Ensemble Learning for Face Recognition in Suspect Identification using Cloud Environment

Submitted: 21st August 2024; accepted: 24th September 2024

Shilpa Chaudhari, Rajarajeswari S, Archana Rane

DOI: 10.14313/jamris-2026-022

Abstract:
Facial recognition technology finds applications in security, surveillance, and social media. Existing research explores the use of machine learning and deep learning for face recognition, emphasizing the need for improved accuracy. This paper proposes a system for suspect identification using facial recognition. The system leverages ensemble learning by integrating seamlessly with OpenAI's advanced technologies and is supported by a robust cloud infrastructure. The comparison of the proposed ensemble model to individual models like VGG-Face, Facenet, Facenet512, Deepface, DeepID, ArcFace, and SFace uses multiple detectors and the Labelled Faces in the Wild (LFW) dataset. The results show that the ensemble model offers the most efficient processing time across all sample sizes. In contrast, models like VGG-Face and DeepID exhibit a steeper increase in processing time, suggesting lower scalability. For instance, at a sample size of 50, the local test completes in 61.3 seconds, while the cloud API test takes 67.2 seconds. This highlights the faster processing speed of the local test across all sample sizes. FaceNet, VGG-Face, and ArcFace models are chosen in ensemble model where in all of them have accuracy above 95% in every face detector test. Facenet512 model has 98.4% among the selected ensemble model whereas ensemble of these models shows 98.8 accuracy.

Keywords: *Face recognition, Accuracy test, Deep learning, ensemble learning, Cloud API*

1. Introduction

Facial recognition technology has become a powerful tool with applications in social media [19], security [20], and surveillance [18]. The traditional method of using hand-drawn sketches for suspect identification is labor-intensive, time-consuming, and often a bottleneck in the investigation process. Despite the spread of modern recognition techniques, the inefficiencies inherent in manual sketching processes—limited scalability, significant time investment, and resource constraints—have hindered the timely and effective identification of suspects [21]. These challenges underscore the urgent need for a revolutionary application that transcends the selection of individual facial features to enable rapid, comprehensive face sketch recognition. Existing research explores various machine learning and deep learning models for face recognition [22, 23].

The proposed method of this paper identifies seven prominent models: VGG-Face [24], FaceNet [25], FaceNet512, DeepFace [26], DeepID, ArcFace [28], and Sface [27]. These models are trained on large datasets (like Labelled Faces in the Wild (LFW)) to achieve high accuracy (above 90%). VGG-Face learns distinctive features for the face recognition process. FaceNet maps face images to high-dimensional features for similarity recognition. FaceNet512 utilizes a higher dimensional space for improved face recognition performance. DeepFace extracts facial features for multi-task learning encompassing face detection, alignment, and recognition. DeepID learns hierarchical representations of faces for high accuracy in face verification and identification. ArcFace and Sface enhance the discriminative power within the face feature space through specific loss functions during training. These models are widely used and have significantly contributed to advancements in face recognition. Their performance varies based on factors like datasets, evaluation protocols, and application (in terms of accuracy, precision, recall, and F1-score).

Ensemble learning combines multiple models to surpass the performance of any single model. It aggregates the diverse predictions from chosen models, leveraging the "wisdom of the crowd." The proposed face recognition model utilizes ensemble learning to improve its performance by harnessing the collective strengths of the seven models while mitigating individual weaknesses. Cloud environments offer convenient data storage and retrieval from anywhere. Services like Amazon S3 provide an interface for cost-effective retrieval of large datasets. Combining these S3 buckets with AWS segmentation services ensures authorized user access to specific data stored within the bucket.

The proposed system designs and develops a recognition model using machine and deep learning-based ensemble learning by leveraging advanced facial recognition against criminal databases with efficient cloud-based storage, management, and security of suspect faces. This approach ensures scalability, reliability, and cost-effectiveness for face recognition application workloads. It establishes a cloud-based centralized database to facilitate easy global access to crucial information for law enforcement agencies. The developed model is tested on AWS cloud for

portability using the respective APIs. The integration of deep learning algorithms and cloud infrastructure for database matching and verification, aiming at a significant improvement in the efficiency of the identification process. Our specific contributions include the following: (1) design and development of recognition model using machine and deep learning-based ensemble learning; (2) establishment of a cloud-based centralized database to facilitate easy global access; (3) ensuring the scalability, reliability, and cost-effectiveness for face recognition application workloads using S3 buckets with AWS segmentation services; and (4) a performance analysis of the developed face recognition model.

This paper is structured as follows: Section II discusses related work on face recognition using deep learning. Section III details the proposed methodology for ensemble learning in a cloud environment. Section IV presents result analysis, followed by the conclusion in Section V.

2. Related Works

The Dual-Scale Markov Network (DSMN) and Multi-Information Fusion algorithm [1] represents an innovative contribution to the field of face sketch-photo synthesis and recognition considering diverse facial features and variations. The algorithm integrates information from multiple sources, enhancing the synthesis and recognition accuracy. The accuracy of the DSMN relies on the quality of the initial sketch. If the sketch is poor, the synthesized photo or recognition accuracy suffers. The information fusion stage, might be computationally expensive, making it challenging for real-time applications.

The authors of [2] achieved promising performance in matching composite sketches generated by eyewitnesses, to mugshot photographs. It relies on three key techniques: Active Shape Model (ASM) for pinpointing facial landmarks on both the sketch and photo, Multiscale Local Binary Patterns (MLBP) to extract distinctive features from each facial component, and component similarity fusion to determine the overall match between the sketch and photo. This component-based approach achieved significant improvements when compared to a leading commercial face recognition system and a simpler method that analyzed the entire face as a single unit. The promising performance suggests that this method has the potential to be a valuable tool for law enforcement agencies in identifying and apprehending suspects.

