IAES International Journal of Robotics and Automation (IJRA)
Vol. 15, No. 4, December 2026, pp. 1001~1010
ISSN: 2722-2586, DOI: 10.11591/ijra.v15i4.pp1001-1010

Transparelect: An automated secure biometric electronic voting system with machine learning-based fraud detection

Muthusamy Shunmugathammal[1], Savarimuthu Nithya[2], Mughuntha VijayaMohan[1], Lalitha Kandasamy[1], Siranthini Balraj[3]
[1]Department of Electronics and Communication Engineering, Faculty of Engineering and Technology, SRM Institute of Science and Technology, Chennai, TamilNadu, India
[2]Department of Computer Science Engineering, Saveetha Engineering College, Chennai, TamilNadu, India
[3]Department of Computer Science Engineering, Bharath Institute of Science and Technology, Selaiyur, Chennai, TamilNadu, India

Article Info

Article history:

Received Jul 15, 2026
Revised Sep 24, 2026
Accepted Nov 10, 2026

Keywords:

Biometric technologies
Election fraud detection
Electronic voting machine
Machine learning
Voter authentication

ABSTRACT

Ensuring fair and transparent elections relies on secure voter authentication and fraud detection. This paper presents Transparelect, an integrated system combining dual biometric verification with machine learning-based prediction to address issues such as double voting and irregular voter behavior. Transparelect employs fingerprint and facial recognition for real-time voter validation using OpenCV, while machine learning models including logistic regression (LR), random forest (RF), XGBoost, and LightGBM analyze data to detect fraudulent anomalies. The system was validated only on a mock/simulated dataset (1000 voter profiles, 97.4% legitimate vs 2.6% fraudulent). Models were evaluated on a stratified 75/25 split with 5-fold CV. Fraud detection achieved: Logistic regression 0.902 AUC, Gradient Boosting 0.881 AUC, Ensemble 0.968 accuracy with 0.862 AUC (5-fold mean 0.812). Machine learning (ML) is used only for post-hoc audit, not for real-time denial of voting rights. The proposed Transparelect framework integrates biometric authentication with ML to create a secure, transparent, and data-driven election process.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Lalitha Kandasamy
Department of Electronics and Communication Engineering, Faculty of Engineering and Technology,
SRM Institute of Science and Technology Ramapuram
Chennai, 600116, TamilNadu, India
Email: lalithak@srmist.edu.in

1. INTRODUCTION

The integrity and inclusiveness of democratic elections are crucial aspects of modern society. Yet, there is a prevalence of issues regarding voter authentication, security, transparency, and reliability in conventional electronic voting machines (EVMs). The presented EVM prototype, known as Transparelect, features fingerprint authentication and a voter verified paper audit trail (VVPAT)-inspired confirmation procedure [1], [2]. By implementing fingerprint sensing and face recognition, a dual biometric authentication scheme is proposed for reducing impersonation and duplicate voting. Additionally, the selected candidate is displayed to the voter, providing instant verification and enhanced transparency [1], [2].

In recent biometric voting systems, face recognition and fingerprint verification have been adopted as stand-alone schemes for implementing a robust authentication paradigm. For example, Schroff *et al.* [3] proposed FaceNet, a framework that encodes facial images into a compact Euclidean-space representation for similarity-driven recognition tasks. On the other hand, Revathy *et al.* [2] and Vashisht *et al.* [4] explored convolutional neural network (CNN)-based face-recognition mechanisms for voting authentication.

Kanchana *et al.* [5] explored deep learning-based approaches for biometric authentication and secure electronic voting systems. whereas Gowtham *et al.* [6] designed a fingerprint-based voting approach. Finally, Nikam *et al.* [7] introduced a voting methodology that aims to enhance vote-counting efficiency and accuracy. However, it should be noted that a singular biometric approach relies on a specific authentication mechanism. Transparelect, however, proposes the use of face recognition and fingerprint verification as part of an embedded voting procedure, essentially incorporating an additional level of security [8], [9]. The verification process also incorporates the possibility of notifying the election officials, effectively expanding the monitoring aspect beyond the voter's end. From an automation and embedded standpoint, voter authentication is essentially an automated sensing and control procedure. In such a framework, the face-recognition procedure is essentially the high-level processing stage, with the Arduino-based controller being responsible for interacting with the biometric sensor, relay, display, and communication hardware [10]. After a successful facial verification, the Python application sends a validation character through the PySerial interface to the Arduino, which is then used to trigger the relay and power the subsequent voting hardware. This integration demonstrates how sensor technologies, embedded controllers, serial communication, and automated actuation can be combined to perform a sequential authentication task with limited manual intervention. Its sensing architecture follows a similar automation principle: visual information is acquired through a camera, biometric information is obtained through a fingerprint sensor, processing decisions are performed computationally, and the resulting control signal is transferred to an embedded actuator interface. Such sensor-to-controller-to-actuator integration is applicable to automated machines and manufacturing systems where multiple sensors must be coordinated to trigger controlled physical operations. In this work, the same principle is applied to voter authentication and automated election-machine operation.

