Clark – An AI tool to find a missing person

T.N. Sawant, Saisupraja Bhaskaran, Jayashree Poojary, Saakshi Shirke, Mohit Yadav

Department of EXTC, Bharati Vidyapeeth College of Engineering, Navi Mumbai, Maharashtra, India-400614
tukaram.sawant@bvcoenm.edu.in, saisuprajah@gmail.com, jayashreepoojary1506@gmail.com, saakshishirke21@gmail.com, yadavmohit2001@gmail.com

Abstract **- The number of missing persons is abundant in many parts of India as well as in every corner of the globe. Whenever a missing person complaint is filed at the police station, there are a lot of steps involved in order to find the person. These steps include filing a report, getting photographs of the missing person and circulating the data to various locations throughout the country. This involves a lot of manual processing and consumes a lot of time and energy. So, our research will be geared towards reducing this manual process and also implementing an innovative idea that will be efficient for finding the missing person with increased accuracy. By using different inputs, this research paper aims to help track the missing person. We will use machine learning and a Postgres database to store all the photos of people who have already been reported missing.**

Keywords **– Missing cases, Photographs, Locations, Postgres, Database.**

I. INTRODUCTION

As we know that the traditional process of finding a missing person involves a lot of manual steps, in this paper we have implemented some upcoming technologies that can be helpful in reducing the energy and time needed in finding the missing person. We will be building an application on which the missing person's photo and other details can be uploaded. This data will be stored and segregated in a database. Once the image and details of the missing person is uploaded by that individual's family member or by police, that photo will be matched with the photos present in the database of already missing people. From the other end, if any individual finds the missing person and uploads the details from their end, feature matching takes place and returns appropriate results if matched. We will be using codes for running scans on the image and finding out unique features of the person. If the HaaR features of both the images are similar to each other, the application will further run a scan through the other details in the backend of the system. If these details also match, then the app concludes that person is matched. Basically, this research is executed in order to track the missing person with the help of various inputs that will be taken. The aim is to make sure the critical details of the missing person are used along with their current location details. This will enable us to find them along with their other details like age, location and phone number to fasten up the process of finding the person who has been missing. Using the latest technologies like Machine Learning and other advanced softwares, this novel application "Clark" can be proved to be one of the most useful and efficient by making sure that we get the process of finding a missing person easily done. The implementation includes different technologies such as image processing, database management system and machine learning concepts. We can also include a color detection feature to expand this research for different racial limits.

II. LITERATURE SURVEY

S. Sharma, Karthikeyan Shanmugasundaram, Sathees Kumar Ramasamy used the dataset of FAREC which can have a large storage capacity. This paper also gives importance to face alignment. Therefore, the accuracy and False Acceptance Rate (FAR) is also observed by this method. Dlib is used which is an open source library used in this paper gives the best solution for face recognition [1]. Xiujuan Ren, Junhang Ding, Jinna Sun, Qingmei Sui used Random forest Algorithm for classification as it gives 95% accurate results. The ratio of the number of training data to the prediction accuracy of the model is also observed. And the data is synthesized by the transformation of present data in the case where there is insufficient data [2]. Vijay Kumar Sharma used AdaBoost classifier, Local Binary Pattern (LBP), Open CV, Feature extraction and Classification. Local Binary Patterns (LBP) is used to extract the distinctive features of the face like nose, mouth, eyes in the feature extraction phase. Implementation of the system is done using Python and OpenCV library [3]. Ahonen, Timo, Abdenour Hadid, Matti Pietikaine utilized the advantage of Local Binary Patterns where face image is divided into various parts and the features are extracted. These features are then converted into enhanced vector and then used for further processing [4]. S. Schulter, C. Leistner, et al., used the joint classification-regression random forest. Basically, they presented an approach for object detection that is able to work with the aspect ratios of an object and have also used sliding window approach for regression and to learn the object probability [5]. K. Taunk, S. De, et al., reviewed on Nearest Neighbour Algorithm to learn and classify. The approach is simpler and the K nearest points are used to segregate the unlabeled data. This approach comes handy when there is difficulty in determining the parametric approximations and unknown quantities [6].

