Improving Library Resource Access with Face Recognition-Enhanced VPN Security

S.Balachandran, *Member, IAENG,* and J.Dominic

***Abstract*—In the digital age, where data security is paramount, managing the security of library repositories poses significant challenges due to the increasing threat of unauthorized access and data breaches. Face recognition technology has emerged as a robust and efficient method for authentication. This research article aims to develop a face recognition-based Virtual Private Network (VPN) authentication system, leveraging pre-trained Convolutional Neural Networks (CNNs) to provide secure and seamless access to library repositories. The integration of CNNs in VPN authentication addresses critical security concerns and provides a sophisticated user identity verification method. Unlike traditional methods that rely on passwords or tokens, this system utilizes biometric data, which is inherently unique and difficult to replicate. This approach not only enhances security but also streamlines the user experience, reducing the complexity and time required to access VPN services. To train the CNN-based face recognition module, we built a dataset comprising 15,000 facial images from 300 library users. After comprehensive training and evaluation, the model achieved a remarkable final validation accuracy of 98.01% and a final validation loss of 0.0621, indicating its robust reliability and effectiveness in accurately identifying authorized users. This research study underscores the potential of pre-trained CNNs in revolutionizing network security, providing a strong foundation for future developments in this critical area.**

***Index Terms*—Face Recognition, VPN Authentication, Convolutional Neural Networks, Biometric Security, Deep Learning, Library Repositories, Identity Verification**

I. Introduction

THE proliferation of digital technologies has profoundly transformed various aspects of our daily lives, leading to significant advancements in security and convenience. Among these innovations, face recognition technology has emerged as a powerful tool for authentication and access control, particularly in securing sensitive systems. Face recognition technology, which determines individual identities through the analysis of facial features, represents a fusion of artificial intelligence (AI) and computer vision. Significant advancements in machine learning and neural networks have led to remarkable evolution in this field. Artificial Intelligence (AI) is dedicated to developing systems that can execute tasks traditionally requiring human intelligence. Within AI, CNNs are a class of deep learning algorithms particularly effective for processing images. CNNs excel at extracting features organized in a hierarchical manner from images, making them well-suited for face recognition tasks.

We use CNNs to enhance the accuracy and efficiency of authentication by addressing variations in face images, such as image blur, which can affect clarity. By employing advanced processing techniques within the, we improve recognition performance, ensuring reliable and accurate authentication. Our research focuses on developing a face recognition-based VPN authentication system utilizing CNNs. This system aims to provide a seamless and highly secure authentication process, overcoming the vulnerabilities of traditional VPN methods, such as username-password combinations, which are often susceptible to breaches and can be cumbersome.

We have integrated face recognition technology into a VPN framework. In this setup, a VPN server operates within a virtual machine, managing access to a library repository of databases and additional VMs. Face recognition serves as the authentication mechanism, enhancing security and streamlining the user experience. Our objective is to develop a VPN-based framework that enables seamless and secure interconnection between libraries, removing the dependency on third-party agencies. The system incorporates AI-based face recognition technology to authenticate users, ensuring secure and personalized access to the VPN server. Additionally, we implement mirrored RAID (Redundant Array of Independent Disks) at two locations for data redundancy and reliability, providing data security and availability. This setup ensures that librarians and staff can focus on data entry without concerns about system reliability or security breaches.

The VPN-based access will enable library staff and users to securely access resources from any location and at any time. By leveraging VPN technology, our system will facilitate inter-library collaboration and provide access to a wide range of digital e-resources, including IEEE, Elsevier, Springer, and Wiley etc. We will employ modern technologies including Machine Learning, Artificial Intelligence, and Deep Learning to efficiently manage and retrieve library repositories. The integration of face recognition technology into VPN authentication constitutes a major breakthrough in cybersecurity, providing a more secure, efficient, and user-friendly solution compared to traditional methods. As face recognition technology evolves, it will likely enhance security and convenience across various domains. Our research contributes valuable insights and practical solutions for the future development of face recognition technology.

Manuscript received September 18, 2024; revised February 13, 2025.

S. Balachandran is a Research Scholar of Hindustan Institute of Technology and Science, Chennai, Tamil Nadu, India. (e-mail: itsbaala@gmail.com).

J. Dominic is a Chief Librarian of Central Library of the Hindustan Institute of Technology and Science, Chennai, Tamil Nadu, India (e-mail: jdom16@gmail.com).

II. Literature Survey

The integration of artificial intelligence (AI) and machine learning (ML) in digital libraries has notably enhanced content classification and recommendation systems. Research indicates that Neuro-Fuzzy systems and Support Vector Machines can achieve over 97% accuracy in categorizing and recommending digital content, improving upon traditional methods by automating classification and managing large-scale content more effectively [1]. Biometric authentication systems have advanced significantly, with extensive evaluations of various biometric traits [2]. Facial recognition, in particular, leverages Principal Component Analysis (PCA) for dimensionality reduction, improving the accuracy of facial feature processing and person authentication [3]. Deep learning techniques, such as CNNs, are increasingly used for facial emotion recognition, achieving high accuracy in real-time applications [4].

VPN and proxy protocols in managing library networks, evaluating their performance in security, data handling, and efficiency. VPNs excel in secure, encrypted connections for sensitive data, while proxies offer enhanced privacy and performance through caching. This study provides insights into the strengths and limitations of each, helping libraries make informed choices for secure and efficient network management [5]. VPN-based cloud strategy using SoftEther VPN and Microsoft Azure to securely manage distributed library repositories. The prototype demonstrated strong performance, security, and scalability, supporting numerous concurrent users. Future studies could explore different VPN and cloud options to enhance its adaptability for library management [6].

The approach for detecting copy-move forgery in digital images combines Discrete Cosine Transform (DCT) and Gray-Level Co-occurrence Matrix (GLCM) features with block matching. By leveraging Stationary Wavelet Transform (SWT) for image decomposition, the method extracts composite features and applies block matching to locate forgery regions. Evaluated on the CoMoFoD dataset, this technique achieves impressive results with an F1-score of 95.12% at the image level and 92.86% at the pixel level, demonstrating its capability to accurately distinguish authentic from tampered images [7]. The integration of facial recognition with encryption methods, like fully homomorphic encryption, enhances security by maintaining user confidentiality [8].

CNN fragility when dealing with imbalanced data has been addressed by diversifying latent features, which improves robustness and performance [9]. CNN-based algorithms are also used in UAV applications for efficient low-resolution facial detection, demonstrating significant improvements in speed and accuracy [10]. Advances in image classification through CNN and clustering-based techniques provide enhanced object recognition capabilities [11]. Data augmentation techniques, such as image resizing and grayscale conversion, are employed to enhance dataset robustness and model training [12]. The combination of Role-Based Access Control (RBAC) and Attribute-Based Access Control (ABAC) models offers secure and adaptable solutions for remote data access, integrating two-factor authentication for added security [13]. AES256 and MD5 encryption improve password security by increasing the time required for cracking [14]. RobFaceNet, a CNN model designed for face recognition, incorporates attention mechanisms to extract facial features effectively, outperforming models like ArcFace in accuracy and efficiency [15]. Cloud computing has transformed software development, providing scalable platforms that enhance software quality and efficiency while addressing adoption challenges [16]. Deep learning models, such as Nave Bayes and FFNN, are used to improve VPN security by predicting and preventing attacks, achieving high accuracy in detecting intrusions [17]. Facial emotion recognition has also advanced through CNNs, with improved accuracy on datasets like FER-2013 and CK+, enhancing sentiment analysis capabilities [18].

Data augmentation strategies, including image mixing and deletion, are used to prevent overfitting and enhance model robustness [19]. VPNs utilizing IPSec and L2TP tunnelling protocols offer significant network security improvements but also present inherent risks [20]. The energy consumption of IPSec VPN tunnels is analyzed to balance encryption complexity with environmental impact, highlighting the importance of energy-efficient configurations [21]. The IBNNER model, a biaffine-based approach for Chinese nested NER in medical texts, uses BERT embeddings, biaffine mapping, and CNNs for enhanced data distribution. Tested on CMeEE and CLUENER2020 datasets, it improves recall by 5.04% over BERT and outperforms BiLSTM with 7.37% higher precision, 10.51% recall, and 8.95% F1-score [22]. Performance analyses of IPSec VPN over TCP/UDP with different encryption algorithms provide recommendations for optimizing VPN deployment and configuration [23].

Solving the Navier equation for elastic behavior in solids combines domain decomposition and meshless methods with deep learning. Using the Schwarz wave relaxation method, this model applies a parallelized meshless approach with neural networks, demonstrating high accuracy and efficiency in numerical tests and advancing deep learning in solid mechanics simulation [24]. An enhanced Swin-Unet model segments green tomatoes in agricultural settings, addressing visual similarity and occlusion challenges. With an optimized Attention Gate (AG) to highlight tomato-specific features and an Atrous Spatial Pyramid Pooling (ASPP) module for multi-scale extraction and edge smoothing, the model achieves 97.5%-pixel accuracy, 92.4% Dice similarity, and 85.9% IOU, outperforming previous models and advancing applications in smart agriculture [25]. Deep learning, especially in computer vision, faces challenges like overfitting and prolonged learning times due to numerous parameters and inadequate training samples. This work proposes a two-tiered DCNN approach for image classification: the first level enhances training images by removing unnecessary details, and the second focuses on edge detection to further reduce learning time. Inspired by human vision, this approach aims to expedite CNN training with optimized, lighter samples for real-time use. Results show a 94% reduction in learning time on the Animals10 dataset, achieving 99.2% validation accuracy, outperforming conventional DCNNs for real-time classification [26].

Graphical authentication methods, such as FacetPass, enhance security through multi-faceted graphical passwords and automatic facial recognition, outperforming traditional systems like VAP codes and EvoPass [27]. OpenCV-based systems are used for COVID-19 screening, verifying mask usage and vaccination status with high accuracy [28]. The integration of CNN and LSTM models automates attendance tracking with notable accuracy improvements [29]. The

combination of LBPH and CNN for face recognition shows significant improvements in accuracy and efficiency over earlier methods [30]. Deep learning frameworks, such as Keras and TensorFlow, are essential for automated face recognition systems but may face issues like silent bugs affecting reliability [31]. A touchless face recognition system for smart attendance, using an 11-layer CNN with ReLu activation, achieved 96.2% accuracy on a 1,890-face dataset. Designed for office and college settings, it employs Viola-Jones for multi-face detection and includes a web app for attendance tracking and report generation. [32].

A Hybrid Quantum-Classical Convolutional Neural Network (HQCNN) is proposed to classify challenging spectrogram images from the Allen Telescope's SETI dataset. By using quantum nodes with superposition and entanglement, HQCNN captures intricate features more precisely than traditional models. Experiments on Google Collaboratory with Pennylane, Keras, and TensorFlow show HQCNN achieves 90.19% accuracy, underscoring the promise of Quantum Machine Learning in complex signal processing [33]. The use of adversarial 3D patches to challenge face recognition systems has improved robustness against physical attacks [34]. RSA encryption is emphasized for securing large data transmissions in VPNs against multiple attack vectors [35]. Predictive models for supercapacitor degradation using CNNs and Informer frameworks demonstrate improved prediction accuracy [36]. RBAC-based authentication schemes in smart factories optimize authentication processes for multiple devices, reducing overhead and enhancing security [37].

III. Methodology

The proposed approach involves four essential steps: data preparation, model training, real-time face recognition, and VPN server connection. The first step involves data preparation and the collection of face samples, establishing a comprehensive dataset for training. The second step focuses on training a CNN based model specifically designed for face recognition, enabling accurate and robust identification. In the third step, the system performs real-time face recognition, leveraging the trained model to authenticate users. Finally, in the fourth step, the authenticated user is securely connected to the VPN server. This integrated approach combines state-of-the-art machine learning techniques with secure data handling protocols, resulting in a dependable and user-centric authentication system, as depicted in Fig. 1

Step 1: Data preparation and face sample collection

The first step involves the collection and preparation of face samples, crucial for training the CNN based model. Using OpenCV, a series of face images were captured through a webcam and systematically organized into a structured directory, where each folder corresponds to a unique user ID. This organization allows the model to effectively learn and distinguish between different faces. The collected images underwent preprocessing, including resizing, grayscale conversion, and data augmentation using TensorFlow's Image Data Generator. These techniques ensured a robust dataset that accounts for variations in pose, lighting, and expression, laying a solid foundation for the next phase of training.

Step 2: Training a CNN-Based Face Recognition Model

In the second step, the development of the face recognition model is carried out using Convolutional Neural Networks (CNNs) with TensorFlow and Keras. In order to extract and learn facial features from the given dataset, the CNN architecture has been meticulously built. Multiple convolutional layers and pooling layers make up the model, which aids in capturing key features and decreasing the spatial dimensions. The augmented face samples are used to train the model, and optimization methods like the Adam optimizer are used to minimize the loss function. Once the model training is completed, the trained model is saved as a .h5 file, ensuring easy deployment in the subsequent steps. Throughout this phase, TensorFlow's capabilities are leveraged to ensure efficient training, resulting in a model that is capable of high-accuracy face recognition.

