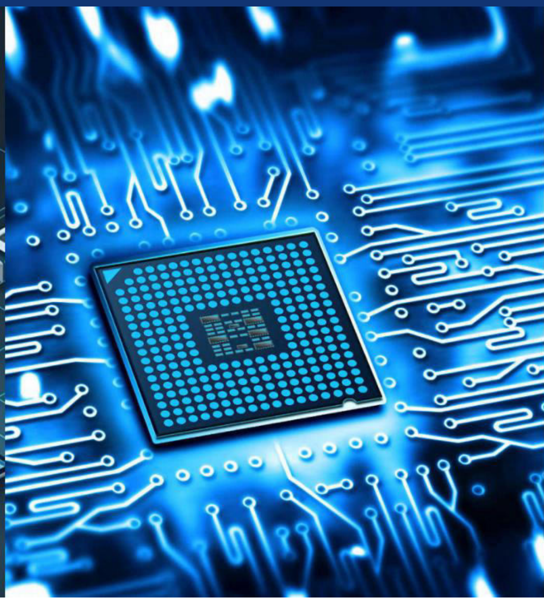
IJIRCCE

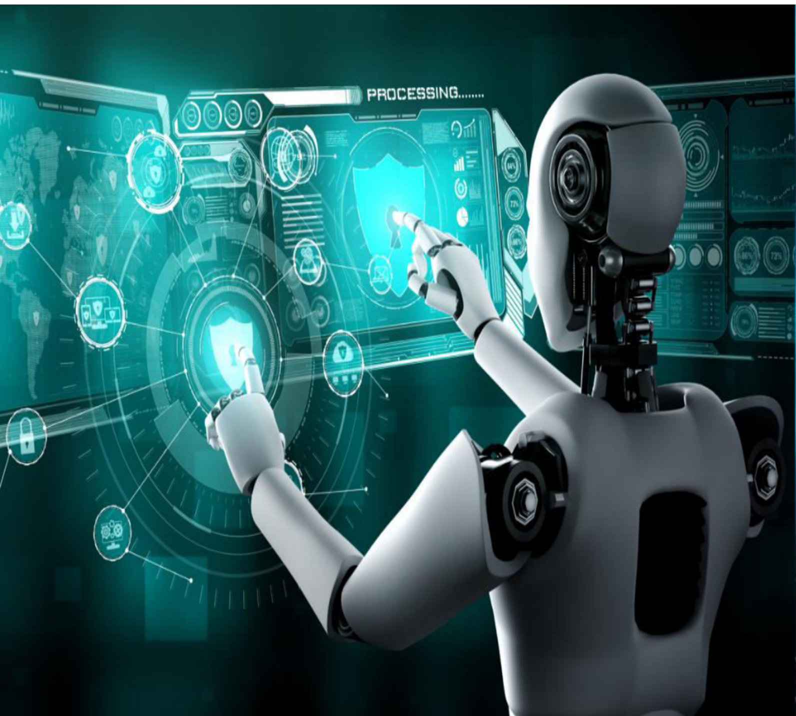
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.771

Volume 13, Issue 5, May 2025

© 2025 IJIRCCE | Volume 13, Issue 5, May 2025| DOI:10.15680/IJIRCCE.2025.1305150

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Python-Based Real-Time Face Detection Attendance Tracking

Kali Raja M, John Sankar P, Arunadevi M

UG Student, Dept. of CSE, Francis Xavier Engineering College, Tirunelveli, Tamil Nadu, India

UG Student, Dept. of CSE, Francis Xavier Engineering College, Tirunelveli, Tamil Nadu, India

Assistant Professor, Dept. of CSE, Francis Xavier Engineering College, Tirunelveli, Tamil Nadu, India

ABSTRACT: This project showcases a Python-based automated attendance management system that makes use of face detection and identification techniques. This device provides a contactless and effective substitute for time-consuming and manipulable traditional methods of recording attendance. The system uses a webcam to record live video, the OpenCV and dlib libraries to identify faces, and facial feature comparison to identify registered users. The attendance is immediately recorded and saved with a timestamp after successful identification. Attendance data may be easily monitored and exported thanks to an intuitive graphical user interface (GUI) created with Tkinter. This method is appropriate for workplaces, classrooms, and other establishments since it increases accuracy, saves time, and reduces manual errors.