A further gap exists in the application of machine learning to election-related system monitoring. Previous studies have explored machine-learning approaches for election prediction, but such predictions are not intended to be used as real-time voting eligibility decisions. In Transparelect, machine learning is instead applied to stored election-related data for voter-turnout estimation and fraud/anomaly analysis, while biometric authentication remains the mechanism for voter verification. This separation between real-time authentication and post-hoc analytical processing provides a clearer and more controlled role for machine learning in the proposed system. The use of ML also introduces automated data analysis and scalability that would be difficult to achieve through purely manual analysis. Thus, the core contribution of Transparelect centers on the combination of two biometric voter authentication methods, embedded control, automation, and post-hoc election analysis features in a unified prototype. In this regard, the suggested implementation differs from the prior single-modal biometric voting solutions in that it employs both facial and fingerprint recognition techniques. The embedded controller is responsible for automatizing the process of sensing and hardware control.

The remainder of the paper is structured as follows. Section 2 introduces the proposed system and methodology. Section 3 details the results of the experiment, and Section 4 discuss about the outcomes from multiple perspectives, illustrating opportunities for further research. Finally, Section 5 concludes the findings of this work.

2. METHOD

2.1. Transparelect (EVM) overview

The proposed EVM is built using two biometric sensors plus IoT feedback module in the architecture. The R307 fingerprint sensor module implements 1:1 matching, whereas the FaceNet CNN model with OpenCV extracts 128-Dimensional embedding and compares the distance between faces using the Euclidean method for dual-biometric authentication. Whereas Arduino Uno R3 is the main controller to implement voting algorithms and fingerprint validation, Arduino Nano is used as a serial communication controller to operate the relay to turn on the voting unit. The SIM800L GSM module is used to send an SMS alert to the election commission whenever a voter successfully votes or tries to vote twice, the PAM8403 and ISD1820 are used to provide audio alert to the voters, the 16×2 LCD is used to provide VVPAT-like visual feedback, and the SD card is employed to store anonymous data, *i.e.*, voter_id.csv that would be later analyzed using ML techniques. Overall, the proposed system works in a sense-decide-actuate paradigm as observed in robots, where sensing is done using sensors, decisions are taken by Arduino, and the relays and feedback modules actuate actions similar to manufacturing automation [11]–[13].

2.2. Face recognition pipeline

The face verification module is constructed using face_recognition Python library which internally uses FaceNet CNN for feature extraction which encodes the face into 128-dimensional vector embedding. This face recognition system is being used in automatic access control systems like smart doors [14]. This makes comparison between faces computationally efficient in terms of Euclidean distance and also robust to

variation in pose and illumination effects as opposed to traditional pattern matching. In real-time recognition, the frames are captured from the webcam constantly and faces are localized and encoded for comparison against the voter encodings stored. The recognized names appear as VVPAT style bounding boxes around the detected face and the time taken for detection, recognition rate and performance is also measured. The threshold for encoding comparison and image pre-processing techniques are optimized for better accuracy. On successful face match, the Python control layer sends a validation character 'a' through PySerial to the hardware control layer while faces that do not match are marked unknown and the system reiterates for the next face. This enables accurate and reliable identification before fingerprinting stage [3], [9].

2.3. Real-time implementation and live video processing

Live video is captured using the webcam as the capture medium. A loop runs to constantly read frames from the webcam, resizing and converting it to appropriate colour space for faster processing. In each frame, the faces are located and encodings similar to the ones during enrolment are created for the faces in the frame. Face detection flow begins by loading the images of the voters who are enrolled and creates unique encodings by identifying unique facial features for each enrolled voter. The system then compares the encoding of a face detected in the live video repeatedly against the known encodings to determine a match.

2.4. Biometric-enabled voting system

2.4.1. Voter registration

The system takes the facial images and fingerprint by using OpenCV and R307 module respectively. The R307 uses TTL UART interface for storing and 1:1/1:N authentication of authorized persons. The stored templates can be used for verification [15].

