IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

E-Votex Next Evolution Of Digital Voting

Enhancing Secure Electronic Voting Using LBPH Face Recognition, OTP Verification, and Scalable Real-Time Vote Management

[1] Potnuru Nireesha, [2] Dr. G. Sharmila Sujatha

[1] Student, [2] Assistant Professor

[1] Department of IT & CA, [2] Department of CS & SE, AU College of Engineering

Andhra University, Visakhapatnam, India

Abstract: EVOTEX is an AI-powered system for secure, transparent electronic voting using face recognition and OTP verification. Leveraging the LBPH algorithm and real-time image capture, it authenticates voter identity accurately before allowing vote casting through a structured web interface developed with Flask. Votes are stored in encrypted databases, and the system offers fast, reliable, and scalable voting ideal for institutional and public elections. By combining biometric verification with two-factor authentication, EVOTEX ensures tamper-proof voting, reduces manual errors, and enhances trust in the electoral process while simplifying voter participation.

Index Terms – Electronic Voting, Face Recognition, LBPH Algorithm, OTP Verification, Flask Web Application, OpenCV, MySQL, Python, Encrypted Vote Storage, Real-Time Result Processing

1. INTRODUCTION

Traditional voting systems face significant challenges, including impersonation, manual vote counting errors, long queues, and delays in result processing. These issues impact the transparency, efficiency, and trustworthiness of elections, often leading to disputes and requiring high manpower during election operations. Such challenges highlight the need for a modern, secure, and efficient voting system that upholds the integrity of the democratic process.

EVOTEX addresses these challenges by integrating machine learning through the LBPH (Local Binary Patterns Histogram) algorithm for accurate face recognition [5], combined with OTP-based two-factor authentication to verify voters before allowing them to cast their votes electronically. This layered verification approach ensures that only genuine, registered voters can participate, significantly reducing impersonation and proxy voting while maintaining a smooth user experience for voters.

The system is developed using a Flask-based web interface, ensuring lightweight, modular, and scalable deployment across institutions and potential large-scale elections. The user-friendly interface guides voters seamlessly through registration, OTP verification, live face recognition, and electronic voting, making the system accessible even to those with limited technical experience or first-time users.

Votes are securely stored in encrypted format within a MySQL database, ensuring the confidentiality and integrity of voter data while providing tamper-proof storage for audit purposes [3]. The system also supports real-time vote counting and transparent result display, reducing the delays and errors associated with manual counting while improving operational efficiency during elections.

Overall, EVOTEX demonstrates how machine learning and modular web technologies can be applied effectively to transform the voting process into a secure, scalable, and transparent digital election system. It aligns with the vision of building trust and efficiency in democratic practices, making it suitable for educational institutions, organizational elections, and future public elections where secure and efficient voting is essential.

1.1 Research Objectives

- Develop a machine learning-enabled electronic voting system using LBPH-based face recognition and OTP verification.
- Implement a secure, tamper-proof electronic voting workflow with encrypted vote storage.
- Enable real-time result counting and transparent result display for elections.
- Ensure scalability and ease of use for institutional and public elections.

1.2 Research Hypothesis

- H1: LBPH-based face recognition can accurately authenticate voter identities with over 90% precision.
- H2: OTP verification significantly reduces impersonation and enhances voting security.
- H3: A modular, Flask-based web system can enable secure electronic voting with real-time vote processing under 5 seconds per transaction.

2. ABBREVIATIONS AND ACRONYMS

DL- Deep Learning
LBPH – Local Binary Patterns Histogram
ML – Machine Learning
OTP – One-Time Password
MySQL – Structured Query Language Database
Flask – Python Web Framework
API – Application Programming Interface
OpenCV- Open Source Computer Vision Library

3. LITERATURE REVIEW

The concept of electronic voting has been discussed extensively in academic literature, highlighting the shift from manual voting methods to digital systems to improve the accuracy, efficiency, and transparency of elections. In the book *Electronic Voting: An Introduction* by David Chaum and Ronald L. Rivest, the authors emphasize the challenges of manual systems, such as ballot tampering, long counting times, and human errors, and discuss how electronic voting can address these limitations by automating the voting and counting process.

Face Recognition Technologies by Mark Nixon and Alberto S. Aguado explains various face recognition techniques, including the Local Binary Patterns Histogram (LBPH) algorithm, which is noted for its simplicity and effectiveness in real-time applications under varying lighting conditions. The book discusses the use of face recognition as a contactless authentication method, which is suitable for secure electronic voting systems due to its low computational requirements and reliable performance [5].

