International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 06 | June 2020 www.irjet.net p-ISSN: 2395-0072

Home Security Using Object Detection and Face Recognition

Mr. Justin KJ[1], Mr. Pranav NK[2], Mr. Alvin Reji[3], Dr. Keerthi Kumar H M[4]

[1-4]Dept. of Information Science and Engineering, Yenepoya Institute of Technology, Moodbidri, Karnataka, India

---

Abstract – *Security issue is the major problem faced by the houses, offices and other institutes and consequently, they need several specially trained personnel to attain the desired security. These personnel, as human beings, make mistakes that might affect the level of security. A proposed solution to the above mentioned matter is a Security System using Face Recognition, which can detect the people coming to restricted or high-security areas, and help in minimizing human error. Smart security system have a major role in modern daily life. The proposed security system has been developed with an intension to prevent robbery in highly secure areas like home environment. The face recognition which helps to detect the people entering that area. The face recognition and detection process is implemented by principal of performing image processing on captured video frames of data, and calculating the difference between the previously captured frames with the running frames in terms of pixels in the captured frames. Moreover this system also detect the objects carried by the people while entering to that area, which can ensure the security by identifying whether the person is a harmful person or not. An effective security system for home and similar institutions can be developed.*

Key Words: **Face detection, Face recognition, Object recognition, LBPH, Har cascade, TenserFlow, etc.**

1. INTRODUCTION

A face recognition system is an application of computer vision and image processing which is capable of performing two major tasks of identifying and verifying a person from an image. This system aims to develop an automated system to recognize the persons entering a private area by using facial recognition technology. This system incorporates a camera that captures input image, an algorithm to detect a face from the input image, encode it and recognize the face and display the name of the person if the face is there in the dataset. A webcam is used to detect the images and the images are sending to the database.

An efficient Face Detection and Recognition technique which is independent of variations in features like colour, hairstyle, different facial expressions etc. are used. If the image matches with the image in the database then the face is labelled with his/her name and if it does not match the image displays unknown. A database with the multiple faces of the family members of that particular house is stored and the input image is cross checked with these images. The system will also detect the objects in the image along with the faces. As part of the security feature system will detect whether the object is harmful or not.

1.1 Objective

Few objectives need to be achieved in order to accomplish this project. These objectives will act as a guideline and will make the system to be implemented for certain situations:

1. Create an effective system that capture and detect human faces and display the name of the person entering a private area
2. Also identify the object from the captured image.
3. From the detected objects we can say that whether the person is carrying any harmful objects or not.

Hence we can develop an efficient home security system.

2. LITERATURE SURVEY

[1] This paper has proposed a system using PCA algorithm for face recognition on hardware platform for its dimensionality reduction and simplicity. The proposed system is a home security system for detection of intruders and also provide security for door access control by using facial recognition for home environment. Human body is an intruder identified by the system with in a home environment. The live videos are captured using web camera and system identifies the motion detection of the intruder.

[2] This paper has implemented a system to recognize face by a variation of LBPH. They use a method of regression of local binary features to get the landmark of face image whose computational complexity is very low. They utilize these landmark points which can be trained to align the face, to extract the facial features. By calculating the Local Binary Patterns Histogram of these landmark points along with its neighbourhood pixels, an effective facial feature can be extract to realize face recognition. The face recognition rate can be improved and also increase the calculating speed of LBPH.

[3] This paper have proposed an intensive study of OpenCV platform and its inbuilt libraries. The proposed system also acts as a home security system for both person detection and provide security. The camera will capture the series of images as soon as the person press switch and by pressing single button the entire face recognition is completed. Face recognition consist of feature extraction from the facial image, classification or recognition and feature reduction.

[4] This paper introduced a system which proposes an automatic locking door using face recognition. There are lot of algorithms are introduced for detection and recognition and does not require human cooperation for identifying or verifying. By using the algorithm the nearest value is generated for the selection of face to be displayed. Face recognition is performed by 2 methods - First method is extraction of some basic parts of a face such as eyes, nose, mouth, and chin, with the one stored in the database .The Viola Jones algorithm is used for this. Second method is based on Principal Component Analysis method.

