile Comput. Sustain. Informat. (ICMCSI)*, Jan. 2024, pp. 797–802.
[16] J. Vijayarajan, "Multi-biometric system based on the fusion of fingerprint and finger-vein," *ELCVIA Electron. Lett. Comput. Vis. Image Anal.*, vol. 23, no. 1, pp. 32–46, Jul. 2024.
[17] H. Kavitha, K. S. Anudeep, K. M. Prajwal, S. Gowda, and G. N. Venkateshprasad, "Finger vein recognition system using bilinear fusion of multiscale features," in *Proc. Int. Conf. Data Sci. Netw. Secur. (ICDSNS)*, Jul. 2024, pp. 1–7.
[18] S. Fairuz, M. H. Habaebi, and E. M. A. Elsheikh, "Pre-trained based CNN model to identify finger vein," *Bull. Electr. Eng. Informat.*, vol. 8, no. 3, pp. 855–862, Sep. 2019.
[19] H. Heidari and A. Chalechale, "Biometric authentication using a deep learning approach based on different level fusion of finger knuckle print and fingernail," *Exp. Syst. Appl.*, vol. 191, Apr. 2022, Art. no. 116278.
[20] R. Ramachandra and S. Venkatesh, "Fingervein verification using convolutional multi-head attention network," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2024, pp. 6163–6172.
[21] M. Hammad, M. A. Wani, K. A. Shakil, H. Shaiba, and A. A. A. El-Latif, "Deep cancelable multibiometric finger vein and fingerprint authentication with non-negative matrix factorization," *IEEE Access*, vol. 12, pp. 120638–120660, 2024.
[22] V. Gurunathan, R. Sudhakar, T. Sathiyapriya, and J. Soundappan, "Finger vein authentication using vision transformer," in *Proc. Int. Conf. Sci. Technol. Eng. Manage. (ICSTEM)*, Apr. 2024, pp. 1–5.
[23] B. Subramaniam, S. V. Krishnan, S. Radhakrishnan, and V. E. Balas, "Fusion of finger vein images, at score level, for personal authentication," *Acta Polytechnica Hungarica*, vol. 21, no. 6, pp. 53–68, 2024.
[24] M. Mufeez Ibrahim, C. Sujith, M. Florence S, and S. Rajagopal, "Enhancing ATM security: A finger vein biometrics approach," in *Proc. 2nd Int. Conf. Netw. Commun. (ICNWC)*, Apr. 2024, pp. 1–4.
[25] H. Krishnan and S. Khare, "Finger vein recognition using deep learning," *Math. Statistician Eng. Appl.*, vol. 71, no. 3, pp. 352–360, 2022.
[26] H. Zhang, W. Sun, and L. Lv, "A frequency-injection backdoor attack against DNN-based finger vein verification," *Comput. Secur.*, vol. 144, Sep. 2024, Art. no. 103956.
[27] P. Zhao, Y. Song, S. Wang, J.-H. Xue, S. Zhao, Q. Liao, and W. Yang, "VPCFormer: A transformer-based multi-view finger vein recognition model and a new benchmark," *Pattern Recognit.*, vol. 148, Apr. 2024, Art. no. 110170.
[28] M. V. Madhusudhan, V. Udayarani, and C. Hegde, "An intelligent deep learning LSTM-DM tool for finger vein recognition model USING DSAE classifier," *Int. J. Syst. Assurance Eng. Manage.*, vol. 15, no. 1, pp. 532–540, Jan. 2024.
[29] M. Li, Y. Gong, and Z. Zheng, "Finger vein identification based on large kernel convolution and attention mechanism," *Sensors*, vol. 24, no. 4, p. 1132, Feb. 2024.
[30] G. K. Kyeremeh, M. Abdul-Al, R. Qahwaji, and R. A. Abd-Alhameed, "Verification technology for finger vein biometric," 2024, *arXiv:2405.11540*.
[31] H. Zhang, W. Sun, and L. Lv, "Multi scale-aware attention for pyramid convolution network on finger vein recognition," *Sci. Rep.*, vol. 14, no. 1, p. 475, Jan. 2024.
[32] M. Chaa, R. Khelil, and D. Khenfer, "Multimodal biometric system utilizing palm vein and finger vein recognition," Ph.D. thesis, Univ. Kasdi Merbah Ouargla, Springfield, MO, USA.
[33] *Finger Vein*. Accessed: Oct. 17, 2024. [Online]. Available: https://www.kaggle.com/datasets/ryeltsin/finger-vein
[34] F. Dhib, M. Machhout, and A. Taoufik, "Pre-processing image algorithm for fingerprint recognition and its implementation on DSP TMS320C6416," *Int. J. Softw. Eng. Appl.*, vol. 9, no. 4, pp. 65–79, Jul. 2018.
[35] S. Mascarenhas and M. Agarwal, "A comparison between VGG16, VGG19 and ResNet50 architecture frameworks for image classification," in *Proc. Int. Conf. Disruptive Technol. Multi-Disciplinary Res. Appl. (CENTCON)*, vol. 1, Nov. 2021, pp. 96–99.
[36] W. Xu, L. Shen, H. Wang, and Y. Yao, "A finger vein feature extraction network fusing global/local features and its lightweight network," *Evolving Syst.*, vol. 14, no. 5, pp. 873–889, Oct. 2023.
[37] Y. Chen, J. Jie, J. Chai, H. Zheng, and X. Wu, "Dual-channel heterogeneous network for finger vein recognition based on transfer learning and improved coordinate attention," *Int. J. Comput. Sci. Math.*, vol. 19, no. 4, pp. 299–317, 2024.
[38] X. Ma and X. Luo, "Finger vein recognition method based on ant colony optimization and improved EfficientNetV2," *Math. Biosciences Eng.*, vol. 20, no. 6, pp. 11081–11100, 2023.
[39] M. Rajalakshmi and K. Annapurani, "A deep learning based palmar vein recognition: Transfer learning and feature learning approaches," in *Proc. Int. Conf. Deep Learn., Comput. Intell.* Nanjing, China: Southeast Univ., 2024, pp. 581–591.
[40] B. Hou, H. Zhang, and R. Yan, "Finger-vein biometric recognition: A review," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–26, 2022.

