FACE DETECTION AND RECOGNITION USING MTCNN AND FACENET

C. Abhiram
School of Computer Science and Engineering
VIT-AP University
Amaravathi, India
abhiram.22bce8988@vitapstudent.ac.in

G. Sai Shivananda
School of Computer Science and Engineering
VIT-AP University
Amaravathi, India
shivananda.22bce7263@vitapstudent.ac.in

N. V. Sriraj
School of Computer Science and Engineering
VIT-AP University
Amaravathi, India
sriraj.22bce9276@vitapstudent.ac.in

***Abstract*—Face detection and recognition are critical components in modern computer vision applications, including security, surveillance, and authentication systems. This paper presents a robust and efficient face verification system that integrates Multi-task Cascaded Convolutional Networks (MTCNN) for face detection and alignment, with FaceNet for generating discriminative facial embeddings. MTCNN ensures accurate detection and landmark localization even under challenging conditions such as occlusions and poor lighting. FaceNet maps facial images into a high-dimensional embedding space using triplet loss, enabling precise identity verification through cosine similarity. The proposed system is evaluated on a curated dataset of football players, achieving an accuracy of 81.25% with a perfect precision score for positive identifications. The system's performance is further analyzed using confusion matrix, ROC curve, and classification metrics. This study demonstrates the effectiveness of deep learning-based facial recognition and highlights the importance of embedding-based similarity measures in enhancing verification accuracy. Ethical considerations such as privacy, bias, and secure deployment are also discussed, emphasizing the need for responsible implementation.**

Keywords—Facial Recognition, MTCNN, FaceNet, Deep Learning, Facial Embeddings

I. INTRODUCTION

The primary aim of this project is to develop a robust and efficient face recognition system capable of identifying individuals with high accuracy under diverse conditions, including varying poses, lighting, facial expressions, and image resolutions. By leveraging advanced deep learning techniques, this system is designed to offer real-time recognition for applications such as security authentication, attendance tracking, and surveillance systems. The integration of cutting-edge frameworks like TensorFlow, Keras, and OpenCV ensures both precision and speed, making the solution suitable for practical deployment.

The objectives of this project include developing a deep learning model that can detect and recognize faces in both real-time video streams and still images. Image preprocessing techniques such as grayscale conversion, histogram equalization, and augmentation will be employed to enhance feature detection and improve model training. The project also emphasizes optimizing the model for real-time use, incorporating secure data storage, and testing performance across a variety of datasets to ensure accuracy, speed, and robustness under real-world conditions.

This face recognition system encompasses the entire development cycle—from data collection and preprocessing to model training and deployment. It covers advanced concepts such as facial alignment, feature embedding generation, and model evaluation using multiple metrics. The system is built to handle real-time video as well as static images, addressing challenges such as occlusions (e.g., masks or glasses) and varying light conditions. Additionally, the project explores the use of encryption and secure facial data management to maintain privacy and data integrity.

The motivation for this project arises from the growing demand for contactless and secure identity verification solutions. Traditional methods like PINs and passwords are increasingly vulnerable to breaches. Facial recognition presents a more secure, automated, and user-friendly alternative. With advancements in AI and deep learning, it has become feasible to develop systems that offer high accuracy even in challenging environments. This technology holds transformative potential across industries such as law enforcement, smart homes, healthcare, and education, particularly in attendance tracking and access control.

Facial detection and recognition, as a subfield of computer vision, have seen significant advancements through the use of deep learning. Detection involves locating faces within images or videos, while recognition focuses on identifying individuals based on extracted features. Convolutional Neural Networks (CNNs) and models like YOLO, SSD, and Faster R-CNN have enabled real-time, high-accuracy face detection. On the recognition side, models such as FaceNet, VGG-Face, and ArcFace produce unique embeddings to distinguish between individuals, making them suitable for both verification and identification tasks.

Key features of the system include:

- Real-time Facial Detection and Recognition: The system identifies and verifies faces instantly from live video streams and static images.
- Deep Learning-Based Feature Extraction: Utilizes Convolutional Neural Networks (CNNs) to extract deep facial features for accurate recognition.
- Integration with OpenCV: Ensures seamless video capture, face detection, and image processing functionalities.
- High Accuracy Across Conditions: The model is tested under different lighting, angles, and facial expressions to ensure consistent performance.

- Scalability and Flexibility: The system architecture allows easy integration into larger applications such as attendance systems, security solutions, or IoT-based platforms.

One of the key technologies highlighted in this project is the Multi-task Cascaded Convolutional Network (MTCNN), which offers robust and efficient facial detection and alignment through a three-stage architecture: Proposal Network (P-Net), Refine Network (R-Net), and Output Network (O-Net). This cascade approach allows for accurate bounding box prediction and facial landmark localization, even under complex conditions. By incorporating such deep learning-based techniques, the system ensures precise detection and recognition, laying the foundation for reliable deployment in real-world applications.

