HOUSE MONITORING SYSTEM

A Project Report

Submitted for the Degree of

Bachelor of Technology in (CSE) of

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

by

AKSHITH KM	**ATP18CS006**
PRAJEESH PK	**ATP18CS023**
SARATH MUKUNDAN A	**ATP18CS024**

Under the guidance of

Mr. ROSHAN SANU Y

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

AHALIA SCHOOL OF ENGINEERING & TECHNOLOGY

Ahalia Campus , Palakkad , Kerala-67855

(http://www.ahalia.ac.in/)

June 3,2022

ISO 9001:2015 Certified Institution. Approved by AICTE & Affiliated to A. P. J. Abdul Kalam Technological University
Ahalia Health, Heritage & Knowledge Village, Palakkad - 678557. Ph: 04923-226666, www.ahalia.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the report entitled “HOUSE MONITORING SYSTEM” submitted by **AKSHITH KM (ATP18CS006), PRAJEESH PK (ATP18CS023), SARATH MUKUNDAN A (ATP18CS024)**, for the award of the **Degree of Bachelor of Technology** in the **APJ Abdul Kalam Technological University** is a bonafide record of the work carried out by them under my guidance and supervision at Ahalia School of Engineering & Technology.

Guide:	**Head of the Department :**
Mr ROSHAN SANU Y	Dr.S.GUNASEKARAN

Submitted for the project viva-voce on 29-06-2022.

Internal Examiner	**External Examiner**

AHALIA SCHOOL OF ENGINEERING & TECHNOLOGY
ISO 9001:2015 Certified Institution. Approved by AICTE & Affiliated to A. P. J. Abdul Kalam Technological University
Ahalia Health, Heritage & Knowledge Village, Palakkad - 678557. Ph: 04923-226666, www.ahalia.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We, **AKSHITH KM (ATP18CS006), PRAJEESH P K (ATP18CS023), SARATH MUKUNDAN A (ATP18CS024)**, hereby declare that this Project entitled "HOME MONITORING SYSTEM" is the record of the original work done by us under the guidance of "**Mr. ROSHAN SANU Y",** M.Tech, Assistant professor, Dr. S.GUNASEKARAN The Head of Department, Professor, Department of Computer Science and Engineering, Ahalia School of Engineering and Technology. To the best of my knowledge, this work has not formed the basis for the award of any degree/diploma/ associateship/fellowship/or a similar award to any candidate in any University.

Place: **Signature of the Student**

Date:

COUNTERSIGNED

Mr. ROSHAN SANU Y

Assistant Professor

Department of CSE

ACKNOWLEDGEMENT

First and foremost we would like to thank the **GOD ALMIGHTY** for this infinite grace and help, without which this project would not have reached its completion. We wish to express our sincere gratitude to **Dr. P R Sreemahadevan Pillai M, Ph.D.**, Principal, Ahalia School of Engineering and Technology for his timely support throughout the course of this project. We extend our sincere thanks **to Dr. Krishna Kumar Kishor Ph.D**., Vice Principal, for extending all facilities.

We express our sincere and heartfelt gratitude to **Dr. S.Gunasekaran, Ph.D.**, Professor and Head of the Department, Department of Computer Science and Engineering for his help and excellent encouragement throughout the project. We express our deepest heartiest and sincere thanks to our Project Guide **Mr. Roshan Sanu Y** Assistant Professor, for his exhilarating supervision, timely suggestions, and encouragement during all phases of this work. We take this opportunity to thank our Project Coordinator **Mr. Roshan Sanu Y,** Assistant Professor and our mentor who had been always there with us. Also, we would like to thank all the Teaching faculty members and Supporting Staff of our department.

We would like to convey our gratitude to our **Parents** whose prayers and blessings were always with us. Last but not least we would like to thank **our friends and others** who directly or indirectly helped us in the successful completion of this work.

TABLE OF CONTENT

ABSTRACT

House security is one of the major issues currently. An efficient house monitoring system prevents property thefts and offences by keeping an eye on the house. To recognise and confirm a person's face from an image or video source, facial recognition systems are frequently utilised.

Surveillance camera systems are constantly expanding, which not only makes them more convenient but also generates enormous amounts of monitoring data that are extremely difficult to store, analyse, and retrieve. The technology in the smart monitoring system allows it to both detect and pre-alarm the presence of unknown people. Here, the proposed system would identify the intrusion within seconds and alert the security.