[5] This paper has developed a system for object detection through object classification and detection using cifar-10 data set with intended classification and detection of airplain images and used convolutional neural network on keras with tensorflow support the experimental results shows the system requires limited time only to train, test and create the model in the computing system. The Deep Neural Network (DNN) is the latest advancement in the deep learning made task simple for image recognition by as deeply as possible learning. The most popular technique used in improving the accuracy of image classification is the Convolutional neural network (CNN).

3. SYSTEM REQUIREMENTS AND SPECIFICATION

System Requirements Specification (SRS) is a document or set of documentation that describes a variety of elements that attempts to define the intended functionality required. And the features and behaviour of a system or software application. System Requirement Specification is a focal report, which outlines the establishment of the item progression handle.

3.1 HARDWARE REQUIREMENTS

Hardware requirement analysis is to define and analyse a complete set of functional, operational, performance, interface, quality factors, design, criticality and test requirements. This project works on a system having Intel core i3, i5 or above processors with an internal memory of 4 GB or above and need a hard disk of 60 GB and above. To capture the videos and images a high quality camera is required.

3.2 SOFTWARE REQUIREMENTS

The system requires Windows7 or above operating system. The programming language used is Python programming language. To run the python codes a python IDE is required and here we used Pycharm.

3.2.1 Python Programming Language

Python is an interpreted, high-level, general-purpose programming language. Its object-oriented approach help programmers to write clear, logical code for small as well as for large-scale projects. Python is a dynamically typed and garbage-collected language. A multiple programming paradigms, which includes procedural, object-oriented, and functional programming. Due to its comprehensive standard library Python is often described as a "batteries included" language.

3.2.2 PyCharm

PyCharm is a Python IDE with complete set of tools for the development of Python programs. Pycharm codes easily and faster in a smart and configurable editor with code completion and code folding and also split windows support.

3.2.3 Tkinter

Tkinter is a better user interface developer with Python binding to the Tk GUI toolkit. It is the general used Python interface to the Tk GUI toolkit, and is Python's de facto standard GUI. Tkinter contains standard Linux, Microsoft Windows and Mac OS X installs of Python.

4. SYSTEM DESIGN

This system is an automated security system with face recognition. It shows how the face biometric based security system works and will facilitate the house owner to maintain the security of the house by detecting the person coming to the house along with the objects that person carrying.

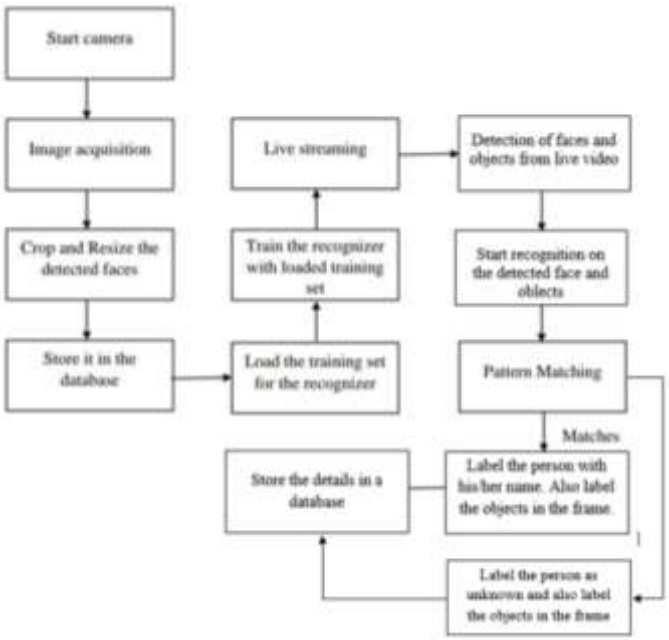

Fig -1: System Architecture

The working of the system is depicted in the figure above. Initially for creating dataset, faces of family members,

relatives and friends will be live streamed. Faces are detected from each frame using Haar Cascade Classifier. Images of each person will be captured. These images will be then sent for pre-processing where they will be cropped to obtain the region of interest (ROI). These cropped images will be resized for face recognition and then the RGB images will be converted into gray scale. These images are then stored in the database and will be considered as the dataset for further face recognition. After creating the dataset, the features from each images will be extracted and stored as histograms. The system will be then trained with the dataset created.

