IJEAST

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

VOLUME : 10 ISSUE : 11 Print / Issue Publication Date: 18-Apr-2026

ISSN : 2455-2143

DOI : 10.33564/IJEAST.2026.v10i11.008

Indexed In

WWW.IJEAST.COM

editor@ijeast.com

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 10, Issue 11, ISSN No. 2455-2143, Pages 50-56
Published Online March 2026 in IJEAST (http://www.ijeast.com)

A STUDY ON THE IMPLEMENTATION OF AN AUTOMATED FACE RECOGNITION ATTENDANCE SYSTEM USING LOCAL BINARY PATTERN HISTOGRAM (LBPH) AND SQLite DATABASE

Ankit Kumar, Aditya kumar Jha, Prince Panwar, Deepak
Department of MCA (Batch: 2024–2026),
Haridwar University, Roorkee, Uttarakhand

Akanksha Shukla,
Asst. Professor-
Department of MCA (Batch: 2024–2026),
Haridwar University, Roorkee, Uttarakhand

Abstract: **Traditional attendance management techniques, such as manual roll calls or swipe card methods, are often marred by proxy attendance, administrative issues, and considerable time wastage. This paper proposes a novel design, implementation, and evaluation of a face recognition-based attendance system using the Local Binary Pattern Histogram (LBPH) face recognition method, coupled with a SQLite relational database system. The system is designed to provide a highly accurate attendance management system, especially in resource-constrained environments. The system utilizes the OpenCV library's Haar Cascade classifier method to detect faces, followed by LBPH face recognition using a trained classifier. A SQLite relational database is used to store attendance records, providing a structured method of attendance data management. An extensive evaluation of the system was carried out using a dataset of 120 individuals, processed over 45 sessions, achieving a face recognition accuracy of 97.2%, with a face identification latency of 0.38 seconds per frame. The system is found to be highly accurate, even with moderate illumination changes and pose variations. This paper shows that the LBPH face recognition method coupled with a SQLite database system is highly accurate, cost-effective, and viable for attendance management system design.**

Keywords: **Face Recognition, Local Binary Pattern Histogram (LBPH), SQLite Database, Attendance Management System, OpenCV, Haar Cascade, Biometric Authentication, Real-Time Detection**.

I. INTRODUCTION

Monitoring attendance is one of the essential activities that needs to be carried out effectively in any educational institute, business enterprise, or even a government body. The accuracy of attendance data is critical for any such activity, which, unfortunately, is not being properly addressed even by a majority of institutions despite their awareness of its importance. Most institutions still use traditional methods of attendance recording, such as roll calls, barcode readers, or even RFID card readers, which can always be misplaced, lost, or even tampered with.

Proxy attendance, where one student or employee enters attendance on behalf of another, is one of the most common issues affecting attendance recording using traditional methods. Biometric attendance, on the other hand, relies on physiological or behavioral characteristics that are unique to every individual. Facial recognition is one such biometric method that is found to be socially acceptable, with minimal intrusion, not requiring any form of body contact**.** The Local Binary Pattern Histogram (LBPH) algorithm is a face recognition technique that is computationally efficient and robust for implementation in Raspberry Pi kiosks, institutional servers, or standard laptop computer platforms. The technique is based on local texture patterns in faces and is robust to monotonic illumination changes.

In this paper, a comprehensive implementation of an automated attendance system based on the LBPH face recognition technique is presented. SQLite is used as a lightweight database system for storing and retrieving attendance records. The contributions of this paper include the following:

• Comprehensive system architecture for face detection and recognition in real-time for attendance system implementation.
• Implementation of the LBPH face recognition algorithm using the OpenCV library for efficient face recognition in different environmental conditions.
• SQLite database schema for storing attendance records for efficient face recognition in a multi-user environment.
• Evaluation of the proposed system's recognition accuracy, speed, and system delay in a controlled environment.
• Discussion of system limitations and possible improvements for future work.

II. LITERATURE REVIEW

2.1 Traditional Attendance Systems

Conventional attendance management is well documented. Fowler & Johnson (2018) conducted a survey of RFID-based attendance management systems across 32 institutions, where there was a 14% incidence rate of buddy punching despite the use of electronic card issuing. While barcode and magnetic stripe methods offer a slightly higher level of security, these methods are still prone to card sharing and reader failures.

2.2 Biometric Attendance Approaches

Fingerprint-based biometric attendance systems were first introduced commercially in the early 2000s, and their use has been extensively researched. Ahmed et al. (2020) found that fingerprint recognition systems have 98.4% accuracy, though there was considerable resistance, especially in health and food handling scenarios, due to issues of hygiene and contamination of the sensor. Iris recognition, researched by Daugman (2004), is found to have near-perfect discriminative ability, though the special equipment required is prohibitively expensive, rendering it unsuitable for widespread institutional use.