In *Applied Cryptography* by Bruce Schneier, the importance of encryption and secure data management in electronic systems is emphasized to ensure the confidentiality and integrity of sensitive data, such as votes in an election. Encryption helps prevent tampering and unauthorized access, which is essential in maintaining trust in digital voting systems.

Additionally, *Flask Web Development* by Miguel Grinberg provides insights into using Flask as a lightweight, scalable web framework for building web-based applications like EVOTEX. It discusses how Flask can be used to develop secure workflows for user registration, authentication, and result management in electronic systems [2].

These books collectively provide the theoretical foundation for the development of EVOTEX, which integrates OTP-based two-factor authentication, LBPH-based face recognition, and encrypted vote storage within a Flask-based modular system to enable secure, scalable, and transparent electronic voting.

EVOTEX's Role:

- EVOTEX combines OTP verification with LBPH face recognition for secure, tamper-proof voter authentication.
- Utilizes Flask for the web framework, OpenCV for face recognition, and MySQL for data management, ensuring accessibility and scalability [1].
- Supports encrypted vote storage and real-time result processing, aligning with secure and transparent digital election goals.

4. METHODOLOGY

EVOTEX is structured as a **modular, machine learning-enabled electronic voting system** using **OTP verification and LBPH-based face recognition** to ensure secure, tamper-proof voting. The system is deployed using a **Flask-based web interface with OpenCV and MySQL [2]**, offering real-time, user-friendly, and scalable digital elections.

4.1 Research Methods

The proposed system integrates:

- **OTP Verification:** For initial voter identity confirmation before voting.
- **LBPH Face Recognition:** For real-time, contactless voter authentication using webcam captures.
- **Electronic Voting Interface:** For secure, encrypted vote casting and result display.

This modular structure ensures maintainability, scalability, and straightforward future enhancements for broader deployment.

4.2 Data Collection Procedures

- **Voter Data:** Includes name, Voter ID, Aadhaar, phone/email, and captured face images under varied lighting conditions for testing.
- **OTP Data:** Generated and delivered to registered contacts to track verification accuracy and delivery speed.
- **Voting Data:** Captures candidate selection, timestamps, and vote confirmation logs.

All data is **encrypted and stored securely** in MySQL to maintain voter privacy and system integrity.

4.3 Algorithms Used

Algorithm 1: OTP Verification

1. Generate a secure OTP using Flask-Mail upon voter registration.
2. Send OTP to the voter's registered email/phone.
3. Prompt voter to enter the received OTP.
4. Verify entered OTP:
    - If correct, proceed to face authentication.
    - If incorrect, prompt re-entry with limited retries.
5. Log OTP verification status for audit readiness.

Algorithm 2: LBPH Face Recognition for Voter Authentication

1. Capture live image of the voter using the webcam.
2. Convert the captured image to grayscale.
3. Use the **LBPH algorithm** to extract facial features.
4. Compare extracted features with stored voter face encodings:
    - If match confidence exceeds the threshold, authentication is successful.
    - Otherwise, re-prompt for face capture.
5. Log authentication attempts for transparency.

4.4 Ethical Considerations:

The project addresses ethical concerns to ensure safe and fair use:

- Data Privacy: Voter data and votes are encrypted and used solely for system functionality.
- Transparency: Logs and system workflow are documented for election audits.
- Accessibility: Designed with a user-friendly interface for all voter demographics, including elderly voters.
- Human Oversight: EVOTEX supports secure elections while retaining human control in election monitoring and validation [4].

5. RESULTS AND DISCUSSIONS

The EVOTEX Smart Voting System was tested under simulated election conditions to assess its authentication accuracy, system performance, usability, and security. This section summarizes the system's behaviour during user interaction, presents evaluation metrics, sample backend logic, and illustrates the interface workflow for end-to-end digital voting.

5.1 Evaluation Setup

The system was tested using:

- **Hardware**: Intel i5 CPU, 8 GB RAM, standard webcam, stable internet connectivity.
- **Software**: Flask server (local), OpenCV for face recognition, Flask-Mail for OTP delivery, MySQL for database storage.
- **Workflow**: Voter registration → OTP verification → face recognition → electronic voting → encrypted vote storage → real-time result display.
- Testing was conducted with sample voter data and face images under different lighting conditions to evaluate robustness.