The face recognition algorithm used here is Local Binary Pattern Histogram (LBPH). During recognition, the faces will be detected from the frame and apply the LBP recognizer on those images where they undergo feature extraction process where the histograms of each detected image will be created. These histograms will be then compared with the already present histograms of trained images and compute the distance between the two histograms using any distance formula. If the distance between two histograms is less than the defined threshold, the recognized persons live stream will be labelled with his/her name and store the data in the database.

The object detection is done by using the Tenserflow. Every Object Detection Algorithm has a different way of working, but they all work on the same principle that is feature extraction. They extract features from the input images and the class of the image can be determined using these features.

The whole system can be divided into two parts- Training and Testing

1. Training:

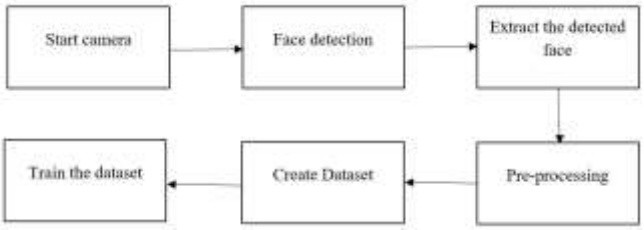

Fig -2: Training model

In this section, we create the dataset by detecting the faces in a live streaming video and capture samples of a single Person. Those images will be pre-processed and then will be stored in the database. This dataset will be further trained for recognition.

2. Testing:

The actual purpose of this system will be served in this section. The faces will be detected from the live streaming video from the camera placed in the entry of houses. The recognizer will be applied on the detected face in order to label the face with name. The LBP recognizer will compute the histograms and then compares it with the trained dataset. If the person is recognized, the system will label the face with name of the respective person and it will be marked as unknown in the else case.

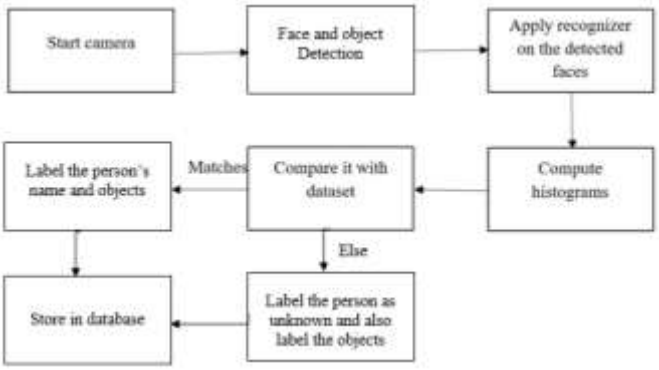

Fig -3: Testing model

5. SYSTEM IMPLEMENTATION

5.1 Dataset Creation

Images of people are captured using a web cam after registration. Multiple images of single person will be acquired with varied gestures and angles. These images undergo pre-processing. The images are cropped to obtain the Region of Interest (ROI) which will be further used in recognition process. Next step is to resize the cropped images to particular pixel position. Then these images will be converted from RGB to gray scale images. And then these images will be saved as the names of respective person in a folder.

5.2 Face Detection

Face detection technique is a method that identifies or locates human faces in the digital images. A typical example of face detection is the instance that detects faces in the picture when we take photographs through our smartphones. Face detection is performed by using classifiers. A classifier is essentially an algorithm that decides whether a given image is a face or not. A classifier needs to be trained on thousands of images with and without faces. OpenCV already has two pre-trained face detection classifiers that can be used in a program.