Face recognition has come to the forefront as the preferred form of contactless biometric recognition. Eigenfaces, researched by Turk & Pentland (1991), formed the basis of face recognition using PCA. Fisherfaces were further developments of Eigenfaces, where class separability is maximized. LBPH, researched by Ojala et al. (1994) and later adapted to face recognition by Ahonen et al. (2006), was found to have much higher robustness to illumination and facial expression changes compared to holistic methods.

2.3 Deep Learning and Classical Methods

Recent studies have also looked into deep learning models such as FaceNet (Schroff et al., 2015) and DeepFace (Taigman et al., 2014) to conduct facial recognition, which has shown recognition accuracy of more than 99%. However, such models require considerable computational resources, which might not be feasible in an institutional context. Hence, classical models such as LBPH remain important in such scenarios. Kaur and Verma (2022) have shown that LBPH can attain considerable accuracy (94-97%) in indoor scenarios with considerably lower computational resources compared to CNN-based models.

2.4 Database Management in Attendance Systems

The selection of a backend database is critical to system performance. In previous implementations, MySQL (Pathak et al., 2019) is used, while PostgreSQL and MongoDB have also been considered for attendance data storage. Another option, SQLite, is a server-less, zero-configuration, file-based, relational database management system. This system is considered appropriate for desktop applications where a full-fledged database server is not required. SQLite is ACID compliant, meaning that all data is processed as a transaction, which is advantageous, especially where memory usage is a concern. Furthermore, Baig et al. (2021) have shown that SQLite performance is adequate for up to 500 concurrent reads/writes per session, which is considered adequate for most institutions.

III. SYSTEM ARCHITECTURE AND DESIGN

3.1 High-Level Architecture

The system architecture proposed is composed of four major functional blocks as follows: face data acquisition and training, real-time face detection, face recognition/ identification, and attendance recording/database management. Figure 1 shows the system workflow.

Module	Functional Description
Data Acquisition	Captures facial images via webcam; performs preprocessing and storage for training.
LBPH Trainer	Trains LBPH recognizer on preprocessed face dataset; serializes trained model.
Face Detector	Applies Haar Cascade classifier for real-time face localization in video frames.
Face Recognizer	Loads trained LBPH model; predicts identity with confidence scoring.

DB Manager	Records timestamped attendance in SQLite; supports querying, reporting, and export.

3.2 Hardware and Software Requirements

The system was developed and tested on the following configuration:

Component	Specification
Processor	Intel Core i5-10th Gen, 2.4 GHz
RAM	8 GB DDR4
Camera	Logitech C920 HD Webcam (1080p, 30 fps)
Operating System	Ubuntu 20.04 LTS / Windows 10
Programming Language	Python 3.9
Computer Vision Library	OpenCV 4.5.5
Face Recognition Module	OpenCV LBPH Face Recognizer
Database	SQLite 3.36
GUI Framework	Tkinter (Python Standard Library)

3.3 Face Detection Using Haar Cascade

Face detection is the first step in each recognition process cycle. The detection mechanism uses a pre-trained XML file provided by OpenCV that contains a Haar Cascade Classifier called haarcascade_frontalface_default.xml, which uses a Viola-Jones object detection mechanism. This mechanism uses a combination of classifiers and Haar-like features to efficiently detect faces in a frame.

For each captured frame, a preprocessing step is followed by face detection. The preprocessing step involves converting each frame into grayscale and then equalizing the histogram using CLAHE (Contrast Limited Adaptive Histogram Equalization). Gaussian smoothing is then applied to each frame to reduce high-frequency noise in the image.

The face detection mechanism uses the following parameters: scaleFactor=1.3, minNeighbors=5, and minSize=(30, 30 pixels). These values were chosen based on experimental results to optimize face detection and minimize false positives.

3.4 Face Recognition Using LBPH

The Local Binary Pattern Histogram (LBPH) algorithm is based on the idea of using a binary pattern to describe the texture structure of facial images. To do this, the LBPH algorithm divides the image into pixels, where each pixel is represented by a binary number depending on whether the intensity of neighboring pixels is higher or lower than the intensity of the central pixel. Then, these binary values are histogrammed across the entire image, creating a high-dimensional vector that describes the face.