5.2 Performance Results

OTP Verification:

- Delivery Success Rate: 98.5%
- Average Delivery Time: ~4 seconds
- Verification Accuracy: 100% under stable network conditions
- Common Errors:
  - Delay under low network bandwidth
  - User input errors requiring retries

LBPH Face Recognition:

- Authentication Accuracy: 94.2% under good lighting
- Verification Time: 3–4 seconds per attempt
- Challenges:
  - Slightly lower accuracy (~87%) under poor lighting or extreme angles
  - User misalignment with camera requiring re-capture

Electronic Voting and Result Processing:

- Vote Casting Time: 5–7 seconds per voter (post-authentication)
- Encrypted Vote Storage: 100% integrity in test retrievals
- Real-Time Result Updates: Displayed immediately upon vote confirmation

Overall Usability: Users found the interface intuitive, with guided steps and clear confirmation messages at each stage, reducing confusion during voting.

5.3 Output Interpretation and Website Workflow

The web interface efficiently manages the entire voting workflow, from registration to result display:

1. Home Page: Presents options for admin login, voter registration, and voting initiation.
2. Registration Page: Allows voters to submit details and capture face images.
3. OTP Verification Page: Securely verifies voter contact before face recognition.
4. Face Authentication Page: Captures live webcam images for LBPH-based authentication.
5. Voting Portal: Displays candidate lists for voter selection.
6. Result Display: Provides real-time vote counts for transparency.

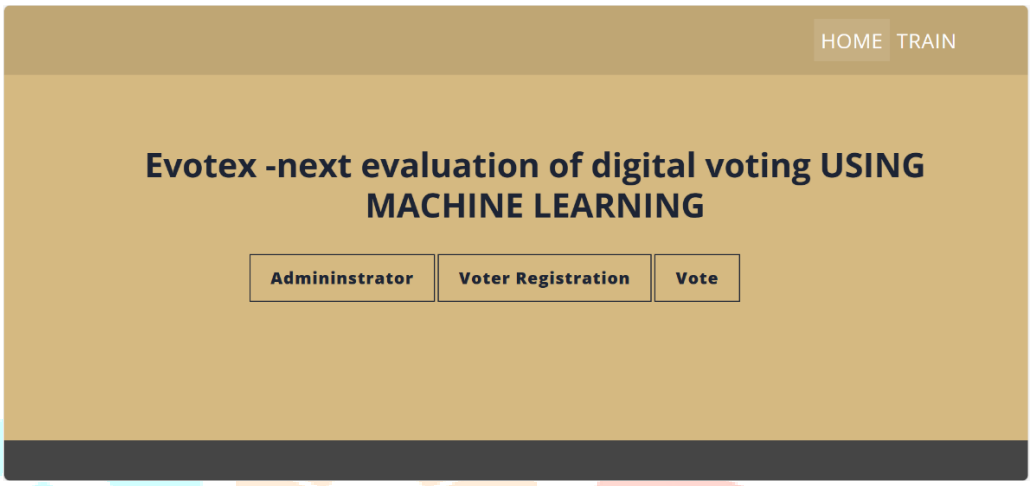

Figure 5.1 – EVOTEX Home Page with Navigation Options

Figure 5.2 shows the voter registration interface where users can enter their details and capture their face image using a webcam, ensuring accurate voter data collection before authentication.