Step 3: Face Recognition

In the face recognition process, the system initiates by loading a pre-trained model designed specifically for facial recognition. This model is critical in classifying detected faces by correlating them with user IDs stored in the system. Each registered user has a unique ID, which the model associates with facial data to ensure accurate identification. Following model preparation, a live video stream is captured using OpenCV, which provides real-time frames for face analysis. This dynamic feed allows the system to respond immediately when a face appears in the camera view. For face detection, the system employs a Haar Cascade classifier, which identifies faces by locating patterns that match facial features, such as the relative positions of the eyes, nose, and mouth. When a face is detected, a bounding box is drawn around it, enabling the system to isolate only the face region for further analysis. Once isolated, the detected face image undergoes preprocessing. This involves resizing it to 128x128 pixels and normalizing the pixel values to enhance prediction accuracy.

The pre-processed face is then passed to the model, which generates probabilities for each registered user in the system. The model identifies the class (user) with the highest probability and compares this probability to a set threshold (e.g., 95%) to ensure confidence in recognition. When a face is recognized with confidence above this threshold, the system saves the recognized face image for reference and displays the user's primary ID and name on the interface. This instant feedback provides real-time recognition results to the user. This cycle continues until the system successfully recognizes a face or the session is terminated, ensuring a seamless and user-friendly experience in face recognition and display of identification data.

Step 4: Connect to VPN Server

The system securely stores VPN credentials encrypted using the Advanced Encryption Standard (AES-256) within a relational SQL database, linking each set of credentials to a user's primary ID. The database schema is meticulously designed, with encrypted credentials stored in varbinary(MAX) format, ensuring compatibility and scalability. Upon successful face recognition, the system queries the SQL database to retrieve the associated credentials based on the recognized user's primary ID.

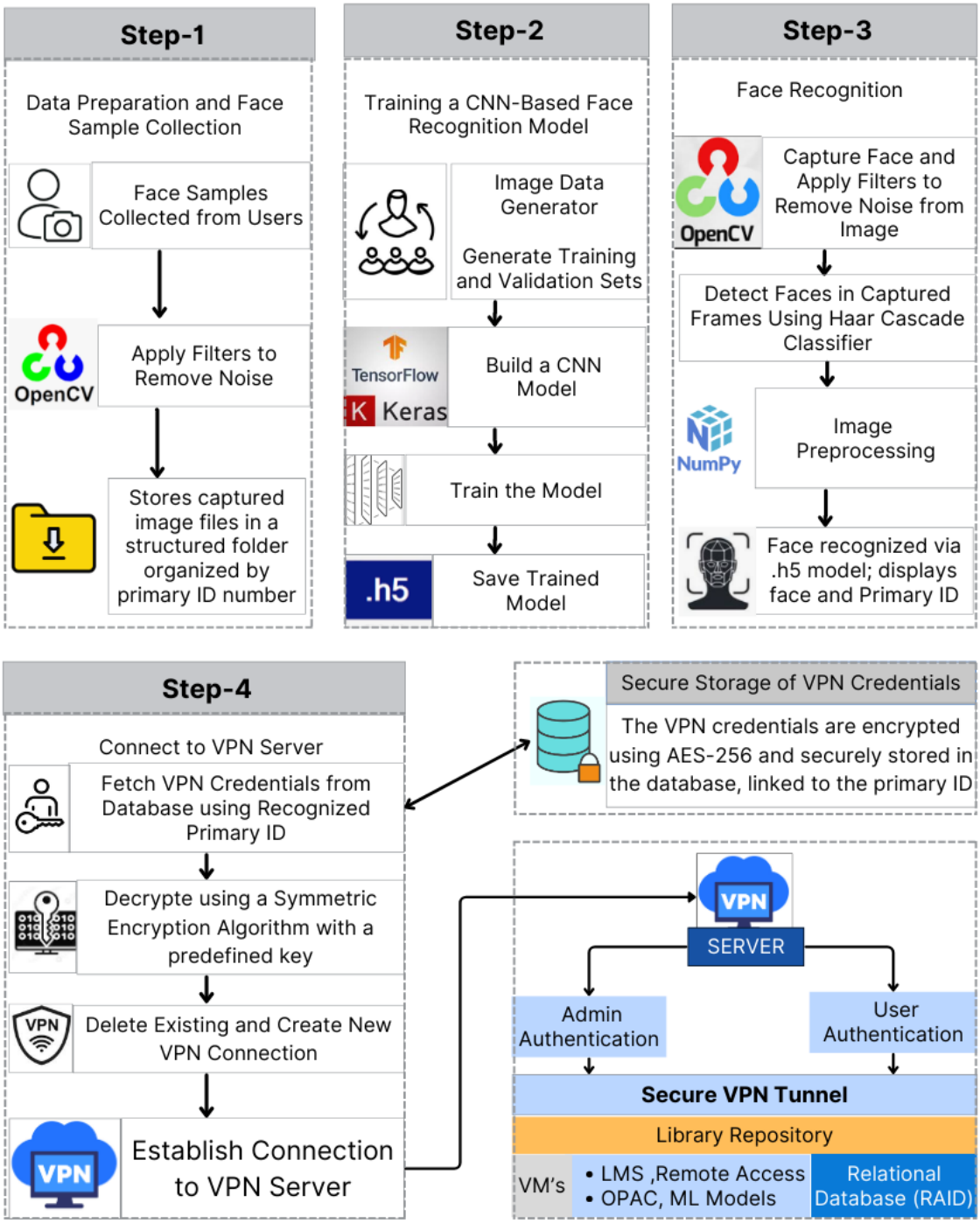

Fig. 1. Architectural Design of a Face Recognition based VPN Authentical System for Secure Access to Library Repositories

A symmetric encryption algorithm with a predefined key decrypt the credentials, ensuring the confidentiality of sensitive information. Before establishing a new VPN connection, the system deletes any existing VPN connections to ensure a clean start. The decrypted credentials are then used to create a new VPN connection, enabling secure access to resources. The system provides authenticated users with appropriate access rights through the VPN server. Advanced protocols, such as L2TP/IPsec with Certificate, L2TP/IPsec with Pre-Shared Key, and PPTP, are employed to ensure robust encryption and secure communication channels. Additionally, AES-256 encryption safeguards data during transmission, while SHA-256 and RSA algorithms maintain data integrity and facilitate secure key exchanges, ensuring the security of client-server communication. The SQL database also plays a crucial role in managing structured storage for user credentials and face embeddings, enabling efficient querying and secure management. To enhance reliability, the system employs a RAID configuration for data redundancy, mirroring data across multiple disks to ensure high availability and fault tolerance. This minimizes the risk of data loss due to hardware failures, providing a dependable storage solution for sensitive information. Role-Based Access Control (RBAC) policies are enforced at the database level to assign specific roles and permissions to users, ensuring access is restricted only to authorized resources. Comprehensive logging of all database queries and access actions supports tracking and auditing, further enhancing system security. The database also supports high-performance querying to facilitate real-time access during user authentication and VPN connection processes.

A secure VPN tunnel is established, allowing authenticated

users to access virtual machines (VMs) and the library repository. This tunnel ensures that all data exchanges are encrypted, preserving confidentiality and integrity. The VPN server continuously monitors active user sessions, verifying that only legitimate users maintain access to sensitive resources.

Through the combined implementation of robust encryption standards, SQL database management, RAID configurations, and access control techniques, the system ensures secure resource sharing and library management. This layered approach guarantees that sensitive data is protected at every stage, providing a reliable and efficient solution for library resource access and VPN connectivity.

A. Image Capture

The initial step in our face recognition-based VPN authentication system involves the capture of facial images. This process is critical as it establishes the foundational data upon which the entire recognition system relies. Facial image capture is executed through a custom developed graphical user interface (GUI) application using tkinter for GUI and OpenCV for real-time image processing.

The application captures images from a webcam, which are subsequently processed and saved for training and testing purposes. The webcam user interface instantiates to initiate the image capture process. The system continuously captures frames until the specified and required number of facial image samples were obtained. The captured images are converted into grayscale with the mathematical representation as given in equation (1).

$$Igray = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B \quad (1)$$

where R, G, and B represents the red, green, and blue colour channels of image respectively. This grayscale conversion contributes to simplify the computational requirements and moreover enhances face detection accuracy.

B. Face Detection and Region on Interest (ROI) Image Cropping

Face Detection: The system uses a robust Haar Cascade classifier to detect faces in captured frames. This classifier identifies the coordinates of the Region of Interest (ROI) for detected faces. Specifically, x, y represent the top-left corner of the bounding box, and w, h indicate the width and height of the box, respectively.

ROI Extraction and Preprocessing: Once a face is detected, the ROI is extracted using equation (2). The extracted face is resized to a standardized dimension of 64×64 pixels using an appropriate interpolation method to maintain uniformity across the dataset. This ensures that variations in image size do not impact model training and testing.

$$ROI = Igray[y: y + h, x: x + w] \quad (2)$$

Data Organization: The processed ROI face images are stored in a hierarchical directory structure organized by library user IDs. This directory structure facilitates easy retrieval for training and testing. Each image is saved with a unique filename following the pattern 'user_id_count.jpg,' where 'count' represents the sequence number of the image.

C. Data Collection and Description

The dataset used in this study was meticulously gathered to develop and evaluate the face recognition-based VPN authentication system. The data collection process included capturing face images from four distinct categories of library users: male students, female students, male faculty members, and female faculty members.

Each library participant contributed 50 face images, of which 10 images were reserved for testing and 40 images were utilized to train the model. TABLE I provides specifics on the dataset distribution. This balanced and representative dataset collection is done to ensure the reliability and effectiveness of the face recognition model, providing a solid foundation for the development of the VPN authentication system.

TABLE I
FACE SAMPLE DATASET DISTRIBUTION FOR LIBRARY USERS

Description	Count	Face Captured per Count	Total Face Images	Training Images 80% (40 per person)	Testing Images 20% (10 per person)
No. of Students (Male)	100	50	5000	4000	1000
No. of Students (Female)	100	50	5000	4000	1000
No. of Faculty (Male)	50	50	2500	2000	500
No. of Faculty (Female)	50	50	2500	2000	500
Total	300	-	15000	12000	3000

D. Data Preparation & Data Augmentation

Data augmentation methods are applied to expand the diversity of the training dataset, it enhances the robustness of the model. The key transformations contain Rotation, Translation, Shearing, Zooming and Horizontal flipping. The images are augmented with rotation operation by rotating the images by an angle θ using the rotation matrix this transformation exposes the model to different image orientations, which improves its ability to generalize. The mathematical expression for image rotation operation is given by

$$\text{Rotation}(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \quad (3)$$

The image translation operation is performed by horizontally and vertically shifting the images with Δx and Δy along the x and y axes, respectively. The mathematical expression for image translation operation is given by

$$Translation(\Delta x, \Delta y) = \begin{bmatrix} 1 & 0 & \Delta x \\ 0 & 1 & \Delta y \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

The image shearing operation is performed with a shear factor s, allowing the model to handle distortions and is represented mathematically as

$$Shearing(s) = \begin{bmatrix} 1 & S \\ 0 & 1 \end{bmatrix} \quad (5)$$

The image zooming operation is performed by considering α is the zoom factor for zoom in or out, which helps in learning facial features at different scales. The mathematical representation of zoom operation is

$$Zoom(\alpha) = \begin{bmatrix} \alpha & 0 \\ 0 & \alpha \end{bmatrix} \quad (6)$$

The image flipping operation-based augmentation is useful for learning facial features that are invariant to horizontal orientation.

$$Flip = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \quad (7)$$

E. Convolutional Neural Network Architecture

In our research, we have considered implementing eight distinct Convolutional Neural Network (CNN) models for face recognition. These include custom basic CNN model and transfer learning performed with seven selectively chosen pre-trained CNN models namely MobileNetV2, EfficientNetB0, DenseNet201, Xception, EfficientNetV2S, and ResNet50.

TABLE II
DATASET DISTRIBUTION BETWEEN TRAINING AND TESTING PHASES

Category	Training Percentage	Testing Percentage
No. of Students (Male)	80%	20%
No. of Students (Female)	80%	20%
No. of Faculty (Male)	80%	20%
No. of Faculty (Female)	80%	20%
Total	80%	20%

The TABLE II outlines the face image collection as percentage distribution between the training and testing datasets, totaling 15,000 images across all participants. A train-test split ratio of 80% of the images (12000) used for training and 20% (3000) set aside for testing divides the dataset into training and testing sets. The preparation of the dataset is crucial for the successful training of a CNN. Images are organized into directories according to their class labels. This labelling allows the model to learn the unique characteristics of each class. The dataset is divided into training and validation sets to ensure that the model undergoes evaluation using new, unfamiliar data. Usually, an 80/20 split is utilized to divide the dataset, using 80% of the images for training and 20% for validation.

Selecting appropriate pre-trained CNN model for performing transfer learning itself requires strategical analysis, the goal is often to achieve the best performance with the least computational cost. The selection strategy depends on the specific requirements of the deployment environment, such as available memory, computational power, and the desired level of accuracy.

