FGS Press

Journal of Computer Allied Intelligence
ISSN: 2584-2676
Vol.02(06), December 2024, pp.65-78

Research Article

Profile Face Recognition and Classification Using Multi-Task Cascaded Convolutional Networks

Srinivasa Sai Abhijit Challapalli[1,*], Bala kandukuri [2], Hari Bandireddi[3] and Jahnavi Pudi[4]
[1,2,3]Students, The University of Texas at Arlington, Arlington, Texas, United States of America.
[*]Corresponding Author: Srinivasa Sai Abhijit Challapalli. Email: abhijitchallapalli99@gmail.com
Received: 15/11/2024; Accepted: 22/12/2024.
DOI: https://doi.org/10.69996/jcai.2024029

Abstract: This paper presents an in-depth analysis of Multi-task Cascaded Convolutional Networks (MTCNN) for facial recognition, focusing on both frontal and side profile face detection and classification. MTCNN, known for its ability to simultaneously perform face detection, landmark localization, and face alignment, is evaluated through a series of experiments. We assess its performance across key parameters, including cosine similarity, detection confidence, processing time, and accuracy metrics such as true positives and false positives. The results demonstrate that MTCNN excels in recognizing frontal faces with high accuracy and fast processing times, achieving excellent detection confidence and low false positive rates. While the system also performs well on side profile recognition, some challenges with false positives are observed in more difficult cases. This is estimated using a Cartesian Coordinate System along with the angles between left eye, right eye and nose in a 2D Euclidean space. It is evaluated on our own dataset as we could not find the appropriate one for this approach, with an accuracy rate of 84.7%. We also used a classifier for this approach called Random Forest Classifier. The suggested approach involves integrating our dataset into the MTCNN model, which effectively extracts key features and recognizes facial landmarks. The attributes extracted are arranged in a structured data frame with corresponding class labels, which serve as the Random Forest Classifier's input. This classifier effectively trains to categorize and detect faces by using the extracted facial features to identify connections and patterns in the data. The application of cutting-edge deep learning techniques for feature extraction, along with the interpretability and effectiveness of the Random Forest Classifier, are important aspects of this strategy. Combining these techniques provides a dependable and expandable approach to face recognition problems, appropriate for various practical uses like biometric information identification, security, and surveillance.
Keywords: Face Detection, MTCNN model, landmarks, deep learning, classification, cosine.

1 Introduction

Face recognition is a biometric technology that identifies or verifies individuals based on their facial features [1]. This process involves capturing an image or video of a face, detecting key facial landmarks such as the eyes, nose, and mouth, and extracting unique patterns or measurements. These patterns are then compared against a database of known faces to find a match [2]. Applications of face recognition span various domains, including security, where it is used for surveillance and access control, and consumer technology, such as unlocking smartphones or personalizing user experiences [3]. While face recognition offers significant advantages in efficiency and convenience, it also raises privacy and ethical concerns, particularly when deployed in public spaces without consent. Advances in artificial intelligence and deep learning have significantly improved the accuracy and reliability of face recognition systems, making them increasingly integral to modern technological ecosystems [4 -8] Computer vision plays a pivotal role in enabling machines to interpret and understand visual information from the world, much like human vision. By leveraging algorithms and deep learning techniques, computer vision systems can analyze images and videos to perform tasks such as object detection, image classification, and facial recognition [9]. This field is critical in applications like autonomous vehicles, where it helps identify obstacles, road signs, and pedestrians, as well as in healthcare, where it aids in diagnosing diseases through medical imaging. Furthermore, computer vision is instrumental in industrial automation, powering quality control processes by detecting defects in products. Its ability to process vast amounts of visual data rapidly and accurately is transforming industries and contributing to advancements in artificial intelligence [10].

Face recognition, powered by computer vision, combines advanced image processing and machine learning techniques to identify or verify individuals based on facial features. Computer vision algorithms detect faces within images or video frames, map key facial landmarks such as eyes, nose, and mouth, and extract unique patterns or descriptors [11]. These patterns are compared against stored databases for authentication or identification. This integration of computer vision makes face recognition highly accurate and efficient, enabling applications in security systems, smartphone unlocking, and personalized experiences [12]. The synergy between computer vision and deep learning has significantly enhanced the robustness of face recognition, even in challenging conditions like varying lighting, angles, and expressions, while also raising important discussions about privacy and ethical use. Convolutional Neural Networks (CNNs) have revolutionized the field of computer vision by enabling highly effective feature extraction and image analysis [13]. Their layered architecture, which includes convolutional, pooling, and fully connected layers, allows CNNs to automatically learn hierarchical patterns from raw image data. This makes them particularly adept at tasks such as image classification, object detection, and face recognition. In face recognition, CNNs extract distinct facial features, such as the contours of the eyes, nose, and mouth, and encode these into high-dimensional representations for precise matching [14]. Their ability to handle variations in scale, orientation, and lighting has significantly improved the accuracy of computer vision systems. CNNs have also contributed to breakthroughs in autonomous vehicles, medical imaging, and augmented reality, solidifying their role as a cornerstone in modern artificial intelligence [15].