The following is how this study is structured: In Section 2, related investigations are discussed and the body of existing literature is reviewed. The methodology used in this study is described in Section 3 and the experimental framework is explained in Section 4. A detailed discussion of the findings is provided in Section 5, which also emphasizes the results. Section 6 wraps up the study by highlighting the main conclusions and ramifications.

II. LITERATURE SURVEY

Facial Recognition (FR) technology has undergone remarkable evolution over the last two decades, primarily due to advancements in deep learning, the availability of large-scale annotated datasets, and increased computational power through GPUs and cloud services. Early approaches relied heavily on handcrafted features and statistical techniques. Classical methods like Eigenfaces and Fisherfaces worked on linear projections of facial images but performed poorly when exposed to real-world variations such as changes in lighting, pose, occlusion, and facial expression. These limitations made it difficult to generalize performance across diverse user populations and environmental conditions [1].

With the rise of Convolutional Neural Networks (CNNs), face recognition systems saw significant improvements. CNNs are capable of learning hierarchical, abstract features from pixel-level data without manual intervention. These models drastically outperformed traditional methods by leveraging deep feature maps that capture structural and textural facial properties. Modern systems can now recognize faces across large variations in lighting, pose, and background clutter, making them suitable for unconstrained environments. Further enhancement in recognition capabilities has been achieved through hybrid approaches, where deep learning is combined with synthetic data generation techniques like Generative Adversarial Networks (GANs). These methods allow augmentation of training datasets with synthetic images to simulate real-world scenarios such as occlusion, blur, or facial aging. Abdul-Al et al. proposed a multi-modal framework integrating CNNs with Principal Component Analysis (PCA) and sequential models to improve generalization in varied conditions [2]. However, challenges such as adversarial robustness, bias against minority demographic groups, and privacy violations still pose ethical and technical barriers.

One of the most pressing issues in recent times is face recognition under occlusion, particularly due to the widespread use of face masks. Malakar et al. addressed this by selectively occluding the lower part of the face during training while preserving the upper half. They used CNN-based classification along with Speeded-Up Robust Features (SURF), achieving a 4–6% improvement in recognition accuracy on benchmark datasets like LFW and CASIA-WebFace [3]. Similarly, Chen et al. proposed a novel approach using super-resolution and frequency-domain feature extraction to recognize masked faces. Their hybrid framework combining spatial and frequency components yielded up to 99% accuracy on RMFRD, a masked face dataset, showing the efficacy of frequency cues in occluded scenarios [10].

In the domain of real-time surveillance, speed and resource efficiency are key. Techniques involving frame down-sampling, score accumulation, and face tracking have been developed to reduce computational overhead while maintaining accuracy. A study focusing on criminal face detection in wild video environments demonstrated how these approaches can balance precision and throughput, although reliance on proprietary datasets limits reproducibility [4]. In another development, CNN-based models were deployed on Jetson Nano, achieving 86.3% accuracy at 6.5 FPS, a milestone for edge computing. These implementations validate the feasibility of deep models on resource-constrained devices for real-time applications [7]. In poorly lit environments, traditional FR systems often fail. Liu et al. tackled this by developing a 3D Morphable Model (3DMM) that integrates Multi-Scale Retinex-based lighting compensation. Their model significantly improved face verification accuracy under extreme lighting conditions, showcasing the value of environment-aware pre-processing [8]. For detecting small or distant faces—commonly found in surveillance footage—Li et al. proposed an enhanced version of MTCNN called RFE-MTCNN, which integrates Inception-V2 and Receptive Field Blocks. This architecture demonstrated robust performance in recognizing small facial regions in complex and cluttered scenes, especially for edge-based deployment [9].

While many systems focus solely on recognition accuracy, there's growing interest in contactless social interaction applications. Valverde et al. introduced a real-time, edge-oriented social distance monitoring system using MTCNN. While their model excelled at accurate distance estimation, it did not handle identity recognition, opening avenues for integrating biometric capabilities in similar systems [11]. Putro et al., on the other hand, designed a lightweight CNN that runs in real-time using only CPU resources. By avoiding GPU dependency, their solution broadens the scope of face detection for practical deployment on mobile or embedded systems [13].

Low-resolution images, especially those extracted from surveillance footage or old photographs, present another major challenge. Li et al. and Horng et al. tackled this issue by incorporating upsampling layers and resolution-aware convolutional kernels into their CNN architectures. Their models performed exceptionally well on datasets such as AR, GT, and LFW, validating the importance of resolution normalization for feature extraction in small face scenarios [12][16]. Additionally, Abdelmaksoud et al. focused on the Single-Sample-Per-Person (SSPP) challenge by combining 3D reconstruction, Super-Resolution GAN (SRGAN), and DeblurGAN to enhance and augment limited training samples. While computationally intensive, their model significantly

improved recognition accuracy for underrepresented individuals [18]. Occlusion and pose variation continue to be among the most difficult problems in FR. Yang et al. proposed Faceness-Net, a deep learning model that uses facial part responses (e.g., eyes, mouth) and their spatial configurations to detect occluded or partially visible faces. The model achieved state-of-the-art performance across several public benchmarks and was particularly strong in real-time applications with partial face visibility [19]. Complementing this, Khan et al. introduced a unified detection and recognition framework based on Faster R-CNN optimized for edge computing in smart classrooms. Their approach demonstrated how smart surveillance and attendance systems can benefit from embedded AI [17].