TABLE III depicts the traits summary of selected pre-trained CNN models in terms of size, number of model parameters, depth and Top-5 Accuracy performance achieved with ImageNet dataset. The custom CNN model is relatively

TABLE III
PERCENTAGE DISTRIBUTION TRAINING AND TESTING DATASETS

CNN Model	Size	Parameters	Depth	Top-5 Accuracy
Basic CNN	6	1.6M	5	--
MobileNetV2	14	3.5M	105	90.1%
EfficientnetB0	29	5.3M	132	93.3%
DenseNet201	80	20.2M	402	93.6%
Xception	88	22.9M	81	94.5%
EfficientNetV2S	88	21.6M	132	96.7%
ResNet50	98	25.6M	107	92.1%

simple model, with 1.6 million parameters and a depth of 5 layers, demonstrates competitive performance, suggesting its efficacy for tasks requiring a balance between model complexity and accuracy. The MobileNetV2 model is a lightweight architecture, consisting of 3.5 million parameters and a depth of 105 layers, makes it highly efficient and suitable for real-time applications. For applications where model size and computational efficiency are critical, MobileNetV2 offers a smaller footprint with decent accuracy 90.1%. On the other hand, Xception and DenseNet201 are suitable for scenarios where slightly higher computational resources are available, but accuracy is still a priority.

Models like EfficientNetV2S and EfficientNetB0 demonstrate the ability to balance accuracy and model size effectively. EfficientNetV2S in particular stands out with the highest Top-5 accuracy (96.7%) while maintaining a manageable size (88 MB), making it the perfect option for uses that demand high accuracy without significantly

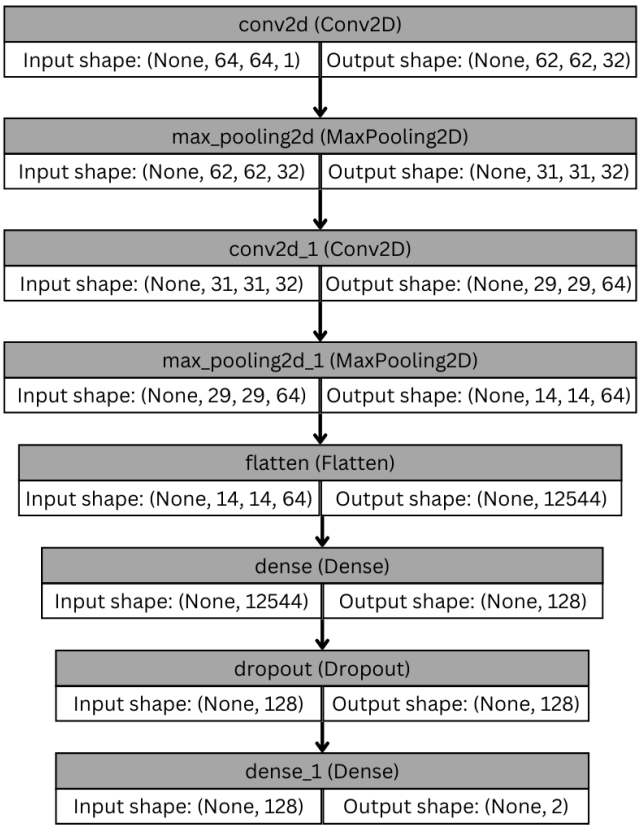

Fig. 2. Detailed architecture of the Custom CNN

increasing the computational load. The models listed provide a range of options to suit different needs, from lightweight

models like MobileNetV2 to high-performance models like EfficientNetV2S. Based on these strategies we have selected these pre-trained CNN model to develop our face recognition module.

Fig. 2 illustrates the detailed architecture of the CNN used in this study. The network's multiple layers, including convolutional, max-pooling, dropout, and dense layers, are depicted in the diagram. The image includes information on the number of filters, kernel sizes, and the shape of data flowing through each layer, providing a comprehensive view of the model's structure and its connections.

The 2D convolutional layers are responsible to extract facial features by applying convolution operation. For a convolution operation, the output is calculated as depicted by

$$(I * K)(i,j) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I(i+m, j+n) \cdot K(m,n) \quad (8)$$

where (i,j) are the coordinates of the output pixel, K is the kernel and I is the input image. The feature maps' spatial dimensions are decreased by the pooling Layers. For MaxPooling, the output is given by

$$P(i,j) = \max_{m,n} \big(I(i+m, j+n)\big) \quad (9)$$

where P is the pooled output and (m,n) are the coordinates within the pooling window. Flattening Layer is responsible to convert 2D facial feature maps into 1D feature vectors as given by

$$Flatten(x) = reshape(x) \quad (10)$$

where x is the input tensor. Then the dense layers perform fully connected operation using the following mathematical relation

$$y = \sigma(Wx + b) \quad (11)$$

where y represents the result, W the weight matrix, x the input vector, b the bias, and σ the activation function, such as ReLU or softmax. We used dropout to reduce overfitting by randomly setting a fraction p of the input units to 0.25 as probability. The mathematical relation of dropout layer is given by

$$Dropout(p) = input \cdot Bernoulli(p) \quad (12)$$

where $Bernoulli(p)$ is random variable with probability p.

F. Transfer Learning and Architectural details

Fig. 3 illustrates the enhanced architecture of the pre-trained CNN model, highlighting the addition of three critical components that optimize the system's performance and generalization capabilities: the GlobalAveragePooling 2D layer, the Dense layer with 1024 units, and the Dropout layer with a rate of 0.5.

GlobalAveragePooling 2D Layer: This layer plays a pivotal role in reducing the spatial dimensions of the feature maps while retaining the most significant information. By aggregating features across spatial dimensions, it minimizes overfitting and enhances the compactness of the learned representation. This contributes to better handling of unseen data.

Dense Layer with 1024 Units: Acting as the fully connected layer, it transforms the summarized features into a high-dimensional space, enabling the model to learn intricate patterns and correlations in the data. This layer is critical for detecting complex relationships and improving classification accuracy.

Dropout Layer (Rate: 0.5): To mitigate overfitting, this layer introduces a regularization mechanism by randomly setting 50% of the units in the preceding layer to zero during training. This ensures the model does not overly rely on specific features, resulting in better generalization.

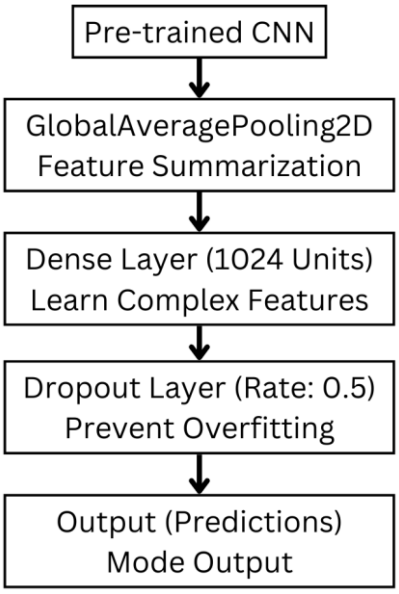

Fig. 3. Enhanced Pre-Trained CNN Architecture

Fig. 3 demonstrates how these layers synergistically enhance the architecture, optimizing the CNN's ability to process facial features efficiently and generalize effectively to new data. This refinement is a cornerstone of our system's high performance, as reflected in the validation accuracy and loss metrics.

G. Face Recognition Mechanism for Secure VPN Connections

The proposed system integrates advanced technologies to create a robust solution for face recognition and VPN management, combining secure encryption mechanisms and state-of-the-art image recognition capabilities. At the core of the face recognition component are Convolutional Neural Networks (CNNs), which excel in image classification tasks by leveraging multiple convolutional layers to automatically extract and learn relevant features from images.

The convolution operation within CNNs is mathematically defined as the sum of element-wise multiplications between the input image and convolutional filters, followed by a non-linear activation function, typically ReLU. To ensure data security, the system employs the AES encryption algorithm in Cipher Feedback Mode (CFB), a symmetric encryption technique that effectively converts a block cipher into a stream cipher, enabling secure encryption of data streams.

The encryption process uses an Initialization Vector (IVec) of 16 bytes, ensuring unique output for every operation. User data, including usernames and passwords, is encrypted using AES-256 and stored in an SQL database as a concatenation of the IVec and ciphertext. The SQL database schema is

meticulously designed, with encrypted credentials stored in varbinary(MAX) format for compatibility and scalability. The decryption workflow mirrors the encryption process, ensuring seamless recovery of original credentials. When encrypted data is retrieved from the database, the stored IVec and ciphertext are separated, and the AES decryption process restores the original plaintext using the encryption key.

The decryption process is mathematically expressed by equation (13)

$$Decrypted\_text = AES\_key(Encrypted\_text) \quad (13)$$

The AES algorithm guarantees that sensitive data remains secure throughout the decryption process. The decrypted credentials are then used to establish a VPN connection. The system automates VPN management using Windows command-line tools and PowerShell scripts, which handle tasks such as creating, configuring, and managing VPN connections. After successful face identification by the CNN module, the system queries the SQL database to verify the user's identity. The encrypted VPN credentials are decrypted using the AES algorithm and securely utilized for establishing the VPN connection. This multi-step verification ensures that only authenticated users can access the VPN, providing a high level of security.

The AES encryption process in CFB mode encrypts data in blocks using key sizes of 128, 192, or 256 bits. It performs transformations such as substitution, permutation, and mixing to achieve secure encryption. In CFB mode, an Initialization Vector (*IVec*) is first encrypted using the AES algorithm, and the final ciphertext is generated by XORing the encrypted *IVec* with the plaintext. This process is repeated for each data block, with the ciphertext of one block serving as the *IVec* for the next. The encryption and decryption processes are defined by the following Encryption equation (14) and Decryption equation (15) respectively.

$$C_i = P_i \oplus E_K(IVec) \quad (14)$$

$$P_i = C_i \oplus E_K(IVec) \quad (15)$$

Here, C_i represents the ciphertext, which is the encrypted output generated from the plaintext data (P_i). $E_K(IVec)$ refers to the AES encryption of the *IVec* or feedback value, where E_K denotes the AES encryption function with a specific key K. The *IVec* ensures that identical plaintexts produce different ciphertexts by acting as the initial input for the encryption process.

In CFB mode, the feedback mechanism secures streaming data by ensuring that each block of data is encrypted using both the previous block's ciphertext and the *IVec*. This approach maintains data confidentiality while enabling continuous data encryption and decryption, making it ideal for applications that require robust security for streaming data.

Fig. 4 illustrates the integrated encryption-decryption workflow, combining AES-256 encryption, SQL storage, face recognition, and VPN management. This visualization details the data flow, from user input through encryption, secure storage, decryption, and VPN connection, emphasizing the system's reliability and efficiency in maintaining data security.

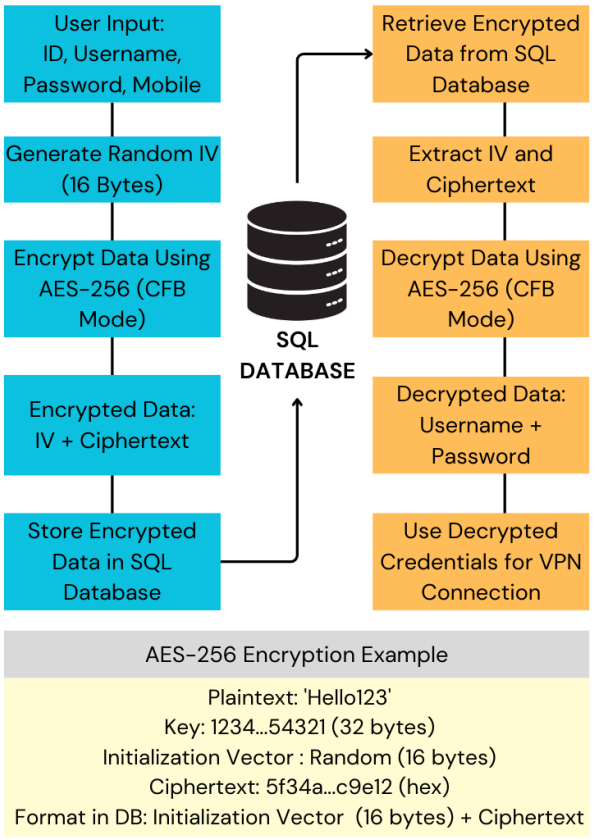

Fig. 4. Encryption and Decryption flowchart with AES-256 and VPN.

IV. Results and Discussion

This system combines biometric face recognition with secure VPN access to ensure efficient and robust remote access to library resources. The system comprises four primary components:

Step 1 (Face Capture Module): The initial step involves capturing and preparing the dataset using a robust face capture module, ensuring high-quality input for subsequent model training. Using OpenCV and Haar Cascade Classifiers, the system detects and crops facial regions from video streams, systematically organizing the images into a structured dataset. This ensures the reliability and robustness of the input data for training MobileNetV2, a lightweight CNN architecture.