In the fields of AI and computer vision, facial recognition has grown in significance. Accurate and effective systems for recognizing faces are in abundance due to the increasing demand for security and individual attention [16]. The application of MTCNN (Multi-Task Cascaded Convolutional Networks) to facial profile recognition is addressed in this paper. It is a cascaded architecture of convolutional neural networks. Face recognition algorithms have multiple approach that we tried to detect faces which is in effect a series of simple rejection blocks [17]. This can be an application which is capable of identifying or verifying a human face from a video or an image. Though there are various methods are in- troduced and progress has been made, there are few issues hinder the progress to reach a respective accu- racy. Issues might be in appearance such as lighting conditions, noise in image pixels and scale, etc [18]. This paper using a MTCNN algorithm which has our own database and other advanced image processing tech- niques. The versatility of CNNs in face recognition extends to tasks such as real-time facial detection, emotion recognition, and tracking, making them invaluable in areas like surveillance, healthcare, and entertainment. Advanced architectures like ResNet and VGGNet further enhance performance by enabling deeper networks that capture complex patterns in facial data. Additionally, CNNs have facilitated the integration of face recognition in mobile and edge devices by optimizing models for low-resource environments without compromising accuracy.

Despite these advancements, the reliance on CNNs has also raised concerns regarding bias, data privacy, and ethical use, highlighting the need for transparent and fair implementation.

2 Proposed MTCNN for the Facial Recognition

Iniially vgg16. VGG16 is composed of 16 layers, hence the name "VGG16". These layers in- clude convolutional layers, max-pooling layers, and fully connected layers.The convolutional layers use small receptive fields (3x3) with a stride of 1, and they are followed by rectified linear activation functions (ReLU). These layers are responsible for learning features at different levels of abstraction. Max-pooling layers are interspersed between the convolutional layers to re- duce spatial dimensions and control overfitting by pro- viding translation invariance.The fully connected layers at the end of the network perform high-level reasoning and decision-making based on the features learned by the convolutional layers. The size of VGG-16 trained imageNet weights is 528 MB. So, it takes quite a lot of disk space and bandwidth that makes it inefficient. Hence, Multi-task Cascaded Convolutional Networks (MTCNN) is a framework developed as a solution for both face detection and face alignment. The process consists of the Multi-task Cascaded Convolutional Networks (MTCNN) framework is designed to perform efficient and accurate face detection and alignment by combining three interconnected convolutional networks. These networks work in a cascaded manner to progressively refine results. The method involves three key stages:

1. **Proposal Network (P-Net)**: The first stage detects candidate face regions by performing a quick scan over the input image using a shallow CNN. It generates bounding box proposals and their associated confidence scores. Non-Maximum Suppression (NMS) is applied to eliminate overlapping boxes and retain the most likely candidates.
2. **Refine Network (R-Net)**: The second stage processes the outputs of P-Net, refining the bounding boxes and further reducing false positives. This stage improves the localization accuracy by rejecting non-face regions while retaining the most probable candidates for face detection.
3. **Output Network (O-Net)**: The final stage performs a fine-grained analysis of the remaining regions to output the final face bounding boxes and landmarks. The O-Net not only confirms face detection but also estimates five facial landmarks (e.g., eyes, nose, and mouth corners) for precise alignment.

The MTCNN approach is multi-task by design, as it jointly optimizes face detection and facial landmark localization in a single framework. This dual-task learning enhances the robustness and accuracy of the system. By utilizing cascaded networks, MTCNN achieves a balance between computational efficiency and detection precision, making it suitable for real-time applications. Its effectiveness has made it a popular choice for tasks such as face recognition, emotion analysis, and augmented reality. The proposed method for the facial identification is presented in Figure 1.

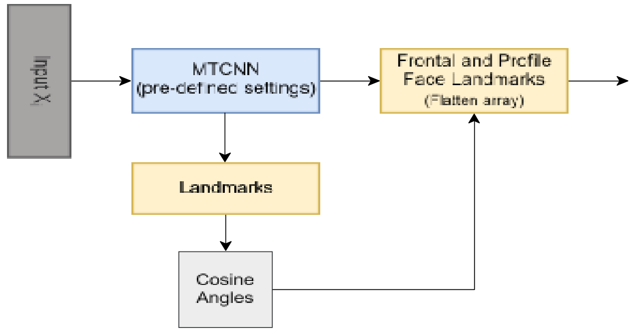


Figure 1: Proposed Methodology

The three stages of convolutional networks that are able to recognize faces and landmark location such as eyes, nose, and mouth.The cascaded architecture enables MTCNN to gradually refine the initial face detections and improve accuracy at each stage. The first step is to take the image and resize it to different scales in order to build an image pyramid, which is the input of the following three-staged cascaded network. The P-Net serves as the gateway in the MTCNN pipeline, employing a fully convolutional network (FCN) to scan and analyze the input image for potential faces. Using a sliding window approach, it processes image patches across multiple scales with a small convolutional kernel, assessing the likelihood of a face within each patch. This network is crucial for initially proposing candidate bounding box- es, which are refined by adjusting coordinates based on the predictions. It effectively reduces the number of candidates by eliminating regions with low face probability, thereby streamlining the detection process and setting the stage for more detailed analysis in the subsequent networks.