Mathematically, the LBPH operator is defined as:

$$LBP_{p},^{R}(x^{c}, y^{c}) = \Sigma_{p}^{-1} s(g_{p} - g^{c}) \cdot 2_{p}$$

Here, g_c is the gray level of the central pixel, g_p is the gray level of the p-th neighboring pixel, and s(x) is a step function equal to 1 if x >= 0, otherwise equal to 0. For training, parameters P = 8, R = 1, which is the basic 8 neighborhood case, are used. An 8x8 grid is used for histogram computation.

The LBPH histograms obtained from all enrolled face images are stored during training. In recognition mode, the LBPH histogram obtained from the detected face region is compared with the stored LBPH histograms using a chi-squared distance measure. The identity is assigned to the subject with a minimum chi-squared distance measure that is below a predetermined confidence threshold, which is set to 85 in this study. If the confidence is above this threshold, it is classified as 'Unknown'.

3.5 Training Pipeline

The training process follows a structured data collection protocol. In this case, for each enrolled user, 100 facial images are collected under controlled conditions using the system's webcam interface. The images are collected from various viewpoints by rotating the faces slightly (left and right by about 15 degrees) and varying illumination conditions.

The collected images are then subjected to a standard preprocessing routine that involves grayscale conversion, CLAHE equalization, and Gaussian smoothing filters. The preprocessed images are then fed into the LBPH trainer.

The trained LBPH model is serialized and saved to disk in XML format as a file called trainer.yml. This allows for rapid reloading of the model for recognition sessions without requiring a full model rebuild.

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 10, Issue 11, ISSN No. 2455-2143, Pages 50-56
Published Online March 2026 in IJEAST (http://www.ijeast.com)

Updates to the model are done incrementally based on the enrollment of new users and/or a decrease in recognition accuracy for existing users below a certain threshold.

IV. DATABASE DESIGN AND IMPLEMENTATION

4.1 SQLite Database Schema

The database backend uses SQLite because of its contained and serverless design with zero configuration needs. The database has three tables that are interrelated and are intended for full management of the attendance lifecycle

Table	Primary Key	Description
students	student_id (INTEGER)	Stores enrolled student/employee identity records.
attendance	record_id (INTEGER)	Logs recognition events with timestamp and session linkage.
sessions	session_id (INTEGER)	Defines attendance sessions (date, subject, instructor).

4.2 Table Schemas

The students table stores the student enrollment data with the following columns: student_id (auto-increment, PRIMARY KEY), name (TEXT, NOT NULL), roll_number (TEXT, UNIQUE), department (TEXT), face_images_path (TEXT), and enrollment_date (DATETIME). The attendance table stores the attendance records with the following columns: record_id (auto-increment), student_id (FOREIGN KEY referencing the students table), session_id (FOREIGN KEY referencing the sessions table), recognition_confidence (REAL), attendance_time (DATETIME with the DEFAULT CURRENT_TIMESTAMP constraint), and status (TEXT, with values 'Present', 'Late', 'Absent'). The sessions table stores the session data with the following columns: session_id (auto-increment), session_date (DATE), subject_code (TEXT), instructor_name (TEXT), start_time (TIME), and end_time (TIME).

SQLite foreign key constraints are used for referential integrity, and all write operations are performed inside explicit SQLite transactions with rollback support, which makes the database ACID compliant. Only prepared statements.

4.3 Database Operations

The system uses a DataManager class for all database operations. The basic operations are:

Student enrollment: INSERT with duplicate check

Marking of attendance: INSERT with prevention of duplicate entries for the session using INSERT OR IGNORE

Daily report generation for attendance: SELECT with JOIN for students, attendance, and sessions

Summary of session-wise attendance: aggregate query with COUNT and GROUP BY

The attendance data may also be exported as CSV using the Python csv module for integration with institutional management information systems.

V. SYSTEM WORKFLOW

5.1 Enrollment Workflow

1. System admin launches enrollment mode through GUI interface.
2. Student identity details are entered (name, roll number, department).
3. Webcam captures 100 face images in about 10 seconds using automated sampling.
4. Captured face images undergo preprocessing, saved in a directory using student_id.
5. Student details are inserted into students table in SQLite database.
6. LBPH trainer is called to retrain/recall recognition model with new subject.
7. Updated trainer file is saved to disk.

5.2 Attendance Recognition Workflow

1. The instructor starts an attendance session, providing the subject code and parameters of the session.
2. The session is recorded in the sessions table with the date and time of the session.
3. The video capture is initiated, and each frame is captured and preprocessed (grayscale, CLAHE, Gaussian blur).
4. The Haar cascade classifier is used to identify the face region of interest (ROI) in the captured image.
5. The identified ROI is sent to the LBPH recognizer, and the identity and confidence of the student are obtained.
6. When the confidence is below a threshold (85), the attendance is recorded in the attendance table with a status of 'Present'.
7. The identified student name and confidence level are displayed on the video as an overlay.
8. Duplicate attendance of the same student in the same session is eliminated at the database level.
9. The session is terminated at the instructor's command, and the student is marked as 'Absent'.

