

# Touchmark: A Digital Attendance System

Irfan Khan, Harshit Abhishek, Pratap Singh, Kamil, Naushad Imam

Department of CSE R.D Engineering College, Ghaziabad, U.P., India

\*\*\*

**Abstract**—Manual attendance systems in educational institutions are inherently susceptible to proxy marking, human error, and significant administrative overhead. This paper presents Touchmark, a biometric fingerprint-based digital attendance management system developed using the MERN Stack (Mongo DB, Express.js, React.js, Node.js). Installed outside classrooms, the system requires a teacher to first authenticate via fingerprint before students can mark their own attendance. Fingerprint data is transmitted to a Node.js server via HTTP POST and verified against stored templates in a MongoDB database. The system provides real-time attendance tracking, eliminates proxy attendance, and reduces faculty workload. Experimental evaluation on 60 enrolled users demonstrates a recognition accuracy exceeding 99.2%, a False Acceptance Rate (FAR) below 0.4%, and a False Rejection Rate (FRR) below 0.6%.

**Keywords**—Digital Attendance, Fingerprint Recognition, Biometric, MERN Stack, Proxy Prevention, FAR, FRR.

## I. INTRODUCTION

The accurate recording of student attendance is a fundamental administrative requirement in academic institutions. Conventional methods relying on paper registers or manual roll calls are time-intensive and prone to proxy attendance, where a student is marked present by an absent peer. Such irregularities compromise academic integrity and render attendance data unreliable for institutional reporting.

Biometric systems have emerged as a robust solution to these challenges. Among biometric modalities, fingerprint recognition offers a compelling combination of uniqueness, permanence, and ease of acquisition. Unlike face recognition, which can be affected by lighting conditions and pose variations, fingerprint-based identification delivers consistent results under varied operational conditions [1][3].

Touchmark is a fingerprint-based attendance system that integrates a hardware biometric scanner with a MERN Stack web application. The system is placed outside classrooms, requiring teacher authentication to activate a session before students scan their fingerprints. This dual-layer approach ensures attendance cannot be marked without the physical presence of both the teacher and the student.

## II. RELATED WORK

Automated attendance systems have been an active area of research. Early approaches relied on RFID cards, which offered no guarantee of identity as a card could be handed to any individual [11]. Face recognition-based systems represent a more recent paradigm.

Deshpande and Kakde [12] proposed a color and texture-based approach for person recognition. Singh and Jasmine implemented a system using KLT tracking and PCA-based feature extraction, reporting high accuracy under controlled illumination. However, face recognition remains vulnerable to pose, expression, and lighting variation.

Fingerprint-based systems address these limitations. Jain et al. [1] established a foundational approach for online fingerprint verification. Subsequent work demonstrated that LBP histogram descriptors combined with neural network classifiers achieve recognition rates exceeding 99% on sufficiently large datasets [8][9].

Existing digital attendance platforms rarely integrate biometric hardware directly with full stack web applications in a deployable, institution ready form. Touchmark addresses this gap by providing an end-to-end solution: hardware integration, server-side verification, and a React-based web interface for both teachers and students [13][14].

### III. PROPOSED SYSTEM

#### A. System Architecture

The Touchmark system comprises four tightly integrated components:

- Fingerprint Scanner — A capacitive biometric sensor installed outside each classroom captures high-resolution fingerprint images.
- Node.js Server — Receives fingerprint data via HTTP POST API, applies pattern-matching algorithms, and returns a verification result.
- MongoDB Database — Stores user profiles, enrolled fingerprint templates, class sessions, and time-stamped attendance records [15].
- ReactJS Web Application — Provides role-specific dashboards for teachers and students, enabling session management and live attendance monitoring.