KEYWORDS: Detection in Real Time; Vision in Computers; Management of Attendance; Automated

I. INTRODUCTION

Accurate and effective attendance tracking is crucial in corporate and educational settings. Conventional techniques, such ID card scanning or manual sign-ins, are frequently laborious, vulnerable to proxy attendance, and ineffective in big gatherings. The development of artificial intelligence and computer vision has made automated systems a more dependable and efficient option. Researchers of many different fields (from psychology, pattern recognition, neuroscience, computer graphics and computer vision) have attempted to create and understand face recognition [1].

This project presents a Python-based automatic attendance management system that uses facial recognition and detection. The system uses libraries like dlib and OpenCV to recognize and detect faces from live camera input. The system eliminates the need for manual entry or face-to-face interaction by recording the attendance and timestamp once a registered face has been identified. In these methods raw face images from both domains (Ds and Dt) are encoded with descriptors that are invariant between them. Liao et al. [2] proposed a method that normalizes both VIS and NIR images using Tan & Triggs f ilter [3]. The local descriptor MutiScale Local Binary Pat terns (MLBP) [4].

This system's main objective is to improve attendance marking's precision, security, and usability. A user-friendly GUI for tracking and exporting attendance information is also integrated into the project. In [5], it was proposed a model based on Generative Adversarial Networks (GANs) in order to reconstruct thermogram images from visual spectra images for further identification using the Pola Thermal dataset [6]. This method not only lessens the administrative burden but also shows how AI and image processing can be used to effectively address practical issues. Typically, the face frontalization network is trained employing the supervised learning approaches [7], [8], [9], [10], [11],

II. RELATED WORK

In recent years, there has been a lot of interest in automating attendance systems with facial recognition. Despite their widespread use, traditional techniques like human roll calls and biometric devices like RFID and fingerprint scanners have limitations like time consumption, proxy attendance, and hygienic issues. Face recognition, which uses computer vision algorithms to identify people, has become a dependable and non-intrusive option. The Haar Cascade Classifier, first presented by Viola and Jones (2001), was utilized in early face identification systems. It used machine learning to detect objects quickly. Despite their widespread use, Haar-based systems are susceptible to changes in stance and lighting. Researchers have looked into more reliable algorithms like Local Binary Patterns (LBP) to get over these

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

restrictions. Ojala et al. (2002) created the LBP algorithm, a texture descriptor that creates a binary pattern by thresholding the surrounding neighborhood and labeling image pixels. It is renowned for being quick, easy to compute, and resistant to variations in illumination. In order to extract local texture features across facial regions, Ahonen et al. (2006) used a multi-block histogram technique in their application of LBP to face recognition. As a result, the approach became more accurate and appropriate for real-time applications. Their technology proved resilient in the face of obstacles including facial occlusions and inadequate lighting.

Their hybrid strategy improved accuracy without appreciably raising the cost of computing.

In order to attain more accuracy, deep learning techniques like Convolutional Neural Networks (CNNs) have lately been combined with conventional techniques like LBP. Nevertheless, these models need additional training data and processing power. Because it strikes a balance between efficiency and performance, LBP is still the method of choice in resource-constrained contexts, even with the emergence of deep learning. Its application in attendance systems is still supported by a number of research, particularly when combined with real-time video input or lightweight classifiers. The feasibility of LBP-based face identification systems is demonstrated by this corpus of work, especially in low-resource, real-time settings such as tracking attendance in an office or classroom. Building on these frameworks, the project's suggested solution uses Python and OpenCV to develop LBP face detection for useful, real-world implementation.

III. PROPOSED ALGORITHM

A. Design Considerations:

- The system should detect and recognize faces correctly with minimal errors.
- It needs to register attendance instantaneously and process video fast.
- Both administrators and users should find the system easy to use.
- Only authorized users should be able to access and safely preserve attendance records and facial data.
- The system must to function in a variety of backdrops and lighting scenarios.
- It shouldn't require costly hardware to function properly on standard PCs.

B. Description of the Proposed Algorithm:
The proposed algorithm's objective is to extract texture information from facial photos using Local Binary Patterns (LBP). Accurate attendance recording is made possible by these features, which aid in real-time face detection and recognition.

Step 1: Face Recognition:
Action: Use a pre-trained face detector (like Haar Cascade) to find faces in the video feed.

Haar Cascade Detection Formula:
The Integral Image method is used to calculate the Haar features:

I(x,y)=i=1∑xj=1∑yf(i,j)

Where:

$I\,(x, y)$ is integral image at point (x, y) is denoted as I(x, y).