To ensure proper safety and security our project "House Monitoring System" aims at providing safe and secure environment around our home where this is implemented. This app contains a database of known people and their names which are set to be trained.

The app provides security by detecting the faces which appear on the camera using HOG model. The face from the live feed is encoded and compared with the encodings of the trained images and then classifying them as known or unknown person.

If a known person is detected in the live feed, then the log of the person is maintained, else if an intruder or unknown person is detected in the camera, then an alert through SMS will be sent to the owner/user. The alert is sent using Twilio API. Along with the alert, the photo of the unknown person is also captured and stored in the folder of unknown people and log is updated in the system.

LIST OF FIGURES

LIST OF ABBREVIATIONS

S.NO.	WORD	FULL FORM
1.	ESR	Explicit Space Regression
2.	HVL	Home Visitor Logger
3.	RCPR	Robust Cascaded Pose Regression
4.	CRF	Conditional Random Field
5.	PSF	Point Spread Function
6.	NSPR	Noise-to-Signal Power Ration
7.	DBMS	Database Management System
8.	PIR	Passive Infrared
9.	XML	Extensible Markup Language
10.	SMQT	Succesive Mean Quantisation Transfer
11.	SNoW	Sparse Network of Winnows
12.	SVM	Support Vector Machine
13.	HOG	Histogram Of oriented Gradients

CHAPTER 1

INTRODUCTION

Face detection which is also known as facial detection is an AI related technology which detects and identifies human faces from images. This technology is used in various fields which include biometrics, security, law enforcement to ensure security and for surveillance of people in real time. From simple computer vision methods through developments in machine learning to increasingly complex artificial neural networks and associated technologies, face identification has advanced, leading to ongoing performance increases. It currently serves as the foundation for a number of crucial applications, such as face tracking, face analysis, and facial recognition. The application's ability to conduct sequential tasks has a substantial impact on face detection.

In facial analysis it identifies different parts of image which have to be focused to categorize features like age, emotion, gender using facial expressions. It basically maps the facial features using mathematical functions and stores the data as face print. After the identification, the new faceprint is compared with stored face print to detect pre-trained faces.

Face detection applications use algorithms and ML to locate a person's face within large images, which often include other non-facial features such as landscapes, structures and other parts of the human body parts such as feet or hands. The face detection algorithm is performed by searching for eyes which is easier to detect. After the detection of eye, it detects for the neighbouring parts like eyebrow, nose, mouth, nostrils. After concluding the detection, it performs additional test to confirm that it has detected the face. To improve the accuracy the algorithm needs to be trained with large datasets which includes hundreds of positive and negative images. The training improves the ability of algorithm to determine the faces accurately. The more accurate the model is, the more will be the ability of the model to detect faces.

The various advantages of face detection for users are improved security which improves the efforts of surveillance and for tracking criminals with facial detection. It's a great choice because face recognition technology is simple to implement. The majority of face recognition systems are compatible with the majority of security software, therefore there is no need to spend additional money on integration.

This project is built with the help of face_recognition module in python which is created using dlib's state-of-the-art face recognition built with deep learning. The HOG + Linear SVM face detection from Dlib is quick and effective. The Histogram of Oriented Gradients (HOG) descriptor is not invariant to changes in rotation and viewing angle because of the way it functions.

The Dlib open source C++ application and library package is released under the Boost licence, which is flexible. Dlib provides a broad range of capabilities in a number of machine learning fields, including classification and regression, numerical methods like quadratic programme solvers, a variety of image processing tools, and a variety of networking capabilities, among many other features. Dlib also offers powerful capabilities for monitoring objects, estimating their poses, categorising seen objects as faces, and face recognition

Although the Histogram of Oriented Gradients (HoG) + Linear Support Vector Machine (SVM) method in Dlib can recognise faces quickly from the front, it is only partially capable of doing so for faces posed at acute angles. Additionally, passport-style profile faces are supported, although there is very little room for mistake. HoG is a fast, reductive algorithm that divides a picture into individual groups of pixels from which it extracts "features" that may be linked to certain sorts of objects. In order to achieve this, it first builds a low-level histogram that depicts the lineaments of objects in the picture based on pixel intensity and the degree to which it abruptly drops off.