A comprehensive survey of various 3D face reconstruction techniques is explored in [3], including deep learning, epipolar geometry, one-shot learning, 3D morphable models, and shape-from-shading methods. It delves deeper into the analysis of deep learning-based reconstruction techniques, dominant technique with high accuracy, detail, and robustness. Reconstructing faces from historical photographs or rare medical scans might be challenging due to the scarcity of training data from similar domains. It may require additional cues like depth information for better accuracy for handling complex lighting condition.

The role of facial reconstruction (FFR) identifies unknown individuals by utilizing a combination of scientific principles and artistic skills to recreate a likeness of the deceased based on their skeletal remains [4]. The process involves employing various techniques such as 2D/3D computer-aided methods, manual sculpting, and even clay modeling. An important aspect of FFR is the understanding of facial tissue depth variations those are influenced by factors such as age, sex, and ancestry. However, the accuracy of FFR is currently hampered by limitations in the existing database on tissue depth variations, particularly for non-caucasian populations. This highlights the need for further research and data collection to improve the accuracy of FFR for a wider range of demographics.

An approach to facial image editing [5] leverages a hybrid Convolutional Neural Network (CNN) architecture to manipulate specific facial attributes. Notably, it incorporates a pre-trained facial recognition model to extract key features from the image. This allows the framework to edit aspects like age, gender, expression, and hair color while maintaining a realistic appearance and preserving the underlying facial identity. The current study focuses on editing faces of Asian descent. Further research is needed to determine the generalizability of this approach to a wider range of ethnicities.

3D face reconstruction leverages a neural network to not only predict the 3D face shape but also assess the confidence of the reconstruction [6]. The authors demonstrate that their approach surpasses shape-averaging techniques in terms of reconstruction accuracy, particularly on the MICC dataset. The neural network tends to favor high-quality face images for reconstruction, specifically those with frontal poses, clear visibility, and natural lighting conditions. Conversely, images containing occlusions like sunglasses, hats, or hair can lead to decreased confidence scores in the reconstruction process. This highlights the need for the model to be more robust to variations in image quality and pose for real-world applications.

The study of [7] investigates eye-tracking data of participants viewing freehand sketches. Interestingly, the research identified consistent patterns in how people fixate on various parts of the sketch, both within individual sketches and across sketches of the same category or related groups using a sketch-specific data augmentation technique. This method significantly improves the accuracy of deep learning models in recognizing freehand sketches.

The automation of facial composite production and identification processes focuses on EvoFIT system, a software program that allows witnesses to build facial composites by selecting features from a database [8]. The study compared a standalone version of EvoFIT, which does not require operator guidance, to the full system with a human operator using the Shape Tool. They evaluated the resulting composites using a root mean square error (RMSE) measure

to assess their similarity to the target face, the study explored the potential for matching composites generated using EvoFIT against a database of other composites.

Coupled Information-Theoretic Encoding (CITE) utilizes Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) alongside a Randomized CITE Tree Algorithm to extract informative features from different modalities: photos and sketches [9]. These features are then fed into a Linear Support Vector Machine (SVM) for classification. Notably, the CITE descriptors outperform popular facial recognition features like Local Binary Patterns (LBP) and Scale-Invariant Feature Transform (SIFT). CITE method significantly improves verification rates at low false acceptance rates compared to existing methods. This improvement is further enhanced by incorporating PCA, LDA, and SVM for information fusion, solidifying CITE as a powerful approach for face photo-sketch recognition.

CITE allows the system to capture the essence of facial structure despite the inherent differences between these image types [19]. The technique of synthesizing pseudo photos from query sketches and using RS-LDA for matching demonstrated certain limitations, especially when dealing with significant shape distortion between photos and sketches in the training set.

A real-time deep neural network architecture called DiFRuNNT for disguised face verification [11] consists of two neural networks: CNN to predict 20 facial key points in the image, and a secondary network to classify subjects based on angles and ratios calculated from these predicted points. The achieved accuracies are 67.4% for prediction and 74.8% for classification, respectively.

Conceptual categorization and evaluation metrics given in [12] provides comprehensive survey of relevant publications on face recognition systems under morphing attacks as well as discusses technical considerations, tradeoffs, open issues, and challenges in the field.

Review of deep learning methods in face recognition covers various deep learning architectures, loss functions, databases, protocols, and application scenes [13]. It highlights the rapid evolution and significant impact of deep learning on face recognition, discussing challenges and promising directions.

Discriminative approaches without explicit age modeling achieves excellent performance in face verification across age progression [14]. They find that gradient orientation, particularly in a hierarchical structure called the gradient orientation pyramid (GOP), combined with SVMs, achieves excellent performance. Empirical study on age gaps impact on recognition algorithms, providing insights into age-related challenges.

The conventional pipeline for face recognition involves four stages: detect, align, represent, and classify for improving the alignment and representation steps by incorporating explicit 3D face modeling to apply a piecewise affine transformation. They derive a face representation using a nine-layer deep neural network with over 120 million parameters, employing locally connected layers without weight sharing. Trained on a massive facial dataset containing over four million images from more than 4,000 identities, their method couples accurate model-based alignment with a large facial database, yielding remarkable generalization to faces in unconstrained environments. Even with a simple classifier, their approach achieves an accuracy of 97.35% on the LFW dataset, reducing the error of the current state of the art by over 27% and approaching human-level performance closely.

