Comparitive analysis of Haar Cascade and MTCNN for Efficient Face Recognition

Dr Shiva Kumar C
Computer Science Engineering
Jain Deemed to be University
Bengaluru, India
k.shiva@jainuniversity.ac.in

Aere Pavan Sankar
Computer Science Engineering
Jain Deemed to be University
Bengaluru, India
21btrcs108@jainuniversity.ac.in

Agile Yaswanth Sai Simha Reddy
Computer Science Engineering
Jain Deemed to be University
Bengaluru,
21btrcs109@jainuniversity.ac.in

***Abstract*—Face detection is a fundamental task in computer vision, widely applied in security, authentication, and surveillance systems. This study compares the performance of Haar Cascade, a conventional machine learning-based approach, and MTCNN, a deep learning-based technique, using real-world image datasets and structured CSV data. The evaluation focuses on three key parameters: accuracy, detection speed, and real-time applicability. Experimental results highlight the trade-offs between computational efficiency and detection precision, offering insights into the strengths and limitations of each method. This analysis aims to assist in selecting the appropriate algorithm for various real-time face detection applications.**

Keywords—Face Detection, Haar Cascade, MTCNN, Computer Vision, Real-Time Processing, Accuracy, Speed, Machine Learning, Deep Learning, Algorithm Comparison, Image Processing, Face Net.

I. INTRODUCTION

Artificial Intelligence (AI) is revolutionizing various domains, including healthcare [1], security [2], and marketing [3], by significantly enhancing state-of-the-art applications. Intelligent systems are increasingly preferred over traditional methods due to their efficiency and automation capabilities. AI-powered software enables users to create, train, and test datasets effortlessly, offering advanced face detection and recognition solutions [10]. One such method is the Multi-Task Cascaded Convolutional Neural Network (MTCNN) [4], which processes video feeds or CCTV footage to detect faces with high accuracy. These detected faces can then be recognized using the Face Net module [5], forming the foundation of an automatic face recognition attendance system. Manually taking attendance in large classrooms is labour-intensive and time-consuming, making AI-based systems an effective alternative for seamless attendance recording.

However, facial recognition technology [12] encounters several challenges that impact its performance. Factors such as pose variation, facial hair, lighting conditions, background clutter, and facial expressions can significantly influence detection accuracy [11]. Various methods, including the Eigenfaces method, Fisher faces method, Local Binary Patterns Histograms (LBPH), and Haar Cascade, are implemented in the OpenCV library to improve face detection and identification. These techniques, when optimized, contribute to building efficient, real-time face recognition systems. In this paper we are going compare two algorithms i.e., Haar Cascades and MTCNN and present the results in terms of speed and accuracy by using real time datasets [13]. While Haar Cascade is a machine learning-based approach known for its fast execution, MTCNN leverages deep learning techniques to improve detection accuracy by addressing variations in pose, lighting, and occlusions [2]. This architecture enhances accuracy in detecting faces across various conditions but requires significantly higher computational power compared to Haar Cascade [6]. Additionally, integrating edge AI technologies has been proposed to enhance real-time processing capabilities in facial recognition systems [13]. Facial recognition technology has undergone significant advancements over the years, transitioning from traditional feature-based approaches to modern deep learning techniques. Early methods such as Eigenfaces and Fisher faces relied on Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) for facial feature extraction. However, these techniques were highly sensitive to variations in illumination, occlusions, and facial expressions, limiting their effectiveness in real-world applications [11]. The introduction of Haar Cascade brought substantial improvements by leveraging Haar-like features and an integral image approach, making face detection computationally efficient and suitable for real-time applications. Despite its efficiency, Haar Cascade often struggles with non-frontal faces and dynamic lighting conditions, necessitating more advanced solutions. Haar Cascade remains a popular choice for low-power devices due to its lightweight architecture, making it suitable for real-time applications such as automated attendance systems, surveillance, and embedded systems [9]. Its ability to detect faces quickly makes it effective in controlled environments where lighting and pose variations are minimal. However, in dynamic and unconstrained environments, MTCNN offers superior performance by accurately detecting faces under varying poses, occlusions, and complex backgrounds. This makes MTCNN a preferred choice for applications requiring high accuracy, such as advanced security systems, mobile authentication, and human-computer interaction [4]. The adoption of facial recognition technology in various sectors raises concerns regarding security, privacy, and ethical implications. The storage and processing of biometric data pose risks related to data breaches and unauthorized access. Additionally, AI-based face detection models have been found to exhibit biases, leading to disparities in recognition accuracy across different demographic groups [7]. To mitigate these challenges, researchers are exploring techniques such as differential privacy, adversarial training, and federated learning, which enhance security while preserving user privacy. Addressing these ethical considerations is crucial for ensuring the responsible deployment of facial recognition systems in real-world applications. However, in dynamic and unconstrained environments, MTCNN offers superior performance by accurately detecting faces under varying poses, occlusions, and complex backgrounds [8].