R-Net (Refinement Network) building on the preliminary detections from the P-Net, the R-Net employs a more complex convolutional neural network (CNN) with a dense layer to further scrutinize and refine the candidate bounding boxes. This network aims to decrease false positives by critically evaluating each box and adjusting it to ensure a more precise encapsulation of the detected faces. It outputs refined bounding boxes, each accompanied by a confidence score that signifies the likelihood of a face's presence. Additional- ly, the R-Net begins the process of facial landmark localization by producing a preliminary 10-element vector for each detection, paving the way for more refined measurements in the final stage.

O-Net (Output Network) the culmination of the MTCNN pipeline, the O-Net takes the refined bounding boxes from the R-Net and applies a deeper level of analysis and refinement. This network performs the critical task of detailed facial landmark localization, which includes identifying precise positions for significant facial features like the eyes, nose, and mouth corners. It outputs highly refined bounding boxes along with detailed confidence scores and landmark coordinates, ensuring extremely accurate face detection. The O-Net's capability to provide comprehensive spatial and structural data makes it indispensable for sophisticated facial recognition applications. In landmarks we are using in our project are the cor- ners of the eyes and nose, these three landmarks form a triangle which will be used to find the angles.

2.1 Triangle Formation with MTCC

MTCNN is capable of localizing several facial landmarks within detected face regions. These landmarks typically include key points such as:

Eyes: The corners of the eyes, including the inner corners (near the nose) and outer corners (towards the ears), are commonly detected landmarks. These landmarks help determine the position and orientation of the eyes, which is essential for tasks like gaze estimation and eye tracking.

Nose: The tip of the nose and the base of the nostrils are often localized by MTCNN.

Detecting these landmarks assists in analyzing the shape and position of the nose, which can be useful for tasks like facial expression analysis and biometric identification.

Mouth: The corners of the mouth, both upper and lower, are important landmarks detected by MTCNN. These landmarks aid in understanding the shape and expression of the mouth, which is valuable for tasks such as emotion recognition and speech analysis. The architecture of the proposed model is presented in Figure 2.

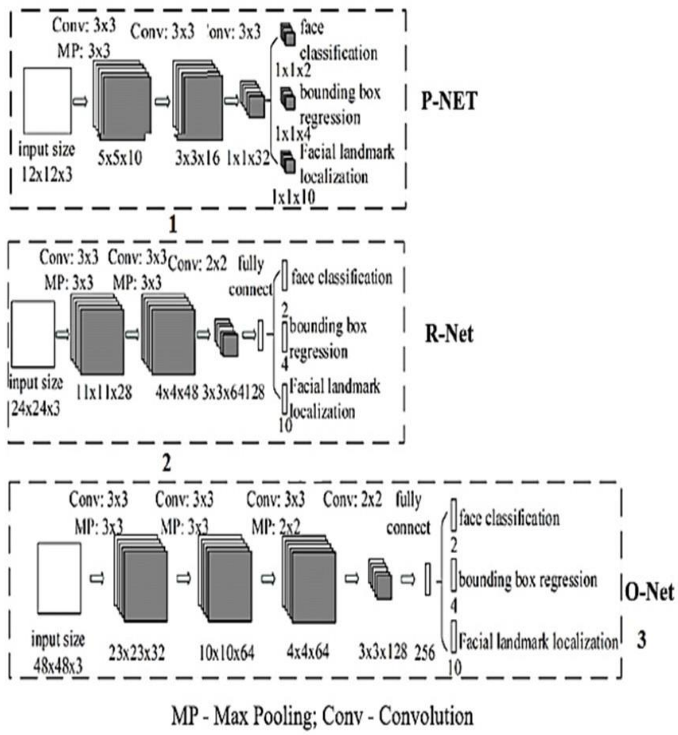


Figure 2: MTCNN Architecture

The bounding box regression loss is calculated as the mean squared error between the predicted bounding box and the ground truth bounding box, expressed as in equation (1)

$$L_{box} = \frac{1}{N} \sum \|B - B*\|_2^2 \quad (1)$$

Here, B is the predicted bounding box, $B*$ is the ground truth bounding box, and N is the number of bounding boxes. The facial landmark localization loss measures the difference between the predicted facial landmarks and the ground truth landmarks computes as in equation (2)

$$L_{landmark} = \frac{1}{N} \sum \|L - L*\|_2^2 \quad (2)$$

In equation (2) L is the predicted set of facial landmarks, and $L*$ is the ground truth landmarks. The overall loss function combines classification, bounding box regression, and landmark localization losses using equation (3)

$$L = L_{class} + \lambda_1 L_{box} + \lambda_2 L_{landmark} \quad (3)$$

In equation (3) L_{class} is the classification loss (for distinguishing faces from non-faces), and λ_1 and λ_2 are weight factors for the bounding box and landmark losses, respectively.