2D-to-3D integrated face reconstruction approach significantly improves accuracy of face recognition with changing pose, illumination, and expression (PIE) [16]. It offers an efficient and automatic framework for 3D face reconstruction and recognition, overcoming challenges of variant factors like PIE. Experimental results show that their synthesized virtual faces significantly enhance recognition accuracy, especially when dealing with changes in PIE conditions.

Identity-Preserving Face Recovery from Portraits (IFRP) recover photorealistic faces from artistic portraits while preserving identity poses a significant challenge due to potential distortions or loss of fine details. It comprises two main components: the Style Removal Network (SRN) and the Discriminative Network (DN). SRN and DN recover latent photorealistic faces while preserving identity. It introduces a method for recovering realistic faces from unaligned stylized portraits while preserving identity, achieving state-of-the-art results.

3. Proposed Face Recognition

Face recognition process flow involves a user, recognition model, AWS segmentation, and an S3 storage component as shown in Figure 1. The sequence begins with the user providing an image, which is then fed into the recognition model and uploaded to an AWS S3 bucket for storage. The recognition model initiates a recognition task by retrieving the image from AWS S3, facilitated by the AWS Segmentation service that performs pre-processing tasks like image segmentation or feature extraction. Once the recognition process is completed, the model calculates a match percentage and generates metadata related to the image, sending these results back to the user. This workflow integrates cloud storage and processing, with the core functionality residing in the deep learning model deployed on the AWS Lambda cloud platform for on-demand execution. The model focuses on extracting key features from the input image, particularly those corresponding to facial elements like the eyes, nose, and mouth. By comparing these extracted features against a database of facial images, the model attempts to identify potential matches between the sketch and real suspects.

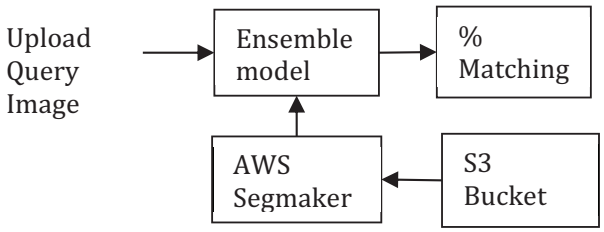

Figure 1. Face Recognition Process

The recognition module receives results from the deep learning model, which typically include similarity scores for each potential suspect in the database, indicating how closely the sketch resembles a particular suspect's facial features. Acting as a bridge between the user-query image and the suspect database, the recognition module leverages the power of deep learning to identify potential matches based on facial recognition. The accuracy of seven existing algorithms in Deepface, including VGG-Face, FaceNet, FaceNet512, DeepFace, DeepID, ArcFace, and SFace, is tested. This process enhances recognition tasks by effectively leveraging cloud computing capabilities for scalable and efficient image analysis. The evaluation begins with meticulous dataset selection. Each image undergoes standard preprocessing steps such as normalization of size and color intensity, and alignment using detectors like Retinaface, MTCNN, fast MTCNN, dlib, yolov8, yunet, centerface, mediapipe, ssd, and OpenCV. Testing of the seven models involves comparing an input image to each existing image, with predictions recorded to calculate the models' accuracy. These seven models run in an ensemble model processed in parallel for the same input image. The most frequently occurring output image generated by each individual model is then put through a voting system, where the mode of all output images is taken as the final result. Given that each facial recognition model was trained on different datasets by different developers at different times, varying results are expected. Thus, this method of taking the mode value of the results provided by each model ensures greater accuracy and consistency. A facial recognition model broadly works on the same pipeline as shown in Figure 2.

Facial Recognition Process Flow is as follows: (1) Image Input: The process begins with an image provided by the user as shown in Figure 2; (2) Storage: The image is uploaded to a designated storage location, an Amazon S3 bucket in this case; (3) Image Preprocessing: The Amazon Segmentation service retrieves the image from storage and performs preprocessing tasks like segmentation or feature extraction; (4) Recognition Model: A deep learning model, deployed on the AWS Lambda platform, analyzes the preprocessed image. The model extracts key features from the image, particularly those corresponding to facial elements. These features are compared against a database of facial images to identify potential matches; (5) Results Generation: The model calculates a match percentage for each potential match in the database. Metadata related to the image is also generated; and (6) Output: The results, including match percentages and image metadata, are sent back to the user.

Following Deep Learning Model Details are considered. (1) The core functionality lies in a deep learning model trained to extract facial features. (2) The model is evaluated against various existing algorithms to ensure optimal accuracy. (3) An ensemble model approach is used, where multiple models process the image in parallel and the final result is determined through a voting system.

This workflow leverages cloud storage (S3) and processing (Lambda) for scalable and efficient image analysis. It demonstrates a typical facial recognition pipeline where an image is uploaded, processed, analyzed for facial features, compared against a database, and results are delivered.

3.1. Face Recognition and Alignment

Face detection involves identifying and aligning faces within an image. Alignment is straightforward once the face and eyes are detected. Various algorithms used for face detection are as follows: (1) RetinaFace: Discusses its architecture and the specific features that enable RetinaFace to handle different scales and orientations in face detection; (2) MTCNN: Explains the multi-task cascaded framework and its efficiency in detecting detailed face attributes alongside face detection; (3) FastMTCNN: Focuses on the optimizations that make FastMTCNN a faster alternative to MTCNN while maintaining comparable accuracy; (4) Dlib: Compares the HOG+SVM-based approach and the CNN-based detector in Dlib, highlighting scenarios where each is preferable; (5) YOLOv8: Describes how YOLOv8 adapts the YOLO object detection framework for fast and effective face detection; (6) YuNet: Provides details on YuNet's architecture and its effectiveness in real-time face detection applications; (7) CenterFace: Analyzes the method's approach to detecting face centers and scales, particularly in crowded environments; (8) MediaPipe: Explores the integration of MediaPipe's face detection in multimodal pipelines and its real-world applications; (9) SSD: Discusses the application of the SSD framework for face detection and its performance across various datasets; and (10) OpenCV: Outlines the use of Haar feature-based cascade classifiers and their suitability for entry-level face detection tasks. A detector detects an image and aligns it as shown in Figure 3.

