iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL
FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 Issue: VIII Month of publication: August 2021

DOI: https://doi.org/10.22214/ijraset.2021.37270

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VIII Aug 2021- Available at www.ijraset.com

Missing Person Detection Using AI

Basavaraj L P[1], Manohar R[2], Surendra K V[3], Vijaykumar D. Damodar[4], Prof. Arun Kumar G H[5]
[1, 2, 3, 4, 5]CS&E Dept., BIET, Davanagere, Karnataka, India

Abstract: In the world, a countless number of people are missing every day which includes kids, teens, mentally challenged, old-aged people with Alzheimer's, etc. Most of them remain untraced.[1] This paper proposes a system that would help the police and the public by accelerating the process of searching using face recognition. When a person goes missing, the people related to that person or the police can upload the picture of the person which will get stored in the database. The face recognition model in our system will try to find a matching the database with the help of face encodings. It is performed by comparing the face encodings of the frames in the video streaming to the face encodings of the images in the database. If a match is found, it will be notified to the police and the people related to that person with the details of the person found. This project is really helpful as concerned for the government in order to lessen the time and man power spent in finding the missing persons. The face recognition model that we have developed maintains an accuracy of 94%.

Keywords: MachineLearning, Python, Tensorflow, Keras, Image Augmentation.

I. INTRODUCTION

A missing person can be characterized as the one who can be a child or an adult -- who is lost, voluntarily or involuntarily. There are various categories of missing cases of which only 43% of missing cases' reasons are known, 99% are juvenile runways, 2500 cases are due to family problems and around 500 cases are kidnapped by strangers (which include both teens and adults). Women add about 52% of missing cases and males 48%.

In India, there are no budgets allocated to finding missing people‖, claimed by an official source. A person, if goes missing, faces many challenges, some of them are subjected to death (murder), abusement or rape. People who are concerned with the missing person such as friends, relatives, parents and guardians will be exposed to stress and worries because of not knowing if the missing person is alive or dead. In our system, the image of the person given by the guardian at the time of missing is stored in the database. The public will be given the authority to upload photographs of any person in uncertain situations. Automatic detection of match for this picture among the already existing images in the database will be done through our application. This helps the police department to find the missing person in any place in India. When a suspicious person is detected, the image of the person at that instance of time is compared with the images uploaded by the guardian/police department at the time of missing through the face recognition model. If a match is found,a notification will be sent to the police department and the guardian in the form of an electronic mail along with the location of where the person is detected. If the image is not found, a new model for that person's image will be created in the database with the uploaded picture. By this way, it decreases the time taken for searching a person's detail after he is found.

II. LITERATURE SURVEY

1) *Appearance Based Approaches*: As mentioined in the paper published by (Dass, Rani, & Kumar, 2012), appearance based (or holistic matching) methods use the whole face region as the raw input to a recognition system. The face recognition problem is firstly transformed to a face space analysis problem and then several well-known statistical methods are applied to it. They are quite prone to the limitations caused by facial variations such as illumination, 3D poses and expressions. [2]
2) *Eigen face Technique:* The Eigen face technique is one among the mostly used face recognition processes. The Principal Component Analysis (PCA) is a technique effectively used to achieve dimensionality decline. Face recognition and detection mostly use Principal Component. Mathematically, Eigen faces are the principal components through which into feature vectors can be obtained from the face. The feature vector data can be obtained as of covariance matrix. These Eigen vectors are used to calculate the difference between numerous faces. The faces are categorized by the linear grouping of maximum Eigenvalues. Every face can be measured as a linear grouping of the Eigen faces. The face, having the largest eigen values of the eigen vectors can be approximated. It has been addressed in (Patel & Yagnik, 2013) paper that Eigen face is an applied method for face identification. Execution of an Eigen face recognition scheme has become easy because of the ease of its algorithms. The accuracy of Eigen faces rest on numerous things. [3] The Eigen face method finds an approach to make ghost-like faces that

characterize the bulk of variance in an image dataset. This method is built on an evidence theory method that decomposes face pictures into a minor set of feature images called –Eigen faces‖, which are in fact the principal components of initial training set. The problem of Eigen face is, it is profound for lightening environments and location of the Head. Drawback is outcome of the eigenvalues and eigenvectors are phase consuming.