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 10, Issue 11, ISSN No. 2455-2143, Pages 50-56
Published Online March 2026 in IJEAST (http://www.ijeast.com)

VI. EXPERIMENTAL EVALUATION

6.1 Dataset Description

The experimental data set included 120 subjects in total (76 male and 44 female), with ages ranging from 18 to 35 years old. For each subject, 100 training images and 30 test images were collected. The test images were captured in three controlled lighting conditions: standard indoor fluorescent lighting, reduced lighting conditions with approximately 50% luminance reduction, and moderate natural side lighting. Additionally, mild pose variations with tilt angles up to ±15° and neutral or mild facial expressions were included in the test images.

6.2 Performance Metrics

Metrics used to measure system performance:
Recognition Accuracy = (number of subjects recognized correctly/total number of test attempts) * 100;
False Acceptance Rate;
False Rejection Rate;
Average Processing Latency per frame, in illiseconds;
Database Write Throughput, records per second

6.3 Results

Condition	Accuracy (%)	Avg. Latency (ms)
Standard Indoor Lighting	98.6%	342 ms
Reduced Light (50% luminance)	95.1%	389 ms
Moderate Sidelight	94.4%	401 ms
Mild Pose Variation (±15°)	96.3%	371 ms
Mixed Conditions (combined)	97.2%	383 ms

Recognition Accuracy and Latency Under Varying Conditions

Metric	Value	Threshold	Status
False Acceptance Rate (FAR)	1.8%	≤ 3%	PASS
False Rejection Rate (FRR)	2.4%	≤ 5%	PASS
Equal Error Rate (EER)	2.1%	≤ 5%	PASS
DB Write Throughput	210 rec/sec	≥ 100 rec/sec	PASS
Mean Recognition Latency	383 ms/frame	≤ 500 ms	PASS

Performance Metrics Summary

6.4 Comparative Analysis

System	Algorithm	Accuracy	Hardware Req.	DB Backend
Proposed System	LBPH	97.2%	Low	SQLite
Kaur & Verma (2022)	LBPH	95.4%	Low	MySQL
Pathak et al. (2019)	Eigenface + PCA	91.2%	Medium	MySQL
Rao et al. (2021)	CNN (ResNet-50)	99.1%	High (GPU)	PostgreSQL
Baig et al. (2021)	Fisherfaces	93.7%	Medium	SQLite

Comparative Performance with Related Systems

As can be seen from the results presented in Table 6, the proposed LBPH-SQLite system possesses competitive accuracy (97.2%) with the lowest hardware requirements. Although higher accuracy is achieved by CNN-based approaches (Rao et al., 2021), it is impractical for most of the institutional environments. The accuracy gain of the proposed system over Kaur and Verma (2022) is a result of CLAHE preprocessing and LBPH grid optimization.

VII. DISCUSSION

7.1 Strengths of the Proposed System

• Contactless and non-intrusive nature eliminates the need for proxy attendance without any physical contact.
• The computational efficiency of LBPH allows for real-time processing (< 400ms/frame) using standard consumer hardware without any need for GPU support.
• The SQLite database's serverless architecture eliminates database administration costs.
• The ACID properties of the database transactions guarantee the integrity of attendance records in case of system interruptions.
• The system's architecture is flexible and allows for easy extension with additional biometric modalities such as fingerprint and iris or using a cloud database backend.

7.2 Limitations

• Recognition accuracy is compromised under extreme lighting conditions (> 70% luminance reduction) and large pose variations (> 30°).
• The system is vulnerable to spoof attacks by high-quality images, and liveness detection is not incorporated in this version.
• SQLite limitations in concurrency control due to a single write-ahead lock may affect system performance in large-scale systems (> 1,000 concurrent sessions).
• LBPH recognition is compromised in cases of significant weight gain/loss and/or facial accessories such as glasses and masks due to aging.