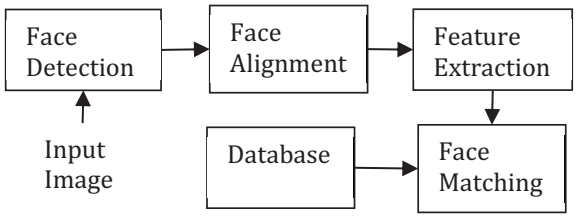

Figure 2. Face Recognition Stages

Figure 3. Facial Detection and Alignment

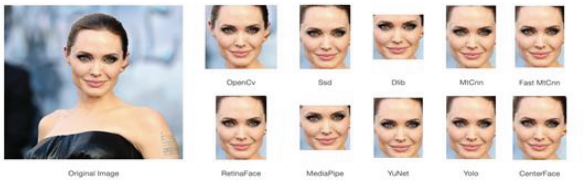

Figure 4. Facial Detection by Different Detectors

Figure 5. Nodal Points on Two Different People

Subsequently, different detectors perceive faces in distinct ways. For instance, Figure 4 illustrates how each of the previously mentioned detectors perceives the same face from a single image. While each cropped image includes the entire face, the variation lies in the amount of background retained in the image.

3.2. Feature Extraction

Features such as eyes, nose, mouth, etc. are extracted from the aligned face. This is achieved by using nodal points [14]. As seen in Figure 5, the ratio of the nodal points of both the humans are different, it is not possible for them to be similar unless they are identical twins.

Face Representation: Each face is represented as a high-dimensional vector in a feature space. This vector is obtained from the output of a deep neural network (e.g., VGG Face, FaceNet) that has been trained to map facial images into a compact embedding space.

3.3. Feature Matching

Encoded representations are compared to a database or gallery of known identities, using methods such as cosine similarity and Euclidean distance to determine the similarity between two vectors in a multi-dimensional space.

Distance Metric: To determine how similar or dissimilar two facial feature vectors are, the Euclidean distance between them is computed. For two feature vectors x and y, each of dimension n, the Euclidean distance d is calculated using the formula given in Equation 1 where *Sum of Squared Differences* is computed as given in Equation 2.

$$Euclidean\ distance: d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (1)$$

$$\text{Sum of Squared Differences} = \sum_{i=1}^{n}(x_i - y_i)^2 \quad (2)$$

Cosine Similarity in Face Recognition

1. Direction Over Magnitude: Cosine similarity focuses on the orientation of the vectors rather than their magnitude. This is useful for face recognition because it measures how similar the patterns of the features are, regardless of their scale. This can help in scenarios where the length of the feature vector may vary due to different factors like the scale of images or variations in lighting conditions.
2. Normalization: Cosine similarity normalizes the feature vectors, making it robust to variations in the length of the vectors. This normalization can be beneficial when comparing face embeddings that may be affected by varying image conditions.
3. Similarity Measurement: Cosine similarity is used to compute the similarity score between feature vectors. A higher cosine similarity indicates that the vectors are more similar, which can be interpreted as the faces being more likely to belong to the same individual.

Feature Vectors: For two facial feature vectors x and y, each of dimension n, the cosine similarity is defined as given in Equation 3.

$$\text{Cosine Similarity}(x.y) = \frac{x.y}{||x||\ ||y||} \quad (3)$$

where:

1) x·y is the dot product of the vectors.
2) ‖x‖ and ‖y‖ are the magnitudes (or norms) of the vectors.

Dot Product: The dot product x·y is calculated as given in Equation 4.

$$x \cdot y = \sum_{i=1}^{n} x_i.y_i \quad (4)$$

where xi and yi are the components of vectors x and y, respectively.

Vector Norms: The norm (or magnitude) of a vector x and y is calculated as given in Equation 5.

$$||x|| = \sqrt{\sum_{i=1}^{n} x_i{}^2} \ and \ ||y|| = \sqrt{\sum_{i=1}^{n} y_i{}^2} \qquad (5)$$

Combining these components, the cosine similarity is computed as given in Equation 6.

$$\text{Cosine Similarity}(x, y) = \frac{\sum_{i=1}^{n} x_i . y_i}{\sum_{i=1}^{n} x_i{}^2} \sum_{i=1}^{n} y_i{}^2 \qquad (6)$$

This value ranges from -1 to 1, where 1 indicates that the vectors are identical in direction, 0 indicates orthogonality (no similarity), and -1 indicates opposite directions.

The ensemble model includes seven recognition models, each running simultaneously. The focus on each model individually is given as follows. (1) VGG-FACE: The VGG Face model extracts discriminative features from facial images, significantly enhancing facial recognition technology. Trained on a vast dataset, these networks learn complex feature hierarchies crucial for facial identification, effectively handling varying expressions and lighting conditions. VGG Face emphasizes features critical for accurate individual identification, supported by rigorous training and optimization. Additionally, it employs a joint Bayesian framework to model these features' distribution, enabling robust verification of face pairs in diverse real-world scenarios. This approach ensures VGG Face's reliability and effectiveness in high-security applications, setting a new standard in the field of facial recognition.