Sooch and Anand explored the synergy between emotion recognition and facial keypoint detection, proposing a shared learning mechanism to simultaneously improve both tasks. Although not directly targeting face verification, their approach highlighted the benefit of multi-task learning in improving the robustness of shared feature representations [14]. Recent trends are also moving towards Transformer-based architectures, particularly in models like InsightFace and YOLO variants. These models have shown strong potential in balancing speed and accuracy for identity recognition tasks. Anusudha's work on integrating YOLO with InsightFace has led to a real-time face recognition pipeline that performs well in uncontrolled environments such as public places or crowded indoor venues [20].

Despite these technological breakthroughs, key limitations remain. The generalization gap—i.e., performance drop when moving from benchmark datasets to real-world environments—persists due to domain shifts, lighting variability, motion blur, and demographic imbalances. Bias in training datasets leads to disparities in recognition performance across age, race, and gender groups. This has serious implications for deployment in sensitive domains like law enforcement, border control, and healthcare. Moreover, ethical and privacy concerns must be carefully considered. Unregulated facial recognition systems raise red flags about surveillance overreach, data misuse, and lack of user consent. Future research must include fairness-aware learning, federated training, and differential privacy to ensure that systems respect user rights while maintaining performance.

This reveals a clear progression from traditional handcrafted techniques to highly intelligent, deep learning-powered facial recognition systems. From real-time deployment on edge devices to the incorporation of frequency domain features and hybrid models, the field is advancing rapidly. However, challenges such as dataset bias, occlusion handling, pose invariance, and ethical compliance continue to demand attention. Emerging approaches that combine transfer learning, domain adaptation, and model compression are expected to bridge the gap between research and real-world deployment. Ethical AI frameworks must also evolve in parallel to ensure that facial recognition systems are not only accurate and fast but also secure, fair, and transparent.

III. Methodology

This study employs a robust face verification pipeline integrating MTCNN for face detection and FaceNet for feature embedding. Initially, the dataset comprising image pairs is prepared, ensuring each pair represents either the same or different individuals. MTCNN is used to detect and align faces within each image by cropping and resizing the detected facial regions. These preprocessed face images are then passed through the FaceNet model to generate 128-dimensional embeddings that encapsulate the unique facial features. Cosine similarity is calculated between the embeddings of image pairs to assess their likeness. A predefined threshold determines whether two faces are similar or not. Finally, the model's accuracy is evaluated by comparing the verification results against the ground truth labels, providing insights into the effectiveness of the face verification system.

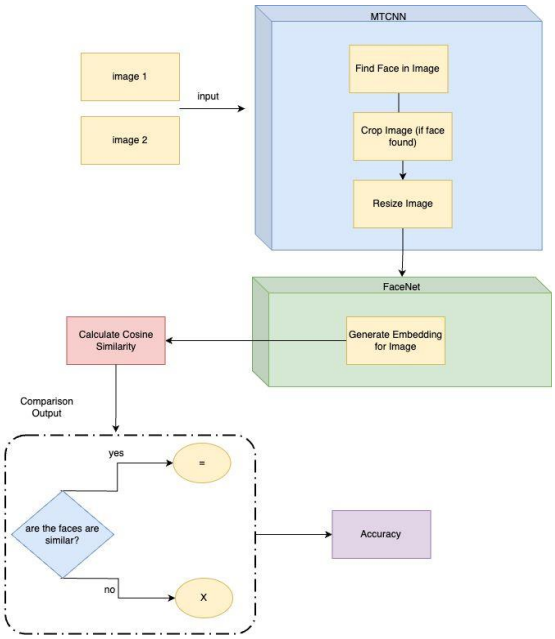

Fig 1: Proposed Model Architecture

A. Dataset Description

The study uses a dataset consists of directories for both training and validation data, under the "Football players" folder, taken from Kaggle. +Within these directories, there are subdirectories corresponding to different football players: Haaland, Mbappé, Messi, Neymar, and Ronaldo. These subdirectories likely contain images of each player, organized for training and validation tasks.

The dataset is structured as follows:

- The "train" folder contains subdirectories for each player, with training images of Haaland, Mbappé, Messi, Neymar, and Ronaldo.
- The "valid" folder mirrors the "train" folder, containing the same subdirectories for each player, but with images designated for validation purposes.