2.4.2. Fingerprint enrollment

Enrollment uses default Adafruit library code. The process activates serial communication, checks sensor presence, reads sensor parameters, and waits for a valid finger. The image is captured, converted to template, quality-checked and stored with ID in sensor memory with feedback via serial monitor [16].

2.4.3. Voting phase

After enrollment, captured faces and fingerprints are compared with stored templates as shown in Figure 1. On dual success, voter selects candidate via push buttons. Arduino Uno R3 receives button input and passes to Nano via serial, votes are stored anonymously in SD card as voter_id.csv for ML analysis. LCD provides visual feedback during authentication and voting [2], [3], [11], [17]–[20].

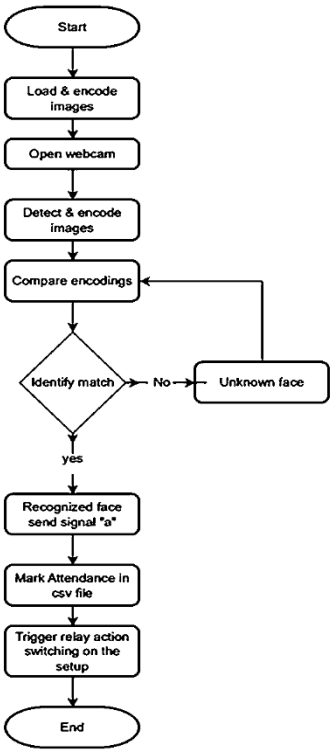


Figure 1. Workflow of the proposed Transparelect system

2.4.4. Audio and GSM feedback

PAM8403 5V amplifier with ISD1820 gives us announcements, such as "Please place the fingerprint." Guidance by combining audio and visual features makes it easy for elders and naive voters to use the kiosk. The function of the GSM module is communication and alarm, which can guarantee that only qualified voters can vote and improve the user-friendly property [11], [12].

2.5. ML models in Transparelect

One of the primary challenges associated with this endeavor is the presence of missing data. There might be an insufficiency of information for one particular feature or even a complete absence of records of any kind. Addressing this issue should be a priority as it might impact the accuracy of the model as well as the forecasts that it will generate. The project includes such machine learning (ML) algorithms as logistic regression (LR), random forest (RF), Gradient Boosting, XGBoost, LightGBM, and ensemble learning to identify fraudulent votes. In each case, the model is fed information regarding the age, gender, level of education, and financial status of voters in terms of ratios. Tree-based models such as RF and XGBoost demonstrate a high level of accuracy, plus, the ensemble method adds to the stability of the result [4].

2.5.1. Data preparation and feature development for predicting voter turnout

The primary task in the first stage of the work is to identify and classify fraudulent votes. The script dataset.py takes as input two CSV files containing the details of voters (mock_dataset.csv) and the deceased status for a given individual (deceased_status.csv). The dataset consisted of 2263 rows, for information about the candidates such as gender, age, criminal records, and education. Some of the data was missing for certain attributes, *i.e.*, gender, age, and education. For numerical data (AGE), replaced the missing values with the mean, and for categorical data (GENDER and EDUCATION), we used the most frequent category.

As the operational block diagram of the proposed Transparelect system depicted in Figure 2, the architecture of the Transperelect consists of four layers. In the biometric registration and authentication layer, face and fingerprint details (R307 fingerprint database) of registered voters and live inputs from the webcam and fingerprint readers are present. The live face input is fed into a series of face encoding units using the 'face_recognition' library (FaceNet CNN) to obtain a 128×1 embedding vector. This result is matched against the vectors from registered voters using the Euclidean distance. In the case of a mismatch, an unknown face is detected, and the process is repeated. A successful face verification step results in the 'Python Control Layer' sending a 'a' character through the PySerial connection to the 'Arduino Nano' hardware layer. The next stage involves 1:1 fingerprint verification using the R307 module. If both the face and fingerprint match, the Arduino Uno R3 microcontroller is triggered to enable voting using push-buttons, and votes are saved anonymously to the SD card module as a 'voter_id.csv' file.