III. EXISTING SYSTEM AND LIMITATIONS

In the existing system, face recognition systems have been developed in order to find the uniqueness of the facial features of the datasets uploaded using the Dlib algorithm which is an open-source library [7]. In one of the approaches, Radio Frequency Identification (RFID) tag is used. This tag can be used at Police stations and other places where number of people going missing is more. This method can have high

usability but, the only concern is that the person has to keep the RFID Tag with oneself. Random forest algorithms are used in face modeling to deliver better results than the previously used classification models. As these are trained datasets, the training process undergoes all the necessary steps involved. Limitations in these face detection system were that they were not effective for the machine learning model to segregate the facial features of the uploaded pictures due to accuracy constraints [8]. Due to limitations of database capacity, only a limited number of datasets could be trained at a given particular time. Some algorithms used to also consume a lot of time to train the model. The effectiveness of the facial detection system is somewhat proportional to the image quality of the images that are being uploaded. Lower the image quality, lower is the accuracy of detection and so will be the further process taking place with lesser accuracy. It is possible for the resulting verified results to differ on a wider scale, because of distortions that arise in the images. To improve this, we are using the K Nearest Neighbor (KNN) model to train our datasets. As KNN algorithm is simpler to use and handles large amount of data. It can also be used for both Classification and Regression problems [9].

IV. METHODOLOGY

We are working with various Python libraries such as OpenCV, Numpy, Pillow, Uvicorn, Python multipart, Database API and Face encoding API, and so on. We have included Dlib which is a library written in C++ and it is basically intended to estimate the location of 68 coordinates (x, y) that map the facial points on a person's face [10]. Feature extraction is processing or transforming the data into numerical features and then storing them in the database. This is done to ensure that the numerical features are compatible with machine learning algorithms. It is used when raw data cannot be analyzed in that form. In terms of improving the system for random modeling, the KNN model does not include a training period. Thus, this model ensures time efficiency and needs less time to train. This is because the data itself is a model that will be the reference for future prediction. Therefore, it is very time efficient. [11] We are implementing various Visual Studio codes and Docker (a major use of Postgres for Database). In order to develop, test, ship, and run an application, Docker is used. In the procedure mentioned below, we have concisely described the ideal steps that the user will come across while interacting with the interface of our application "Clark"

A. Procedure

- User login
- New case registration
- Upload images and details

} Input

- Images feature match → Internal process
- Information of similar people displayed
- Person's location details shown
- Missing person found

