E-Voting using Face Verification

P.Hema Venkata Ramana[1],Munagapati Hema Sree[2],Medepalli Alekhya[3],Bevara Sai Ratna Shaanmmukhii[4]

Department of Computer Science and Engineering, Andhra Loyola Institute of Engineering and Technology,
Jawaharlal Nehru Technological University - Kakinada

[2]hemasree.m.1999@gmail.com

Abstract—India is the world's largest democracy with nearly 900 million eligible voters. The election period in India spans over nearly six weeks for general elections and there is no alternative system working for eligible voters who are at outstations or performance. For this, the face landmark estimation algorithm is used. This algorithm identifies major landmarks that exist on the faces including the iris of an eye, the inner edge of eyebrows, the top of the chin, lips, etc. **willing to cast their vote but not able to do so due to location** constraints. Moreover, there is no special provision made for whom it is difficult to vote in person at the polling station. Service voters have to use postal ballot and go through a tedious process to cast their vote and this process is also prone to human errors. Also in COVID-19 pandemic situations it is difficult to conduct traditional elections.

Our system is an E-Voting system which uses face verification, is designed especially for Service voters for whom it is difficult to cast their votes through the existing system. It provides an efficient, convenient, and secure mechanism for voters to cast their votes. The design of this system will make the voting process more convenient and may, therefore, lead to improving the turnout.

Keywords— Face Verification*, *Encryption*, *Vote Security*, *VotingSystem*, *KNN Algorithm

I. INTRODUCTION

In paper-based voting, people cast their votes with the help of ballot papers. This is mainly used for the people who are on government duty and may not be physically present to cast their vote. So, the service voters can cast their votes with the help of ballot papers. In this process, there can be an error in counting the votes. Sometimes votes are also manipulated and some voters can't be physically present in the voting booth center. Also in COVID-19 pandemic situations it is difficult to conduct traditional voting system.

Creating E-Voting using Face Verification provides security, saves time and also there is no need for the voter to go to the polling station to cast his/her vote and can easily cast his vote by using advanced technology and facial detection.

II. ABOUT THE PROPOSED WORK

Literature Survey

There are many algorithms and classification techniques for every problem in Machine Learning. Choosing a right method is essential to create a customized solution for different problems. This section overlooks similar existing advantages, disadvantages, similarities, measures for evaluating the algorithm, sample value of evaluations and examines the algorithms used and drawbacks. Capturing and identifying facial image of the voter is main task. Facial Recognition is done using Machine learning. A face is initially detected by using KNN algorithm. The whole process of face recognition is divided into 4 following parts:

Detecting faces in Image
Isolating and projecting faces
Face identification

Detecting faces in Image: It is the first step in facial recognition where faces in the image are detected. If there are more than two faces in the image, then the image is discarded(as the only voter should in the frame).

Isolating and Projecting faces: The next step is to warp the faces including eyes, lips, etc perfectly in the frame. Aligning faces accurately during both training and testing gives a boost in database who has the closest measurement with the test image provided.

3) *Face Identification:* The last step is to identify the person from our

The classification of the image is done using KNN (k-nearest neighbour) classifier. KNN is a supervised learning algorithm which classifies the given instance based on the majority of K-nearest neighbor. This algorithm relies on the distance between the feature vectors, and it also has labels associated with each image, so that it can return the name of the person who is being classified. The distance can be calculated with the help of Manhattan distance or Euclidean Distance. The KNN algorithm classifies by finding the most common class among the k-closest examples.

Let X_i be the point which represents the class C_i for all i belongs to n. Here, Xt is the point which represents the class Ct whose label is not known. It should be noted that X_i, Xt ∈ X and C_i, Ct C
∀ i ∈ n.

```
Algorithm 1: KNN Algorithm1:    procedure KNN
2:    Define the distance metric -d (eg. Euclidean Distance, Manhattan Distance, Minkowski Distance etc)
3:    Calculate $d(X_t , X_i)\ \forall\ i \in n$.
4:    Sort all the distances in non-decreasing order.
5:    Define value for ‘K’.
6:    Select the first ‘K’ points from the sorted list.
7:    Let $K_i$ denotes the number of points belonging to the$i^{th}$ class among K points.
8:    If $K_i \geq K_j\ \forall\ i \neq j$ then assign label of class ‘Ci’ to‘Xt’.
9: end procedure
```

In the context of this paper, *Xi* represents the embeddings (fea-tures) for the individual class *Ci* in the array *X* of size ‘*n*’. The value of K is set to 1 so that the only one class which closely resembles the*Ct* is selected. Also, Euclidean distance formula is used as a distancemetric which is given by:

$$d(Xt, Xi) = \sqrt{\sum_{i=1}^{n} (Xt-Xi)^2}$$

Where:

X_t = 128-d feature vector of the test image

X_i = 128-d feature vector of the image in the database

Project Objective

Primary objective of our project is to overcome the disadvantages of traditional voting system and to make the voting process more easy and safe. Through our project we ensure to provide an efficient, convenient and secure mechanism in voting. Not only security we can also eliminate some of the miscellaneous acts such as vote rigging and other such malpractices.