This structure allows for easy access to player-specific images for the task of image classification or face verification. The images within each player's subdirectory are likely used to train and validate models to recognize and verify faces of these specific football players.

B. Face Detection using MTCNN

Face detection is a critical step in face recognition systems, as it involves locating faces within images before any further analysis, such as feature extraction or verification, can take

place. In this methodology, the Multi-task Cascaded Convolutional Networks (MTCNN) is used for face detection. MTCNN is a deep learning-based framework that detects faces in images through three stages: Proposal Network (P-Net), Refine Network (R-Net), and Output Network (O-Net).

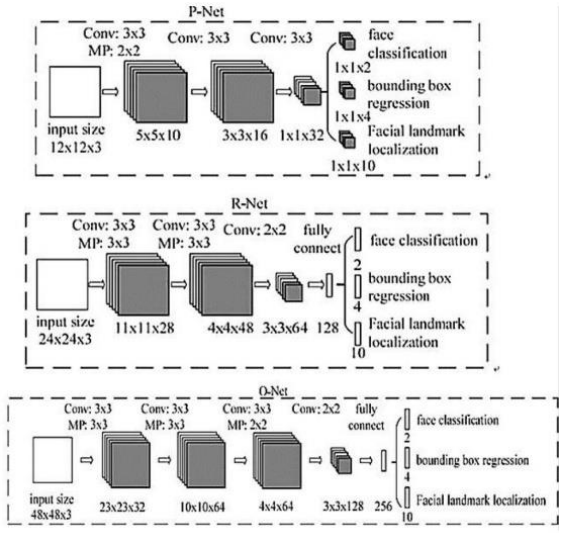

Fig 2: Architecture of MTCNN

The MTCNN model not only detects faces but also estimates facial landmarks such as eyes, nose, and mouth, which are important for face alignment before feature extraction.

Steps in Face Detection with MTCNN:

1. P-Net (Proposal Network): The first stage of MTCNN, P-Net, performs a sliding window approach over the input image. It uses convolutional layers to propose candidate face regions. These proposals are evaluated using a bounding box regression and classification to determine the likelihood of a face being present. The output of P-Net is a set of candidate regions with corresponding confidence scores.

$$Confidence(r_i) = f(x_i; \theta) \quad (1)$$

Where r_i represents the proposed region (bounding box), x_i is the region's feature representation, and θ are the parameters of the convolutional network.

2. R-Net (Refine Network): After the P-Net generates the face proposals, the R-Net refines these proposals by further classification and bounding box regression. R-Net takes the proposed bounding boxes from P-Net, extracts feature maps, and evaluates them to classify whether the region contains a face. It also refines the bounding box locations for more accurate localization.

$$Confidence(r_i) = f(x_i; \theta) \quad (2)$$

Where x_i is now the feature extracted from the proposed region after the initial step by P-Net, and θ refers to the parameters in R-Net.

3. O-Net (Output Network): The final stage, O-Net, further refines the face proposals by providing an even more accurate classification of the detected face, along with the precise facial landmarks. The bounding box is adjusted further to minimize errors, and the output includes the positions of the eyes, nose, and mouth.

$$Confidence(r_i) = f(x_i; \theta) \quad (3)$$

Where this final layer ensures the bounding box has been localized correctly and gives more precise landmark information.

C. Embedding Generation using FaceNet

After successful face detection and alignment using MTCNN, the next critical phase in the face verification pipeline is generating embeddings using FaceNet. These embeddings are numerical representations of faces in a high-dimensional space where spatial proximity reflects facial similarity. Below is a detailed step-by-step outline of the embedding generation process using FaceNet:

1. Input Preprocessing: The cropped and aligned face image obtained from MTCNN is resized to 160×160 pixels, which is the standard input size for FaceNet. The image is then normalized, typically to a pixel value range of [−1,1], to match the distribution the model was trained on.
2. Passing Through the Network: The preprocessed image is fed into the FaceNet model, a deep convolutional neural network designed to extract rich and discriminative facial features. The network processes the image through multiple convolutional and fully connected layers to produce a 128-dimensional feature vector, known as the face embedding.
3. Embedding Normalization: The output embedding is normalized using L2 normalization, which scales the vector to have a unit norm:

$$\hat{f}(X) = \frac{f(X)}{||f(X)||} \quad (4)$$

This ensures that all embeddings lie on a unit hypersphere, enabling reliable comparisons using cosine similarity or Euclidean distance.

4. Training with Triplet Loss: FaceNet is trained using the triplet loss function, which ensures that embeddings of the same identity are close together while those of different identities are far apart. The loss function is defined as:

$$||f(A) - f(P)||^2 + \alpha < ||f(A) - f(N)||^2 \quad (5)$$

where, A represents the anchor image, P is the positive image (i.e., an image of the same identity as the anchor), N is the negative image (i.e., an image of a different identity), and α is the margin that enforces a minimum distance between the positive and negative pairs to improve embedding separability.