U. SUMALATHA received the B.E. and M.Tech. degrees from Viswesvaraya Technological University, Belagavi. She is currently pursuing the Ph.D. degree with the Department of Information and Communication Technology, Manipal Institute of Technology, Karnataka. Her current research interests include deep learning, biometrics, multimodal biometric systems, biometric template security, privacy enhancing technologies, cryptographic protocols, and their applications.

K. KRISHNA PRAKASHA (Senior Member, IEEE) received the B.E. and M.Tech. degrees from Viswesvaraya Technological University, Belagavi, and the Ph.D. degree in network security from Manipal Academy of Higher Education (MAHE), Manipal, India. He is currently an Associate Professor with the Department of Information and Communication Technology, Manipal Institute of Technology, MAHE. He has more than 35 publications in national and international conferences and journals. His current research interests include information security, network security, algorithms, real-time systems, and wireless sensor networks.

SRIKANTH PRABHU (Senior Member, IEEE) received the M.Sc., M.Tech., and Ph.D. degrees from IIT Kharagpur. He is currently a Professor with the Department of Computer Science and Engineering, Manipal Institute of Technology, MAHE, Manipal. He has more than 150 publications in national and international conferences and journals. His current research interests include pattern recognition, pattern classification, fuzzy logic, image processing, and parallel processing.

VINOD C. NAYAK received the M.B.B.S. and M.D. degrees from the Kasturba Medical College, Manipal Academy of Higher Education (MAHE), Manipal, India. He is currently a Professor with the Department of Forensic Medicine, Kasturba Medical College, MAHE. His current research interests include public health, epidemiology, traffic medicine, suicidology, toxicology, medical ethics and laws about medicine, medical education, endocrinology, and forensic pathology.