Step 2 (CNN Models and Evaluation Metrics): This step evaluates various CNN models, including MobileNetV2, which is implemented using TensorFlow and Keras libraries. The models are optimized through hyperparameter tuning, and their performance is evaluated based on metrics such as accuracy, loss, precision, recall, and F1 scores. The effectiveness of each model is highlighted, showcasing the robustness of MobileNetV2 in achieving high accuracy with minimal computational overhead.

Step 3 (Face Recognition): The trained CNN model is analysed for its real-world applicability, focusing on its accuracy, adaptability to diverse conditions, and robustness against variations in lighting and angles. The implementation leverages pre-processing techniques such as normalization and resizing, ensuring efficient and accurate facial recognition in real-time.

Step 4 (SQL Verification and VPN Server Integration): The final step integrates the face recognition module with a secure backend system. AES-256 encryption ensures that user credentials, including VPN access information, are securely stored in an SQL database. The Routing and Remote Access Service (RRAS) configured on Windows Server 2019, coupled with MS-CHAPv2 and PPTP, guarantees secure and efficient communication. Active Directory integration further enhances user management and access control, enabling seamless and reliable access to digital resources.

The primary objective of this system is to provide a secure, efficient, and user-friendly solution for accessing academic resources remotely. By combining advanced biometric authentication, secure credential management, and encrypted VPN integration, the system addresses the growing need for robust access control in academic and research environments.

Step-1: Face Capture Module: Performance Analysis and Dataset Preparation

In order to ensure building efficient face recognition module in this research, we trained and evaluated various CNN models to efficiently detect faces, with the reported evaluation metrics being Accuracy and Loss. To identify the most robust and efficient face detection CNN model, we experimented different pre-trained CNN models and tested them on the same dataset. In the first step the face capture module is designed to efficiently manage and visualize the image capture process which serves as central need for training the CNN model to perform facial recognition task.

This face capture module systematically counts and categorizes each captured image to ensure a comprehensive analysis of the capture process. During the image capture, unsuccessful attempts are methodically recorded whenever frames cannot be read or when faces are not detected. This method ensures that both successful and unsuccessful captures are accurately tracked, providing a reliable measure of the capture process performance. A straightforward data structure is employed to manage the counts of successful and failed captures for each session. These counts form the basis for generating statistical summaries and visual representations, offering clear insights into the effectiveness of the face capture module.

TABLE IV
STATISTICAL DESCRIPTION OF IMAGE CAPTURE MODULE

Session (ID)	Total Captures	Successful Captures	Failed Captures	Capture Time (secs)
Session 1 (ID: CLBALA7917)	58	50	8	5.6
Session 2 (ID: CLSIVA9713)	50	50	0	5.2
Session 3 (ID: CLRAJA6706)	50	50	0	5.0
Session 4 (ID: CLSRI8308)	56	50	6	5.2

TABLE. IV illustrates the statistical description of the image capture module, highlighting its robust performance and efficiency across multiple sessions. The module consistently achieved 50 successful captures in each session, with Sessions 2 and 3 demonstrating a perfect 100% success rate and no failures. Session 1 recorded the highest total captures at 58, with 8 failures, while Session 4 had 56 total captures and 6 failures, indicating room for optimization when managing higher capture volumes. Capture times remained stable across sessions, ranging between 5.0 and 5.6 seconds, showcasing the module's efficiency and reliability under varying capture loads.

The numerical data and statistical summaries are seamlessly integrated into the application's user interface, allowing users to interactively view results and analyse their capture sessions. This robust approach ensures effective tracking and assessment of the face capture process, supporting the overall reliability of the system.

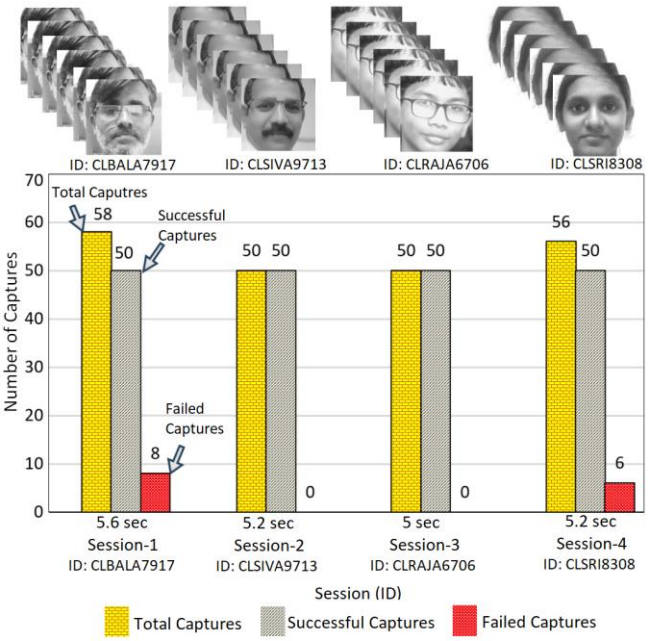

Fig. 5. Numerical statistical analysis of Image Capture Module.

The image capture performance is also visually depicted for four unique instances in Fig. 5, which provides a comprehensive bar chart illustrating the total captures, successful captures, and failed captures for each session, along with the time taken for each. This visual feedback enables users to immediately assess the effectiveness of their capture sessions, facilitating quick evaluation of system performance. Each session's metrics are visually distinguishable, and the clear representation of failures helps identify areas for improvement.

Each capture session is initiated by entering a unique ID number, and the system organizes facial images in a directory specific to this ID, maintaining a structured dataset. Using a Haar Cascade classifier, the module detects faces within frames from the camera. When a face is detected, the module crops and saves the region of interest, incrementing the count of successful captures. Frames without faces or unreadable frames are logged as failures, incrementing the failed capture count. The module's graphical output, alongside real-time image displays, provides a comprehensive analysis of capture success and failure, supporting the capture module's reliability and accuracy in facilitating the training process for facial recognition. This robust approach ensures that users can effectively track and evaluate the capture process, reinforcing the overall performance and reliability of the system.

Step-2: CNN Models and Evaluation Metric

To assess the performance of the most effective CNN

model for face recognition, we have focused on optimizing several key parameters. We fine-tune hyperparameters, including batch size, learning rate, and number of epochs, to improve model performance. Techniques such as grid search and random search are used to identify the optimal hyperparameter values, guided by validation performance. The batch size of 8 was used, and the number of epochs was set to 50. Each epoch consists of multiple iterations over the dataset, updating the model parameters based on gradients computed from the loss function. Adam optimizer is used for training the CNN models and it is mathematically represented as

$$\theta_{t+1} = \theta_t - \eta \frac{m_t}{\sqrt{v_t + \epsilon}} \tag{16}$$

where θ represents model parameters, η is the learning rate, m_t and v_t are moment estimates, and ϵ is a small constant for numerical stability. The learning rate was fixed as 0.001.

A. Evaluation Metrics

Face Recognition with the CNN and its performance is evaluated using the following metrics. The metrics used are Accuracy and Loss. Accuracy assesses the fraction of correctly predicted instances among all instances. This metric is mathematically expressed by

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} \tag{18}$$

The Loss metric quantifies the discrepancy between the true labels and the predicted probabilities. Since the proposed task is a multi-class classification, loss function chosen is Categorical cross-entropy

$$Loss = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} \; y_{ij} \log(\hat{y}_{ij}) \tag{19}$$

In the formula for categorical cross-entropy loss, N represents the number of samples, and C denotes the number of classes. The term y_{ij} is a binary indicator (0 or 1) that denotes whether class j is the true class for sample i. Meanwhile, $\hat{y}_{ij}$ signifies the predicted probability of class j for sample i.

B. Face recognition Performance analysis of CNN

This study evaluates the performance of several convolutional neural network (CNN) architectures based on accuracy, loss, and training efficiency. TABLE V summarizes the results, while Figs. 6–12 illustrate the training and validation accuracy curves for each model, providing insights into their behavior over 50 epochs.

TABLE V
STATISTICAL DESCRIPTION OF IMAGE CAPTURE MODULE

Model	Final Accuracy		Final Loss	
	Training	Val	Training	Val
Basic CNN	0.9122	0.9730	0.2164	0.1024
MobileNetV2	**0.9866**	**0.9801**	**0.0366**	**0.0621**
EfficientnetB0	0.2838	0.2703	1.3826	1.3854
DenseNet201	0.9792	0.7222	0.1477	0.0152
Xception	0.9837	0.9722	0.0040	0.2273
EfficientNetV2S	0.9324	0.6486	0.2412	1.6827
ResNet50	0.2431	0.2500	1.3864	1.3863

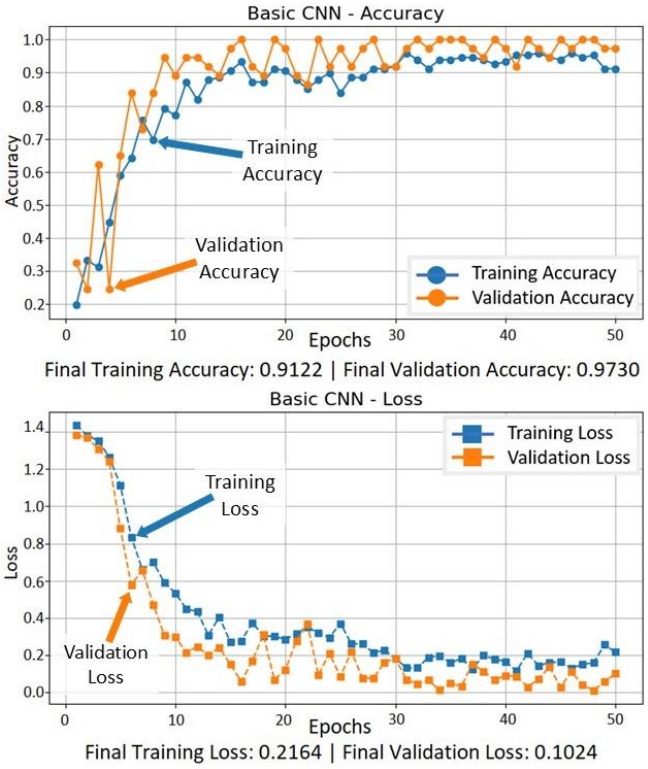

Fig. 6. Training Metrics for Basic CNN Model.

The performance analysis plot of basic CNN model illustrated in Fig. 6 shows achieving a heighest training accuracy of 91.22% and a validation accuracy of 97.30%. It records a training loss of 0.2164 and a validation loss of 0.1024, showing strong baseline performance for simpler architectures with consistent generalization.

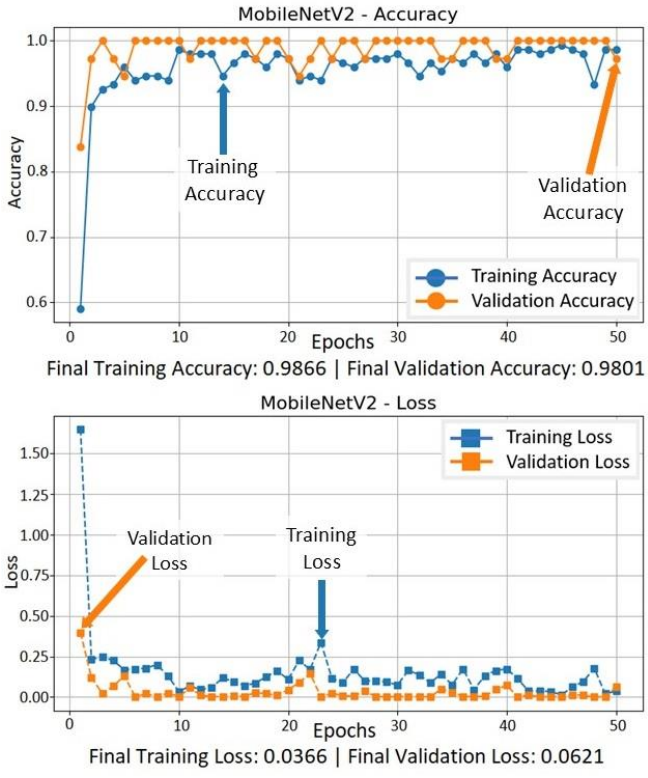

Fig. 7. Training Metrics for MobileNetV2 CNN Model.

The performance analysis plot of MobileNetV2 CNN model illustrated in Fig. 7 shows its excellence with a training accuracy of 98.66% and a perfect validation accuracy of 98.01%. It minimizes training and validation losses to 0.0366 and 0.0621, respectively, demonstrating exceptional robustness and suitability for real-time applications.