(2) ARCFACE: ArcFace is an advanced facial recognition system distinguished by its innovative use of an angular margin penalty added to the loss function during training. This method maximizes the distinctiveness between the learned features of different individuals (inter-class discrepancy) while maintaining consistency in features of the same individual across various images (intra-class compactness). ArcFace is trained on extensive datasets containing a wide array of facial images, allowing the neural networks to effectively learn and abstract complex hierarchies of facial features into higher layers. By integrating the angular margin penalty, ArcFace enhances feature embeddings by decisively separating embeddings of different classes and bringing those of the same class closer together. This mechanism significantly improves the discriminative power of the model, making it exceptionally capable of handling challenges in facial recognition, such as variations in lighting, expression, and other dynamic environmental factors.

(3) FACENET: FaceNet, developed by Google, utilizes deep CNNs to map face images into a compact Euclidean space where distances represent facial similarity. This model measures similarity by calculating the distance between points representing faces. The innovation of FaceNet lies in its use of the triplet loss function during training, which minimizes the distance between an anchor image and a positive image (same person) while maximizing the distance between the anchor and a negative image (different person). This approach enables FaceNet to accurately distinguish between individuals under varying conditions such as changes in expression, lighting, and camera angles. Its ability to maintain high accuracy despite these variations makes it highly effective for applications requiring reliable facial recognition.

(4) Facenet512 has a high value in accuracy calculation compared to Facenet.

(5) DEEPFACE: Deepface [15] Developed by Facebook, DeepFace represents a significant advancement in facial recognition technology. It utilizes a sophisticated nine-layer neural network with over 120 million connection weights, trained on an extensive dataset comprising four million facial images from more than 4,000 distinct identities. This system employs a novel approach by aligning faces using three-dimensional models, allowing it to adjust for variations in head position, orientation, and lighting conditions before processing the images through its deep neural net. The core functionality of DeepFace lies in extracting and utilizing detailed facial features from aligned images. It learns a compact representation of each face, simplifying the comparison and identification of faces across various conditions. By focusing on these representations, DeepFace achieves a high level of accuracy in facial recognition tasks, effectively capturing and analyzing subtle facial features crucial for reliable identification.

(6) DEEPID: DeepID represents a significant advancement in facial recognition technologies, focusing on the extraction and utilization of deep hidden identity features. Developed at The Chinese University of Hong Kong, DeepID employs deep CNNs to learn a hierarchy of features from a substantial dataset of facial images. These features, which become increasingly abstract at higher layers of the network, enhance the discriminative power of the model, making it particularly effective in distinguishing between different identities. Additionally, DeepID's application of a joint Bayesian framework to model these features allows for more accurate face pair verification, providing robustness against typical variations encountered in real-world scenarios, such as changes in expression and lighting conditions. This approach improves the accuracy of facial recognition systems and extends their applicability in security and personal identification, addressing pressing challenges faced by current technologies.

(7) SFACE: SFace is a lesser-known term in the context of popular face recognition technologies and might refer to a specific implementation or model within the research community. If it follows the conventions of other deep learning-based face recognition systems, SFace would likely employ a deep neural network to learn a representation of facial features useful for recognition tasks. The "S" could potentially stand for a specific feature of the model, such

as secure, simple, or scalable, indicating a focus on those aspects of the facial recognition process. Since SFace is not a widely recognized term like DeepFace or DeepID, the specifics of its architecture and functionality would depend on the context in which it is referenced, including the particular research paper or implementation that defines it.

4. Results and Discussion

The development environment leverages AWS as the cloud infrastructure to achieve scalability, elasticity, and potential cost reductions. The following AWS services are considered for implementation. (1) Amazon S3: For storing the facial image dataset and potentially the application code. (2) AWS Lambda: To execute the deep learning model for facial recognition in a serverless environment, enabling on-demand processing without server management. (3) Amazon SageMaker: As a managed service for building, training, and deploying machine learning models. SageMaker may be explored for deep learning model development as needed. (4) Amazon EC2 Instance: If AWS Lambda proves insufficient for deep learning tasks, an EC2 instance can be considered for model training or inference.

The system requires integration with a deep learning library (e.g., TensorFlow, PyTorch) to facilitate facial feature recognition, matching, and image generation capabilities. Deep learning serves as the core technology underpinning facial recognition functionality. Pre-trained deep learning models are utilized to analyze constructed sketches and extract relevant features for comparison against a facial image database. A deep learning model for facial recognition is to be developed using TensorFlow or PyTorch and trained on a dataset of facial images with labeled features. Model training can be executed on an EC2 instance or potentially through SageMaker.

A database API interface is used to connect to a criminal database containing suspect information and facial recognition data. Snowflake serves as the database management system for storing suspect information and potentially sketch data, providing a scalable and secure solution.

This study compares the accuracy of several models, including Facenet512, Facenet, VGG-Face, ArcFace, SFace, DeepFace, and DeepID, using the LFW dataset. The outcome of this research is a comparison of the accuracy of each model. Multiple face detectors, such as Retinaface, MTCNN, fastMTCNN, dlib, yolov8, yunet, centerface, mediapipe, ssd, and OpenCV, are used for testing to determine the best-performing model on the dataset [1]. The testing method involves verifying the accuracy of each model by comparing each image with another, yielding two outcomes: threshold and distance. Threshold values can be modified based on the model: 0.68 for VGG-Face, 0.4 for FaceNet, 0.3 for FaceNet512, 0.23 for DeepFace, 0.015 for DeepID, 0.68 for ArcFace, and 0.593 for SFace. Depending on the threshold, the distance value determines whether the result is true or false; if the distance exceeds the threshold, the result is false, and vice versa.