CHAPTER 2

LITERATURE SURVEY

2.1 Managing and Monitoring House Visits with Home Visitor Logger

AUTHORS: Paul-Marc BOUGHARIOS, Charbel HABIB, Semaan GEORGES and Mustapha HAMAD

Home automation is a common and particularly when it comes to home monitoring systems. The use of camera and sensor technology to achieve many applications in presence control and security systems is nothing more than a burgeoning field sample that is changing people's lives. With the entrance of computers into the engineering industry, as well as ever-increasing production costs, their involvement in projects became a need in the end. System developers can create complicated and unique solutions thanks to their capacity to conduct a number of jobs while having easy-to-access workspaces. The majority of the study is focused on constructing systems that automate manual operations like opening and shutting windows or blinds, or providing control over various domestic equipment like lighting and music systems. The administration and monitoring of house visitors is one of the most odd sorts of applications tackled in this field. The administration and monitoring of house visitors is one of the uncommon sorts of applications addressed in this sector. “Home Visit Logger” HVL provides a DBMS of the house visitors. This database management system allows the owner to keep track of the house visit in great detail, including a photo of the visitors as well as the date and time of the visit. Using a human detecting circuit, the device detects and captures house visits. If visitors are retained on the website, it also contains Image Processing using MATLAB to offer default attention. The application also links to the Internet, where the user's website is monitored online through a web site. The webpage is updated on a regular basis by the user's home system base station.

2.1.1 House Visit Loging Module:

This is made up of 3 modules, which is integrated in a way that separates the performance of the interface between the modules. In this way, each module can be completely redesigned, unless other modules are modified in any way. The first module "Audio / Image Module." Includes a Discovery Circuit and a recording device. The Acquisition Cycle detects the presence of a person near the door, activates system to include a person photography process. The device captures the photo of the home visitor, then passes on to next module. The next module, called the final module, "Photo Process Module," processes the picture acquired by the first module and then provides us with the fourth and final module. Face identification, automated recognition of family and acquaintances, and picture augmentation and blurring (for low-resolution cameras) are examples of image processing. "Communication / Communication Module" is the final module. Last module has an important role in the interaction with the other modules, it reads the trigger from the sensor circuit, takes a picture from the camera, passes it to the image processing module, and reads the module effect and saves it locally. It also enables Internet connectivity by updating the website database regularly.

2.1.2 Sensing/Imaging Module :

Imaging module is to detect guests at door, and taking a photo of the faces of the students. The two components, transmission circuit and imaging device, are manipulated by third module. The sensor is made up of a PIR infrared component and a Fresnel lens that filters infrared radiation before allowing it to reach the human body. When a person walks in front of a sensor, the sensor generates a heartbeat that is polarised depending on whether the subject walked in front of it or walked away from it.

2.1.3 Image Processing Module:

This module gives system tools that it needs to access, compare, process images captured by the Sensing/Imaging Module. It is maintained by the third module and heavily relies on the ML-IA/PT libraries. The module is made up of three MATLAB sub-modules that are built and executed within a C# wrapper function. The sub-modules are the picture capture, image enhancement, and image comparison sub-modules.

Sub-module A: The Image Acquisition sub-module is responsible for saving the image given by the Imaging Module from the imaging equipment. This initiates an investigation into the probed interfaced imaging device, or more particularly, the frame grabber driver, which is set on the trigger of the first module. The picture is then read in a format that the frame grabber driver recognises.

The following are the steps in the image acquisition process:

• Experimenting with the Imaging Device;

• Connecting the Imaging object to a MATLAB Video Input ;

• Using Video Input object to configure it;

• Getting the object started;

• Activating the object;

• Store image to hard drive from Memory.

This module disposes of the video object and de-allocates the memory space to reduce up system resource once the appropriate images have been saved. Performed because the system installation aim for less amount of hardware reliability: resource management is critical.