Related Work

Face recognition technology has been widely explored for attendance systems, leveraging machine learning and deep learning models to improve accuracy and efficiency.

Traditional methods, such as Haar Cascade classifiers, have been extensively used for face detection due to their computational efficiency. However, they often struggle with variations in illumination, pose, and occlusions, limiting their robustness in real-world applications [1].To address these challenges, more advanced deep learning-based methods, such as Multi-task Cascaded Convolutional Neural Networks (MTCNN), have been introduced. MTCNN performs face detection with higher accuracy by leveraging a three-stage cascaded architecture that refines face localization at each stage [2]. Studies have shown that MTCNN outperforms traditional methods like Haar Cascades in terms of precision, recall, and detection speed, particularly in complex environments [3].Several research efforts have proposed hybrid models integrating traditional and deep learning techniques to optimize detection performance. For instance, some systems combine Haar features for initial face localization and deep learning models for refined face recognition, achieving a balance between speed and accuracy [4]. Additionally, OpenCV-based solutions implementing the Local Binary Pattern Histogram (LBPH) and Fisher Faces algorithms have been explored for real-time face recognition applications, but they often fall short when dealing with large-scale datasets [5].Recent advancements in convolutional neural networks (CNNs) have led to the development of highly efficient models for facial recognition tasks. Face Net and Deep Face have demonstrated superior recognition capabilities by mapping facial features into high-dimensional embeddings, significantly improving identification accuracy over conventional methods [6]. Moreover, real-time implementations of face recognition attendance systems leveraging cloud computing and edge AI have been proposed to enhance scalability and performance in large institutions [7].Despite these advancements, challenges such as occlusions, variations in lighting, and computational complexity still persist. Researchers continue to explore hybrid and optimized models to achieve a balance between efficiency and accuracy, making face recognition-based attendance systems more reliable and practical for real-world deployment [15].

II. METHODOLOGY

This research focuses on the comparative analysis of Haar Cascade and MTCNN face detection algorithms using publicly available datasets from Kaggle. The methodology involves five major steps: Dataset Selection, Preprocessing, Face Detection, Performance Evaluation, and Result Interpretation. The *fig:1* will show the clear understanding of the methodology neatly one by one starting from the data collection to analysis. The methodology involves capturing real-time video frames, preprocessing them, and applying Haar Cascade and MTCNN for face detection. Recognized faces are compared with stored embeddings, and attendance is automatically logged in the database.

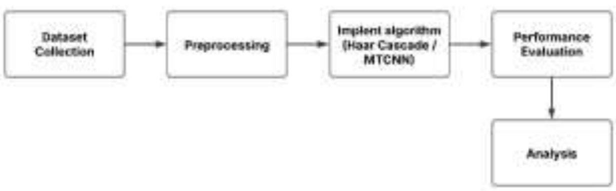

Fig:1

A. Dataset Collection

For a fair comparison, a Kaggle dataset containing labeled face images under different conditions such as varying lighting, facial expressions, occlusions, and poses was selected. This dataset includes high-resolution images, real-world face images, and video frames to test both algorithms comprehensively.