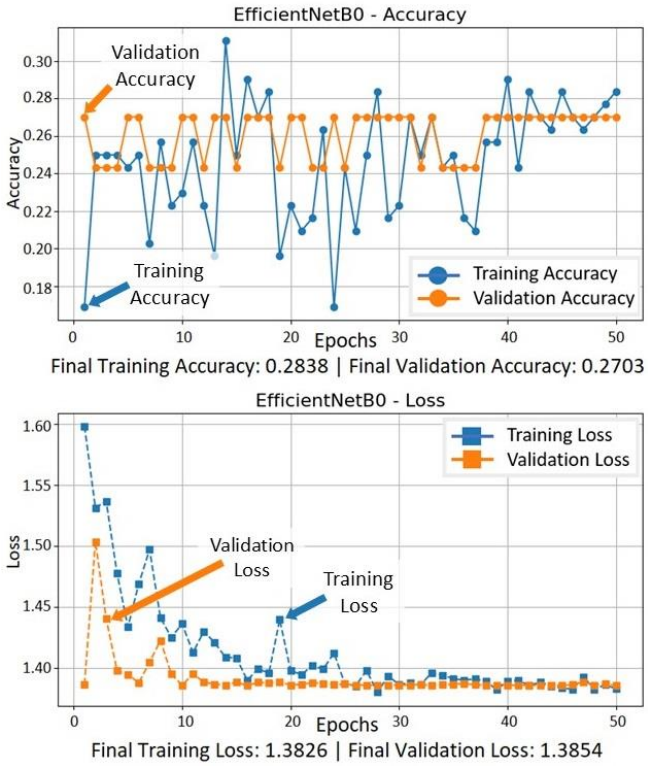

Fig. 8. Training Metrics for EfficientnetB0 CNN Model.

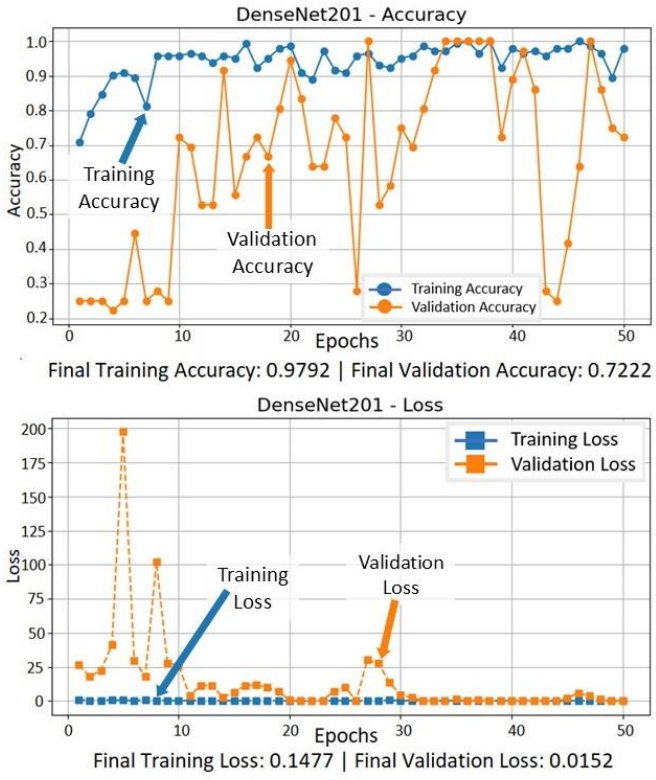

Fig. 9. Training Metrics for DenseNet201 CNN Model.

The performance analysis plot of EfficientNetB0 CNN model illustrated in Fig. 8 shows how it struggles in achieving a training accuracy of 28.38% and a validation accuracy of 27.03%. It records high training and validation losses of 1.3826 and 1.3854, respectively, indicating inefficiency for this task without substantial modifications.

The DenseNet201 CNN model Fig. 9 achieved a high training accuracy of 97.92% but suffered a notable drop in validation accuracy to 72.22%. Despite its low training loss of 0.1477 and validation loss of 0.0152, the significant gap between training and validation metrics reveals overfitting, requiring further tuning to improve its generalization capabilities.

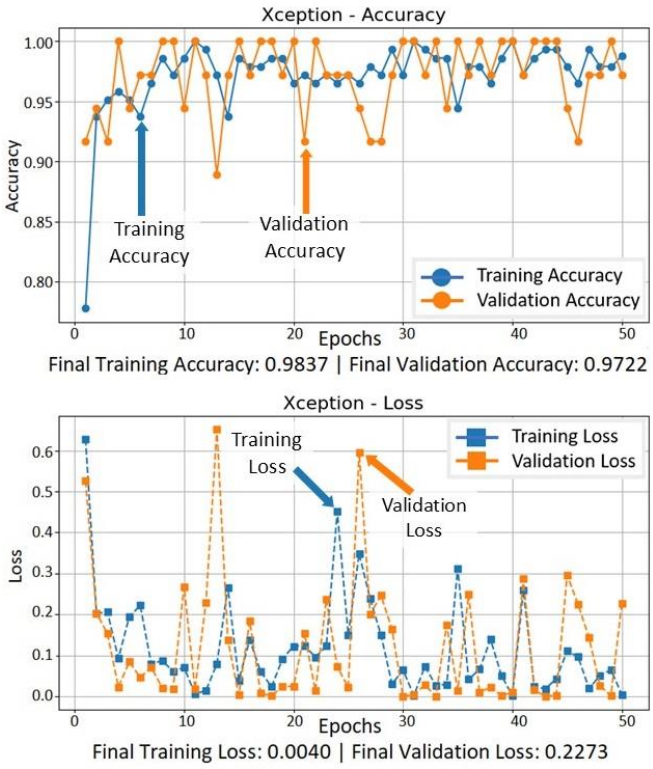

Fig. 10. Training Metrics for Xception CNN Model.

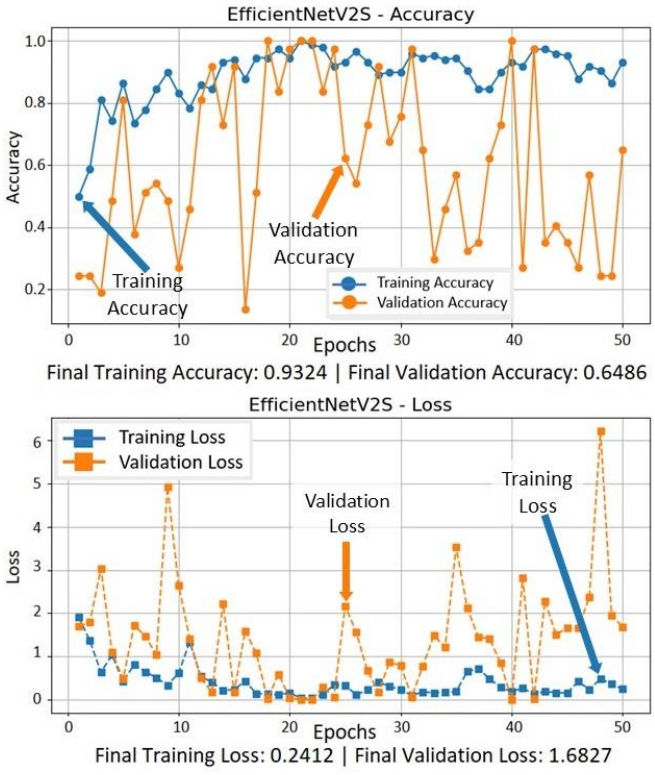

Fig. 11. Training Metrics for EfficientNetV2S CNN Model.

The performance analysis plot of Xception CNN model illustrated in Fig. 10 shows delivering strong performance, with a training accuracy of 98.37% and a validation accuracy of 97.22%. It maintains a training loss of 0.0040 and a validation loss of 0.2273, performing well while requiring further optimization to reduce the validation loss.

The performance analysis plot of EfficientNetV2S CNN model illustrated in Fig. 11 has recorded a training accuracy of 93.24%, while its validation accuracy reached 64.86%. The training loss stood at 0.2412, whereas the validation loss increased to 1.6827. This discrepancy between training and validation metrics indicates overfitting, suggesting that the model requires adjustments to perform better on unseen data.

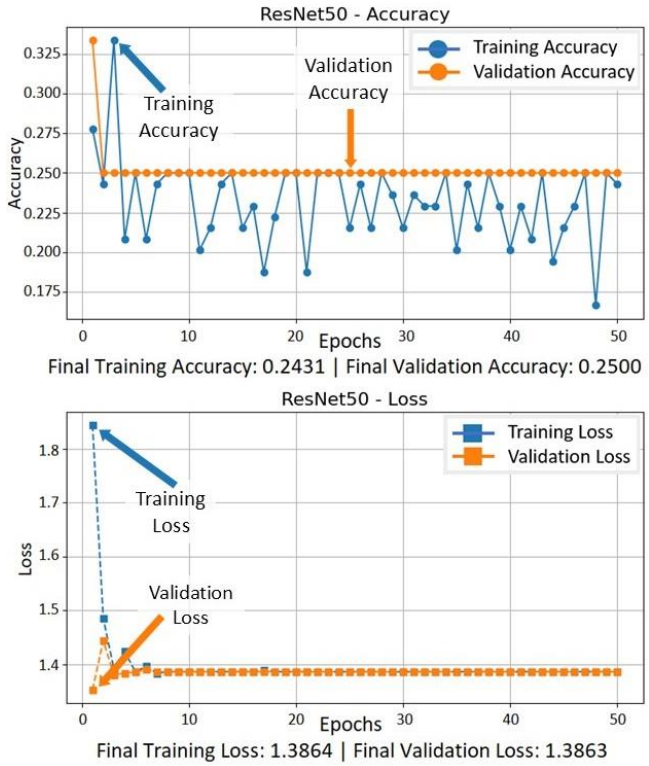

Fig. 12. Training Metrics for ResNet50 CNN Model.

The performance analysis plot of the ResNet50 CNN model, illustrated in Fig. 12 demonstrates limited performance, with a training accuracy of 24.31% and a validation accuracy of 25.00%, along with high loss values of 1.3854 and 1.3853 for training and validation, respectively. This indicates that ResNet50 is not well-suited for the task without substantial architectural adjustments or hyperparameter tuning.

Among the evaluated CNN models, MobileNetV2 demonstrates exceptional efficiency for real-time face recognition, achieving a validation accuracy of 98.01%. This performance is attributed to its lightweight architecture and depth-wise separable convolutions, which optimize computational resources by reducing memory usage and providing faster inference without compromising accuracy. These features make MobileNetV2 particularly suitable for deployment on resource-constrained platforms, ensuring robust performance even under limited hardware capabilities.

Furthermore, the model effectively balances accuracy, speed, and efficiency, making it ideal for real-time applications. Its strong generalization capabilities allow it to deliver consistent results across diverse practical scenarios, including varying lighting and environmental conditions. By achieving an optimal trade-off between computational efficiency and accuracy, MobileNetV2 sets a high standard for performance and resource optimization in face recognition tasks.

TABLE VI
MODEL CLASSIFICATION REPORT SUMMARY

Model	Final Accuracy %	Precision %	Recall %	F1-Score %
Basic CNN	0.97	0.94	0.97	0.95
MobileNetV2	**0.98**	**0.98**	**0.99**	**0.98**
EfficientnetB0	0.27	0.28	0.26	0.27
DenseNet201	0.72	0.72	0.73	0.72
Xception	0.97	0.97	0.97	0.96
EfficientNetV2S	0.64	0.65	0.66	0.65
ResNet50	0.25	0.25	0.25	0.25

C. *Classification Report Analysis*

Various key performance metrics such as Accuracy, Precision, Recall, and F1-Score of classification report analysis provides insights into different aspects of the models' effectiveness in correctly identifying positive instances while minimizing errors. Precision, Recall, F1-Score are clearly described as follows in TABLE VI:

1) **Precision**: Focuses on the accuracy of positive predictions and helps reduce false positives. High precision indicates that when the model predicts a positive instance, it is likely correct.

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)+False Positives (FP)}} \quad (18)$$

2) **Recall**: Measures the model's ability to identify all positive instances, reducing false negatives. High recall ensures that most positive instances are correctly detected by the model.

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)+False Negatives (FN)}} \quad (19)$$

3) **F1-Score**: Balances precision and recall, offering an overall view of the model's performance in handling positive predictions accurately. This metric is particularly useful when there is a need to balance precision and recall.

$$\text{F1} - \text{Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision+Recall}} \quad (20)$$

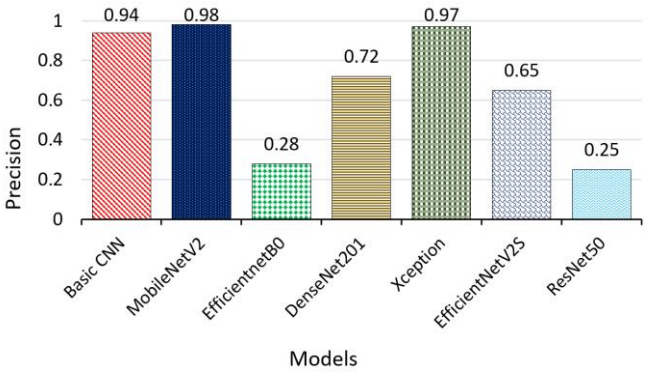

Fig. 13. Comparative Precision Metrics

Fig.13 illustrates the precision of each model, which represents their ability to accurately identify true positives. MobileNetV2 and Xception demonstrate the highest

precision values at 0.98 and 0.97, respectively, indicating their effectiveness in minimizing false positives. Basic CNN achieves a solid precision score of 0.94, while DenseNet201 and EfficientNetV2S perform moderately with scores of 0.72 and 0.65, respectively. In contrast, EfficientNetB0 and ResNet50 exhibit the lowest precision values, at 0.28 and 0.25, respectively, highlighting challenges in accurately handling predictions.