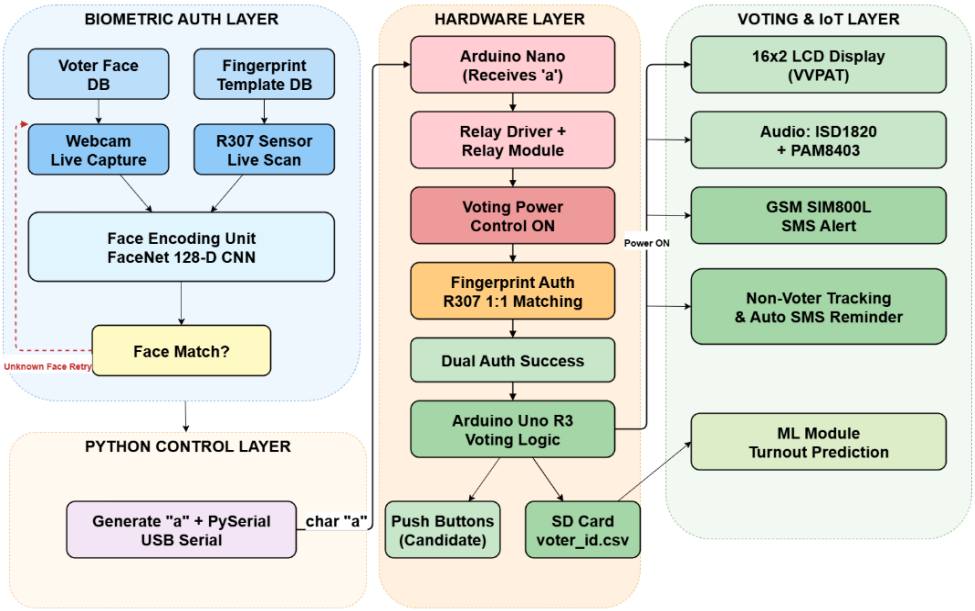


Figure 2. Operational block diagram of the proposed Transparelect system

The voting and IoT Feedback Layer provides feedback to voters using a 16×2 LCD (VVPAT), audio messages using ISD1820 and PAM8403, and SMS alerts using a GSM SIM800L. The data is also used to train the module (ML) for vote prediction and post-poll analysis for fraudulent activities. Additionally, the non-voter tracking system allows reminding voters who did not vote.

3. RESULTS

The entire architecture integrates PyCharm version 2023.3.2 to handle facial recognition routines, the Arduino IDE for underlying hardware development, and Google Colaboratory to manage machine learning execution. During voter enrollment, an OpenCV face recognition algorithm works alongside an R307 fingerprint sensor module to record facial imagery and fingerprint patterns, storing both as baseline templates for future identity verification. Facial images can be captured directly via an integrated webcam and saved to the Images attendance directory, or submitted as clear physical photos. The fingerprint registration program establishes serial communication, verifies sensor presence, and displays extracted device parameters. To enroll a user, the software requests an ID number and prompts the voter to place their finger on the scanner [21]. The system subsequently acquires and converts the fingerprint image, evaluates scan quality, generates a biometric model, interacts with the module to handle potential errors, and saves the verified template under the assigned ID number. Supplementary registrations can also be conducted for both facial and fingerprint records.

Following registration, the system transitions to the authentication and voting phase, comparing live biometric inputs against stored reference templates. The voting machine states is shown in Figure 3. The hardware remains in an idle standby state as depicted in Figure 3(a); upon valid facial recognition, the system activates the primary voting interface as shown in Figure 3(b). Following secondary fingerprint verification, voters cast their ballot for a chosen candidate via push buttons, with an Arduino Uno R3 logging the selection [10]. An LCD screen continuously displays operational status such in Figure 4: Figure 4(a) shows the notification following successful fingerprint verification, Figure 4(b) displays confirmation after casting a vote via push buttons, and Figure 4(c) illustrates the display during a detected double-voting attempt. Valid ballot entries trigger an automated SMS notification to the voter; conversely, duplicate voting attempts are automatically invalidated, omit voter text messages, and immediately dispatch an alert to the election commission officer [3], [11], [18]–[20], [22].