B. Preprocessing

This process is where it is applied before using face detection algorithms like Haar Cascade or MTCNN, to standardize the dataset. This process include the following steps:

a) Resizing: In this all the images were resized to fixed dimension to ensure consistency across different methods.

b) Grayscale Conversion: Since Haar Cascade relies on intensity-based features, images were converted to grayscale before detection.

c) RGB Normalisation: Unlike Haar Cascade, MTCNN requires RGB images, hence pixel values were normalised between 0 and 1 for better accuracy.

d) Noise Reduction: Gaussian filtering is applied to reduce image noise, enhancing detection accuracy for both algorithms.

e) Face Annotations: The dataset's ground truth labels are extracted for accuracy evaluation.

C. Face Detection Using Haar Cascade and MTCNN

a) Haar Cascade Approach:

The Haar Cascade classifier from OpenCV was used with a pre-trained model. The detectMultiScale() function was applied with optimized parameters such as scaleFactor (1.1) and minNeighbors (5) to improve accuracy. Haar Cascade detects faces based on edge-like features (e.g., eyes, nose, and mouth) using a cascade of classifiers. The detection was performed in real-time on each image, and the execution time was recorded. Additionally, the number of detected faces was compared to the ground truth labels to measure accuracy. Haar Cascade is known for its fast speed but lower accuracy, making it ideal for real-time applications with fewer computational resources.

b) MTCNN Approach:

MTCNN (Multi-Task Cascaded Convolutional Neural Network) was implemented using the detect_faces() function. Unlike Haar Cascade, MTCNN operates on RGB images and employs deep learning techniques to locate faces. It consists of three stages: P-Net (Proposal Network), R-Net (Refinement Network), and O-Net (Output Network), refining detections at each step. MTCNN predicts facial landmarks (eyes, nose, mouth) along with bounding boxes, making it more robust to variations in pose and lighting. The time taken for each image was recorded to analyze speed performance. While MTCNN provides higher accuracy, it is computationally heavier and thus slower than Haar Cascade.

Both the methods were tested on the kaggle dataset to comapre their performance in terms of speed, accuracy where the dataset includes many images of different images under various conditions like facial expressions, varying illumination.

D. Performance evaluation metrics

To compare these both the algorithms we have used the two metrics i.e., Speed and Accuracy. In Speed the execution time in seconds for each image is recorded as shown in the fig1, fig2. The average detection time across all images was computed, lower detection time indicates a faster algorithm. Where as in Accuracy False Negatives and Positives were detected MTCNN's accuracy was expected to be higher than Haar Cascade, especially under complex lightning conditions.

III. SYSTEM ARCHITECTURE

This research focuses on the comparative analysis of Haar Cascade and MTCNN face detection algorithms using publicly available datasets from Kaggle. The methodology involves five major steps: Dataset Selection, Preprocessing, Face Detection, Performance Evaluation, and Result Interpretation. The *fig:1* will show the clear understanding of the methodology neatly one by one starting from the data collection to analysis. The methodology involves capturing real-time video frames, preprocessing them, and applying Haar Cascade and MTCNN for face detection. Recognized faces are compared with stored embeddings, and attendance is automatically logged in the database.

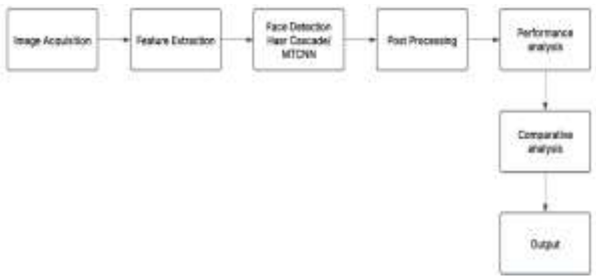

Fig: 2

In the face detection phase, Haar Cascade applies a sliding window approach with predefined classifiers, whereas MTCNN processes images through a three-stage pipeline involving proposal, refinement, and output networks to enhance accuracy. Post-processing techniques like non-maximum suppression eliminate redundant bounding boxes, while thresholding is applied to filter false detections based on confidence scores. The performance evaluation phase measures accuracy, precision, recall, and F1-score, along with visualization methods like ROC and precision-recall curves to assess efficiency. Comparative analysis is conducted to examine speed, accuracy, and robustness under varying conditions such as lighting, occlusions, and pose variations, using statistical analysis and hypothesis testing. The final results present a performance comparison between the two algorithms through graphical visualizations and tabular data. The system stores results, extracted features, and performance metrics in a database for future analysis, and a user-friendly interface provides clear visualization of detection outputs and algorithm efficiency. This structured approach ensures an effective and systematic comparison of Haar Cascade and MTCNN in real-world face recognition applications.