3) *Principal component analysis (PCA)* :In the paper published by (Antony, 2016), the PCA technique converts each two-dimensional image into a one-dimensional vector. This vector then goes through several steps such as Detect Face in Image, Extract Facial Features, Normalize Facial landmarks, and then Recognize Face Image. (Antony, 2016) further stated that the technique selects the features of the face, which vary the most from the rest of the image. In the process of decomposition, a large amount of data is discarded as not containing significant information since 90% of the total variance in the face is contained in 5-10% of the components. This means that the data needed to identify an individual is a fraction of the data presented in the image[4].
4) *Linear Discriminant Analysis (LDA):* (Madane & Khandare, 2015) The linear discriminate analysis (LDA) is a powerful method for face recognition. It yields an effective representation that linearly transforms the original data space into a low-dimensional feature space where the data is well separated. In LDA, the goal is to find an efficient or interesting way to represent the face vector space. However, if the within-class scatter matrix (SW) becomes singular in face recognition and the classical LDA cannot be solved which is the under sampled problem of LDA (also known as small sample size problem). A subspace analysis method for face recognition called kernel discriminate locality preserving projections (MMDLPP) is based on the analysis of LDA, LPP and kernel functions
5) *Hidden Markov Model (HMM):* (Sharma & Kaur, 2016) stated that the first efforts to use Hidden Markov Model (HMM) were introduced by Samaria and Young. HMM has been worked effectively for images with variations in lighting, facial expression, and orientation. Thus, it has an advantage over the appearance-based approaches. For processing images using, HMM, the temporal or space sequences has been considered[5]. Local Binary Patterns Histograms (LBPH) (Sharma & Kaur, 2016) argued that the local binary pattern (LBP) had been designed for texture description. According to (Wahid, 2013), the face area is first divided into small regions from which Local Binary Patterns (LBP), histograms are extracted and concatenated into a single feature vector. This feature vector forms an efficient representation of the face and is used to measure similarities between images. The major advantage of this algorithm is that it produces better recognition rates in controlled environments and it is not profound to illumination. [6] Limitations of Local Patterns Histograms (LBPH) Even though current machine recognition systems have reached a certain level of perfection but still there are many real application conditions which limits their good performance.

III. METHODOLOGY

The following tools were used in the design of the prototype:

1) *MySQL Database:* MySQL database is the mostly used database for various applications notably web applications amongst others. The MySQL database was choice used by the researcher for saving information about system users and user details. The researcher used this database because he was equipped with some experience in using it. Additionally, MySQL database is an open source and relatively simple to use.
2) Opencv: It is a computer vision library, which is equipped with various libraries to perform image-processing techniques. OpenCV was designed in C++. It provides algorithms namely the Eigen Face, Fisher Face as well as LBPH (Local Binary Pattern Histograms) algorithm, which was implemented in this research. Furthermore, OpenCV have machine-learning algorithms that were used to train the image datasets.
3) Web Camera: This was used for capturing images of individuals for testing the prototype. Python Programing Language: Python is a scripting or general-purpose language that was developed by Guido van Rossum. The author opted to use python because of its simplicity and code readability. Additionally, it enables the programmer to express his ideas in fewer lines of code without reducing any readability. Python is also easy for performing OpenCV bindings.
4) Numpy: Is a highly optimized library for numerical operations. Its purpose in the prototype is to handle arrays to and from the OpenCV library.
5) Flask: A python library for user interface design. The researcher used this library to create the Graphical user interface.