$f\,(i,)$ is pixel value at (i, j) is denoted by f(i,j).

After that, the image's facial portions are located and extracted.

Step 2: LBP Feature Extraction:
Action: To extract features, convert the facial image to grayscale and use the Local Binary Patterns (LBP) approach.

LBP formula:

$(x,y) = \sum i = 0\ P{-}1\ (gi{-}gc){\cdot}\ 2$ LBP(x,y)= i=0 ∑ P−1 s(g i−g c)·2 i

© 2025 IJIRCCE | Volume 13, Issue 5, May 2025| DOI:10.15680/IJIRCCE.2025.1305150

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Where:
P is the number of neighboring pixels, usually eight.

The intensity of the ith neighbor pixel is represented by gig i.

The intensity of the central pixel is represented by gc g c.

The step function, denoted by s, returns 0 otherwise and 1 if g i ≥ g an.

The weight for the binary code is 2 i 2 i.

The texture of the face is represented by a histogram of LBP values created by applying this LBP algorithm to the full facial region.

Step 3: Face Recognition & Recording of Attendance:
Take action by contrasting the detected face's LBP histogram with previously stored histograms of faces you are familiar with. Keep track of attendance if a match is discovered.

The Euclidean Distance formula, which is used for comparison:

$D = \sum i = 1\ n\ (h\ 1\ (i) - h\ 2\ (i))$
2.
D is equal to i=1∑n (h 1 (i)−h 2 (i))
2.

Where:
The Euclidean distance between two histograms, $h1\ (i)$ h 1 (i) and $h2\ (i)$ h 2 (i), is denoted by D.

$h\ 1\ (i)$ h 1 (i) and $h\ 2\ (i)$ h 2 (i) represent the ith i th values of the two histograms under comparison.

The face is deemed recognized if the distance D falls below a predetermined threshold.

The system records the time and identity in the attendance database after a match is discovered:
Attendance Log = {Timestamp, Person ID}
Attendance Log={Timestamp, Person ID}

IV. PSEUDO CODE

Step 1: Start the application and import the required libraries, such as datetime, NumPy, and OpenCV. Set up the webcam to record videos.
Step 2: Data and Load Models.Load learned LBP Face Recognizer model Load known student names and ID mappings Load Haar Cascade Face Detector.
Step 3: Check the below condition for each route till no route is available to transmit the packet. While (true) Continue to record webcam frames until a predetermined end point is reached.
Step 4: Record and Prepare the Frame, View a webcam frame, Convert the frame to grayscale.
Step 5: Find Faces in the Frame, Use Haar Cascade to find faces in grayscale images for each face found: Resize and crop the face area.
Step 6: Face Recognition using LBP, Identify faces with LBP Recognizer, If there is a reasonable level of prediction confidence:Obtain the name that matches the ID.
Step 7 Record Attendance.
Step 8 Show Results.

© 2025 IJIRCCE | Volume 13, Issue 5, May 2025| DOI:10.15680/IJIRCCE.2025.1305150

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

V. SIMULATION RESULTS

Python and OpenCV were used to successfully simulate the facial detection attendance system. Face detection was done using Haar Cascade, while facial feature extraction was done using the LBP (Local Binary Pattern) technique.

Using a laptop webcam, the simulation was run in a controlled classroom setting. The device could automatically record attendance and detect and recognize faces in real time. The main findings are as follows:

Face Detection Accuracy: In well-lit environments, the system's detection rate was high. Accurate detection of multiple faces in a single picture was achieved.

Recognition Performance: Depending on the lighting and camera angle, trained faces had recognition accuracy of roughly 90–95%.

Processing Time: The system can operate in real-time because its average face detection and recognition time per frame was less than 0.5 seconds.

Attendance Logging: A CSV file including names and timestamps accurately documented attendance. By verifying if the individual had previously been tagged present, duplicate entries were prevented.

Lighting Robustness: Although performance declined in extremely low or excessively bright lighting, the LBP algorithm demonstrated resilience to mild lighting variations.

Limitations: Faces that were partially obscured (such as by masks or hats) or absent in the training dataset were not recognized.