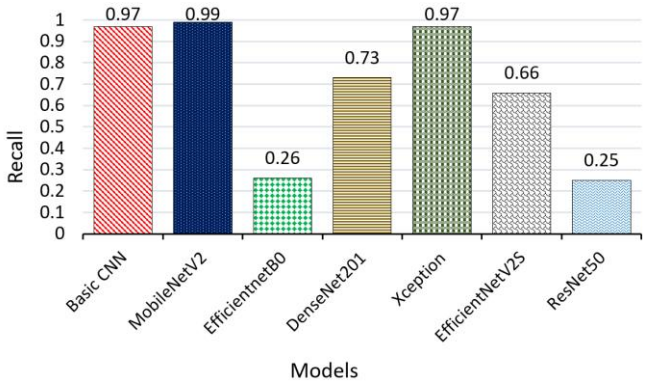

Fig. 14. Comparative Recall Metrics

Fig.14 illustrates the recall performance of each model, emphasizing their effectiveness in identifying relevant instances. MobileNetV2 leads with a recall of 0.99, followed by Basic CNN and Xception, each scoring 0.97. DenseNet201 achieves a recall of 0.73, while EfficientNetV2S records 0.66, indicating moderate detection capabilities. In contrast, EfficientNetB0 and ResNet50 exhibit the lowest recall values, at 0.26 and 0.25, respectively, indicating difficulties in capturing all true positives.

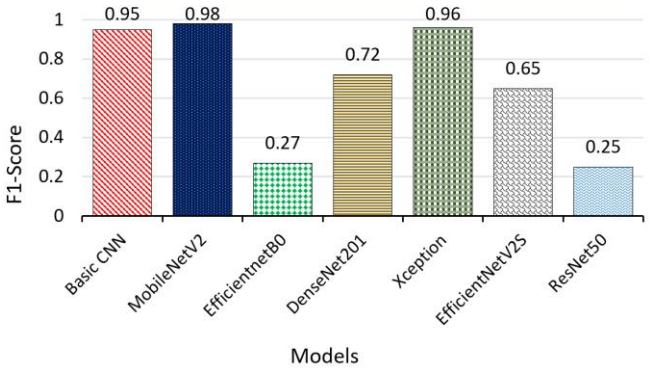

Fig. 15. Comparative F1-Score Metrics

Fig. 15 depicts the F1-scores of the models, reflecting a balance between precision and recall. MobileNetV2 achieves the highest F1-score of 0.98, followed by Xception 0.96 and Basic CNN 0.95, showcasing their consistent performance. DenseNet201 and EfficientNetV2S display moderate F1-scores of 0.72 and 0.65, respectively. EfficientNetB0 0.27 and ResNet50 0.25 rank the lowest, highlighting their overall weaker performance in achieving a balance between precision and recall.

In contrast, EfficientNetB0 and ResNet50 exhibit the lowest F1-scores, underscoring significant challenges in achieving balanced predictive performance. Based on the comparative analysis across Figs. 13 to 15, it is evident that MobileNetV2 CNN consistently outperforms other models across all metrics, making it an excellent choice for applications demanding high accuracy and reliability. These figures provide a holistic assessment of the models' strengths and weaknesses. While MobileNetV2, Xception, and Basic CNN showcase superior capabilities, EfficientNetB0 and ResNet50 emphasize the need for further optimization or alternative methodologies to enhance their effectiveness.

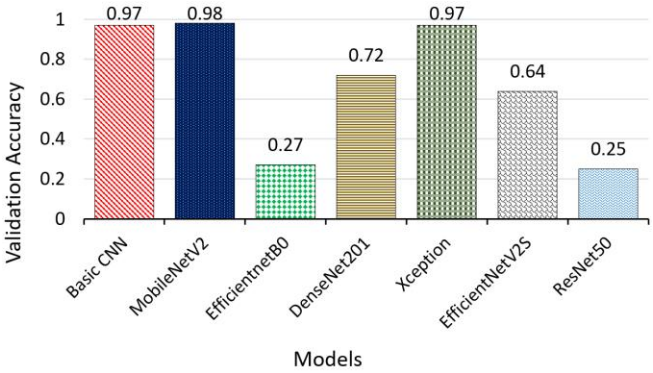

Fig. 16. Validation Accuracy Trends Across Models

D. Comparative Performance Across Models

Fig. 16 presents the validation accuracy trends for all models over 50 epochs. MobileNetV2 consistently achieved rapid convergence with minimal loss fluctuations, underscoring its computational efficiency and effectiveness in feature extraction. The model's lightweight architecture and depth-wise separable convolutions enabled superior performance while maintaining low computational overhead.

This makes MobileNetV2 particularly suitable for real-time applications in resource-constrained environments, such as libraries. Conversely, ResNet50 exhibited stagnation in accuracy, highlighting its inefficiency for this dataset. The model's high complexity and parameter overhead likely contributed to suboptimal performance, particularly given the limited size and diversity of the dataset.

Xception and Basic CNN also demonstrated reliable performance, achieving F1 scores of **0.96** and **0.95**, respectively. While both models showed consistent validation accuracy, their slightly lower recall values suggest occasional challenges in identifying all relevant instances. This limitation may arise from sensitivity to variations in pose or lighting. The trends underscore the robustness and versatility of MobileNetV2, outperforming more complex models while achieving a validation accuracy of **98.01%**, making it the most effective choice for this application.

Step-3: Face Recognition

The system uses a CNN-based facial recognition module built on the MobileNetV2 architecture to deliver high accuracy and robustness. This module achieves 95% accuracy in identifying individuals across various angles and lighting conditions. The high precision ensures reliable and consistent user authentication.

Fig. 17 demonstrates the system's high confidence in recognizing faces with 95% accuracy across diverse conditions, highlighting the system's ability to extract and classify facial features effectively. The heatmap of class probabilities showcases the model's confidence levels. For example, the system identifies the user Sri as CLSRI8308 with a probability of 1, while it assigns probabilities of 0.00 to other users, such as CLBALA7917, CLSIVA9713, and CLRAJA6706. The x-axis represents class names, and the y-axis indicates probabilities, with darker blue shading emphasizing higher confidence levels.

This system employs CNN's feature extraction and

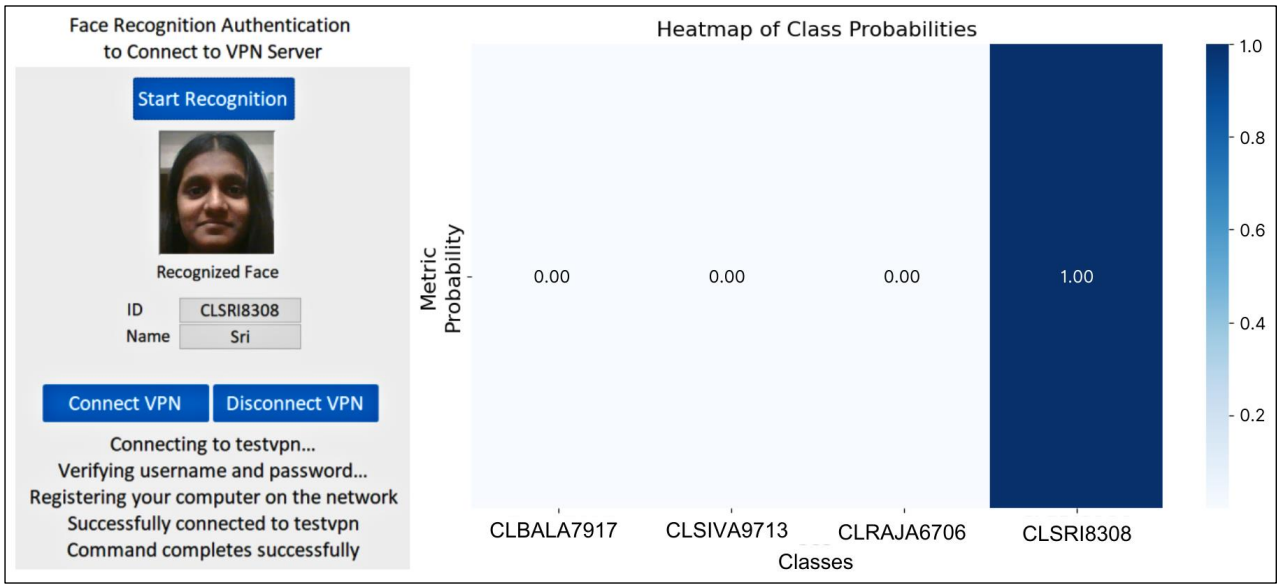

Fig. 17. Face Recognition and Connect VPN Server with heatmap of class probabilities

classification capabilities, which significantly reduce the risk of unauthorized access compared to traditional authentication methods. The face recognition module provides an intuitive, user-friendly interface that enables users to authenticate seamlessly and receive real-time feedback on the process.

The system's adaptability ensures consistent performance across various environments, handling dynamic lighting and angles effectively. MobileNetV2's lightweight architecture optimizes computational efficiency, enabling deployment on resource-constrained devices. Pre-processing techniques like normalization and resizing enhance robustness against image quality variations. This design minimizes latency, providing a seamless and efficient user experience.

Step 4: SQL Verification and VPN Server Integration

The system secures user authentication by combining face recognition with SQL credential verification and VPN integration. After successful face recognition, the system sends the recognized user ID to the SQL server, where it matches the ID against encrypted VPN login credentials stored in the database. Sensitive information, including usernames and passwords, is encrypted using AES-256 encryption with a 32-byte key and decrypted only upon successful face recognition. This approach prevents brute-force attacks and ensures data confidentiality during storage and transmission. The system uses Windows Server with Active Directory (AD) to centrally manage VPN users.

Fig.18 displays the Active Directory Users and Computers dashboard, where users are organized in a security group for streamlined access control. This integration simplifies user management and ensures consistent access policies.

Fig.19 highlights the VPN User Management interface, where the SQL database securely stores encrypted credentials. This design protects sensitive information and

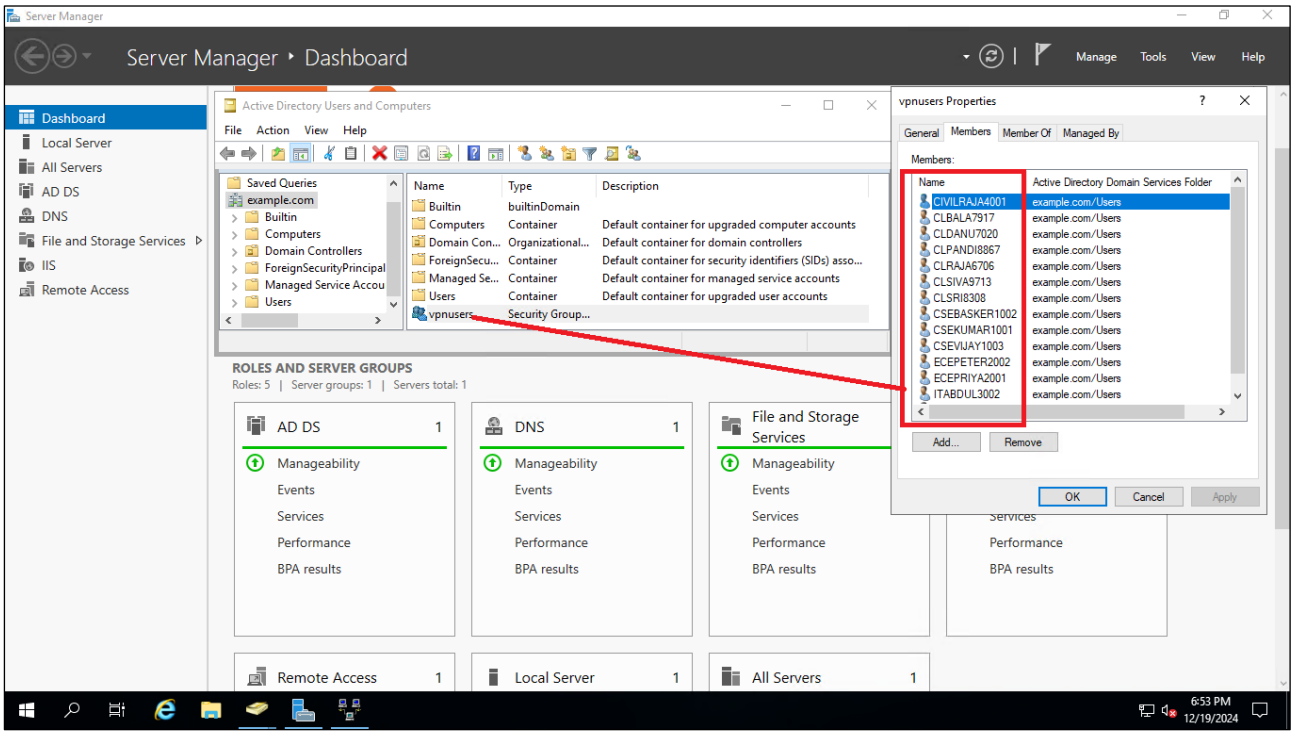

Fig. 18. Active Directory Users and Computers Displaying VPN User Security Group.