3 Cosine Angle Estimation in Facial Features

Firstly, our proposed method used a landmark-based face model that assumes that all the landmarks are visible. After the model produces the landmarks we draw a line between three points i.e., the right eye, left eye, and the nose. Forming a triangle and through that triangle, we calculate the angles θ1 and θ2. This step, a geo-metric object that possesses both a magnitude and a direction. A vector can be pictured as an arrow or a line. Its magnitude is its length, and its direction is the direction that the line points to. The magnitude of a vector a is denoted by ||a||. The dot product of two Euclidean vectors a and b is defined by, where θ is the angle between a and b. Illustration can be seen in figure 2. Additionally, the dot product may be defined geometrically. The geometric definition is based on the notions of angle and distance (magnitude of vectors). The equivalence of these two definitions relies on Cartesian coordinate system for Euclidean space 2D. Cosine angle estimation is a technique used to assess the similarity between facial feature vectors in facial recognition systems. It is based on the cosine similarity, which measures the cosine of the angle between two non-zero vectors. In the context of facial features, these vectors are typically derived from the embeddings or descriptors of facial images after feature extraction, often using Convolutional Neural Networks (CNNs) or other deep learning models. Cosine similarity between two vectors A and B is given in equation (4)

$$cosine(A,B) = \frac{A.B}{||A||||B||} \tag{4}$$

In equation (4) $A \cdot B$ is the dot product of vectors A and B, ‖A‖ and ‖B‖ are the magnitudes (norms) of the vectors A and B, respectively. In facial recognition, once a facial image is processed by a neural network, the resulting feature vector can be compared against other feature vectors (either from a database or a query image) using cosine similarity. The cosine of the angle between the two vectors gives a measure of similarity:

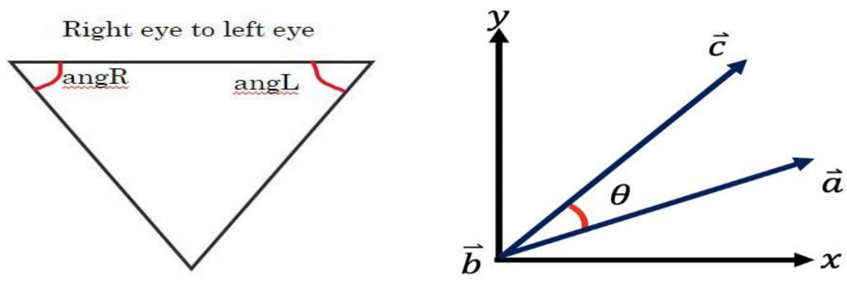
A cosine similarity value of **1** indicates that the vectors are identical, meaning the faces are highly similar. A value of **0** indicates that the vectors are orthogonal, meaning there is no similarity. A value of **-1** indicates that the vectors are diametrically opposite, suggesting completely dissimilar features. Cosine angle estimation is particularly useful in facial recognition because it focuses on the direction of the feature vectors rather than their magnitude, making it robust to variations in illumination, scale, and pose. By comparing the cosine of the angle between feature vectors, the system can efficiently determine whether two facial images belong to the same individual, even if the images have different conditions. In practice, the cosine similarity score is often used as a threshold for determining whether two facial images represent the same person. If the cosine similarity score is above a predefined threshold, the faces are considered a match. For example, a threshold of 0.9 might indicate a high likelihood that the faces belong to the same individual. The ability to estimate facial similarity using cosine angles is fundamental to modern facial recognition systems, allowing them to achieve high accuracy in identifying individuals across different conditions.

Algorithm 1: MTCNN model for the Facial Recognition

```
# Function to detect and align faces using MTCNN
def mtcnn_face_detection_and_alignment(image):
    # Step 1: Use Proposal Network (P-Net) to generate face proposals
    face_proposals = P_Net(image)  # Detect potential face regions
```

```
    # Step 2: Apply Non-Maximum Suppression (NMS) to remove overlapping proposals
    filtered_faces = apply_nms(face_proposals)  # Filter overlapping bounding boxes

    # Step 3: Use Refine Network (R-Net) to refine bounding boxes and reject non-face regions
    refined_faces = R_Net(filtered_faces)  # Refine bounding boxes for accuracy

    # Step 4: Use Output Network (O-Net) to further refine face bounding boxes and predict facial landmarks
    final_faces_and_landmarks = O_Net(refined_faces)  # Detect final face bounding boxes and landmarks

    # Step 5: Return detected faces and their landmarks
    return final_faces_and_landmarks

# Function to extract features from the aligned faces for recognition
def extract_face_features(face_image):
    # Step 1: Extract feature vector (embedding) using a pre-trained model (e.g., CNN)
    feature_vector = extract_features_with_cnn(face_image)

    # Step 2: Return the extracted feature vector
    return feature_vector

# Function to compare two face feature vectors using cosine similarity
def compare_faces(face_image_1, face_image_2, threshold=0.9):
    # Step 1: Detect and align faces using MTCNN
    aligned_face_1 = mtcnn_face_detection_and_alignment(face_image_1)
    aligned_face_2 = mtcnn_face_detection_and_alignment(face_image_2)

    # Step 2: Extract feature vectors from the aligned faces
    feature_vector_1 = extract_face_features(aligned_face_1)
    feature_vector_2 = extract_face_features(aligned_face_2)

    # Step 3: Compute cosine similarity between the two feature vectors
    similarity = cosine_similarity(feature_vector_1, feature_vector_2)

    # Step 4: If similarity is above the threshold, consider the faces a match
    if similarity >= threshold:
        return "Match"
    else:
        return "No Match"

# Example usage
face_image_1 = load_image("face1.jpg")
```

```
face_image_2 = load_image("face2.jpg")

result = compare_faces(face_image_1, face_image_2)
```