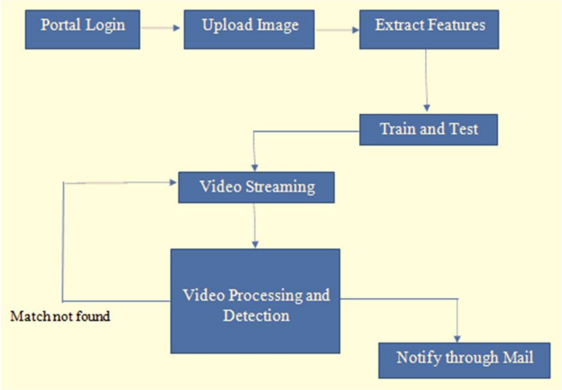


Fig: Workflow Methodology

The proposed system makes use of face recognition for missing people's identification. As shown in the architecture, here the police authorities who finds the suspicious person or missing person uploads the image of that person into the portal. Our algorithm extracts the face encodings of the image and compares the face encodings of the previously existing images in the database. If a match is not found then the image will be pre-processed and trained with the existing model. If the image is already trained, no pre-processing is needed to be done.

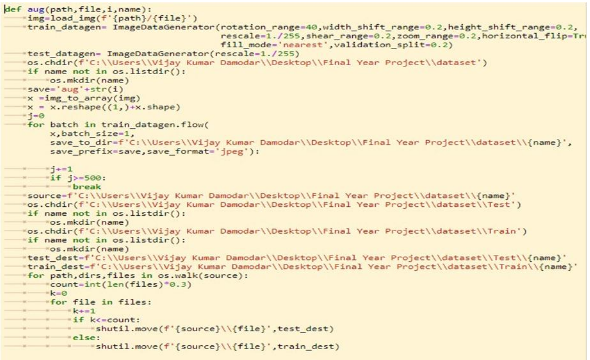
Whenever a trained image is found in the video streaming, the image will be tested with the trained set of images. If our algorithm finds the match, it returns the related data of that person such as name of the person, email address, etc. Then the proposed system will send the notifications to the police station or concernedauthorities.

A. Data Pre-Processing And Data Augmentation

In order to make the most of our few training examples, we will "augment" them via a number of random transformations, so that our model would never see twice the exact same picture. This helps prevent overfitting and helps the model generalize better.

In Keras this can be done via the *keras.preprocessing.image.ImageDataGenerator* class

1) *Rotation_range* is a value in degrees (0-180), a range within which to randomly rotate pictures
2) *Width_shift and height_shift* are ranges (as a fraction of total width or height) within which to randomly translate pictures vertically or horizontally
3) *Rescale* is a value by which we will multiply the data before any other processing. Our original images consist in RGB coefficients in the 0-255, but such values would be too high for our models to process (given a typical learning rate), so we target values between 0 and 1 instead by scaling with a 1/255. factor.
4) *Shear_range* is for randomly applying shearing transformations.
5) *Zoom_range* is for randomly zooming inside pictures
6) *Horizontal_flip* is for randomly flipping half of the images horizontally --relevant when there are no assumptions of horizontal assymetry (e.g. real-world pictures).
7) *Fill_mode* is the strategy used for filling in newly created pixels, which can appear after a rotation or a width/height shift.