After testing, the results shown in Table 1 indicate that the Cosine metric was used. The results demonstrate that the FaceNet512 model achieved the highest accuracy value of 0.984 or 98.4% with the RetinaFace [11, 17] detector and surpassed all other models in every detector. The FaceNet model achieved an accuracy value of 0.974, the VGG-Face model achieved 0.960, the ArcFace model achieved 0.967, the SFace model achieved 0.924, the DeepFace model achieved 0.677, and the DeepID model achieved 0.659.

The Euclidean metric was used to obtain the test results in Table 2. The results indicate that the FaceNet512 model achieved the highest accuracy value of 0.976 or 97.6% with the centerface detector, outperforming all other models in every detector. The highest accuracy values for other models are as follows: FaceNet model at 0.938, VGG-Face model at 0.960, ArcFace model at 0.886, SFace model at 0.814, DeepFace model at 0.690, and DeepID model at 0.665.

From these results, it can be concluded that the Facenet512 model is superior to every other model in terms of accuracy.

Cosine similarity distances of various facial recognition models when tested with a set of 20 and 50 sample images. Is given in Table 3. Lower values indicate closer or more accurate matches. For 20 sample images, ArcFace demonstrates the best performance with the lowest distance at 0.012, suggesting highly accurate model predictions. In contrast, VGG-Face has the highest distance at 0.46, indicating less precision. Other models like Facenet, Facenet512, Sface, DeepFace, and Ensemble exhibit varied performances, with distances ranging between 0.12 and 0.46.

For 50 sample images, VGG-Face starts with the highest distance at 0.42, indicating less precision. Facenet512 shows the best performance with the lowest distance at 0.17. The distances for other models like Deepid, ArcFace, Sface, DeepFace, and Ensemble range from 0.10 to 0.31, reflecting varying levels of accuracy across these models.

Further, Table 4 compares the highest accuracy obtained from each model from these research results and the accuracy of the models previously studied by the creators. The accuracy obtained is lower than what is been declared for models, this occurred because there were differences in the type of dataset used where models used the LFW dataset for the training process as this study used a celebrity dataset of varying ages and ethnicity. This condition has occurred in previous studies where racial differences in the dataset affected the level of accuracy [7, 13]. Ensemble model shows 98.8 accuracy.

The time taken by various facial recognition models (VGG-Face, Facenet, Facenet512, Deepid, Arc Face, Sface, Deepface, Ensemble) as a function of sample size, ranging from 10 to 150 samples is shown in Figure 6. All models show increasing time with larger sample sizes. The Ensemble model is the most time-efficient across all sample sizes, while models like

Table 1. Cosine metric accuracy

Detector	FaceNet512	Face Net	VGG-Face	ArcFace	SFace	DeepFace	DeepID
retinaface	98.4	96.4	95.8	96.6	92.4	67.7	64.4
mtcnn	97.6	96.8	95.9	96	90.5	66.3	63
fastmtcnn	98.1	97.2	95.8	96.4	90	67.4	63.6
dlib	97	92.6	94.5	95.1	69.8	66.5	58.7
yolov8	97.3	95.7	95	95.5	91.9	67.5	65.9
yunet	97.9	97.4	96	96.7	91	66.5	63.5
centerface	97.7	96.8	95.7	96.5	89.3	67.8	63.6
mediapipe	96.1	90.6	92.9	90.3	75.4	64.8	63
ssd	88.7	87.5	87	86.2	84.5	63.8	62.6
opencv	87.6	84.9	87.2	84.6	83.6	63.7	60.1

Table 2. Euclidean metric accuracy

Detector	FaceNet512	FaceNet	VGG-Face	ArcFace	SFace	DeepFace	DeepID
Retinaface	95.9	93.5	95.8	85.2	80.2	67	65.6
MTCNN	95.2	93.8	95.9	83.7	77.4	66.5	63.5
FastMTCNN	96	93.4	95.8	83.5	77.7	66.7	64
Dlib	96	90.8	94.5	88.6	66.3	63.4	60.4
Yolov8	94.4	91.9	95	84.1	73.4	69	66.5
Yunet	97.3	96.1	96	84.9	79.4	65.8	65.2
CenterFace	97.6	95.8	95.7	83.6	77.4	65.5	62.8
MediaPipe	95.1	88.6	92.9	73.2	72.5	61.8	62.2
SSD	88.9	85.6	87	75.8	76.9	63.4	62.5
OpenCV	88.2	84.2	87.3	73	81.1	65.5	59.6

Table 3. Cosine Similarity distance

Sample	FaceNet512	FaceNet	VGG-Face	ArcFace	SFace	DeepFace	DeepID	Ensemble
20	0.19	0.12	0.46	0.35	0.29	0.11	0.12	0.12
50	0.17	0.2	0.42	0.31	0.22	0.2	0.27	0.2

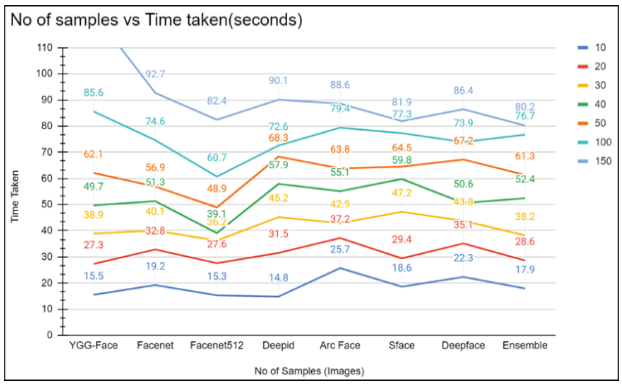

Figure 6. Measured and Declared Accuracy Comparision

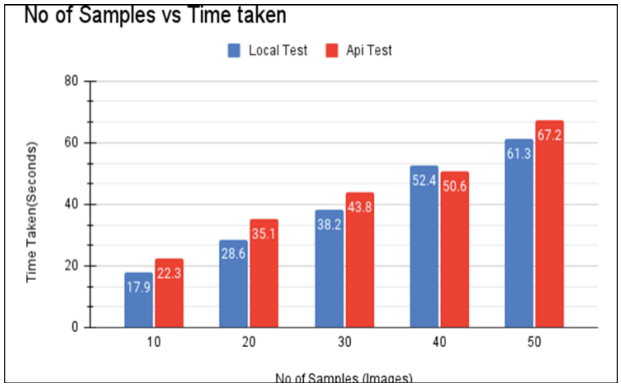

Figure 7. Time Taken local and cloud environment.