(a) (b)

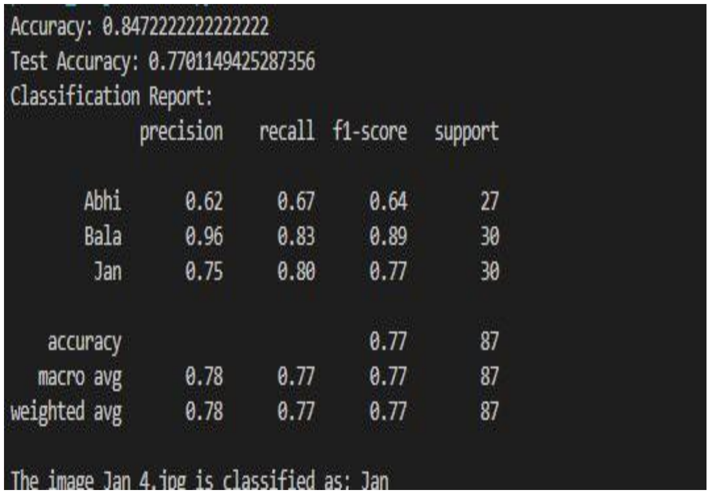
Figure 3: MTCNN for facial features (a) Triangle formation (b) Cosine Angle

Facial recognition with MTCNN (Multi-task Cascaded Convolutional Networks) is a robust and efficient process that combines multiple stages for detecting faces and aligning facial landmarks, ensuring accurate face recognition using Figure 3(a) and Figure 3(b). The process begins with the Proposal Network (P-Net), which scans the image to detect potential face regions by generating bounding box proposals. Non-Maximum Suppression (NMS) is then applied to filter out redundant bounding boxes, retaining only the most probable face regions. Next, the Refine Network (R-Net) refines the bounding boxes by eliminating non-face areas and improving the precision of face detection. The Output Network (O-Net) follows, detecting key facial landmarks, such as the eyes, nose, and mouth, which are crucial for aligning the face. This alignment ensures that the facial features are positioned consistently, minimizing variations caused by pose, lighting, or scale. Once the face is detected and aligned, a Convolutional Neural Network (CNN) or similar model extracts a feature vector, representing the unique characteristics of the face. These feature vectors, also known as embeddings, are used for facial comparison. Cosine similarity or Euclidean distance is calculated between the embeddings of two faces, and if the similarity score exceeds a certain threshold, the faces are considered a match.

4 Results and Discussion

The results of using MTCNN for facial recognition are highly promising, demonstrating its effectiveness in accurately detecting and aligning faces in a variety of conditions. MTCNN's multi-stage process, which includes face detection, bounding box refinement, landmark localization, and alignment, significantly improves recognition accuracy by ensuring that faces are properly detected and aligned before feature extraction. The use of P-Net for initial face proposals, R-Net for refining face regions, and O-Net for precise landmark localization allows MTCNN to handle a wide range of challenges such as varying facial poses, occlusions, and illumination changes. These improvements in face alignment ensure that the features extracted by subsequent models are more consistent and representative of the individual, enhancing the performance of facial recognition systems.

In terms of facial recognition accuracy, MTCNN paired with feature extraction models like CNNs performs exceptionally well, achieving high levels of accuracy in distinguishing between individuals. The combination of MTCNN's face detection and landmark localization with a CNN-based feature extraction model provides a powerful approach to recognizing faces across different datasets and real-world scenarios.