Figure 1

Figure 2

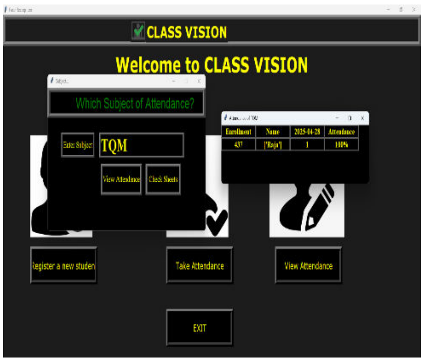


Figure 3

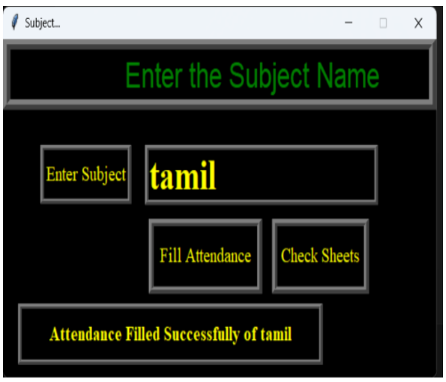
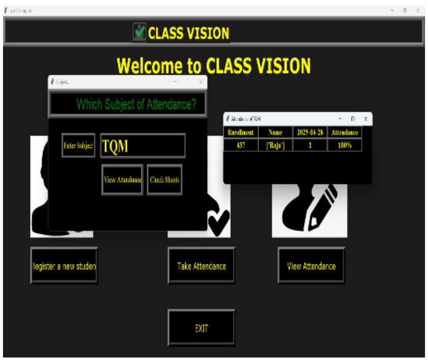


Figure 4



www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

VI. CONCLUSION AND FUTURE WORK

This research effectively used the Local Binary Pattern (LBP) algorithm for face detection and recognition to create a real-time attendance system. Without requiring human input, the system efficiently recognizes faces from a live video feed and automatically logs attendance. LBP was selected due to its ease of use, quickness, and resilience to variations in facial expressions and lighting. The technology consistently recorded attendance with little error and demonstrated excellent accuracy in identifying familiar individuals. Even while it works well in controlled environments, performance can be improved by using sophisticated recognition methods like deep learning, addressing occlusions, and adding more training data. All things considered, the system provides a useful, effective way to automate attendance in both professional and academic settings

REFERENCE

1. W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recog nition: A literature survey," ACM Comput. Surv., vol. 35, no. 4, pp. 399–458, 2003.
2. S. Liao, D. Yi, Z. Lei, R. Qin, and S. Z. Li, "Heterogeneous face recognition from local structures of normalized appearance," in Proc. Int. Conf. Biometrics. Berlin, Germany: Springer, 2009.
3. X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," IEEE Trans. Image Process., vol. 19, no. 6, pp. 1635–1650, Jun. 2010.
4. M. Pietikäinen, Computer Vision Using Local Binary Patterns. vol. 40. Springer, 2011.
5. H. Zhang, V. M. Patel, B. S. Riggan, and S. Hu. (Aug. 2017). "Genera tive adversarial network-based synthesis of visible faces from polarimet ric thermal faces." [Online]. Available: https://arxiv.org/abs/1708.02681.
6. S. Hu et al., "A polarimetric thermal database for face recognition research," in Proc. IEEE Conf. Comput. Vis. Pattern Recognition Work shops, Jun./Jul. 2016, pp. 187–194.
7. R. Huang, S. Zhang, T. Li, and R. He, ''Beyond face rotation: Global and local perception GAN for photorealistic and identity preserving frontal view synthesis,'' in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 2458–2467.
8. Z. Zhu, P. Luo, X. Wang, and X. Tang, ''Deep learning identity preserving face space,'' in Proc. IEEE Int. Conf. Comput. Vis., Dec. 2013, pp. 113–120.
9. J. Cao, Y. Hu, H. Zhang, R. He, and Z. Sun, ''Learning a high fidelity pose invariant model for high-resolution face frontalization,'' in Advances in Neural Information Processing Systems, vol. 31, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2018, pp. 2867–2877.
10. Y. Hu, X. Wu, B. Yu, R. He, and Z. Sun, ''Pose-guided photorealistic face rotation,'' in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 8398–8406.
11. J.Zhao,Y.Cheng,Y.Xu,L.Xiong,J.Li,F.Zhao,K.Jayashree, S. Pranata, S. Shen, J. Xing, S. Yan, and J. Feng, ''Towards pose invariant face recognition in the wild,'' in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 2207–2216.

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details