The final output is a 128-dimensional vector for each face, capturing unique identity features. These embeddings are then used for comparison using distance metrics like cosine similarity or Euclidean distance.

D. Cosine Similarity

Cosine similarity is a key metric used to determine the degree of similarity between two facial embeddings in a face verification system. After the embeddings are generated by the FaceNet model for two different face images, cosine similarity measures how closely the two embeddings align in terms of direction in the vector space. Rather than comparing the magnitude of the vectors, cosine similarity focuses on the orientation, which makes it especially suitable for high-dimensional embeddings like those produced by FaceNet.

In the context of face verification, cosine similarity is used to compare an embedding from a query image with that of a reference image. If the cosine similarity score is close to 1, it indicates that the two images are highly similar and likely belong to the same person. Conversely, a lower similarity score suggests that the images represent different individuals. A predefined threshold is set to decide whether a given pair of faces should be classified as the same or different. This threshold can be adjusted depending on the application to balance between false positives and false negatives. Cosine similarity is preferred in face verification tasks because it is computationally efficient and effective in capturing the subtle differences and similarities in facial features. It enables fast and reliable comparison of embeddings, making it a crucial step in face recognition and verification pipelines.

IV. Experimental Setup

This study utilized TensorFlow and Keras for deep learning implementation, with Python 3.10.12 as the development environment. Face detection was performed using MTCNN via the facenet-pytorch library, and embeddings were generated using FaceNet. Experiments were run on an HPC system with an NVIDIA GeForce RTX 3090 GPU (24 GB GDDR6X RAM), enabling efficient training and inference. This setup ensured high computational performance and methodological reliability.

V. Results and Discussions

The proposed face verification system was rigorously evaluated on a controlled test dataset comprising 96 image pairs, evenly divided into positive pairs (same individual) and negative pairs (different individuals). Each image underwent face detection and alignment using the Multi-task Cascaded Convolutional Neural Network (MTCNN), followed by 128-dimensional embedding extraction via the pre-trained FaceNet model. Pairwise cosine similarity was computed to assess identity similarity and make verification decisions.

A. Cosine Similarity and Threshold Selection

Cosine similarity was employed as the similarity metric of choice due to its proven efficacy in capturing angular distances within high-dimensional embedding spaces. This metric is particularly suited for tasks involving facial feature vectors, as it measures the orientation between two vectors regardless of their magnitude, thereby focusing purely on the directional similarity of the embeddings. An empirical examination of similarity scores across the dataset revealed a distinct bimodal distribution: embeddings from positive pairs (same individual) typically exhibited cosine similarity scores well above 0.8, whereas those from negative pairs (different individuals) clustered below 0.6. This natural separation suggested that the FaceNet embeddings effectively encoded discriminative features.

Fig 3: Image describing the result that both faces are same

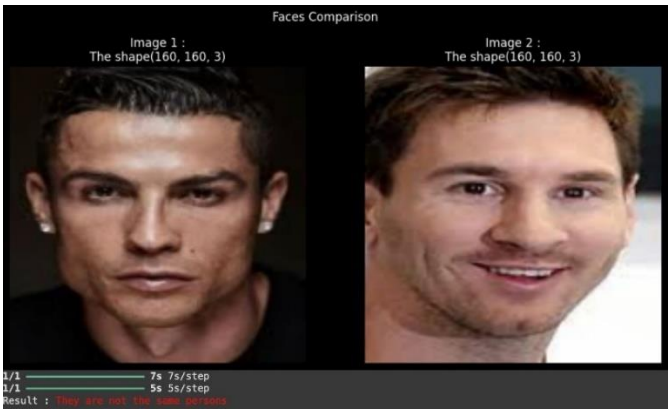

Fig 4: Image describing the result that both faces are different

A decision threshold of 0.7 was empirically selected to maximize inter-class margin while minimizing intra-class misclassifications:

- Pairs scoring $\geqslant$ 0.7 were classified as "same person".
- Pairs scoring < 0.7 were classified as "different persons".

The figure X illustrates the distribution of cosine similarity scores computed from FaceNet embeddings for the test dataset, which contains 96 image pairs equally split between positive and negative samples. The blue bars represent scores for same-person pairs, while the red bars represent different-person pairs. A vertical dashed black line marks the decision threshold at 0.7. The visualization reveals a distinct separation between the two classes: most "different person" scores cluster below 0.3, while the "same person" scores predominantly lie above 0.7.

This separation supports the choice of 0.7 as an effective classification threshold, helping to minimize overlap and reduce false positives. The inclusion of kernel density estimates (KDE curves) further emphasizes the sharp contrast in distribution shapes and locations between the two groups,

confirming the strong discriminative capacity of the cosine similarity metric when paired with FaceNet embeddings.