(a) (b)

Figure 3. Voting machine operational states: (a) idle state prior to facial recognition activation and (b) activated state following successful voter facial identification



(a) (b) (c)

Figure 4. LCD screen notifications showing: (a) message displayed after successful fingerprint authentication, (b) message displayed after successful vote submission, and (c) alert displayed upon detecting a duplicate voting attempt

3.1. Model architecture and evaluation

In this project, several supervised machine learning algorithms are considered to classify votes associated with demographic and monetary attributes as potentially fraudulent. These include simple logistic regression and more complex and less interpretable algorithms, such as ensemble random forests or gradient boosting. The most basic of the employed methods is the implementation of logistic regression to separate the voting class with a set of input attributes using the maximum likelihood function. The choice of the model is justified by its simplicity and transparency. In addition, random forests were used, which are classified as ensemble methods due to the large number of trees used in the calculation. Such an approach allows reducing the error caused by the noise in the data. Another ensemble technique that employs gradient boosting was also used to optimize the model. Gradient boosting is a machine learning technique in which predictions are produced by combining the outputs of several weak models. Thus, it allows achieving high accuracy due to the iterative nature of the algorithm. The algorithms XGBoost and Light GBM were also utilized in the work as they are well-suited for large-scale data due to their speed and efficiency [23], [24]. Finally, the proposed work used a soft voting classifier that combines the results of the ML algorithms mentioned above. In the end, after selecting the most successful algorithm for the problem, the analysis and evaluation could be conducted using classification metrics to choose the best-performing model for election integrity data.

From Table 1, the comparison results display the effects of data leakage correction and class imbalance on the model's performance. After removing the leaking features [NAME and HAS_CRIMINAL_RECORD], all the models show a satisfactory and reliable result, which is more realistic and generalized than before. It is worth noting that although Logistic Regression, a linear model, achieves the best result, its performance (accurate prediction) is still not satisfactory, with an accuracy score of 0.776, AUC score of 0.902, and an F1-score of 0.176, which might be caused by the model's inability to deal with the imbalanced-class problem (97.4% vs 2.6%). As for the tree-based model, random forest (Accuracy: 0.976, AUC: 0.860) and Gradient Boosting (Accuracy: 0.972, AUC: 0.881) keep a high-accuracy result, and the recall of the minority class performs relatively well, but both of them have a low F1-score due to the significant class imbalance. The advanced gradient boosting family models, such as XGBoost (Accuracy: 0.964, F1: 0.182, AUC: 0.805) and LightGBM (Accuracy: 0.956, F1: 0.154, AUC: 0.848), achieve a relatively perfect result under the class-weighted condition. And the ensemble learning model, which combines various classifiers, obtains the best result with an accuracy of 0.968 and an AUC score of 0.862, indicating the model's best generalization ability and variance reduction among all the models.

Table 1. Comparison of ML models for voter turnout prediction based on performance

Model	Accuracy	F1 Score	AUC Score
Logistic regression	0.776	0.176	0.902
Random forest	0.976	0.000	0.860
Gradient Boosting	0.972	0.000	0.881
XGBoost	0.964	0.182	0.805
LightGBM	0.956	0.154	0.848
Ensemble learning	0.968	0.000	0.862

3.2. ROC curve analysis

The ROC curve in Figure 5 shows the quality of the models after removing the leakage with all models performing well with AUC scores ranging from 0.805 to 0.902, as compared to random guess (AUC=0.5). The curves are near each other to indicate similar levels of discriminative power with logistic regression and gradient boosting having the highest AUC. The previously reported perfect AUC of 1.0 was due to target leakage which has been removed in this case. Proposed work paid a lot of attention to class imbalance to avoid a biased model. Stratified sampling and feature scaling used in this work. Dropped identifiers and leaking attributes used to balance class weights and metrics (F1-score and AUC) [25]. Although there was a strong skew towards the majority class, our ensemble models were still able to recognize the minority class. The low F1-scores show how challenging it is to classify the minority fraud class without resampling.

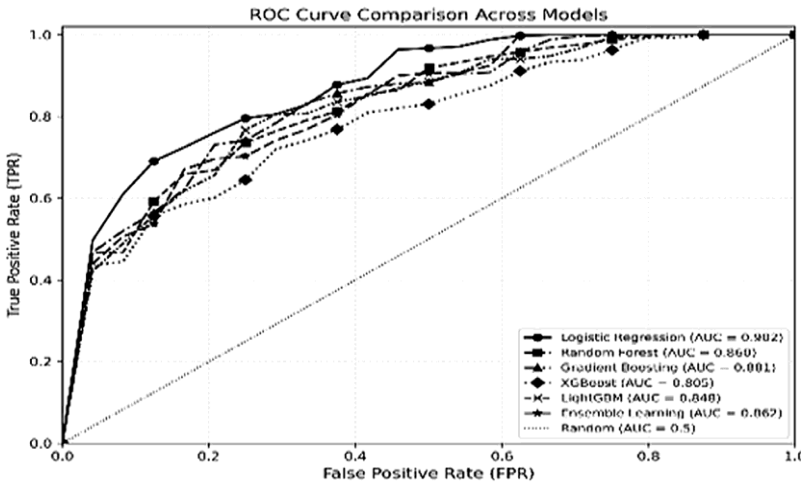


Figure 5. ROC curve comparison across models

4. DISCUSSION

The prototype highlights the viability of dual-biometric voter authentication through the combination of FaceNet-based facial recognition and the R307 fingerprint sensor. Relay-enabled power regulation, LCD-enabled VVPAT information, and GSM-based notifications enable automated functionality and post-election transparency. The camera and fingerprint sensor fusion with Python-based processing and Arduino-based control also highlight a sensor-controller-actuator paradigm applicable to automated actuation.