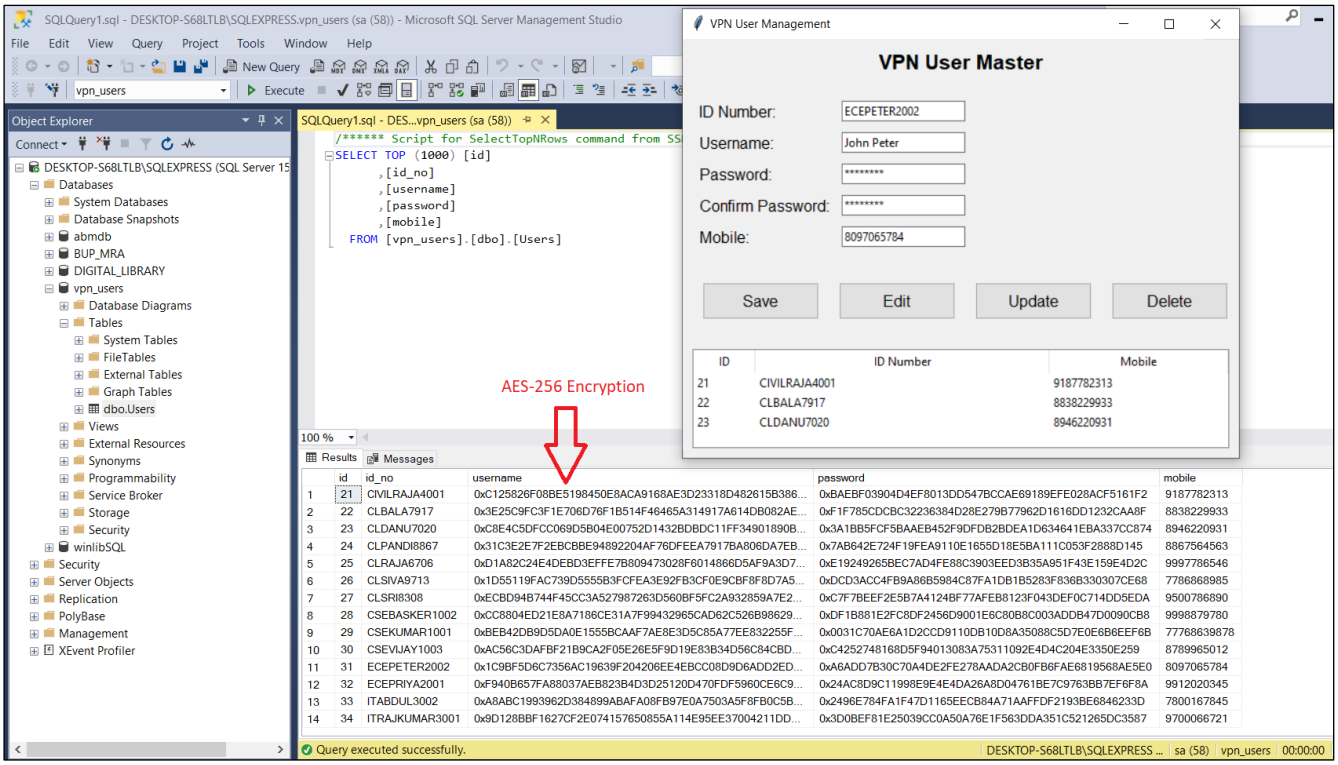

Fig. 19. VPN User Management Interface and AES-256 Encrypted Credentials in SQL Database.

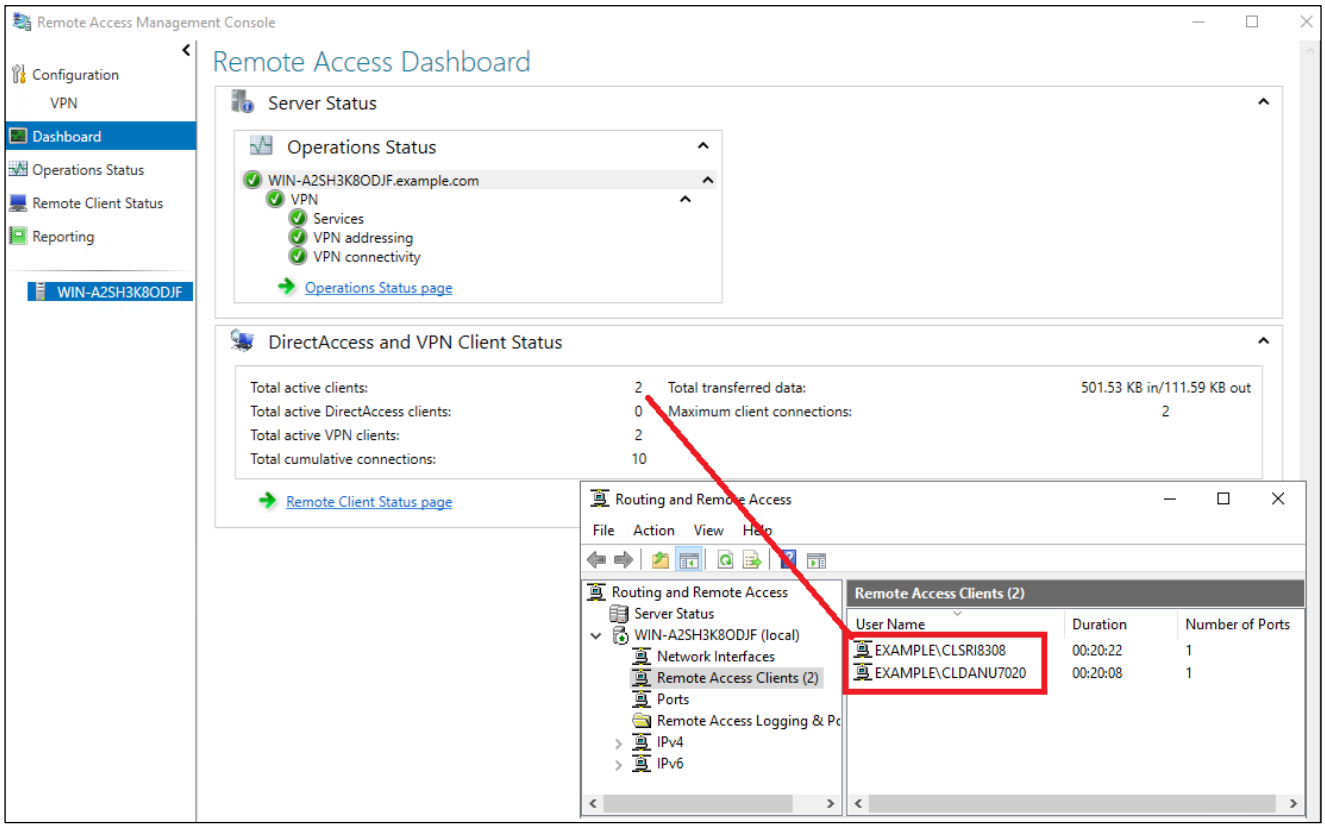

Fig. 20. Remote Access Dashboard Showing Successful VPN Connections.

automates the credential verification process for enhanced security. The VPN connection employs Routing and Remote Access Service (RRAS) on Windows Server, enabling secure remote access.

Fig.20 illustrates the Remote Access Management Console, detailing active VPN clients, connection statuses, and transferred data. The system supports multiple simultaneous connections using the PPTP protocol with MS-CHAPv2 authentication, ensuring secure communication.

Fig.21 shows the Remote Access Client Status, displaying protocol types, IP addresses, and connection durations for each connected client. The robust configuration guarantees stable connections under diverse network conditions. Fig.22 depicts the centralized interface for accessing e-resources

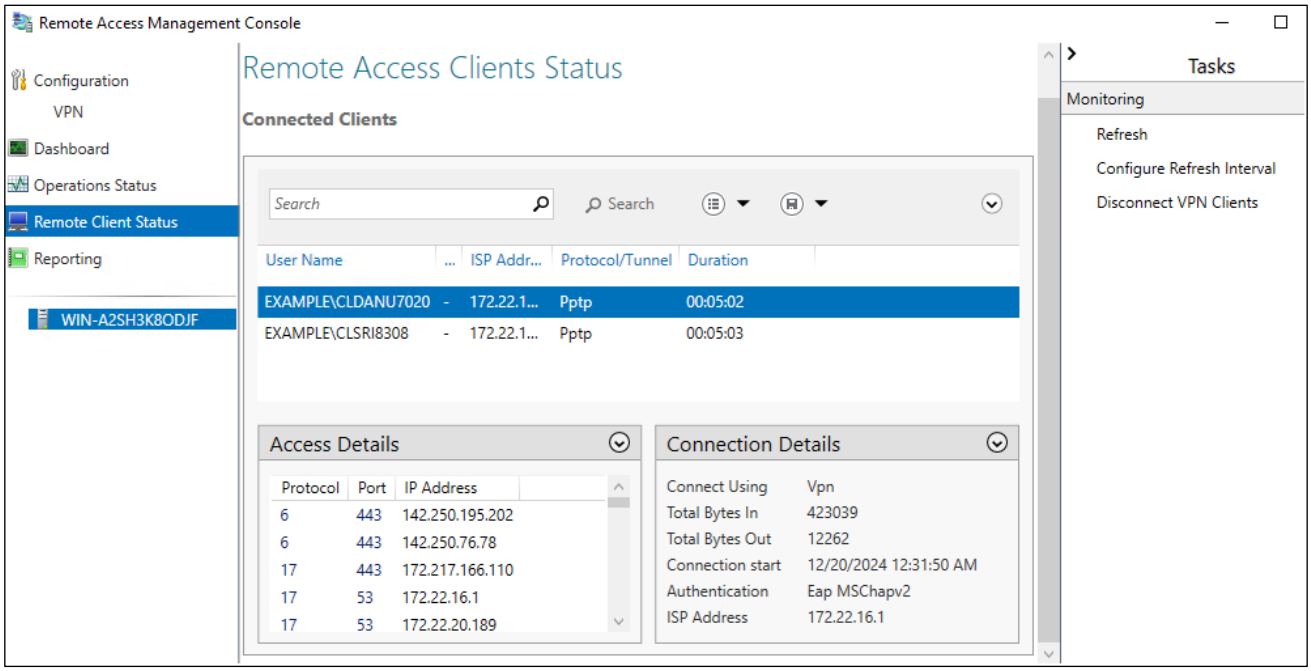

Fig. 21. Remote Access Clients Status with Connection Details.

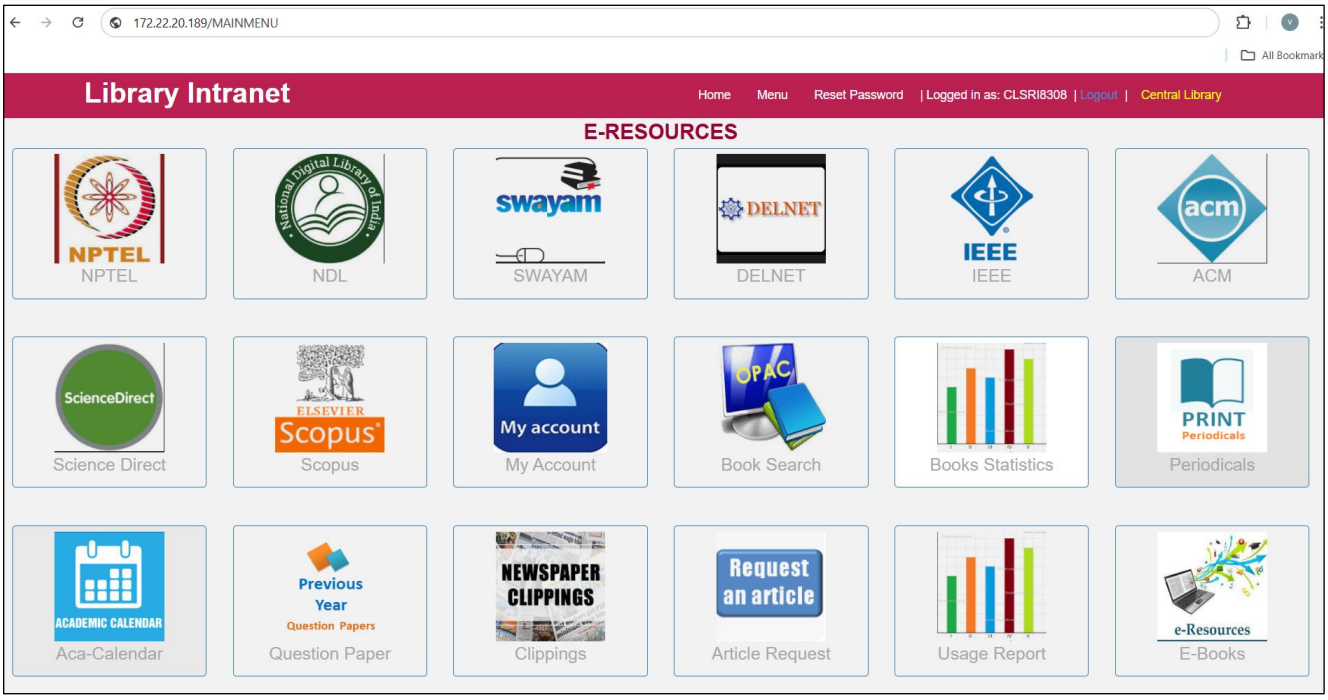

Fig. 22. Library Intranet Interface showcasing access to subscribed e-resources and library repositories through a secure VPN connection.

securely through the VPN. The interface integrates platforms like IEEE, ACM, ScienceDirect, Scopus, OPAC, and Book Statistics, providing seamless navigation for both local and cloud-hosted resource. This design enhances research and academic workflows by ensuring uninterrupted access to essential tools.