} Output

V. BLOCK DIAGRAM

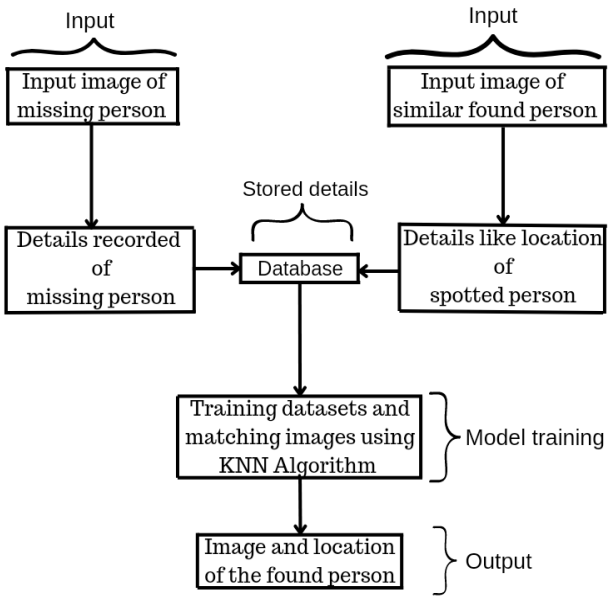

Fig. 1. Overall Workflow of "Clark" Application

Age Group	*Percent of Missing Persons (25% Sample)*	*Percent of General Population*	*Missing Persons as a Percent of the General Population*	*Relative Risk (versus other Gender)*	*Relative Risk (versus Males, 0–14 Years)*
Female					
0–14 years	15.3	8.7	1.0	1.4	1.4
15–19 years	26.7	3.4	4.5	1.4	6.4
20+ years	16.0	39.3	0.2	1.4	0.3
% female	58.0	51.4	0.7	1.3	1.0
Male					
0–14 years	11.2	9.1	0.7	0.7	—
15–19 years	20.0	3.5	3.3	0.7	4.7
20+ years	10.8	36.0	0.2	0.7	0.3
% male	42.0	48.6	0.5	0.8	0.7
Total %	100.0	100.0	0.6		—
Unknown	1				
Total	724	504,560			

Fig. 2. Analysis on missing cases

In fig. 2. We have segregated the data of all the missing persons in a whole year. The above data has been collected by relating the total general population with the risk prone population that can go missing [12]. Includes distance a person has covered, rate at which the vehicle, object or person is moving, analysis of the location from where they have been missing, also checking the background of the missing person takes place.

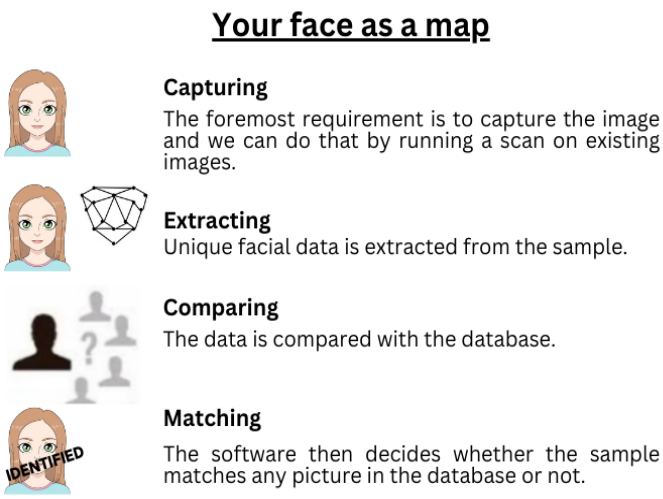

Fig. 3. Facial Detection

Firstly, the image of the missing person is taken as an input by the system. This image can be obtained by images uploaded by the Police or by running a scan on already existing images. Images are further processed for extraction of unique facial features. The features collected are hence compared with the data from the database. The final step, software concludes whether the sample is same as the data from the database.

VI. RESULTS AND DISCUSSION

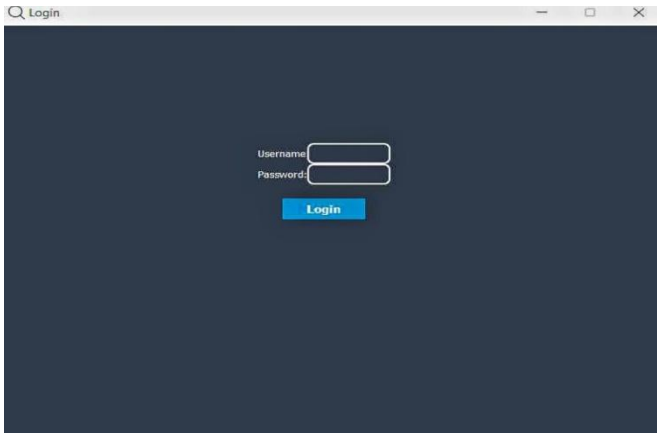

Fig. 4. Login window

This is the interface that the user will interact to as soon as the application is opened. Username and password is used in order to maintain security of users and to avoid unwanted access to application.