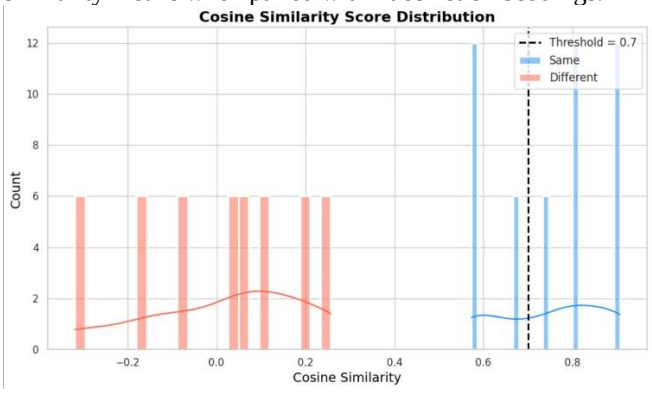

Fig 5: Distribution of cosine similarity scores for positive (same person) and negative (different person) image pairs

B. Confusion Matrix Analysis

The model's classification outcomes were further evaluated through a confusion matrix, which provided insight into how accurately the system distinguished between matching and non-matching pairs. The matrix revealed 48 true negatives (correctly identified different individuals), 30 true positives (correctly identified same individuals), 0 false positives (no incorrect same-person identifications), and 18 false negatives (missed same-person identifications). The absence of false positives is particularly notable, as it indicates the system's strong conservatism in affirming identity matches—an important property in applications where false acceptance carries significant consequences, such as secure access systems or border control.

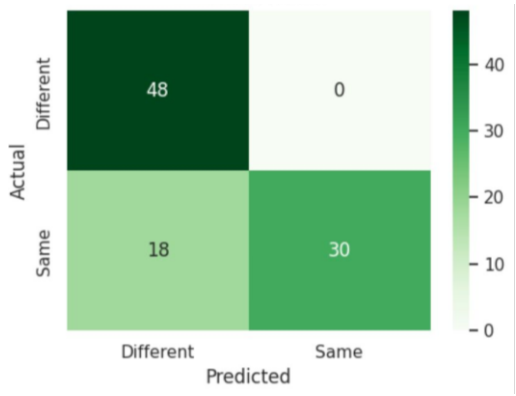

Fig 6: Confusion Matrix

However, the 18 false negatives suggest that the system encountered difficulty in some genuine identity matches. These misclassifications may stem from various factors, including non-frontal facial orientations, occlusions such as glasses or shadows, or dramatic differences in facial expressions. Such conditions can distort the facial geometry and diminish the similarity between otherwise matching pairs, thus lowering their cosine similarity score below the threshold.

C. Performance Metrics

A suite of performance metrics was computed to quantitatively assess the system's classification behavior. The overall accuracy of the system was 81.25%, indicating that it correctly classified approximately four out of every five image pairs. The precision for the "same person" class was 1.00, meaning that every image pair predicted to be a match was indeed a true match—there were no false positives. The recall for the same class, however, was 0.62, reflecting that the system correctly identified only 62% of all actual matches. This disparity suggests a cautious decision boundary that favors specificity over sensitivity.

On the other hand, for the "different person" class, the system achieved a perfect recall of 1.00, successfully identifying all mismatched pairs, though the precision was slightly lower at 0.73 due to the relative class imbalance. The F1-score, which balances precision and recall, was 0.77 for the "same" class and 0.84 for the "different" class. The macro-averaged and weighted F1-scores were both 0.81, reflecting balanced performance across both categories. These metrics collectively highlight the system's strength in avoiding false positives, with room for improvement in detecting all valid matches.

The key performance indicators are:

- Accuracy: 0.8125 (81.25%)
- Precision (Different): 0.73
- Recall (Different): 1.00
- F1-Score (Different): 0.84
- Precision (Same): 1.00
- Recall (Same): 0.62
- F1-Score (Same): 0.77

Table 1: Classification Report

	Precision	***Recall***	***F1-score***	***Support***
Different	0.73	1.00	0.84	48
Same	1.00	0.62	0.77	48
Accuracy			0.81	96
Macro average	0.86	0.81	0.81	96
Weighted average	0.86	0.81	0.81	96

This Figure 6 presents a visual summary of the system's classification performance across two classes: image pairs of the same person and those of different individuals. Each bar represents a key evaluation metric—precision (blue), recall (orange), and F1-score (green)—for the respective class.