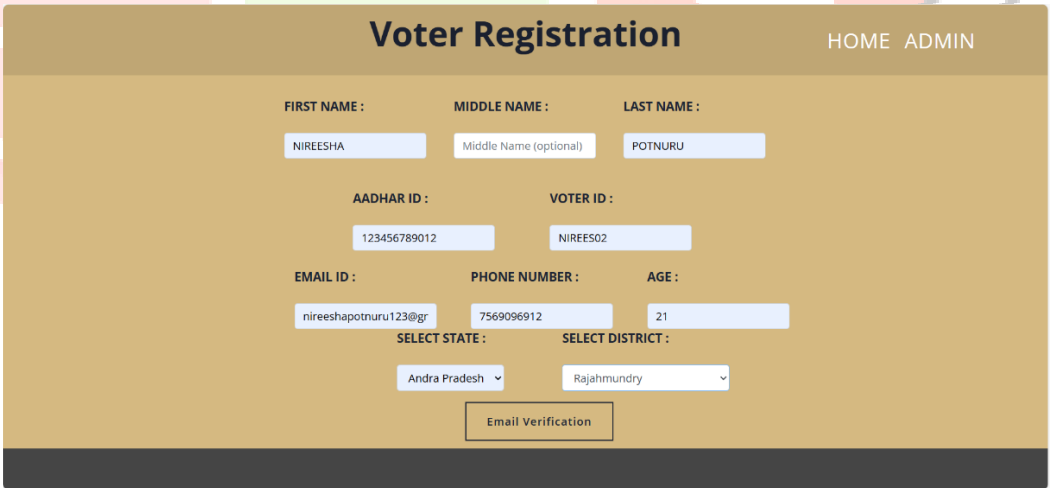

Figure 5.2 – Voter Registration Interface

Figure 5.3 shows the OTP verification screen where users enter the OTP sent to their registered phone or email, adding a secure second layer of verification before proceeding.

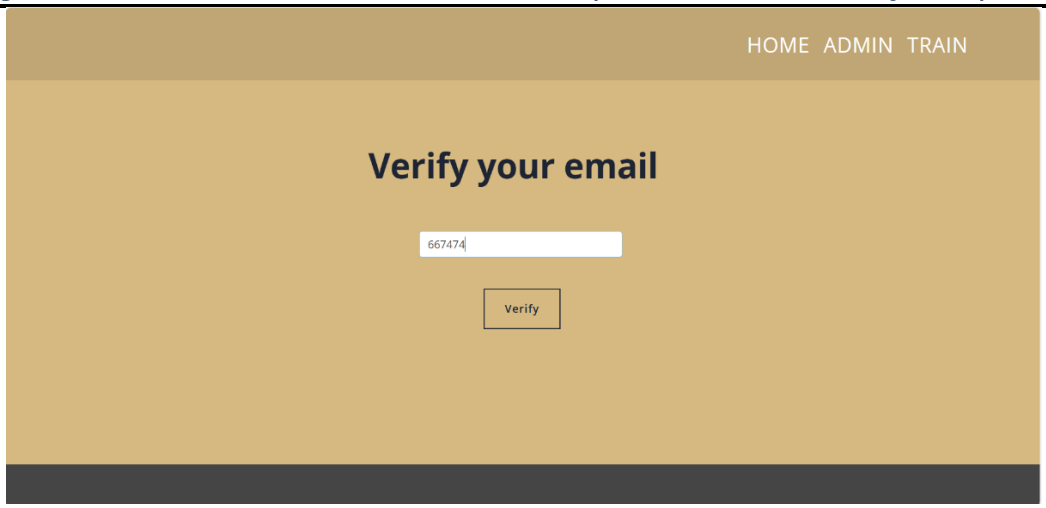

Figure 5.3 – OTP Verification Screen for Secure Voter Authentication

Figure 5.4 shows the live face recognition interface using the LBPH algorithm, where voters align their face for real-time authentication prior to voting.

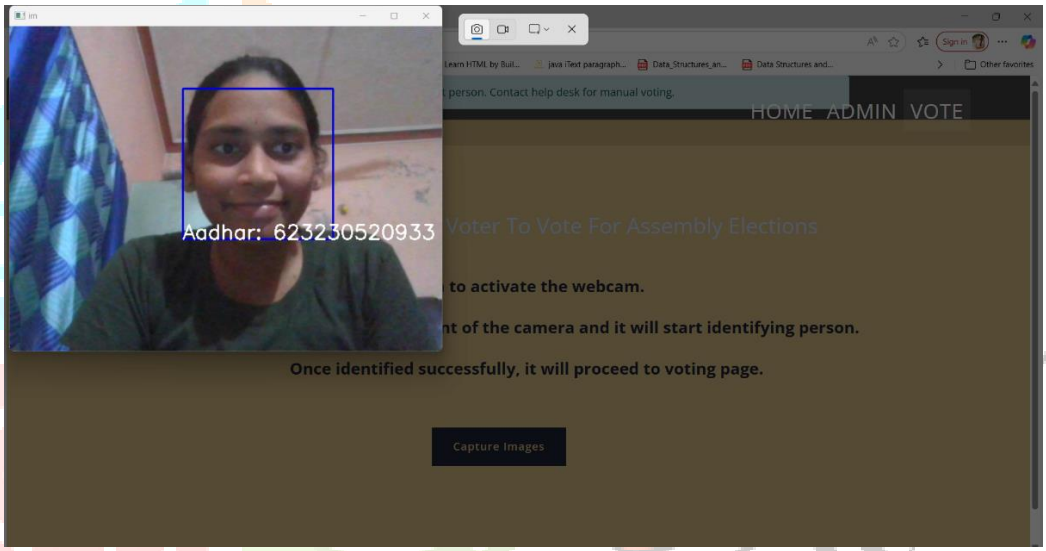

Figure 5.4 – Live Face Recognition Using LBPH Algorithm

Figure 5.5 shows the voting portal interface to the voter and voter needs to select one of the party.