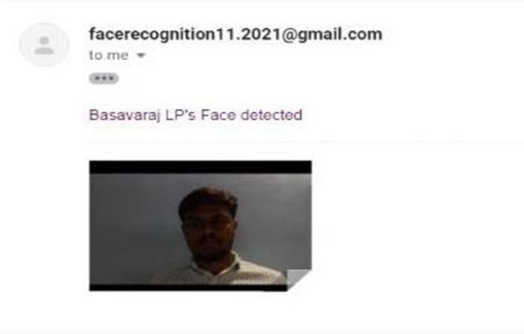
```
def aug(path,file,i,name):
    img=load_img(f'{path}/{file}')
    train_datagen= ImageDataGenerator(rotation_range=40,width_shift_range=0.2,height_shift_range=0.2,
                                      rescale=1./255,shear_range=0.2,zoom_range=0.2,horizontal_flip=Tr
                                      fill_mode='nearest',validation_split=0.2)
    test_datagen= ImageDataGenerator(rescale=1./255)
    os.chdir(f'C:\\Users\\Vijay Kumar Damodar\\Desktop\\Final Year Project\\dataset')
    if name not in os.listdir():
        os.mkdir(name)
    save='aug'+str(i)
    x =img_to_array(img)
    x = x.reshape((1,)+x.shape)
    j=0
    for batch in train_datagen.flow(
        x,batch_size=1,
        save_to_dir=f'C:\\Users\\Vijay Kumar Damodar\\Desktop\\Final Year Project\\dataset\\{name}',
        save_prefix=save,save_format='jpeg'):

        j+=1
        if j>=500:
            break
    source=f'C:\\Users\\Vijay Kumar Damodar\\Desktop\\Final Year Project\\dataset\\{name}'
    os.chdir(f'C:\\Users\\Vijay Kumar Damodar\\Desktop\\Final Year Project\\dataset\\Test')
    if name not in os.listdir():
        os.mkdir(name)
    os.chdir(f'C:\\Users\\Vijay Kumar Damodar\\Desktop\\Final Year Project\\dataset\\Train')
    if name not in os.listdir():
        os.mkdir(name)
    test_dest=f'C:\\Users\\Vijay Kumar Damodar\\Desktop\\Final Year Project\\dataset\\Test\\{name}'
    train_dest=f'C:\\Users\\Vijay Kumar Damodar\\Desktop\\Final Year Project\\dataset\\Train\\{name}'
    for path,dirs,files in os.walk(source):
        count=int(len(files)*0.3)
        k=0
        for file in files:
            k+=1
            if k<=count:
                shutil.move(f'{source}\\{file}',test_dest)
            else:
                shutil.move(f'{source}\\{file}',train_dest)
```

Fig: Image Augmentation code

Fig: Image Augmentation Result

After augmentation, the result set of images are split into train and test sets. Further, the model is trained for real time face detection. The video streaming from the camera will be continuously monitored and tested against the stored model. If the system finds the exact match, the notification containing the details of the person will be sent to the concerned authorities.



facerecognition11.2021@gmail.com
to me

Basavaraj LP's Face detected

Fig: Notification after successful detection

IV. CONCLUSION

The proposed work helps in identification of a person in surveillance area using face information. The face recognition in the images got from surveillance camera is challenging task due to the presence of multiple faces in the given area. In this project a method has been proposed where the algorithm has been modified for the detection of the faces, extraction of the feature information and matching the features. The work can further be extended for improving the recognition accuracy as well as time for large face databases. This system is designed to find the missing people. If the missing person found in the CCTV Video streaming, then track the location of missing person by using the location of camera where the person detected. After missing person found in the CCTV Video streaming, then send the location to the respective authorities. So our system can perform the very important role in security and authentication issues.

The researcher however encountered problems, which are prone to most facial recognition systems. The system was affected by the illumination problem thus variation in lighting conditions. Whenever there was insufficient lighting in the room, the recognition rate declined. Another challenge was that of hardware, facial recognition requires high performance computing hardware and most particularly a high definition camera with a high resolution.

The development of the proposed system was narrowed towards finding missing people. However, the same system can be improved by implementing it on DSP processors and using other Hardware devices like Raspberry Pi.

REFERENCES

[1] Bharath Darshan Balar, D S Kavya, Chandana M, Anush E, Vishwanath R Hulipalled ―Efficient Face Recognition System for Identifying Lost People‖,