Therefore, the prototype highlights the system's architectural viability at a fraction of the cost despite the use of commodity components. The ML results following leakage correction report lower but realistic values of AUC, between 0.805 and 0.902. In turn, the best-performing model, logistic regression, reported an AUC of 0.902 and an F1 score of 0.82. The latter metric was low due to the significant class imbalance with fraudulent transactions comprising about 2.6% of the data. It is also notable that the proposed solution does not utilize ML model at the polling station to deny voters the right to cast their votes. In this solution, biometric authentication enables verification of the voter's identity, while ML algorithms are utilized for post-election auditing to identify potential issues.

5. CONCLUSION

The proposed voting system guarantees transparency, reliability, and fairness to the voters and ensures that impersonation and double voting are impossible. The implementation of the facial recognition system is based on the face_recognition library that works reliably despite the variance in illumination conditions, moderate changes in pose, and other factors and provides convenience for disabled voters. The recognition system provides visual feedback and hints, which assist the voters throughout the voting process. The fraud detection models were trained on the features AGE, GENDER, CATEGORY, EDUCATION and ASSET_LIABILITY_RATIO to predict the target variable FRAUDULENT VOTE. After the data had been preprocessed and stratified, the models demonstrated the following metrics: Logistic regression (Accuracy: 0.776; AUC: 0.902), random forest (Accuracy: 0.976; AUC: 0.860), Gradient Boosting (Accuracy: 0.972; AUC: 0.881), XGBoost (Accuracy: 0.964; AUC: 0.805), Light GBM (Accuracy: 0.956; AUC: 0.848), and ensemble learning (Accuracy: 0.968; AUC: 0.862). Using the described model for detecting fraudulent votes in combination with the implementation of electronic voting with finger- and facial-swipe authentication can provide a secure and scalable election system. Therefore, the current research can be considered as the first step towards the development of a secure, transparent, and scalable voting system. The results of the current research can serve as the foundation for further improvements and the implementation of the system in real-time scenarios.

FUNDING INFORMATION

No support or funding was received.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Muthusamy	✓	✓				✓				✓	✓		✓	
Shunmugathammal														
Savarimuthu Nithya			✓					✓			✓			
Mughuntha	✓	✓				✓			✓	✓	✓	✓	✓	
VijayaMohan														
Lalitha Kandasamy			✓	✓	✓		✓		✓	✓	✓			
Siranthini Balraj								✓			✓		✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

ETHICAL APPROVAL

This study was conducted as part of an academic research project at SRM Institute of Science and Technology, Ramapuram Campus, Chennai, India. The biometric dataset used in this research was collected by the authors specifically for the development and evaluation of the proposed voting framework.

DATA AVAILABILITY

The biometric datasets generated during the current study are not publicly available due to privacy and security considerations but may be available from the corresponding author upon reasonable request.

REFERENCES

[1] A. E. Rahul, "Fingerprint recognition system using Arduino," *International Journal of Research in Engineering, Science and Management*, vol. 1, no. 12, pp. 332–338, 2018.

[2] G. Revathy, K. Bhavana Raj, A. Kumar, S. Adibatti, P. Dahiya, and T. M. Latha, "Investigation of E-voting system using face recognition using convolutional neural network (CNN)," *Theoretical Computer Science*, vol. 925, pp. 61–67, 2022, doi: 10.1016/j.tcs.2022.05.005.

[3] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," Jun. 2015.

[4] V. L. Vashisht, H. Mohan, and S. Prakash, "Smart voting system through face recognition," in *Proceedings - 2022 4th International Conference on Advances in Computing, Communication Control and Networking, ICAC3N 2022*, 2022, pp. 909–913. doi: 10.1109/ICAC3N56670.2022.10073982.