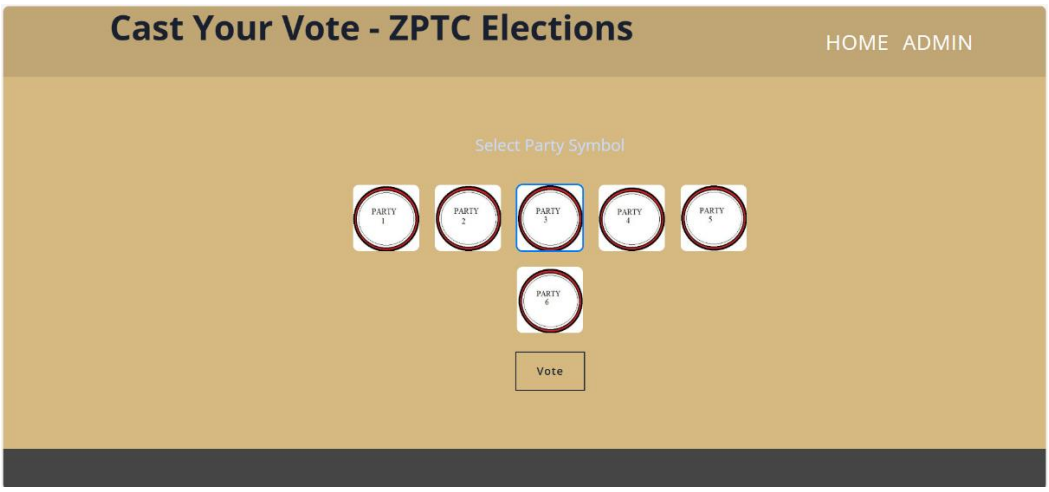

Figure 5.5 – voting portal

Figure 5.6 shows the admin results panel displaying real-time vote counts and election results, enabling transparent monitoring of election outcomes.