VGG-Face and Deepid have steeper time increases, indicating lower scalability.

Figure 7 compares the time taken for processing varying numbers of samples (from 10 to 50) between Local Tests and Cloud API Tests. As the number of samples increases, both testing methods show a gradual rise in processing time. Local Tests consistently take less time than API Tests for the same number of samples, indicating higher efficiency. For instance, at 50 samples, the Local Test completes in 61.3 seconds while the API Test takes 67.2 seconds, highlighting the faster processing of the Local Test across all sample sizes.

Figure 8 depicts responses by various facial recognition models (VGG-Face, Facenet, Facenet512, Deepid, Arc Face, Sface, Deepface, Ensemble) across different sample sizes from 10 to 50 images. The majority of the models consistently deliver between two and three responses, regardless of sample size. Interestingly, Facenet512 reaches a maximum of three responses at the smallest sample size, highlighting an exceptional case in its response pattern.

Table 4. Measure and Declared accuracy

Detector	FaceNet512	FaceNet	VGG-Face	ArcFace	SFace	DeepFace	DeepID	Ensemble
Measured	98.4	97.4	96.7	96.7	93	69	66.5	98.80
Declared	99.6	99.2	98.9	99.5	99.5	97.3	97.4	NA

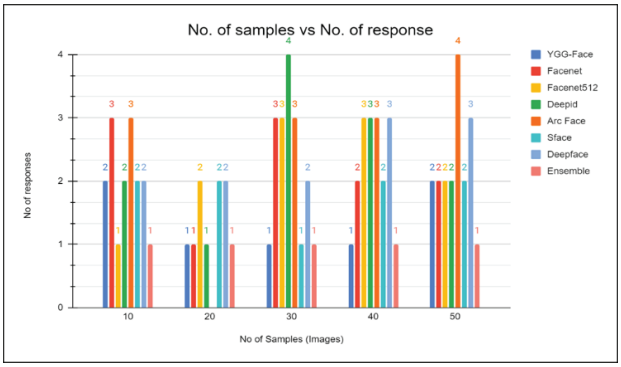

Figure 8. Time Taken local and cloud environment.

5. Conclusion

A system for suspect identification using facial recognition based on ensemble learning integrates seamlessly with OpenAI's advanced technologies and is supported by a robust cloud infrastructure. The comparison of the proposed ensemble model with individual models like VGG-Face, Facenet, Facenet512, Deepface, DeepID, ArcFace, and SFace uses multiple detectors and the LFW dataset with varying ages, gender, and ethnicity. Tests were conducted using multiple face detectors on each model, with the technicality of each image compared to one another. Cosine similarity distances for various facial recognition models across tests with 20 and 50 sample images highlight significant performance variations among the models. When tested, the Facenet512 model has higher accuracy than every other model, which is 0.984 or 98.4% which means it can predict actual faces at 100%. From the results of this study, we concluded that the FaceNet, VGG-Face, and ArcFace models could be used if desired as these have accuracy above 95%. Ensemble of these models shows a 98.8% accuracy. For instance, at a sample size of 50, the local test completes in 61.3 seconds, while the cloud API test takes 67.2 seconds. This highlights the faster processing speed of the local test across all sample sizes.

AUTHORS

Shilpa Chaudhari* – Dept. of CSE, M S Ramaiah Institute of Technology (Affiliated to VTU), Bangalore-560054, India, e-mail: shilpasc29@msrit.edu.

Rajarajeswari S – Dept. of CSE, M S Ramaiah Institute of Technology (Affiliated to VTU), Bangalore-560054, India, e-mail: raji@msrit.edu.

Archana Rane – K. K. Wagh Institute of Engineering Education and Research, Nashik- 422003. India, e-mail: alrane@kkwagh.edu.in.

*Corresponding author

References

[1] S.I. Serengil and A. Ozpinar, "LightFace: A Hybrid Deep Face Recognition Framework," *2020 Innovations in Intelligent Systems and Applications Conference* (ASYU), 2020, pp. 23–27; doi: 10.1109/ASYU50717.2020.9259802

[2] H. Han et al. "Matching Composite Sketches to Face Photos: A Component-based Approach," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 1, 2012, pp. 191–204; doi: 10.1109/TIFS.2012.2228856

[3] Y. Zhong et al., "SFace: Sigmoid-Constrained Hype rsphere Loss for Robust Face Recognition," *IEEE Transactions on Image Processing*, vol. 30, 2021, pp. 2587–2598; doi: 10.1109/tip.2020.3048632.