To achieve this we are using facial data verification which helps usto provide security.

. C. Proposed Work

In this section, the proposed work is elaborated at a high- level scope. Here we can understand the user interface and working nature of the application.

Design Methodology:

The proposed e-voting system is used to cast vote using face verification. A webcam is used to capture the images of the voter that will be used as input to the proposed system, and then the captured image is verified with the images in the database. If the image matches then the voter can cast his/her vote otherwise cannotcast the vote.

III. System Architecture:

In System Architecture, we focus on both back-end and front-end performance.

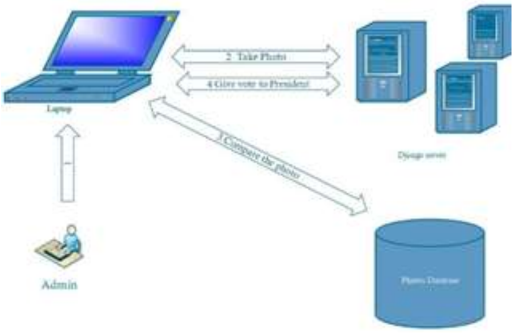


Image Capturing

At first, we begin by taking the consent of the voter by logging into the e-voting system. If they want to cast their vote, the system captures the image of the voter's face using a built-in laptop webcam (or any external camera can be employed). The face image to be correctly processed in the proposed system must contain one face in the frontal position in a uniformly illuminated background.

IV. ALGORITHM

The classification of the image is done using KNN (k-nearest neighbor) classifier. KNN is a supervised learning algorithm which classifies the given instance based on the majority of K- nearest neighbor. This algorithm relies on the distance between the feature vectors, and it also has labels associated with each image, so that it can return the name of the person who is being classified. The distance can be calculated with the help of Manhattan distance or Euclidean Distance. The KNN algorithm classifies by finding the most common class among the k-closest examples.

Let X_i be the point which represents the class C_i for all i belongs to n. Here, Xt is the point which represents the class Ct whose label is not known. It should be noted that Xi, Xt ∈ X and Ci, Ct C ∀ i ∈ n.

```
Algorithm 1: KNN Algorithm1:    procedure KNN
2:    Define the distance metric -d (eg. Euclidean Distance, Manhattan Distance, Minkowski Distance etc)
3:    Calculate $d(X_t , X_i)\ \forall\ i \in n$.
4:    Sort all the distances in non-decreasing order.
5:    Define value for ‘K’.
6:    Select the first ‘K’ points from the sorted list.
7:    Let $K_i$ denotes the number of points belonging to the$i^{th}$ class among K points.
8:    If $K_i \geq K_j\ \forall\ i \neq j$ then assign label of class ‘Ci’ to‘Xt’.
9: end procedure
```

In the context of this paper, Xi represents the embeddings (features) for the individual class Ci in the array X of size 'n'. The value of K is set to 1 so that the only one class which closely resembles the Ct is selected. Also, Euclidean distance formula is used as a distance metric which is given by:

n
$d(Xt, Xi) = \sqrt{} \sum (Xt-Xi)^2 i=1$

Face Detection

After acquiring the image, the system will start to detect the face by applying the KNN (k-nearest neighbor) algorithm. This Algorithm relies on the distance between the feature vectors, and it also has labels associated with each image, so that it can return the name of the person who is being classified. The distance can be calculated with the help of Manhattan distance or Euclidean distance. The KNN algorithm classifies by finding the most common class among the k-closest examples.

Face Verification

The captured voter's image from the webcam is verified with the image within the database using a python program and then the voter can cast his/her vote if he/she is an authenticated voter otherwise the voter cannot cast his/her vote.

Voting

If the voter is an authenticated voter i.e, the captured voter's image is available within the database is proceeded to cast his valuable vote to his desired candidate and then submit his vote. If the voter is not an authenticated voter it is displayed as unknown voter. If the voter already once casted his vote and if at all he again commits to cast his vote again, it is displayed as already casted your vote.

Where:

X_t = 128-d feature vector of the test image

X_i = 128-d feature vector of the image in the database

V. RESULTS AND OBSERVATIONS

KNN was born out of research done for the armed forces. Fix and Hodge – two officers of USAF School of Aviation Medicine – wrote a technical report in 1951 introducing the KNN algorithm

K-Nearest Neighbors algorithm (or KNN) is one of the most used learning algorithms due to its simplicity. KNN is a lazy learning, non-parametric algorithm. It uses data with several classes to predict the classification of the new sample point. KNN is non-parametric since it doesn't make any assumptions on the data being studied, i.e., the model is distributed from the data.

KNN is called as a lazy algorithm because it doesn't use the training data points to make any generalization.