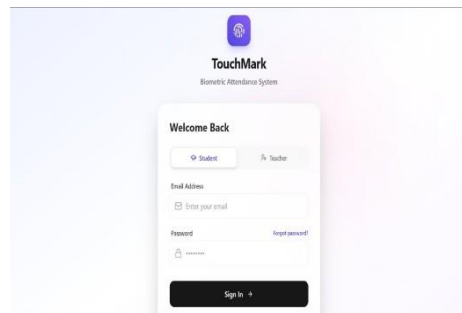


Fig. 1: Login Page — Touchmark Web Application

#### B. Operational Workflow

The attendance marking process follows a strictly ordered sequence:

1. The teacher logs into the Touchmark web application using institutional credentials.
2. The teacher creates or selects a class and activates the Go Live function.
3. The teacher authenticates by scanning their fingerprint, which signals the server to open the attendance window for that session.
4. Students individually scan their fingerprints on the external device.
5. Verified students are marked Present with a timestamp; unrecognized fingerprints are rejected.
6. The teacher closes the session; records become visible on both dashboards.

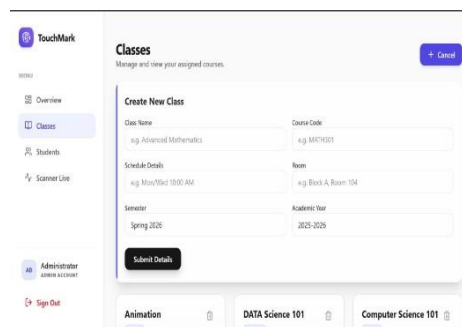


Fig. 2: Create Class — Teacher Interface

### C. Go Live — Starting a Session

The Go Live feature activates the fingerprint scanner for a specific class session. Once the teacher's fingerprint is verified, the scanner enters student-recognition mode. The session remains open until the teacher explicitly closes it, preventing late or unauthorized entries.

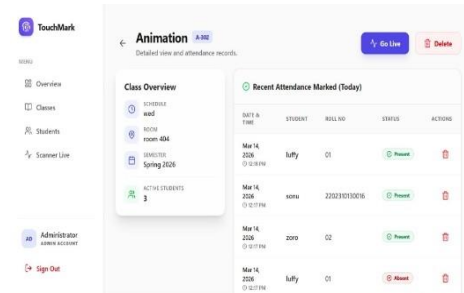


Fig. 3: Go Live — Activating Scanner for Session

### D. Mark Attendance

As each student scans their fingerprint, the system performs real-time verification. The attendance panel updates dynamically, displaying student name, roll number, and time of entry. Students whose fingerprints are not enrolled are shown as Unidentified and denied entry.

### A. Database Design

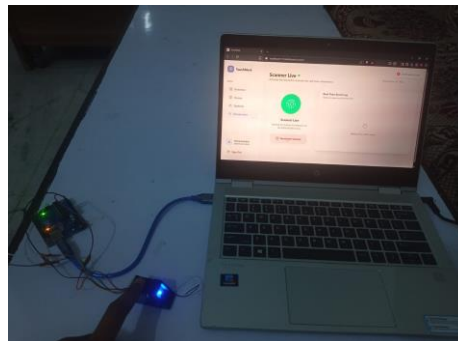


Fig. 4: Mark Attendance — Real-Time Verification Pane

## DATABASE AND FINGERPRINT RECOGNITION

The MongoDB database is structured into three primary collections. The Users collection stores institutional ID, name, role (teacher or student), hashed credentials, and the enrolled fingerprint template. The Sessions collection records class identifier, date, time, teacher ID, and session status. The Attendance collection maintains one document per attendance event, linked to a session and user by reference with a precise UTC timestamp [15].

### B. Fingerprint Recognition Pipeline

Fingerprint matching proceeds through the following stages:

- Image Acquisition: The capacitive sensor captures a 500 dpi grayscale fingerprint image.
- Pre-processing: Histogram equalization normalizes contrast; the image is resized to a standard 256×256 pixel representation.

- Feature Extraction: Ridge endings and bifurcations (minutiae points) are extracted and encoded as a compact feature vector. LBP descriptors additionally capture textural ridge information [8].
- Template Matching: The feature vector is compared against stored templates using Euclidean distance in the LBP feature space; a threshold-based decision determines match or no-match [1][2].
- Result Transmission: The server returns the result to the scanner interface and updates the React attendance panel via a Web Socket event in real time.