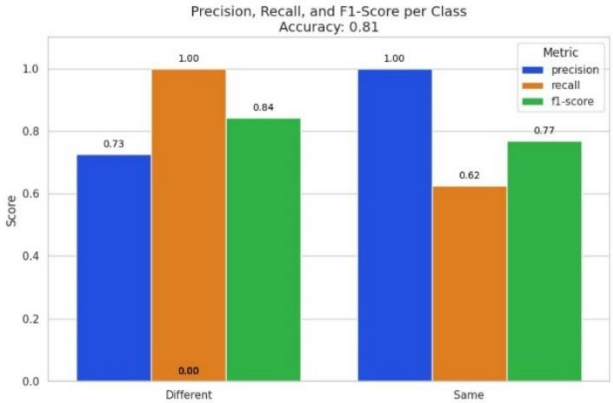

Fig 7: Bar chart showing precision, recall, and F1-score for each class

For the "Different" class (i.e., negative pairs), the model achieves a perfect recall of 1.00, indicating that all

different-person pairs were correctly identified. The precision is 0.73, reflecting that out of all the pairs predicted as "Different," 73% were indeed negative pairs. The F1-score, which balances precision and recall, stands at 0.84, showing strong performance in detecting mismatches. Conversely, for the "Same" class (i.e., positive pairs), the system demonstrates a perfect precision of 1.00, meaning there were no false positives—every prediction of a match was correct. However, the recall is notably lower at 0.62, suggesting that 38% of true matches were missed by the model. The resulting F1-score is 0.77, indicating decent but improvable sensitivity to genuine matches.

The overall system accuracy is 81%, which aligns with the metrics shown. The disparity between recall scores, especially for the "Same" class, underscores the need for further enhancement in handling intra-class variations due to pose, lighting, and occlusion.

D. ROC Curve

The ROC curve depicted above illustrates the trade-off between the true positive rate (TPR) and false positive rate (FPR) at various decision thresholds for the face verification system. The blue line represents the actual performance of the model, while the dashed diagonal line denotes the performance of a random classifier, which serves as a baseline.

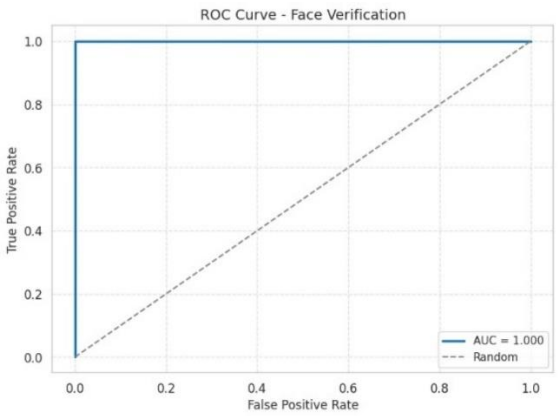

Fig 8: Receiver Operating Characteristic (ROC) curve evaluating the face verification system

In this case, the curve reaches the top-left corner of the graph immediately, indicating perfect classification ability across all thresholds. This behavior is further confirmed by the Area Under the Curve (AUC) value of 1.000, which suggests that the model is capable of perfectly distinguishing between same-person and different-person image pairs within the test dataset. This result implies that the embedding-based similarity measurements, combined with the selected classification threshold, lead to an ideal separation of classes—at least under the evaluation conditions. However, such a perfect AUC, while impressive, may also warrant additional validation on more diverse datasets to ensure that the model generalizes well and is not overfitted to the current testing scenario.

VI. Conclusion

Facial recognition technology has significantly advanced with the integration of deep learning, enabling accurate, efficient, and real-time identification across various applications. From security and biometric authentication to healthcare and retail, facial recognition has revolutionized industries by offering seamless and automated identity verification. The use of convolutional neural networks (CNNs) and deep learning frameworks such as TensorFlow and PyTorch has greatly improved recognition accuracy, even in challenging conditions such as low lighting, occlusions, and varied facial expressions. The successful implementation of a facial recognition system requires a combination of high-performance hardware, such as GPUs for deep learning computations, and optimized software, including machine learning libraries and database management tools. Cloud-based solutions and edge AI further enhance scalability and real-time processing, allowing for more flexible deployment in different environments.

Despite its benefits, facial recognition poses challenges, including privacy concerns, data security risks, and biases in recognition accuracy across different demographic groups. Ethical considerations and regulatory frameworks must be carefully addressed to ensure responsible deployment and prevent misuse. Advances in encryption techniques, federated learning, and bias-mitigation strategies are crucial to making facial recognition systems more secure and equitable. Looking ahead, continuous research and technological innovations will further enhance the reliability, fairness, and security of facial recognition systems. By integrating AI-driven improvements with ethical best practices, facial recognition will continue to play a transformative role in both public and private sectors, shaping the future of automated identity verification and security solutions.

Declarations

Competing interests No conflicting interests are disclosed by the authors.

Data availability Data will be available on request.