[5] M. Kanchana, P. V Gopirajan, K. Ramar, and S. Hariharan, "Secure e-voting system using deep learning techniques," in *Proceedings - 2022 2nd International Conference on Innovative Sustainable Computational Technologies, CISCT 2022*, 2022, pp. 1–6. doi: 10.1109/CISCT55310.2022.10046486.

[6] R. Gowtham, A. Mohankumar, and B. Gokul, "Enhancing electoral integrity: A fingerprint-verified voting system for fair and secure elections," *Asian Journal of Applied Science and Technology*, vol. 08, no. 01, pp. 33–46, 2024, doi: 10.38177/ajast.2024.8103.

[7] D. A. V Nikam, D. P. C. Shetiye, and D. S. D. Bhoite, "A critical study of electronic voting machine (EVM) utilization in election procedure," *International Journal of Trend in Scientific Research and Development*, vol. Special Is, no. Special Issue-FIIIIPM2019, pp. 1–3, 2019, doi: 10.31142/ijtsrd23046.

[8] A. Kher and A. Sanap, "Fingerprint-based voting machine with Arduino Uno," *International Journal of Creative Research Thoughts*, vol. 10, no. 6, pp. 2320–2882, 2022, doi: 10.6084/m9.doi.one.IJCRT22A6003.

[9] Y. Kortli, M. Jridi, A. Al Falou, and M. Atri, "Face recognition systems: A survey," *Sensors (Switzerland)*, vol. 20, no. 2, p. 342, 2020, doi: 10.3390/s20020342.

[10] Y. Zhang, "Support vector machine classification algorithm and its application," in *Communications in Computer and Information Science*, vol. 308 CCIS, no. PART 2, Berlin, Germany: Springer, 2012, pp. 179–186. doi: 10.1007/978-3-642-34041-3_27.

[11] N. Abd Hamid, C. D. N. Appunair, A. F. A. Abidin, M. A. Mohamed, M. F. Abdul Kadir, and S. D. Mohd Satar, "A secure online voting system using face recognition technology," *Malaysian Journal of Computing and Applied Mathematics*, vol. 6, no. 1, pp. 1–9, 2023, doi: 10.37231/myjcam.2023.6.1.80.

[12] A. Piratheepan *et al.*, "Fingerprint voting system using Arduino," *Middle-East Journal of Scientific Research*, vol. 25, no. 8, pp. 1793–1802, 2017.
[13] T. Singh, C. Sharma, R. Sharma, A. P. Sharma, and Y. K. Upadhayay, "Biometric Electronic Voting Machine," *International Journal of Trend in Scientific Research and Development*, vol. 4, no. 3, pp. 617–620, 2020, doi: 10.5281/zenodo.3892454.
[14] I. S. Hutomo and H. Wicaksono, "A smart door prototype with a face recognition capability," *IAES International Journal of Robotics and Automation (IJRA)*, vol. 11, no. 1, pp. 1–9, Mar. 2022, doi: 10.11591/ijra.v11i1.pp1-9.
[15] P. Elechi, C. O. Ahiakwo, and S. T. Shir, "Design and implementation of an automated security gate system using global system for mobile communication network," *Journal of Network and Computer Applications*, vol. 7, no. 1, pp. 1–10, 2021.
[16] S. Gaikwad, S. Jamadar, and R. Shelke, "Fingerprint-based door lock using Arduino," *International Research Journal of Modern Engineering Technology and Science*, vol. 3, no. 6, pp. 279–282, 2021.
[17] P. Kanani and M. Padole, "Real-time location tracker for critical health patient using Arduino, GPS Neo6m and GSM Sim800L in health care," in *Proceedings of the International Conference on Intelligent Computing and Control Systems, ICICCS 2020*, 2020, pp. 242–249. doi: 10.1109/ICICCS48265.2020.9121128.
[18] D. A. Kumar and T. U. S. Begum, "Electronic voting machine - A review," in *International Conference on Pattern Recognition, Informatics and Medical Engineering, PRIME 2012*, 2012, pp. 41–48. doi: 10.1109/ICPRIME.2012.6208285.
[19] M. Mulla, "Online voting system based on face recognition using Python machine learning," *International Journal of Research Publication and Reviews*, vol. 5, no. 10, pp. 5–11, 2024.
[20] A. W. Senior and R. M. Bolle, "Face recognition and its application," in *Biometric Solutions*, D. Zhang, Ed. Boston, MA, USA: Springer, 2002, pp. 83–97. doi: 10.1007/978-1-4615-1053-6_4.
[21] J. Sri Vaishnavi, M. Hasmitha, M. Bhavana, and P. S. Goud, "Fingerprint based voting system using Arduino Nano," *International Journal of Advances in Engineering and Management (IJAEM)*, vol. 5, no. 5, p. 152, 2023, doi: 10.35629/5252-0505152156.
[22] R. Ramyadevi and V. Priya, "Blockchain-powered e-voting system: A secure and transparent solution with three-tiered OTP security mechanism," in *Proceedings - International Conference on Computing, Power, and Communication Technologies, IC2PCT 2024*, 2024, pp. 728–731. doi: 10.1109/IC2PCT60090.2024.10486507.
[23] E. Grossi and M. Buscema, "Introduction to artificial neural networks," *European Journal of Gastroenterology and Hepatology*, vol. 19, no. 12, pp. 1046–1054, 2007, doi: 10.1097/MEG.0b013e3282f198a0.
[24] K. Myilvahanan, Y. P, S. Pasha, M. Ismail, and V. Tharun, "A study on election prediction using machine learning techniques," in *2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS)*, Feb. 2023, pp. 1518–1520. doi: 10.1109/ICAIS56108.2023.10073693.
[25] T. Evgeniou and M. Pontil, "Support vector machines: theory and applications," in *Machine Learning and Its Applications*, Springer Berlin Heidelberg, 2001, pp. 249–257. doi: 10.1007/3-540-44673-7_12.