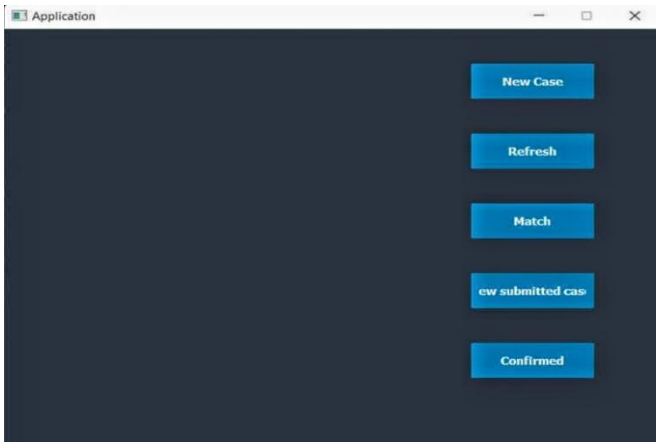

Fig. 5. Options in application

After successful login to the application, the user can either submit a case of missing person, refreshing the data stored, matching the details of submitted cases, view the submitted cases or find any confirmed cases through the application.

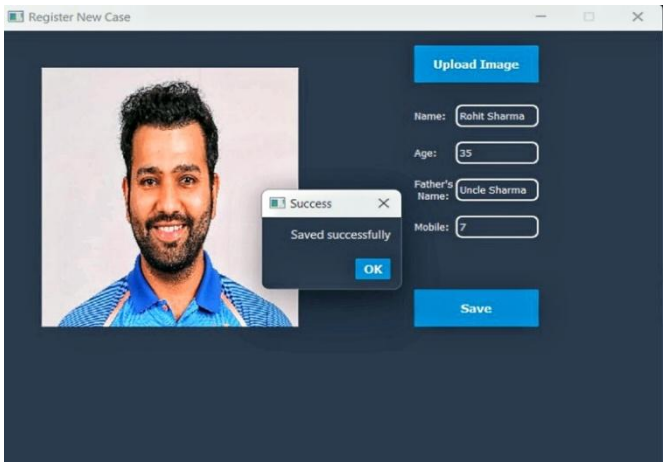

Fig. 6. Details submission and image uploading

New case submission in the application is by uploading image of the missing person. If someone has found the person then the same interface for their details uploading is used. The image getting uploaded successfully on the left side of the screen indicated that the face encoding points have been properly generated from the API. Thus, one can click on save and submit the details.

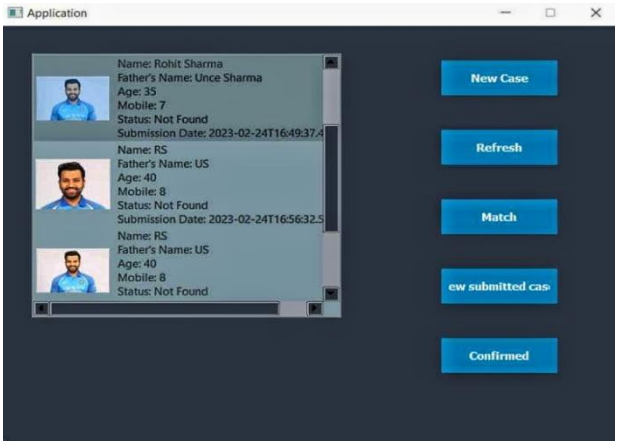

Fig. 7. List of submitted cases

All the cases that are submitted by the user are shown along with their details. The process of feature extraction and face encoding part where the encoding points are extracted from the Application Programming Interface (API) is carried out.

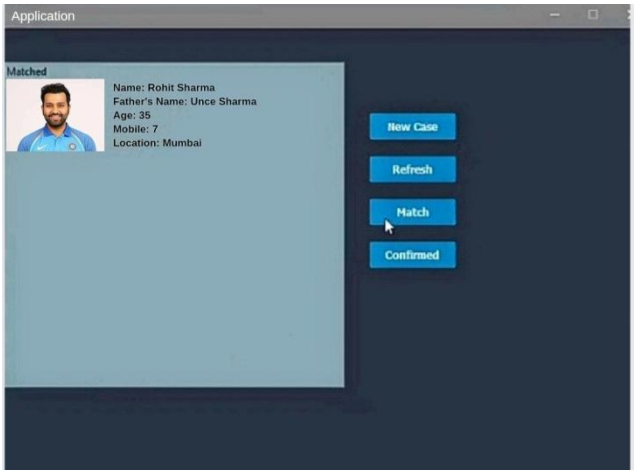

Fig. 8. Found person details shown

If after processing of images (data) stored in Database that was uploaded as an input from the missing person's family, friends or relatives and the data from the person who has found the missing person and incase there is feature matching between the two images the application shows the current location details and image.