References

[1] Priya, S. B. (2024). A Comprehensive Review of Face Recognition Techniques, Trends and Challenges. IEEE Access, 1. https://doi.org/10.1109/access.2024.3424933

[2] Abdul-Al, M., Kyeremeh, G. K., Qahwaji, R., Ali, N., & Abd-Alhameed, R. A. (2024). A Novel Approach to Enhancing Multi-Modal Facial Recognition: Integrating Convolutional Neural Networks, Principal Component Analysis, and Sequential Neural Networks. IEEE Access, 1. https://doi.org/10.1109/access.2024.3467151

[3] Malakar, S., Chiracharit, W., & Chamnongthai, K. (2024). Masked Face Recognition with Generated Occluded Part using Image Augmentation and CNN Maintaining Face Identity. IEEE Access, 1. https://doi.org/10.1109/access.2024.3446652

[4] Surveillance System for Real-Time High-Precision Recognition of Criminal Faces From Wild Videos. (2023). IEEE Access, 11, 56066–56082. https://doi.org/10.1109/access.2023.3282451

[5] Face Recognition Method Based on Siamese Networks Under Non-Restricted Conditions. (2022). IEEE Access, 10, 40432–40444. https://doi.org/10.1109/access.2022.3167143

[6] Real-Time Implementation of Face Recognition and Emotion Recognition in a Humanoid Robot Using a Convolutional Neural

Network. (2022). IEEE Access, 10, 89876–89886. https://doi.org/10.1109/access.2022.3200762

[7] Design for Visitor Authentication Based on Face Recognition Technology Using CCTV. (2022). IEEE Access, 10, 124604–124618. https://doi.org/10.1109/access.2022.3223374

[8] Liu, H., Zheng, N., Wang, Y., Li, J., Zhang, Z., Li, Y., & Lan, J. (2021). Development of a Face Recognition System and Its Intelligent Lighting Compensation Method for Dark-Field Application. IEEE Transactions on Instrumentation and Measurement, 70, 1–16. https://doi.org/10.1109/TIM.2021.3111076

[9] Li, X., Yang, Z., & Wu, H. (2020). Face Detection Based on Receptive Field Enhanced Multi-Task Cascaded Convolutional Neural Networks. IEEE Access, 8, 174922–174930. https://doi.org/10.1109/ACCESS.2020.3023782

[10] Chen, H.-Q., Xie, K., Li, M.-R., Wen, C., & He, J.-B. (2022). Face Recognition With Masks Based on Spatial Fine-Grained Frequency Domain Broadening. IEEE Access, 10, 75536–75548. https://doi.org/10.1109/access.2022.3191113

[11] Valverde, E. C., Oroceo, P. P., Caliwag, A. C., Lim, W., & Maier, M. (2023). Edge-Oriented Social Distance Monitoring System Based on MTCNN. IEEE Transactions on Industrial Informatics, 19, 8654–8666. https://doi.org/10.1109/TII.2022.3217499

[12] Recognizing Very Small Face Images Using Convolution Neural Networks. (2022). IEEE Transactions on Intelligent Transportation Systems, 23(3), 2103–2115. https://doi.org/10.1109/tits.2020.3032396

[13] Putro, M. D., Kurnianggoro, L., & Jo, K.-H. (2021). High Performance and Efficient Real-Time Face Detector on Central Processing Unit Based on Convolutional Neural Network. IEEE Transactions on Industrial Informatics, 17(7), 4449–4457. https://doi.org/10.1109/TII.2020.3022501

[14] Sooch, S. K., & Anand, D. (2021). Emotion Classification and Facial Key point detection using AI. IEEE Access. https://doi.org/10.1109/ACCESS51619.2021.9563289

[15] Li, X., Yang, Z., & Wu, H. (2020). Face Detection Based on Receptive Field Enhanced Multi-Task Cascaded Convolutional Neural Networks. IEEE Access, 8, 174922–174930. https://doi.org/10.1109/ACCESS.2020.3023782

[16] Horng, S.-J., Supardi, J., Zhou, W., Lin, C.-T., & Jiang, B. (2020). Recognizing Very Small Face Images Using Convolution Neural Networks. IEEE Transactions on Intelligent Transportation Systems, 1–13. https://doi.org/10.1109/TITS.2020.3032396

[17] Khan, M. Z., Harous, S., Hassan, S. U., Khan, M. U. G., Iqbal, R., & Mumtaz, S. (2019). Deep Unified Model For Face Recognition Based on Convolution Neural Network and Edge Computing. IEEE Access, 7, 72622–72633. https://doi.org/10.1109/ACCESS.2019.2918275

[18] Abdelmaksoud, M., Nabil, E., Farag, I., & Abdel Hameed, H. (2020). A Novel Neural Network Method for Face Recognition With a Single Sample Per Person. IEEE Access, 8, 102212–102221. https://doi.org/10.1109/ACCESS.2020.2999030

[19] Yang, S., Luo, P., Loy, C. C., & Tang, X. (2018). Faceness-Net: Face Detection through Deep Facial Part Responses. IEEE Transactions on Pattern Analysis and Machine Intelligence, 40(8), 1845–1859. https://doi.org/10.1109/TPAMI.2017.2738644

[20] Anusudha, K. (2024). Real time face recognition system based on YOLO and InsightFace. Multimedia Tools and Applications, 83(11), 31893-31910.