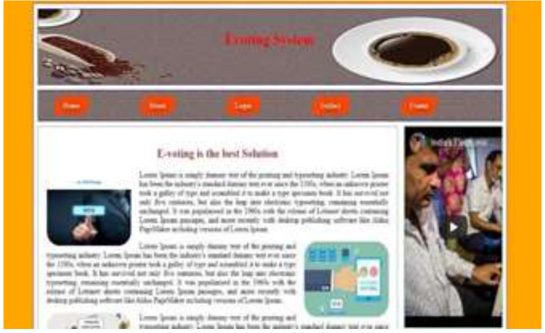
Which implies:

You expect little to no explicit training phase,

The training phase is pretty fast,

KNN keeps all the training data since they are neededduring the testing phase.

Most data do not obey the typical theoretical assumptions, like when we consider a model like linear regression, which makes KNN crucial when studying data with little or no prior knowledge.



Evoting System

E-voting is the best Solution

VI. CONCLUSION

This study focuses on an E-Voting using Face Verification by the system to a user based. This kind of approach will be very useful and the voter/user can save a lot of time. As the voting process is done from internet taking webcam access this approach will save alot of time along with cost cutting advantages.

ACKNOWLEDGMENT

This research project was partially supported by the Department of Computer Science and Engineering, Andhra Loyola Institute of Engineering and Technology, Jawaharlal Nehru Technological University. We are grateful to Assistant Professor Mrs. P. Hema Venkata Ramana for leading us to develop and contribute a paper to the conference.

REFERENCES

S. Kumar and E. Walia, "Analysis of electronic voting system in various countries," International Journal on Computer Science and Engineering, vol. 3, no. 5, pp. 1825–1830, 2011.

D. Nikam, D. Shetiye, and 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 Issue, pp. 1–3, 03 2019.

S. Ravi and D. P. Mankame, "Multimodal biometric approach using fingerprint, face and enhanced iris features recognition," in 2013 International Conference on Circuits, Power and Computing Technologies (ICCPCT). IEEE, 2013, pp. 1143–1150.

D. Menotti, G. Chiachia, A. Pinto, W. R. Schwartz, H. Pedrini, A. X. Falcao, and A. Rocha, "Deep representations for iris, face, and finger- print spoofing detection," IEEE Transactions on Information Forensicsand Security, vol. 10, no. 4, pp. 864–879, 2015.

J. Soldera, G. Schu, L. R. Schardosim, and E. T. Beltrao, "Facial biometrics and applications," IEEE Instrumentation & Measurement Magazine, vol. 20, no. 2, pp. 4–30, 2017.

S. Kumar, S. Nandury, and S. Raj, "An extended client server ar- chitecture in mobile environment," International Journal of Computer Engineering and Applications, vol. 5, pp. 97–107, 02 2014.

T. Ko, "Multimodal biometric identification for large user population using fingerprint, face and iris recognition," in 34th Applied Imagery and Pattern Recognition Workshop (AIPR'05), 2005, pp. 6 pp.–223.

V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 1867–1874.

A. Bansal, C. Castillo, R. Ranjan, and R. Chellappa, "The do's anddon'ts for cnn-based face verification," in The IEEE International Conference on Computer Vision (ICCV) Workshops, Oct 2017.

J. P. Jose, P. Poornima, and K. M. Kumar, "A novel method for color face recognition using knn classifier," in 2012 International Conference on Computing, Communication and Applications, 2012, pp. 1–3.

S. A. Perera, A Comparison of SIFT , SURF and ORB., Jan. 17, 2020. [Online]. Available: https://medium.com/@shehan.a.perera/acomp arison-of-sift-surf-and-orb-333d64bcaaea

D. Tyagi, Introduction to FAST (Features from Accelerated Segment Test),.,Nov. 26, 2019. [Online].

SK Kotamraju, PG Arepalli, SS Kanumalli (2021), Implementation patterns of secured internet of things environment using advanced blockchain technologies, Materials Today: Proceedings, 2021.

Gopi A.P., Patibandla R.S.M. (2021) An Efficient Methodology for Avoiding Threats in Smart Homes with Low Power Consumption in IoT Environment Using Blockchain Technology. In: Choudhury T., Khanna A., Toe T.T., Khurana M., Gia Nhu N. (eds) Blockchain Applications in IoT Ecosystem. EAI/Springer Innovations in Communication and Computing. Springer, Cham. https://doi.org/10.1007/978-3-030-65691-1_16.

RSML Patibandla, SN Mohanty (2021), Need of Improving the Emotional Intelligence of Employees in an Organization for Better Outcomes,Decision Making And Problem Solving: A Practical guide, 2021.

Available: https://medium.com/data-breach/introduction-to-fast-features from-accelerated-segment-test-4ed33dde6d65

S. Khedkar, S. Thube, W. Estate, C. Naka et al., "Real time databases for applications," International Research Journal of Engineering and Technology (IRJET), vol. 4, no. 06, pp. 2078–2082, 2017.

L. Gupta, Java AES 256 Encryption Decryption Example., Dec. 1, 2019. [Online]. Available: https://howtodoinjava.com/security/aes-256-encryption-decryption/

A. Chauhan, Difference between AES and DES ciphers., Nov. 17, 2019. [Online]. Available: https://www.geeksforgeeks.org/difference-betweenaes-and-des-ci