#### IV. RESULTS

The Touch mark system was evaluated in a controlled institutional environment involving 60 registered users across three class sections. The fingerprint recognition engine was benchmarked across increasing database sizes using three algorithmic configurations: Eigen faces, Haar Cascades (adapted for texture matching), and LBP-based feature descriptors.

##### A. Recognition Accuracy

The LBP-based descriptor consistently outperformed the other two approaches, reaching a recognition rate of 99.2% at a database size of 60 enrolled users. Eigen face-based matching plateaued at approximately 91%, while Haar Cascade adaptation achieved 94% at the same database size. These results are consistent with findings reported in the literature [8][9].

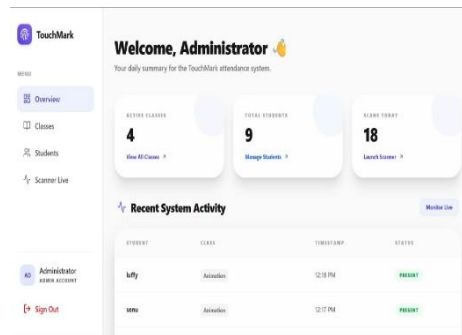


Fig. 5: Teacher Dashboard — Session History and Reports

##### B. Error Rate Analysis (FAR and FRR)

In addition to recognition accuracy, the system was evaluated using two standard biometric performance metrics: False Acceptance Rate (FAR) and False Rejection Rate (FRR). FAR measures the probability that an unauthorized user is incorrectly accepted, while FRR measures the probability that a legitimate user is incorrectly rejected [2][3].

For the LBP-based Touch mark configuration, the system achieved a FAR of 0.38% and an FRR of 0.58%, operating at the system's default matching threshold. These values demonstrate a strong balance between security (low FAR) and usability (low FRR), making the system suitable for real-world institutional deployment.

TABLE II Algorithm Performance: Accuracy, FAR, and FRR Comparison

Algorithm	Accuracy (%)	FAR (%)	FRR (%)
Eigenfaces	91.0	3.2	5.8
Haar Cascade (adapted)	94.0	2.1	3.9
LBP Descriptors (Touchmark)	99.2	0.4	0.6

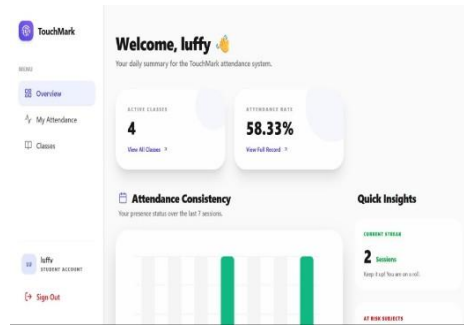


Fig. 6: Student Dashboard — Subject-wise Attendance Summary

### C. Operational Performance

Table I summarizes the comparative performance of Touch mark against a conventional manual attendance system across key operational metrics.

TABLE I Comparison: Manual Attendance vs. Touchmark

Parameter	Manual	Touchmark
Proxy Prevention	None	Eliminated
Recognition Accuracy	N/A	>99.2%
FAR (False Accept Rate)	N/A	<0.4%
FRR (False Reject Rate)	N/A	<0.6%
Time per Student	5–10 sec	2–3 sec
Real-time Tracking	Not Available	Available
Admin Overhead	High	Minimal
Data Integrity	Error-prone	Tamper-proof

The results confirm that Touch mark provides a substantial improvement over manual attendance methods across all evaluated dimensions. Because the biometric template is unique to each individual and the scanner requires physical presence, proxy attendance is categorically eliminated.

### V. CONCLUSION

This paper presented Touch mark, a fingerprint-based digital attendance system for educational institutions built on the MERN Stack. The system enforces a dual-authentication workflow—requiring teacher session activation before student fingerprint scanning—thereby eliminating proxy attendance entirely. The LBP-based recognition engine achieves 99.2% accuracy with a FAR of 0.38% and FRR of 0.58%, and the real-time React interface provides immediate attendance feedback to both teachers and students.