The two classifiers are:

- Haar Classifier
- Local Binary Pattern (LBP) classifier

5.2.1 Haar Classifier

In this system we have used Haar-Cascade Classifier. It works in following steps:

1. Haar Features Extraction

After the tremendous amount of training data (in the form of images) is fed into the system, the classifier begins by extracting Haar features from each image. Haar Features are kind of convolution kernels that is used to detect whether a suitable feature is present on an image or not.

2. Integral Image Concept

The integral image means that to find the sum of all pixels under any rectangle, we simply need the four corner values. This means, to calculate the sum of pixels in any feature window, we do not need to sum them up individually. By using the 4 corner values we need is to calculate the integral image.

3. Adaboost

More than 180,000 features values result within a 24X24 window. However, not all features are useful for identifying a face. To only select the best feature out of the entire chunk, a machine learning algorithm called Adaboost is used. It does so by constructing a strong classifier which is a linear combination of a number of weak classifiers. This reduces the amount of features drastically to around 6000 from around 180,000.

4. Using Cascade of Classifiers

The cascade classifier essentially consists of stages where each stage consists of a strong classifier. This is beneficial since it eliminates the need to apply all features at once on a window. Rather, it groups the features into separate sub-windows and the classifier at each stage determines whether or not the sub-window is a face. In case it is not, the sub-window is discarded along with the features in that window. If the sub-window moves past the classifier, it continues to the next stage where the second stage of features is applied.

5.2.2 Face Detection with OpenCV-Python

Here face detection is performed using Haar-Cascade Classifier with OpenCV. We are using detectMultiScale module from OpenCV. This is required to create a rectangle around the faces in an image. It has got three parameters to consider- scaleFactor, minNeighbors, minSize. scaleFactor is used to indicate how much an image must be reduced in each image scale. minNeighbors specifies how many neighbors each candidate rectangle must have. Higher values usually detects less faces but detects high quality in image. minSize specifies the minimum object size. By default it is (30,30). The parameters used in this system is scaleFactor and minNeighbors with the values 1.3 and 5 respectively.

5.3 Face Recognition Object Detection

Face Recognition is a technique that is used to detect faces of people, whose images are already saved in the data set. There are many face recognition methods. The method that we used here for face recognition is the OpenCV. In face recognition, the image first prepared for pre-processing and then trained the face recogniser to recognise the faces. After teaching the recogniser, we test the recogniser to see the results. The OpenCV face recogniser are of three types, out of which we have used Local Binary Pattern Histogram.

Face recognition process can be divided into four steps-

1. Training the algorithm:

First, we need to train the algorithm. For that, we have to use a dataset with the facial images of the people that we want to recognize. We need to also set a name id for each image, so to recognize an input image the algorithm will use this information and give you an output. Images of the same person must have the same name id.

2. Applying the LBP operation:

The first step of the LBPH is to make an intermediate image that describes the original image, using the facial characteristics. For that, a sliding window concept is used by the algorithm, based on the parameters radius and neighbors. The below image shows this procedure:

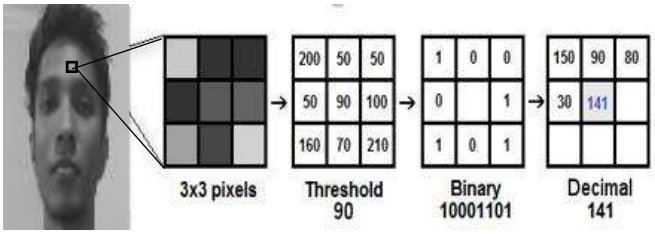

Fig -4: LBP operation

3. Extracting the Histogram:

Now, by using the image generated in the last step, we can use the Grid X and Grid Y parameters to divide the image into multiple grids, as can be seen in the below image. Based on the image below, we can extract the histogram of each region.

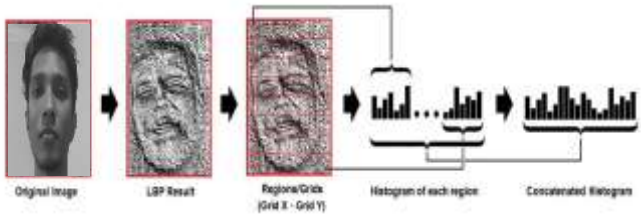

Fig -5: Extraction of histograms.