This multi-layered system combines biometric authentication, encrypted credential storage, and secure VPN integration to deliver comprehensive access control. By leveraging technologies such as Windows Server, RRAS, and AES encryption, it ensures the confidentiality, integrity, and availability of organizational resources. The seamless interaction between these components provides a secure, efficient, and user-friendly solution for remote access to subscribed platforms and library repositories.

V. Future Work

The face recognition system for VPN authentication faces several challenges, including sensitivity to environmental factors such as lighting and camera quality, which can impact accuracy and necessitate optimal conditions for reliable results. Additionally, the CNN model requires ongoing training and maintenance to adapt to new faces and ensure continued effectiveness, highlighting the need for regular updates. Scalability is another concern, as the system may struggle to maintain performance and security with a growing user base, requiring architectural optimizations. To overcome these challenges, future research will aim to improve model accuracy by broadening the scope of diverse datasets, utilizing advanced techniques such as data augmentation,

transfer learning, and ensemble methods, and investigating cutting-edge neural network architectures, including Transformer-based models. Furthermore, integrating the system into cloud environments will enhance scalability and security, leveraging distributed computing resources and implementing robust security measures for reliable authentication in larger deployments.

VI. Conclusion

In this research study, we have developed a face recognition-based VPN authentication framework that leverages pre-trained CNNs to enhance the security and efficiency of network access for library resource management. Using face recognition technology in VPN authentication resolves issues such as password security and ease of use, providing a more secure and user-friendly solution. The fine-tuned MobileNetV2 CNN model, trained with a varied dataset of 15,000 facial images, achieved an impressive perfect validation accuracy of 98.01% and a validation loss of 0.0621, highlighting its proficiency in accurately identifying authorized users. The proposed system fortifies security by utilizing biometric data that is unique and difficult to replicate while simplifying the authentication process, thereby improving the overall user experience. By eliminating the reliance on passwords or tokens, our approach mitigates common security risks associated with these conventional methods and reduces user complexity. The use of mirrored RAID 1 ensures data redundancy and reliability, while VPN access facilitates secure resource utilization. This seamless integration of advanced AI-based face recognition technology into VPN systems represents a significant advancement in cybersecurity, setting a new standard for secure network access. Future efforts will concentrate on enhancing the face recognition model, incorporating additional biometric features, and exploring real-world deployment scenarios to assess the system's scalability and adaptability. Additionally, expanding the dataset and employing more sophisticated machine learning techniques could further enhance the system's performance and robustness. This research contributes to the growing field of secure network authentication, offering a practical and innovative solution for modern cybersecurity challenges. As face recognition technology advances, it has the potential to transform access control mechanisms across different sectors, leading to more secure and efficient systems.

References

[1] Alomran, A. I., and Basha, I., "An AI-Based Classification and Recommendation System for Digital Libraries," Scalable Computing: Practice and Experience, vol. 25, no. 4, pp. 3181-3199, 2024.

[2] Alrawili, R., AlQahtani, A. A. S., and Khan, M. K., "Comprehensive survey: Biometric user authentication application, evaluation, and discussion," Computers and Electrical Engineering, vol. 119, no. Part-A, pp. 109485, 2024.

[3] Ambikapathy, A., Beeta, T. D., Kanna, R. K., Danquah-Amoah, A., Ramya, V. S., and Mutheeswaran, U., "Biometric Application on Facial Image Recognition Techniques," in *2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT)*, vol. 5, pp. 848-851, Feb. 2024, IEEE.

[4] Bakariya, B., Singh, A., Singh, H., Raju, P., Rajpoot, R., and Mohbey, K. K., "Facial emotion recognition and music recommendation system using CNN-based deep learning techniques," Evolving Systems, vol. 15, no. 2, pp. 641-658, 2024.

[5] Balachandran, S., Dominic, J., and Sivankalai, S., "A Comparative Analysis of VPN and Proxy Protocols in Library Network Management," Library Progress International, vol. 44, no. 3, pp. 17006-17020, 2024.

[6] Balachandran, S., and Dominic, J., "Pioneering a Prototype VPN-Based Cloud Strategy for Streamlined Library Management," Library Philosophy & Practice, pp. 7949, 2023.

[7] Prabhu Bevinamarad, and Prakash H. Unki, "Digital Image Authentication and Analysis: Unmasking Copy-move Forgery in Digital Images through Combined DCT and GLCM Features with Block Matching Technique," IAENG International Journal of Computer Science, vol. 51, no.11, pp1672-1685, 2024

[8] Crihan, G., Dumitriu, L., and Crăciun, M. V., "Preliminary Experiments of a Real-World Authentication Mechanism Based on Facial Recognition and Fully Homomorphic Encryption," Applied Sciences, vol. 14, no. 2, pp. 718, 2024.

[9] Dablain, D., Jacobson, K. N., Bellinger, C., Roberts, M., and Chawla, N. V., "Understanding CNN fragility when learning with imbalanced data," Machine Learning, vol. 113, no. 7, pp. 4785-4810, 2024.

[10] Diez-Tomillo, J., Martinez-Alpiste, I., Golcarenarenji, G., Wang, Q., and Alcaraz-Calero, J. M., "Efficient CNN-based low-resolution facial detection from UAVs," Neural Computing and Applications, vol. 36, no. 11, pp. 5847-5860, 2024.

[11] Hemalatha, P., Shankar, G., and Deepak Raj, D. M., "A New Improved Binary Convolutional Model for Classification of Images," Scalable Computing: Practice and Experience, vol. 23, no. 4, pp. 263–274, 2022.

[12] Jones, N. L., "Fast annual daylighting simulation and high dynamic range image processing using NumPy," Science and Technology for the Built Environment, vol. 30, no. 4, pp. 327-340, 2024.

[13] Jyosthna, P. M., Mandapati, A. V., Teja, M. S., Ray, S. K., and Kumar, B. Y. S., "Enhancing Security and Flexibility with Combined RBAC and ABAC Access Control Models," in *2024 10th International Conference on Communication and Signal Processing (ICCSP)*, pp. 576-581, Apr. 2024, IEEE.

[14] Khakim, L., Mukhlisin, M., and Suharjono, A., "Analysis of password after encryption by using the combination of AES256 and MD5 algorithm methods," in *AIP Conference Proceedings*, vol. 3070, no. 1, AIP Publishing, Apr. 2024.

[15] Khalifa, A., Abdelrahman, A. A., Hempel, T., and Al-Hamadi, A., "Towards efficient and robust face recognition through attention-integrated multi-level CNN," Multimedia Tools and Applications, pp. 1-23, 2024.

[16] Khan, H. U., Ali, F., and Nazir, S., "Systematic analysis of software development in cloud computing perceptions," Journal of Software: Evolution and Process, vol. 36, no. 2, pp. e2485, 2024.

[17] Mahdi, R. A. K., and Ilyas, M., "Using deep learning technology to optimize VPN networks based on security performance," Journal of Electrical Systems, vol. 20, no. 4s, pp. 1894-1903, 2024.

[18] Meena, G., Mohbey, K. K., Indian, A., Khan, M. Z., and Kumar, S., "Identifying emotions from facial expressions using a deep convolutional neural network-based approach," Multimedia Tools and Applications, vol. 83, no. 6, pp. 15711-15732, 2024.

[19] Naveed, H., Anwar, S., Hayat, M., Javed, K., and Mian, A., "Survey: Image mixing and deleting for data augmentation," Engineering Applications of Artificial Intelligence, vol. 131, pp. 107791, 2024.

[20] Parenreng, J. M., "Network security analysis based on internet protocol security using virtual private network (VPN)," Internet of Things and Artificial Intelligence Journal, vol. 3, no. 3, pp. 239-249, 2023.

[21] Pillajo, C., Bustos, C., Salazar-Chacón, G., Estévez, A., and Pinto, H., "Energy Consumption Analysis in Networking: IPSec Tunnels Comparison with Different Encryption and Authentication Methods," in *2024 Second International Conference on Emerging Trends in Information Technology and Engineering (ICETITE)*, pp. 1-6, Feb. 2024, IEEE.

[22] Ping Lu, Chongkun Shao, Shan Deng, Jiaying Zeng, and Kaibiao Lin, "IBNNER: A Biaffine Model-Based Chinese Nested Named Entity Recognition Method for Medical Texts," IAENG International Journal of Computer Science, vol. 51, no.11, pp1686-1699, 2024

[23] Ren, B., Kong, W., and Mao, W., "Performance Analysis of IPSec VPN over TCP/UDP under Different Encryption Algorithms," in *Proceedings of the International Conference on Computing, Machine Learning and Data Science*, pp. 1-6, Apr. 2024.

[24] Amattouch Mohamed Ridouan, "An Extended Deep Learning Method for the Navier Equation," IAENG International Journal of Applied Mathematics, vol. 54, no.9, pp1833-1839, 2024

[25] Ru Jiang, Huichuan Duan, Jingyu Yan, and Weikuan Jia, "Green Tomato Segmentation Model Based on Optimized Swin-Unet

Algorithm Under Facility Environments," Engineering Letters, vol. 32, no.11, pp2114-2126, 2024

[26] Safa aldin, S., Aldin, N. B., and Aykac, M., "Enhanced image classification using edge CNN (E-CNN)," The Visual Computer, vol. 40, no. 1, pp. 319-332, 2024.

[27] Sathya, K., Esther, J., Kavitha, S., and Kamalakumari, J., "Facetpass-Intelligent Facial Recognition Authentication System Security and Usability," in *2024 2nd International Conference on Artificial Intelligence and Machine Learning Applications Theme: Healthcare and Internet of Things (AIMLA)*, pp. 1-6, Mar. 2024, IEEE.

[28] Shah, J. J., Ragu, H., David, V., Sasikumar, P., and Subburaj, M., "OpenCV Based Customer Screening System for Prevention of COVID-19 Transmission in Retail Stores," Wireless Personal Communications, vol. 137, no. 2, pp. 685-703, 2024.

[29] Shukla, A. K., Shukla, A., and Singh, R., "Automatic attendance system based on CNN–LSTM and face recognition," International Journal of Information Technology, vol. 16, no. 3, pp. 1293-1301, 2024.

[30] Shukla, R. K., Tiwari, A. K., and Ranjan Mishra, A., "Face Recognition Using LBPH and CNN," Recent Advances in Computer Science and Communications, vol. 17, no. 5, pp. 48-58, 2024.

[31] Tambon, F., Nikanjam, A., An, L., Khomh, F., and Antoniol, G., "Silent bugs in deep learning frameworks: an empirical study of keras and tensorflow," Empirical Software Engineering, vol. 29, no. 1, pp. 10, 2024.

[32] Thalluri, L. N., Babburu, K., Madam, A. K., Kumar, K. V. V., Ganesh, G. V., Rajasekhar, K., and Yaswanth, V. V. N., "Automated face recognition system for smart attendance application using convolutional neural networks," International Journal of Intelligent Robotics and Applications, vol. 8, no. 1, 2024.

[33] Yadav, A., Sridevi, S., Prassana Kumar, R., and Balachandran, S., "Hybrid Quantum-Classical Convolutional Neural Network for Allen Telescope SETI image Classification," in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pp. 1-6, Jul. 2023, IEEE.

[34] Yang, X., Xu, L., Pang, T., Dong, Y., Wang, Y., Su, H., and Zhu, J., "Face3DAdv: Exploiting Robust Adversarial 3D Patches on Physical Face Recognition," International Journal of Computer Vision, pp. 1-19, 2024.

[35] Yeboah-Ofori, A., and Ganiyu, A., "Big Data Security Using RSA Algorithms in A VPN Domain," in *2024 International Conference on Artificial Intelligence, Computer, Data Sciences and Applications (AICDSA)*, pp. 1-6, Feb. 2024, IEEE.

[36] Zhang, H., Yi, Z., Kang, L., Zhang, Y., and Wang, K., "A novel supercapacitor degradation prediction using a 1D convolutional neural network and improved informer model," Protection and Control of Modern Power Systems, vol. 9, no. 4, pp.51-68, 2024.

[37] Zhengnan, X., Guofang, D., and Ruicheng, Y., "RBAC-based one-to-many authentication and key negotiation scheme in smart factory," *IEEE Access*, 2024.