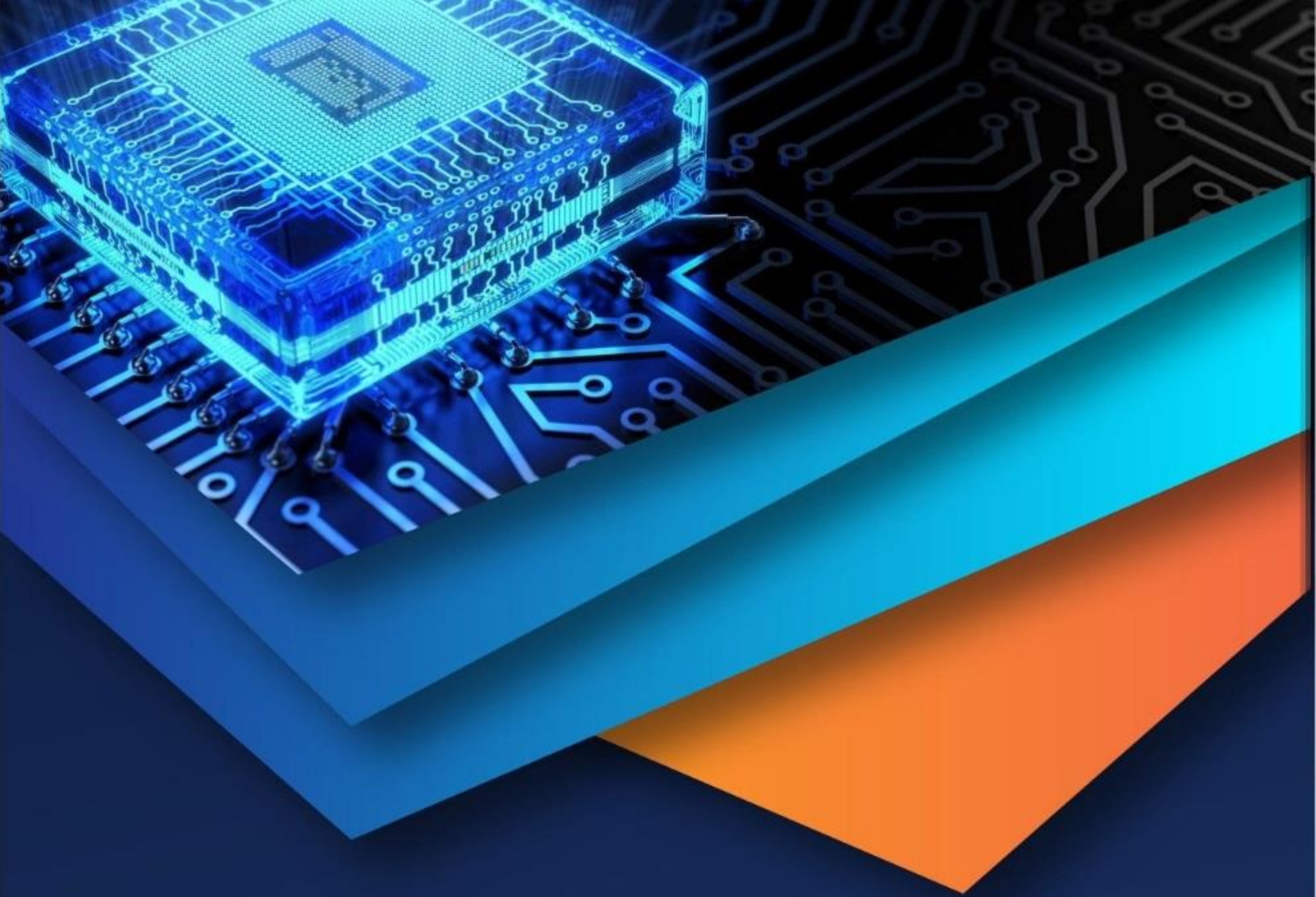
[2] Dass, R., Rani, R., & Kumar, D. (2012). ―Face Recognition Techniques‖ : A Review, 4(7), 70–78.

[3] Patel, R., & Yagnik, S. B. (2013). ―A Literature Survey on Face Recognition Techniques‖, 5(4), 189–194.

[4] Antony, J. (2016). ―Development Phases of Technologies in Face Recognition Systems‖, 1(2), 18–21.

[5] Madane, S. R., & Khandare, P. S. T. (2015). ―A Survey on Face Recognition in Present Scenario‖, (4), 6–10.

[6] Sharma N., & Kaur R. (2016). ―Review of Face Recognition Techniques‖, 6(7), 29–37.



10.22214/IJRASET

INDEX COPERNICUS
45.98

IMPACT FACTOR:
7.129

TOGETHER WE REACH THE GOAL

IMPACT FACTOR:
7.429