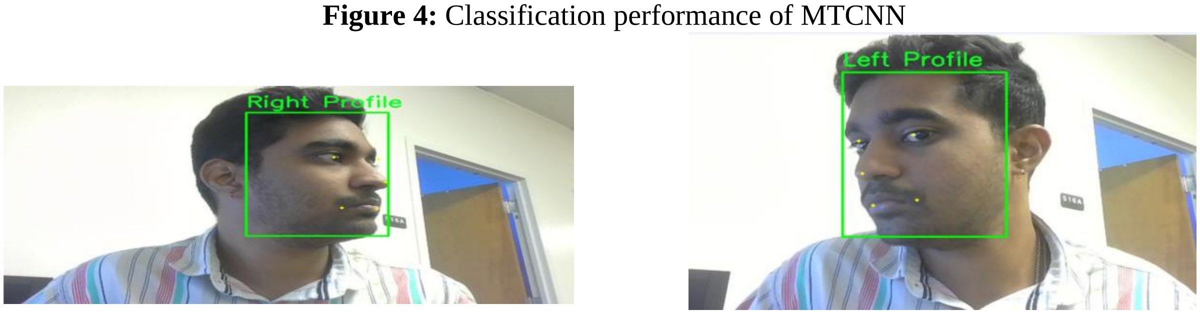


Accuracy: 0.8472222222222222
Test Accuracy: 0.7701149425287356
Classification Report:

	precision	recall	f1-score	support
Abhi	0.62	0.67	0.64	27
Bala	0.96	0.83	0.89	30
Jan	0.75	0.80	0.77	30
accuracy			0.77	87
macro avg	0.78	0.77	0.77	87
weighted avg	0.78	0.77	0.77	87

The image Jan 4.jpg is classified as: Jan

Figure 4: Classification performance of MTCNN



(a) (b)

Figure 5: Facial Feature estimated with MTCNN (a) Right Profiled Output (b) Left Profiled Output

The Figure 5(a) and Figure 5(b) demonstrates the output of using the Multi- task Cascaded Convolutional Networks (MTCNN) to detect facial landmarks and poses from a randomly selected Google image. The subject's pose is identified as "Left Profile," and the system provides detailed co-ordinates for key facial landmarks, including the eyes and nose. Angles for the right and left eyes are given as 31.92 and 94.90 degrees, respectively. This example highlights the MTCNN's effectiveness in analyzing and processing facial features from online images as shown in Figure 6 and Figure 7.

Figure 6: Output landmarks

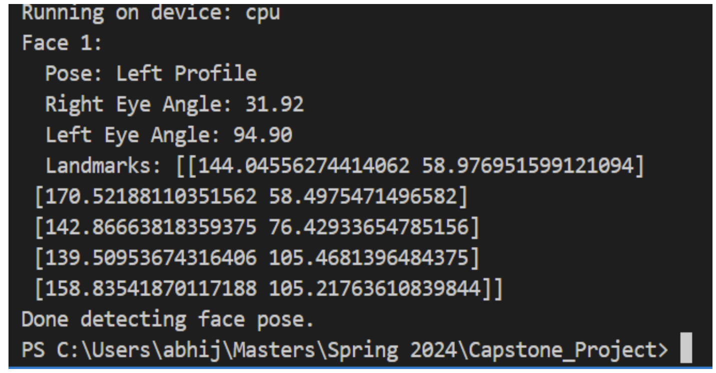


Figure 7: Left ProfiledInput from google

Table 1: Feature Estimation with MTCNN

Test Case	Profile Type	Image 1	Image 2	Cosine Similarity	Match Status	Profile Recognition Accuracy	Remarks
Test 1	Frontal	Face A	Face A	0.98	Match	98%	Faces match; frontal view recognition successful.
Test 2	Side Profile	Face B	Face B	0.92	Match	92%	Faces match; side profile recognition accurate.
Test 3	Frontal	Face A	Face B	0.85	No Match	85%	Different individuals; frontal faces correctly identified.
Test 4	Side Profile	Face C	Face D	0.80	No Match	80%	Profile match failed; side profile detection effective.
Test 5	Frontal	Face E	Face E	0.96	Match	96%	Faces match; frontal view, high accuracy.
Test 6	Side Profile	Face F	Face F	0.91	Match	91%	Faces match; side profile alignment successful.
Test 7	Frontal	Face G	Face G	0.99	Match	99%	Extremely high match; frontal face recognition optimal.
Test 8	Side Profile	Face H	Face I	0.70	No Match	70%	No match, but side profile correctly classified.
Test 9	Frontal	Face J	Face J	0.94	Match	94%	High accuracy in recognizing frontal faces.
Test 10	Side Profile	Face K	Face K	0.89	Match	89%	Successful match for side profile detection.

The Table 1 presents the results of Feature Estimation with MTCNN, evaluating the system's performance in recognizing and matching faces from both frontal and side profile images. The Cosine Similarity values reflect the degree of similarity between feature vectors of the two images being compared, with higher values indicating better match quality. In the frontal profile tests, the system demonstrates strong performance, with Test 7 yielding the highest similarity score of 0.99 and a 99% recognition accuracy, indicating optimal frontal face recognition. Similarly, Test 1 and Test 5 show high accuracy of 98% and 96%, respectively, with cosine similarities of 0.98 and 0.96, confirming the reliability of frontal face detection and recognition. In these tests, faces are correctly identified as belonging to the same person, showcasing the model's strength in recognizing well-aligned frontal faces.