4. Performing the face recognition:

In this step, the algorithm is already trained. Each image from the training dataset is represented by the histogram created. Similarly for the given new input image, we perform the steps again and creates a histogram which represents the image. So we just need to compare two histograms to find the image that matches the input image and return the image with the closest histogram.

5.4 Object Detection

Object Detection is the process of finding real-world object instances like humans, vehicles, bottles, etc. in still images or Videos. The recognition, localization, and detection of multiple objects within an image is done so as to perform better understanding of an image as a whole. Here we use Deep Learning Object Detection as Tensorflow for object detection. Tensorflow is Google's Open Source Machine Learning Framework for dataflow programming for a range

of tasks. Tensors are just multidimensional arrays, an extension of 2-dimensional tables for data with a higher dimension. The different features of Tensorflow makes it more appropriate for Deep Learning.

5.4 GUI Design

It has a user friendly interface which makes it easy to be used by the system owner to view the data and maintain security. GUI used in this project is Tkinter. Tkinter is a feature of Python, using Tk and is Python's standard GUI framework. It is a famous one because of its simplicity and graphical user interface. It is open-source which is licensed by Python.

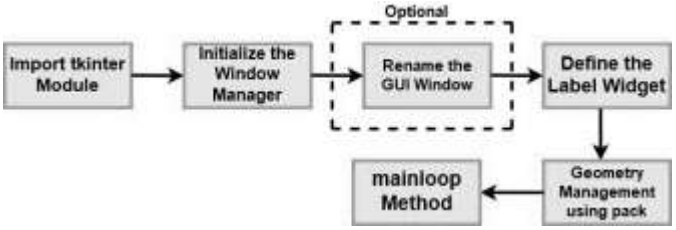

Fig -6: Workflow of Tkinter

6. RESULTS AND SNAPSHOTS

This chapter contains the snapshots of the interface and working screen of the proposed system showing the intermediate results.

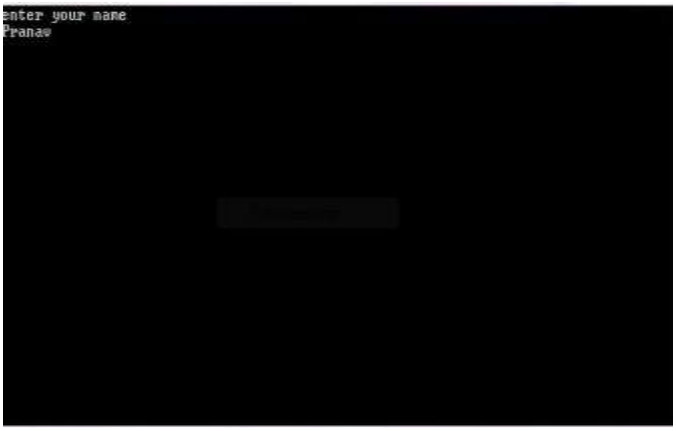

Fig -7: Creating dataset

To store a face into the dataset, first enter the name of the person in the input field of the system as in Fig-7 and then press enter. It will turn on the camera and the image of the person can be captured. Fig-8 shows the face capturing. The captured image will be stored in database and this image will be used to identify the person later.

Fig -8: Capturing the Face for Training

Fig-9 shows the detection and recognition of a known person when he comes in front of the camera. And he will be labelled with his name and it will be shown on the screen and also it will be stored in database.

Fig -9: Face Detection & Recognition

Fig-10 shows the objects that comes into the camera frame. It will be detected and recognized. Then it will be labelled with its name.

Fig -10: Object Detection & Recognition

Fig -11: Face Detection of Unknown Person

If any unknown person comes in front of the camera, the face will be detected and will be labelled unknown as shown in Fig-11. It will also detect the objects in the frame.