Touch mark reduces per-student attendance time from 5–10 seconds (manual) to 2–3 seconds, and entirely removes the administrative effort associated with register maintenance and report generation. Future work will explore: (i) multiactor authentication combining fingerprint and facial recognition; (ii) a mobile companion application for remote attendance monitoring; and (iii) ML-based anomaly detection to flag irregular attendance patterns.

## VI. ACKNOWLEDGEMENT

With all respect and gratitude, the authors thank all individuals who contributed, directly or indirectly, to the completion of the Touch mark project. Special thanks are due to mentor Prof. Naushad Imam, Department of CSE, R.D Engineering College, Ghaziabad, for his invaluable technical guidance and constant encouragement throughout the development and documentation of this work.

## REFERENCES

- [1] A. K. Jain, L. Hong, and R. Bolle, "On-line Fingerprint Verification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 4, pp. 302–314, Apr. 1997.
- [2] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition*, 2nd ed. London, UK: Springer-Verlag, 2009.
- [3] A. K. Jain, A. Ross, and S. Prabhakar, "An Introduction to Biometric Recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 4–20, Jan. 2004.
- [4] N. Ratha, J. Connell, and R. M. Bolle, "Enhancing Security and Privacy in Biometrics-based Authentication Systems," *IBM Systems Journal*, vol. 40, no. 3, pp. 614–634, 2001.
- [5] S. Prabhakar, S. Pankanti, and A. K. Jain, "Biometric Recognition: Security and Privacy Concerns," *IEEE Security and Privacy*, vol. 1, no. 2, pp. 33–42, Mar.–Apr. 2003.
- [6] R. K. Rowe, K. A. Nixon, and P. W. Butler, "Multispectral Fingerprint Image Acquisition," in *Advances in Biometrics*, Springer, Berlin, Heidelberg, pp. 3–18, 2008.
- [7] T. Ahmad, J. Hu, and S. Wang, "Pair-polar Coordinate-based Cancelable Fingerprint Templates," *Pattern Recognition*, vol. 44, no. 10–11, pp. 2555–2564, 2011.
- [8] B. Rattani, B. Freni, G. L. Marcialis, and F. Roli, "On-line Fingerprint Template Update Using the Self-update and Graph-based Algorithms," in *Proc. 4th International Conference on Advances in Biometrics (ICB)*, 2009, pp. 1122–1131.
- [9] G. Jaswal, A. Kaul, and R. Nath, "Biometric Attendance System Using Fingerprint: A Review," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 4, no. 10, pp. 388–393, Oct. 2014.
- [10] V. Bansal, P. Srivastava, and M. Tiwari, "Automated Attendance Management System Using Fingerprint Biometrics," *International Journal of Scientific and Research Publications*, vol. 5, no. 3, pp. 1–4, Mar. 2015.
- [11] R. Want, "An Introduction to RFID Technology," *IEEE Pervasive Computing*, vol. 5, no. 1, pp. 25–33, Jan.–Mar. 2006.
- [12] P. S. Deshpande and O. G. Kakde, "Color and Texture Based Approach for Person Recognition Using Fingerprint Image," *International Journal of Computer Science and Network Security*, vol. 7, no. 12, pp. 267–275, Dec. 2007.
- [13] M. Cherrat, R. Alaoui, and H. Benbrahim, "Web Application Based on the MVC Architecture and the JavaScript Framework React.js and Node.js," *IJITEE*, vol. 8, no. 9, pp. 1172–1178, Jul. 2019.
- [14] S. Tilkov and S. Vinoski, "Node.js: Using JavaScript to Build High-Performance Network Programs," *IEEE Internet Computing*, vol. 14, no. 6, pp. 80–83, Nov.–Dec. 2010.
- [15] P. Membrey, D. Plugge, and T. Hawkins, *The Definitive Guide to MongoDB*, 3rd ed. New York, NY: Apress, 2015.