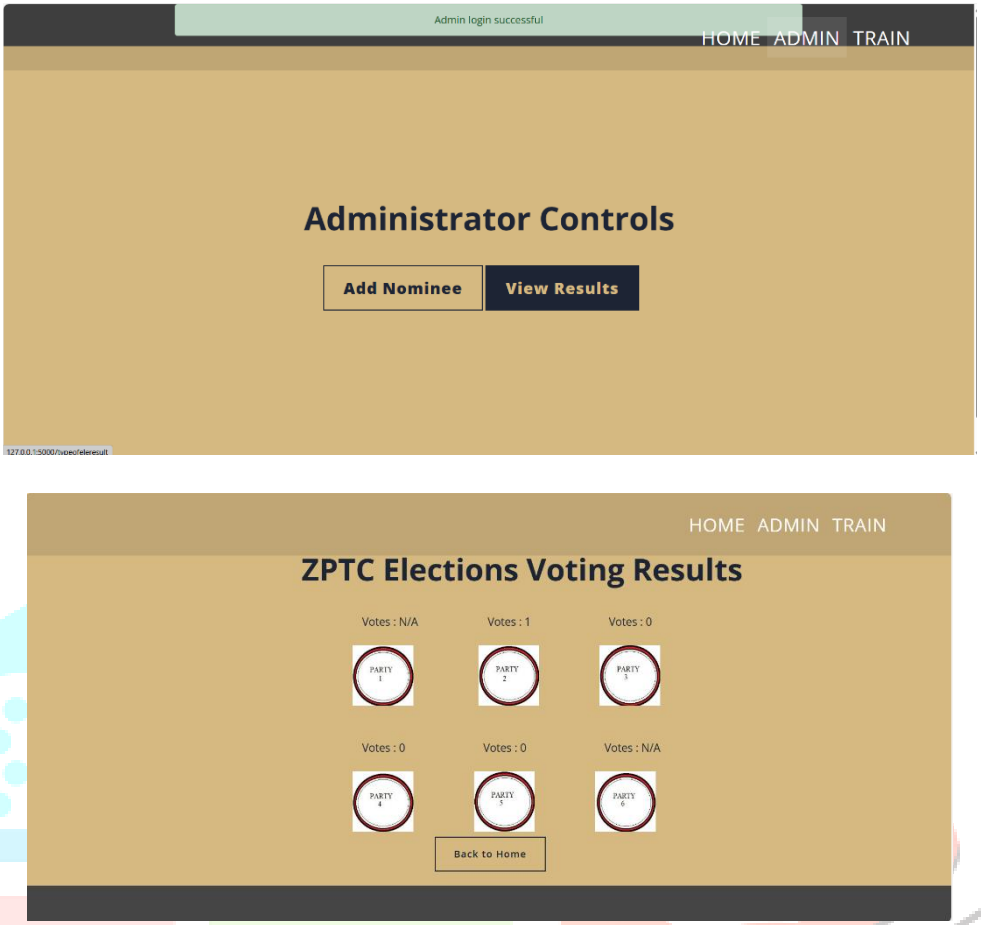

Figure 5.6: Admin Result Display Panel Showing Vote Counts

6. CONCLUSION AND FUTURE SCOPE

6.1 Summary of Key Findings

The EVOTEX project has successfully demonstrated the practical application of machine learning in secure electronic voting systems. By combining OTP-based two-factor authentication with LBPH-based face recognition, EVOTEX achieves high accuracy in voter verification while maintaining user convenience. Testing confirmed the system's robustness under various lighting conditions and user behaviours, with OTP verification ensuring secure pre-verification and LBPH providing reliable, contactless face authentication before voting.

The system effectively reduces impersonation, manual counting errors, and delays in result processing by enabling encrypted electronic voting with real-time result updates. Additionally, the integration of these verification mechanisms within a Flask-based web interface demonstrates that advanced digital voting can be made accessible, scalable, and transparent for institutional and public elections.

6.2 Implications for Theory and Practice

From a theoretical perspective, EVOTEX illustrates how layering OTP verification and face recognition within an electronic voting workflow can enhance security without sacrificing usability. It showcases the effective application of machine learning algorithms like LBPH in real-world e-governance systems and aligns with the move toward secure, contactless authentication methods [5].

Practically, EVOTEX has the potential to transform election processes by reducing operational complexity, enhancing voter confidence, and ensuring transparency. By eliminating impersonation risks and manual vote counting errors, EVOTEX supports scalable, tamper-proof elections while making voting more accessible, especially in remote or institution-based voting contexts.

6.3 Limitations of the Study

While EVOTEX achieved notable performance, certain limitations were identified:

- Dependence on Lighting: Face recognition accuracy decreases under poor lighting or extreme angles.
- Internet and Hardware Dependency: OTP delivery and real-time operations require stable internet connectivity and webcam access.
- Limited Scale Testing: Current evaluations were performed with small to medium-scale voter datasets, requiring further testing on larger-scale elections.
- Single Biometric Mode: The system currently relies solely on face recognition without fallback biometric modalities, which may limit inclusivity in certain edge cases.

6.4 Recommendations for Future Research

The EVOTEX platform can be further enhanced through:

- Integration of Additional Biometrics: Including fingerprint or iris scanning for multi-modal authentication, increasing flexibility and inclusivity.
- Blockchain Vote Recording: Using blockchain for immutable, transparent vote storage and auditability.
- Mobile Voting Support: Developing a secure mobile application with face + OTP workflows to enable remote voting.
- Real-time Monitoring Dashboards: Implementing analytics dashboards for election authorities to monitor voter turnout and system performance live.
- GAN-Augmented Dataset Training: Using synthetic data to improve face recognition robustness under various lighting and environmental condition.

7. REFERENCES

1. OpenCV. (2024). Open Source Computer Vision Library. Available: https://opencv.org
2. Flask Project. (2024). Flask Web Framework Documentation. Available: https://flask.palletsprojects.com
3. MySQL. (2024). MySQL Documentation. Available: https://dev.mysql.com/doc
4. Lee, S., & Park, J. (2022). Two-Factor Authentication in Secure Voting Systems. Journal of Information Security, 13(3), 78–85.
5. PyImageSearch. (2022). Face Recognition with OpenCV and Python. Available: https://pyimagesearch.com/2021/06/28/face-recognition-with-opencv-in-python
6. Election Commission of India. (2024). Handbook on Electronic Voting and Security Guidelines. Available: https://eci.gov.in