VII. CONCLUSIONS

In this research paper, the manual efforts and time that is required to find a missing person is automatically reduced. As we are using KNN algorithm for training the model, the results achieved will be more accurate and effective. This can also be developed as an application that will be deployed on various app stores. The face detection algorithm used will be able to differentiate and identify the uniqueness of various facial features data uploaded on the database. As this paper focuses on the advantages of artificial intelligence technologies, there can be many developments which will enhance the efficiency of this AI tool.

VIII. ACKNOWLEDGEMENT

We are thankful to "Bharati Vidyapeeth College of Engineering" for considering our research paper and considering us through the various stages of the research. It gives us immense pleasure to express our sincere gratitude of our guide for his guidance in selecting the final year project and also for providing us with all the details. We are deeply indebted to our respected Head of The Department of Electronics and Engineering, Dr. P. A. Kharade, for giving us this valuable opportunity to carry out the process and we express our heartily thanks to them for their assistance without which it would have been difficult in finishing this final year project research successfully. We are also thankful to our respected Principal, Dr. Sandhya D. Jadhav. We convey our deep sense of gratitude to all teaching and non-teaching staff of The Department of Electronics and Telecommunication Engineering for their constant encouragement, support and timely help throughout the final year project work.

IX. REFERENCES

[1] S. Sharma, K. Shanmugasundaram, et al., " FAREC — CNN based efficient face recognition technique using Dlib ", International Conference on Advanced Communication Control and Computing Technologies (ICACCCT), Ramanathapuram, India, pp. 192-195, doi: 10.1109/ICACCCT.2016.7831628.

[2] X. Ren, J. Ding, et al., " Face modeling process based on Dlib ", Chinese Automation Congress (CAC), Jinan, China, pp. 1969-1972, doi: 10.1109/CAC.2017.8243093.

[3] V. K. Sharma, " Designing of Face Recognition System ", International Conference on Intelligent Computing and Control Systems (ICCS), Madurai, India, pp. 459-461, doi: 10.1109/ICCS45141.2019.9065373.

[4] Ahonen, Timo, et al., " Face description with local binary patterns: Application to face recognition ", IEEE transactions on pattern analysis and machine intelligence 28.12: 2037–2041, 2006.

[5] S Schulter, C Leistner, et al., " Accurate object detection with joint classification regression random forests ", 2014, Fuel Processing Technology, vol. 91, no. 6, pp. 591-599, 2014.

[6] K. Taunk, S. De, S. Verma and A. Swetapadma, " A Brief Review of Nearest Neighbour Algorithm for Learning and Classification ", International Conference on Intelligent Computing and Control Systems (ICCS), Madurai, India, pp. 1255-1260, doi: 10.1109/ICCS45141.2019.9065747, 2019.

[7] https://medium.com/dataseries/findmissingpeople-usingfacial-recognition-5579462d4911

[8] https://www.researchgate.net/publication/277090732_Missing_persons_and_social_exclusion

[9] http://repository.uonbi.ac.ke/bitstream/handle/11295/90853/

[10] T. Ahmed, P. Das, et al., " A Comparative Study on Convolutional Neural Network Based Face Recognition ", 2020, 11th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1-5, 2020.

[11] S. Angra, S. Ahuja, " Machine learning and its applications: A review ", International Conference on Big Data Analytics and Computational Intelligence (ICBDAC), Chirala, Andhra Pradesh, India, pp. 57-60, doi: 10.1109/ICBDACI.2017.8070809, 2017.

[12] S. B. Arniker, et al., " RFID based missing person identification system ", International Conference on Informatics, Electronics & Vision (ICIEV), Dhaka, Bangladesh, pp. 1-4, doi: 10.1109/ICIEV.2014.7135998.