IV. EXPERIMENTATION AND RESULTS

The experimented results were visualized using graphs and comparative tables in this there are two graphs i.e., for speed and accuracy each one has individual graphs plotted now let us see one by one below. The below figures are generated using Google Colab for analysis.

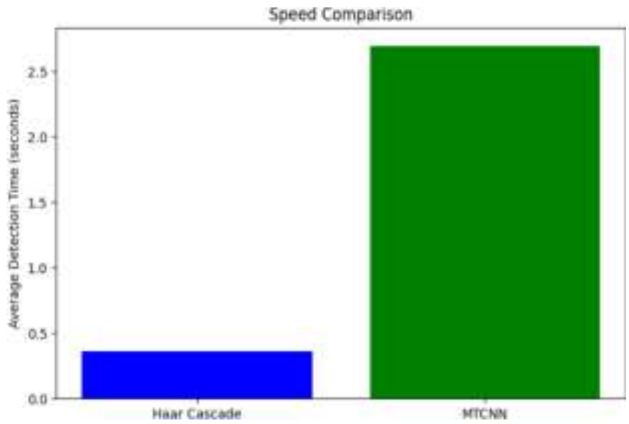

Fig:2

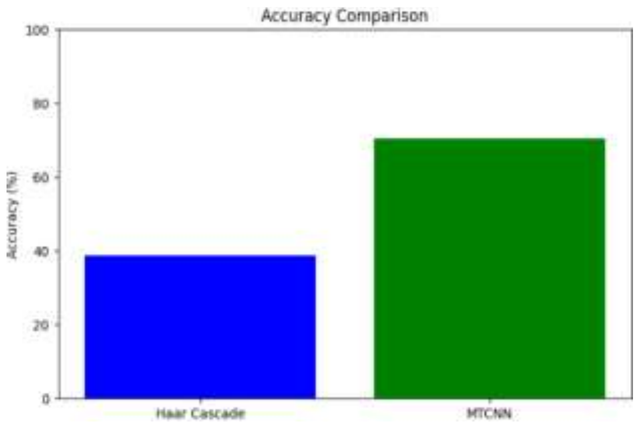

Fig: 3

The first graph (Speed Comparison) i.e., *fig:3* illustrates the average detection time of Haar Cascade and MTCNN. It shows that Haar Cascade (blue) has significantly lower detection time compared to MTCNN (green), making it faster for face detection tasks.

The second graph (Accuracy Comparison) i.e., *fig: 4* presents the accuracy of both algorithms, where MTCNN (green) demonstrates a higher detection accuracy compared to Haar Cascade (blue). This indicates that MTCNN provides more reliable face detection but at the cost of increased processing time.

	Algorithm	Avg Time(s)	Accuracy (%)	Total Detected faces
0	Haar Cascade	0.3639 99	38.88888 9	310
1	MTCNN	2.6959 59	70.37037 0	190

Table: 1

In Table: 1 we can see the average time, accuracy, total faces detected out of 500 (here we took 500 images as dataset). We can see that Haar Cascade average time is less than MTCNN, accuracy is more for MTCNN than Haar Cascade and we can also see that Haar Cascade detected more faces than MTCNN.

We took the images as the dataset randomly from the Kaggle this was said at the earlier itself so for the reference we are displaying the image dataset here below. The above displayed images are sample images displayed over here. These are subjects in various indoor and outdoor setting, these are resized and pre-processed to ensure uniform dimensions suitable to display over here This dataset serves as a sample dataset as a reference.