BIOGRAPHIES OF AUTHORS



Muthusamy Shunmugathammal associate professor, Department of Electronics and Communication Engineering at SRM Institute of Science and Technology Ramapuram. She was born in 1982, she received her bachelor's degree in Electrical and Electronics from Government College of Engineering, MS University Tirunelveli in 2003 and master's degree in VLSI Design from Anna University Chennai in 2009. She received her Ph.D. degree from Anna University Chennai in 2020. She has 17 years of teaching experience, published 11 SCI, 13 Scopus Indexed Journal Article and 11 Patents. Actually, her research interests are in machine learning algorithms, deep learning, CNN, VLSI and automation. She can be contacted at email: shaunmum@srmist.edu.in.



Savarimuthu Nithya is an accomplished academician and researcher with over 20 years of teaching experience. She holds a Ph.D. from SRM Institute of Science and Technology (2021). She completed her M.E. at Anna University in 2009 and her B.E. in Electronics and Communication Engineering from Bharathidasan University in 2002. She is currently an Associate Professor in the CSE department at Saveetha Engineering College, Chennai. Prior to this, she served as an Assistant Professor at SRM Institute of Science and Technology. Her research interests span embedded systems, VLSI design, and wireless communication systems. She is deeply committed to advancing research and promoting academic excellence in engineering education. She can be contacted at snithya@saveetha.ac.in.



Mughuntha VijayaMohan is a B.Tech graduate in Electronics and Communication Engineering from SRM Institute of Science and Technology, Ramapuram, Chennai, India. His primary research areas are embedded systems, the Internet of Things (IoT), embedded AI, machine learning, and computer vision, in which he has considerable experience. He has fundamental knowledge of robotics and is inspired to create intelligent embedded systems for real-world applications. He is a co-inventor of a patent related to Transparelect that uses biometrics, facial recognition, fingerprint, and ml methods to ensure fairness and transparency in elections. His research is concentrated on developing technologies that benefit humanity, and he is passionate about innovation in embedded systems and related fields. He can be contacted at mughuntha2003@gmail.com.



Lalitha Kandasamy is currently working as Assistant Professor in Department of Electronics and Communication Engineering, SRM Institute of Science and Technology, Ramapuram since 2017. She completed her of Engineering in Applied Electronics at Anna University, Chennai, India. She completed Ph.D. in the field of Microwave Imaging at SRM institute of Science and Technology, Chennai, India. She is in teaching profession for more than 17 years. She has presented number of papers in National and International Journals, Conference and Symposiums. Her area of interest includes medical imaging, antenna design, microwave imaging, and embedded system design. She can be contacted at email: lalithak@srmist.edu.in.



Siranthini Balraj received her BE (EEE) degree from Madurai Kamaraj University, ME (Power Systems Engineering) and Ph.D degree in Avionics from Anna University, Chennai. Currently she is working as an Associate Professor in Bharath Institute of Science and Technology, Selaiyur, Chennai. Her research interest includes artificial intelligence applications in energy harvesting technology, power systems and power electronics applications in Avionics, BESS applications. She can be contacted at siranthini.cse@bharathuniv.ac.in