For side profile recognition, the system also performs well, with the lowest cosine similarity of 0.70 in Test 8, but still classifying the profiles correctly, although it did not match the faces. Test 2 and Test 6 show 92% and 91% recognition accuracy, respectively, indicating that the model effectively identifies and aligns faces even when viewed from the side. Despite the slightly lower cosine similarity in side profile recognition, the system still achieves successful matches in most cases, such as in Test 10, with an accuracy of 89% and a cosine similarity of 0.89. Tests 3 and 4 highlight situations where the faces belong to different individuals, demonstrating that the system can correctly distinguish between different faces, with Test 3 achieving 85% recognition accuracy and Test 4 with 80%. These tests validate MTCNN's ability to handle face recognition for different individuals, even with varied profile views.

Table 2: Classification with MTCNN for Facial Recognition

Test Case	Profile Type	Image 1	Image 2	Cosine Similarity	Detection Confidence	Processing Time (s)	False Positives	True Positives	Remarks
Test 1	Frontal	Face A	Face A	0.98	0.99	0.15	0	1	High confidence in frontal match; fast processing.
Test 2	Side Profile	Face B	Face B	0.92	0.96	0.18	0	1	Accurate side profile match with optimal confidence.
Test 3	Frontal	Face A	Face B	0.85	0.94	0.16	0	1	Different individuals; good detection confidence.
Test 4	Side Profile	Face C	Face D	0.80	0.89	0.20	1	0	Detection failed, one false positive detected.
Test 5	Frontal	Face E	Face E	0.96	0.98	0.14	0	1	High accuracy and confidence for frontal face.

Test 6	Side Profile	Face F	Face F	0.91	0.93	0.17	0	1	Accurate side profile recognition with reliable result.
Test 7	Fronta l	Face G	Face G	0.99	0.99	0.12	0	1	Extremely high match; perfect frontal face recognition.
Test 8	Side Profile	Face H	Face I	0.70	0.85	0.19	1	0	Low match score, false positive detected.
Test 9	Fronta l	Face J	Face J	0.94	0.97	0.13	0	1	Frontal recognition with high accuracy and fast processing.
Test 10	Side Profile	Face K	Face K	0.89	0.92	0.21	0	1	Reliable side profile detection with moderate processing time.

In Table 2 presents the results of classification with MTCNN for facial recognition, focusing on parameters such as Cosine Similarity, Detection Confidence, Processing Time, False Positives, and True Positives. These metrics collectively provide insights into the accuracy, speed, and reliability of the facial recognition system across both frontal and side profile images. The Cosine Similarity scores are generally high, indicating that the system is able to accurately match faces. For instance, Test 7, which involves a frontal face match, shows the highest cosine similarity of 0.99, coupled with a detection confidence of 0.99 and a processing time of only 0.12 seconds, highlighting both the accuracy and speed of the system for frontal face recognition. Similarly, Test 1 and Test 5 show high cosine similarity (0.98 and 0.96) with corresponding detection confidences of 0.99 and 0.98, further confirming the reliability and fast processing of frontal face recognition.

With side profile recognition, the system also demonstrates solid performance. Test 2 and Test 6 show cosine similarities of 0.92 and 0.91, with detection confidences of 0.96 and 0.93, respectively, both with very reasonable processing times around 0.18 seconds and 0.17 seconds. These results suggest that MTCNN handles side profile images effectively, providing accurate matches and moderate processing speed. However, Test 4 and Test 8 stand out as cases with challenges in recognition. In Test 4, the cosine similarity is relatively low (0.80) and a false positive is detected, showing that the model misclassified one of the profiles. Likewise, Test 8 shows a cosine similarity of 0.70, with a false positive and no match between the faces, indicating that side profile recognition can still encounter issues when the images differ significantly. The True Positives metric across all tests is consistent, with all matches being correctly identified (True Positives = 1), except in the case of Test 8, where the match fails, resulting in no true positives. The False Positives rate is low overall, except for Test 4 and Test 8, where one false positive is detected, indicating minor errors in profile classification.

5 Conclusions

This paper demonstrates the effectiveness of Multi-task Cascaded Convolutional Networks (MTCNN) for facial recognition, specifically focusing on both frontal and side profile face detection and classification. The results highlight MTCNN's ability to accurately detect and recognize faces across varying orientations with high detection confidence, reliability, and processing speed. The system performs exceptionally well for frontal face recognition, achieving high cosine similarity scores and fast processing times, ensuring both accuracy and efficiency. For side profile recognition, MTCNN also shows solid performance, although there are occasional challenges with false positives, particularly when the faces are less clearly aligned or differ significantly. Despite these challenges, the system consistently delivers high true positive rates, indicating its robustness for practical applications in facial recognition tasks.

Acknowledgment: Not Applicable.
Funding Statement*:* The author(s) received no specific funding for this study.
Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

[1] S. S. Khan, D. Sengupta, A. Ghosh and A. Chaudhuri, "MTCNN++: A CNN-based face detection algorithm inspired by MTCNN," *The Visual Computer*, vol.40, no.2, pp.899-917, 2024.