[4] D. DeepInsight, "deepinsight/insightface," GitHub, 4 Jun. 2021; https://github.com/deepinsight/insightface

[5] X. Ning et al., "Face Editing Based on Facial Recognition Features," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 15, no. 2, 2023, pp. 774-783; doi: 10.1109/TCDS.2022.3182650

[6] J. Xiang and G. Zhu, "Joint Face Detection and Facial Expression Recognition with MTCNN," In *2017 4th International Conference on Information Science and Control Engineering* (ICISCE), 2017, pp. 424–427; doi: 10.1109/ICISCE.2017.95

[7] K. Vangara et al., "Characterizing the Variability in Face Recognition Accuracy Relative to Race," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019; doi: 10.1109/CVPRW.2019.00281

[8] J. Deng et al., "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (CVPR), 2019, pp. 4690–4699; doi: 10.1109/cvpr.2019.00482.

[9] A.A. Poeloemgam et al., "Web-based Face Detection and Recognition using YOLO and Dlib," In *2023 17th International Conference on Telecommunication Systems, Services, and Applications* (TSSA), 2023, pp. 1–6; doi: 10.1109/TSSA59948.2023.10366984

[10] Q. Cao, "Vggface2: A Dataset for Recognising Faces across Pose and Age," In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition* (FG 2018) 2018, pp. 67–74; doi: 10.1109/FG.2018.00020

[11] J. Hu et al., "Rapid Face Detection in Complex Environments based on the Improved RetinaFace," In *Proceedings of the 4th International Conference on Advanced Information Science and System*, 2023, pp. 1–7; doi: 10.1145/3573834.3574552

[12] W. Wu, H. Peng, and S. Yu, "Yunet: A Tiny Millisecond-Level Face Detector," *Machine Intelligence Research*, vol. 20, no. 5, 2023, pp. 656–665; doi: 10.1007/s11633-023-1423-y

[13] J.G. Cavazos et al., "Accuracy Comparison Across Face Recognition Algorithms: Where are We on Measuring Race Bias," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 3, no. 1, 2020, pp. 101–111; doi: 10.1109/TBIOM.2020.3027269.

[14] M.S.S. Bobde and S.V. Deshmukh, "Face Recognition Technology," *International Journal of Computer Science and Mobile Computing*, vol. 3, no. 10, 2014, pp.192–202.

[15] Y.Taigman et al., "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (CVPR), 2014, pp. 1701–1708; doi: 10.1109/**CVPR**.2014.220

[16] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," *2015 IEEE Conference on Computer Vision and Pattern Recognition* (CVPR), 2015; doi: 10.1109/cvpr.2015.7298682.

[17] J. Deng et al., "Retinaface: Single-shot Multi-level Face Localisation in the Wild," *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 5203–5212; doi: 10.1109/CVPR42600.2020.00525

[18] S. Solarova et al., "Reconsidering the regulation of Facial Recognition in Public Spaces," *AI Ethics*, vol. 3, 2023, pp. 625–635; doi: 10.1007/s43681-022-00194-0

[19] M. Mortensen, "Sneaking AI through the Back Door: Constructing the Identity of Capitol Hill Rioters through Social Media Images and Facial Recognition Rechnologies," *Information, Communication & Society*, 2024, pp. 1–17; doi: 10.1080/1369118X.2024.2358164

[20] D. Utegen and B.Z. Rakhmetov, "Facial Recognition Technology and Ensuring Security of Biometric Data: Comparative Analysis of Legal Regulation Models," *Journal of Digital Technologies and Law*, vol. 1, no. 3, 2023, pp. 825–844; doi: 10.21202/jdtl.2023.36

[21] S. Gokulakrishnan et al., "An Optimized Facial Recognition Model for Identifying Criminal Activities using Deep Learning Strategy," *International Journal of Information Technology*, vol. 15, no. 7, 2023, pp. 3907–3921; doi: 10.1007/s41870-023-01420-6

[22] V. Munusamy and S. Senthilkumar, "Face Identification of Suspects Using Sequential-Deep Convolutional Neural Network," In *2024 Second International Conference on Emerging Trends in Information Technology and Engineering* (ICETITE) 2024, pp. 1–3; doi: 10.1109/ACCESS.2024.3523101

[23] N.K. Sharma et al., "Enhancing Facial Geometry Analysis by DeepFaceLandmark Leveraging ResNet101 and Transfer Learning," *International Journal of Information Technology*, 2024, pp. 1–21; doi: 10.1007/s41870-024-01872-4

[24] L. Yu et al.,, "Facial Expression Recognition Based on Improved VGG-face Model and Transfer Learning," In *Proceedings of the 2023 International Conference on Computer, Vision and Intelligent Technology*, 2023, pp. 1–7; doi: 10.1145/3627341.3630376

[25] K.L. Sailaja et al., "Facial Detection and Recognition in Drone Imagery Using FaceNet," In *International Conference on Advances in Distributed Computing and Machine Learning*, 2024, pp. 183–197; doi: 10.1007/978-981-97-1841-2_13

[26] M. Gulhane et al., "Advancing Facial Recognition: Enhanced Model with Improved Deepface Algorithm for Robust Adaptability in Diverse Scenarios," In *2023 10th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering* (UPCON), vol. 10, pp. 1384–1389; doi: 10.1109/UPCON59197.2023.10434721

[27] S. Srinivas and M.P. Selvan, "E-CNN-FFE: An Enhanced Convolutional Neural Network for Facial Feature Extraction and Its Comparative Analysis with FaceNet, DeepID, and LBPH Methods," In *International Conference on Data Management, Analytics & Innovation*, 2024, pp. 339–354; doi: 10.1007/978-981-97-3245-6_23

[28] M.A. Altaha et al., "Facial Expression Recognition based on ArcFace Features and TinySiamese Network," In *2023 International Conference on Cyberworlds* (CW), pp. 24–31; doi: 10.35784/jcsi.7973