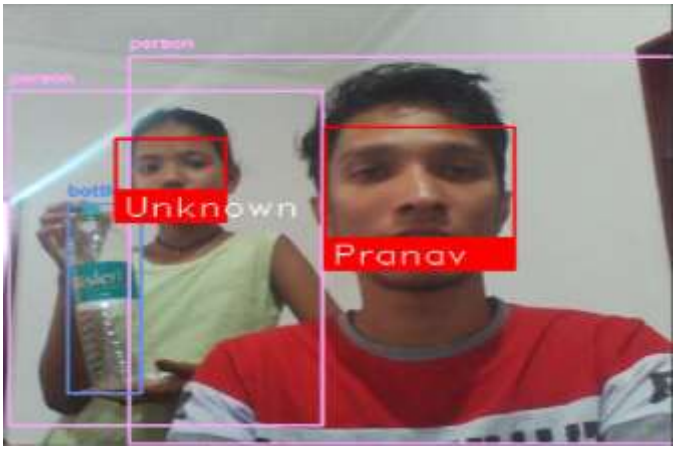

Fig -12: Detecting Multiple Faces and Objects

As shown in Fig-12 multiple faces will be detected by the system and also the objects carried by the people. The system will label the known people with their names and the others as unknown. Also the objects will be labeled with their names. These all data will be stored in the database along with date and time.

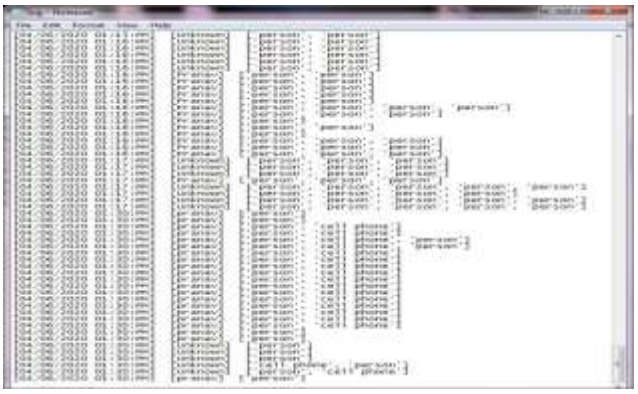

Fig -13: File storing details

Fig-13 shows the details of the people came to home that includes the details like date, time, name of the person, objects detected along with him. And the images of unknown person will be captured in a separate folder.

7. CONCLUSIONS AND FUTURE ENHANCEMENT

In this work, we are able to develop a security system that evaluate the face detection and face recognition. The database created is used as a source for the recognition of face using LBPH. Recognition rate can be increased by increasing the images in the database and the noisy images decrease the recognition accuracy. We can also use this security system in national security sectors and in other private sectors by updating the training set. Instead of capturing images, by using the images from adhar and similar source each and every person entering a private area can be detected. In this way by making some modifications according to requirements we can enhance the proposed system effectively.

REFERENCES

[1] Xiang, G., Qiuyu, Z., Hui, W., & Yan, C. (2016, July). "Face recognition based on LBPH and regression of local binary features". In 2016 International Conference on Audio, Language and Image Processing (ICALIP) (pp. 414-417). IEEE.

[2] Aggarwal, G., Chowdhury, A. R., & Chellappa, R. (2004, August). "A system identification approach for video-based face recognition". In Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004. (Vol. 4, pp. 175-178). IEEE.

[3] Moon, H., & Phillips, P. J. (2001). "Computational and performance aspects of PCA-based face-recognition algorithms". Perception, 30(3), 303-321.

[4] R. Manjunatha, & Dr. R. Nagaraja. "Home Security System and Door Access Control Based on Face Recognition", International Research Journal of Engineering and Technology. Volume: 04 Issue: 03, Mar -2017.

[5] Sudharshan Duth P, Swathi Raj, (2018). "Object Recognition in Images using Convolutional Neural Network", Proceedings of the Second International Conference on Inventive Systems and Control (ICISC 2018).