Fig: 4

V. DISCUSSION

This research mainly focuses on the comparative analysis of the algorithms as we said early. Here the algorithms are Haar Cascade and MTCNN and compared the results and noted in the Table: 1. We took 500 facial images as a dataset from the Kaggle and did the experiment and presented in the form of graphs and table. We can see that average time of Haar Cascade is less than MTCNN because it uses simple feature-based classifiers, while MTCNN involves deep learning with multiple neural network stages that is the advantage of MTCNN because of that it has more accuracy in detecting, but MTCNN detects less faces than Haar Cascade because it applies stricter filtering i.e., it involves false positives and false negatives and extract the features sometimes missing low-quality or partially visible.

VI. CONCLUSION

In this study, we compared the Haar Cascade and MTCNN algorithms for face detection based on accuracy, detection speed, and the number of faces detected. The results indicate that Haar Cascade detects more faces with a significantly lower processing time but at the cost of higher false positives and lower accuracy. On the other hand, MTCNN provides better accuracy by effectively filtering out false detections, but it takes more time to process images due to its deep learning-based approach.

Overall, the choice between Haar Cascade and MTCNN depends on the application requirements. If real-time performance is a priority, Haar Cascade may be suitable due to its speed. However, for applications requiring higher accuracy, such as secure authentication or surveillance, MTCNN is a better choice. Future improvements can focus on optimizing MTCNN for speed while maintaining accuracy or enhancing Haar Cascade to reduce false detections.

REFERENCES

[1] M. Arsenovic, S. Sladojevic, A. Anderla, and D. Stefanovic, "A deep learning-based attendance system using facial recognition," presented at the 15th IEEE International Symposium on Intelligent Systems and Informatics (SISY), 2017, pp. 000053

[2] R. Fu, D. Wang, D. Li, and Z. Luo, "Implementation of university classroom attendance monitoring with deep learning techniques," in 10th International Conference on Intelligent Computation Technology and Automation (ICICTA), 2017, pp. 128–131.

[3] J. Mehariya, C. Gupta, N. Pai, S. Koul, and P. Gadakh, "Utilizing OpenCV and Firebase for automated student attendance tracking," in International Conference on Electronics and Sustainable Communication Systems (ICESC), 2020, pp. 624–629.

[4] P. Gupta, J. Tripathi, M. Sharma, and N. Saxena, "Human face recognition using deep neural networks," MECS Journal, vol. 9, 2018.

[5] R. Rahim, "An investigation into facial recognition using Fisher Linear Discriminant approach," in IOP Conference Series, 2018.

[6] "Application of deep learning for facial attendance tracking," presented at the IEEE International Conference on Mobile Networks and Wireless Communications (ICMNWC), 2021.

[7] J. Redmon and A. Farhadi, "YOLO9000: Advancements in object detection," arXiv preprint, 2017.

[8] L. Zhi-heng and L. Yong-zhen, "Development of a classroom attendance monitoring system using video-based facial recognition," in International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS), 2019.

[9] M. Ritonga, D. T. Sanchez, S. T. Mrudula, R. K. Veluri, M. Jawarneh, and A. Raghuvanshi, "Designing a smart attendance system with deep learning-based facial recognition," in Second International Conference on Emerging Trends in Information Technology and Engineering (ICETITE), 2024.

[10] "Hybrid machine learning and computer vision for automated attendance tracking," in 3rd International Conference on Applied Artificial Intelligence and Computing (ICAAIC), 2024.

[11] "Deep learning-driven attendance system utilizing OpenCV-based face recognition," in 8th International Conference on Smart Structures and Systems (ICSSS), 2022.

[12] "A real-time approach to smart attendance management using Haar Cascade and LBPH," in International Conference on Communication, Circuits, and Systems (IC3S), 2023.

[13] S. Kakarla, P. Gangula, M. S. Rahul, C. S. C. Singh, and T. H. Sarma, "CNN-based attendance tracking using face recognition," 2023.

[14] S. Bhattacharya, G. S. Nainala, P. Das, and A. Routray, "SAMS: An AI-powered attendance system for classrooms," presented at the 18th IEEE International Conference on Advanced Learning Technologies, 2018.

[15] "Implementation of a deep learning approach for attendance automation," in Proceedings of the 7th International Conference on Trends in Electronics and Informatics (ICOEI), 2023.