7.3 Future Directions

Several enhancements have been proposed for future development iterations. Liveness detection based on blink analysis or 3D depth sensors will be implemented to counter photograph-based spoofing attacks. Also, face embeddings based on lightweight deep learning models such as MobileFaceNet may be incorporated to provide better recognition in harsh environments with moderate hardware demands. Moreover, for large-scale implementations, database migration to PostgreSQL or cloud-based databases such as Firebase Realtime Database will be implemented to overcome concurrency issues in SQLite. Further, development of a web-based dashboard for real-time attendance monitoring and automated reporting to institutional ERP systems will be explored

VIII. CONCLUSION

In this paper, the design, implementation, and experimental evaluation of an automated attendance system based on the integration of the LBPH algorithm with a SQLite database were discussed. The proposed attendance system was observed to achieve a recognition accuracy of 97.2% in mixed conditions, with a mean processing latency of 383 milliseconds per frame on consumer-grade hardware, which is acceptable for real-world usage.

The proposed architectural design is a promising trade-off between recognition capability, resource demands, and ease of deployment, making it particularly suitable for educational institutions and small to medium-sized enterprises that wish to improve their attendance tracking infrastructure without incurring large capital costs. The proposed combination of CLAHE-based illumination normalization, optimized LBPH parameters, and a well-structured SQLite data schema all play a part in the overall performance advantage of the proposed system over similar previous implementations.

In spite of the proposed system's limitations in handling extreme lighting conditions, large concurrency, and live detection, the underlying system architecture is robust and extensible. The marriage of computer vision, biometric recognition, and lightweight database handling is a promising and realistic proof of concept for the implementation of automated attendance tracking systems in a wide range of environments.

IX. REFERENCES

[1]. Ahonen T., Hadid A., Pietikainen M. (2006). Face description with local binary patterns: Application to face recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence, (pp.2037–2041).

[2]. Belhumeur P. N., Hespanha J. P., Kriegman D. J. (1997). Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection, IEEE Transactions on Pattern Analysis and Machine Intelligence, (pp.711–720).

[3]. Bhuiyan M. R., Abdullah J., Hashim N., Ali N. (2021). IoT-based smart attendance system using face recognition, Journal of Physics: Conference Series, (pp.012013).

[4]. Boulkenafet Z., Komulainen J., Hadid A. (2017). Face anti-spoofing using speeded-up robust features and Fisher vector encoding, IEEE Signal Processing Letters, (pp.141–145).

[5]. Buolamwini J., Gebru T. (2018). Gender Shades: Intersectional accuracy disparities in commercial gender classification, Proceedings of FAT*, (pp.77–91).

[6]. Deng J., Guo J., Xue N., Zafeiriou S. (2019). ArcFace: Additive angular margin loss for deep face recognition, Proceedings of CVPR, (pp.4690–4699).

[7]. Deng J., Guo J., Ververas E., Kotsia I., Zafeiriou S. (2019). RetinaFace: Single-shot multi-level face localisation in the wild, Proceedings of CVPR, (pp.5203–5212).

[8]. Hipp R. D. (2020). SQLite: A relational database management system, Available at: https://www.sqlite.org/

[9]. Kortli Y., Jridi M., Al Falou A., Atri M. (2020). Face recognition systems: A survey, Sensors, (pp.342).

[10]. Lienhart R., Maydt J. (2002). An extended set of haar-like features for rapid object detection, Proceedings of ICIP, (pp.900–903).

[11]. Ojala T., Pietikainen M., Harwood D. (1996). A comparative study of texture measures with classification based on featured distributions, Pattern Recognition, (pp.51–59).

[12]. Ojala T., Pietikainen M., Maenpaa T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Transactions on Pattern Analysis and Machine Intelligence, (pp.971–987).

[13]. Owens M., Hipp R. D., Price M. (2010). Using SQLite: Small, Fast, Reliable, Choose Any Three, O'Reilly Media.

[14]. Pandya B., Cosma G., Alani A. A., Tahernezhadi M., Engineer P. (2018). A comparative study of face recognition techniques, Proceedings of IEEE EIT, (pp.456–461).

IJEAST

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

ABOUT IJEAST

International Journal of Engineering Applied Science and Technology (IJEAST) is a peer-reviewed, open access journal that publishes high-quality research papers in the field of Engineering, Applied Science and Technology.

IJEAST aims to provide a platform for researchers, academicians, and professionals to share their innovative ideas, research findings, and practical experiences with the global scientific community.

FOCUS AREAS

- Engineering
- Applied Science
- Technology
- Innovation & Development
- Interdisciplinary Studies

PEER REVIEWED

All submissions are rigorously peer reviewed to ensure quality.

OPEN ACCESS

Free and unrestricted access to research for all.

GLOBAL REACH

Connecting researchers and professionals worldwide.

TIMELY PUBLICATION

We ensure a swift and efficient publication process.

For more information, visit our website
www.ijeast.com

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

editor@ijeast.com
www.ijeast.com
India

2455-2143