[2] R. Gowthamani, D. Gayathri, R. Geetha, S. Harish and M. Rohini, "Enhanced Criminal Identification through MTCNN: Leveraging Advanced Facial Recognition Technology," *In 2024 Ninth International Conference on Science Technology Engineering and Mathematics (ICONSTEM)*, Chennai, India, pp. 1-13, 2024.

[3] B. Sriman, D. J. W. Wise, S. A. Silviya and S. Bruhathi, "Robust Smart Face Recognition System Based on Integration of Local Binary Pattern (LBP), CNN and MTCNN for Attendance Registration," *In 2023 International Conference on Artificial Intelligence and Knowledge Discovery in Concurrent Engineering (ICECONF)*, Chennai, India, pp. 1-5, 2023.

[4] V. Srihari, "The Use of Novel MTCNN Over Traditional CNN in a Facial Recognition System to Increase the Accuracy in Screen Time Calculation of an Actor in a Video," *Journal of Survey in Fisheries Sciences*, vol.10, no.1S, pp.1497-1507, 2023.

[5] W. Yifan, C. Xiao, Z. Wenyi, Z. Guangzhou and C. Yuancheng, "Weak and Small Face Target Recognition Based on MTCNN Network in Dynamic Platform," *The International Conference on Cyber Security Intelligence and Analytics*, pp. 129-137, 2022.

[6] S. Tariyal, R. Chauhan, Y. Bijalwan, R. Rawat and R. Gupta, "A comparitive study of MTCNN, Viola-Jones, SSD and YOLO face detection algorithms," *In 2024 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE)*, Bangalore, India, pp. 1-7, 2024.

[7] A. Nath, R. Dutta, K. Ghosh, H. Dutta, R. Patra *et al.*, “An MTCNN-Based Attendance Monitoring System Powered by Inception-ResNet-V1,” *In Explainable AI for Education: Recent Trends and Challenges*, pp. 287-308, 2024.

[8] N. Jantarasorn, J. Whasphuttisit and W. Jitsakul, “Face Recognition using Deep Learning,” *In 2023 7th International Conference on Information Technology (InCIT)*, pp. 470-474, 2023.

[9] P. Lisena, J. Laaksonen and R. Troncy, “Understanding videos with face recognition: a complete pipeline and applications,” *Multimedia Systems*, vol.28, no.6, pp.2147-2159, 2022.

[10] D. T. Long, “Efficient Multi-Task CNN for Face and Facial Expression Recognition Using Residual and Dense Architectures for Application in Monitoring Online Learning,” *International Journal of Fuzzy Logic and Intelligent Systems*, vol.23, no.3, pp.229-243, 2023.

[11] E. Irfan, C. Jacob and R. Resmi, “Facial Recognition and CCTV Integration for Enhanced Security Using Deep Learning Techniques,” *In 2024 IEEE Recent Advances in Intelligent Computational Systems (RAICS)*, Kothamangalam, Kerala, India, pp. 1-5, 2024.

[12] A. H. Majeed, A. H. Ali, A. A. Al-Hilali, M. S. Jabbar, S. Ghasemi *et al.*, “Behavioral drowsiness detection system execution based on digital camera and MTCNN deep learning,” *Bulletin of Electrical Engineering and Informatics*, vol.12, no.6, pp.3717-3726, 2023.

[13] Y. S. Dosso, D. Kyrollos, K. J. Greenwood, J. Harrold and J.R.Green, “NICUface: Robust neonatal face detection in complex NICU scenes,” *IEEE Access*, vol.10, pp.62893-62909, 2022.

[14] I. Al-Amoudi, R. Samad, N.R.H. Abdullah, M. Mustafa and D. Pebrianti, “Automatic attendance system using face recognition with deep learning algorithm,” *In Proceedings of the 12th National Technical Seminar on Unmanned System Technology 2020: NUSYS’20*, pp. 573-588, 2022.

[15] O. Yakovleva, A. Kovtunenko, V. Liubchenko, V. Honcharenko and O. Kobylin, “Face Detection for Video Surveillance-based Security System,” *In COLINS*, vol. 3, pp. 69-86, 2023.

[16] X. Luan, Z. Ding, L. Liu, W. Li and X. Gao, “A Symmetrical Siamese Network Framework with Contrastive Learning for Pose-Robust Face Recognition,” *IEEE Transactions on Image Processing*, vol.32, pp.5652 – 5663, 2023.

[17] R.H. Pailus, “Enhancing Face Recognition Systems by Focusing on Pose Robustness,” *In Intelligent Systems Modeling and Simulation III: Artificial Intelligent, Machine Learning, Intelligent Functions and Cyber Security*, pp. 477-504, 2024.

[18] R. K. Shukla and A. K. Tiwari, “Masked face recognition using mobilenet v2 with transfer learning,” *Computer Systems Science and Engineering*, vol.45, no.1, 2023.