Module for Image Enhancement : To improve the quality of the photo captured by the device in order to make it more acceptable for storage and future querying. Several degradation effects occur as the the ambient illumination conditions of the imaging device surroundings (which are often inadequate) and low-quality picture capturing devices and frame-grabbers are among the factors that are acquired from the imaging device and recorded on the computer. Different tools from the ML-IPT can be used to apply filtering to reduce the granularity of a picture. For linear picture restoring, the expected value operator and is the statistical error function, the Wiener filter finds an estimate function that minimises the statistical error function. For linear image restoration, Wiener filtering is utilised, and the filter looks for an estimate function that minimises the statistical error function, where is the expected value operator and is the degraded picture. This form assumes the Noise-to-Signal Power Ratio (NSPR) is 0 for an image device directly connected to a computer, which is top-probability condition.

2.2: Face Modeling Process Based on Dlib

AUTHOR : Xiujuan Ren, Junhang Ding∗, Jinna Sun, Qingmei Sui

Face recognition technology is widely employed in sectors of security, and other field. As the performance criteria for people face recognition systems become more demanding, the ability to rapidly and precisely locate faces has become a new era necessity. In reality, however, other elements include light alterations, occlusion difficulties, and attitude expressions have all had an impact on the face feature point positioning. Face detection systems typically consist of picture face location, image pre-processing & face detection, as well as significant modules such as face detection and face feature points. The face alignment problem can now be solved using the csf training model. Cascade regression begins at the pixel level, and the picture is progressively regressed to generate a shape approximation. Random forest has gained popularity in the field of computer vision in recent years due to its ability to test the main image classification, behaviour identification, target detection and segmentation, human posture estimation, facial expression recognition, 3D face analysis I, and face feature point positioning with a simple, relatively low computational complexity. ESR and RCPR employ the random fern (one of the trees), global shape regression, and shape index feature techniques to identify the feature point. The facial expression, scale, and attitude features are all included in the form index feature. The key to rectifying the shape is the invariant feature. In addition, ESR improves positioning accuracy by employing cascade regression and a multiple shape initialization technique. The disadvantage is that the model is rather large. The training sample is marked in the facial feature point occlusion information by RCPR, which improves placement accuracy. The notion of training the forest model of the five types of posture points, which can successfully find the various motions, but adds computing complexity and does not apply any form of the restrictions, resulting in inaccurate detection that is sensitive to the block, was introduced by CRF. To tackle the above-mentioned local feature technique, as well as the research results and problems of random forest regression, a feature point localization methodology based on random forest regression is provided. It establishes shape of global optimization model and the local detector of the random forest model. The stochastic forest model is constructed at point in this research, forest model is used to estimate the shape of the training sample. The estimated shape and the true form of the training sample are then fitted using linear and least squares methods. Then apply the global model to optimise the shape of the test sample, and then use the cascade regression form to realise face of face. Feature points are automatically positioned. A random forest regression-based feature point localization technique is proposed. The approach establishes the forest model at each feature point using the face pixel difference feature. The pixel difference feature can significantly reduce computing time while also reducing the impact of expression, attitude, illumination, and other factors. Then, using the forest model regression, you'll be able to the location of the sample feature points can be determined by traversing the tree and averaging the tree output result, which can significantly enhance the algorithm's speed. Least fitting squares approach employed to investigate the situation. The resulting model can be used to optimise the test sample's feature point position, reducing the impact of occlusion and other factors and therefore improving the algorithm's accuracy.

2.2.1 Principle of Face Feature Point Positioning:

The face feature localization method works by first obtaining a face detection box with the ViolaJones detector, and then initialising the shape of the sample. A stochastic forest model is formed at the position of each feature point during training using the pixel difference feature, and the built-in model is then used to estimate the position offset of the training sample feature point and the position offset of each feature point during testing using the built-in model. The linear incremental least squares approach is used to fit the shape increment of the training sample and the real shape increase in order to create a global optimization model. The stochastic forest model is used to estimate the positional deviation of the test samples' feature points, and the offset is acquired in series to provide the test samples' shape increment. The shape increment is then optimised using the learnt global model. After combining the ideal form shape with the initial face shape, the final face shape can be generated, and the facial feature points can be automatically arranged.

2.3 Review and Comparison of Face Detection Algorithm

AUTHOR: Kirthi Dhang, Shanu Sharmaa

The rapid advancement in technology is because computers are becoming more complex. These efficient computers enable human interaction, ushering in this time of discovery in a variety of sectors. Such a fields based on this is face recognition, which is subdomain of object detection. Detection of items in a specific class in an image is known as object detection. Object detection has a wide range of applications in this time, including face recognition, other detection, and so on. Face detection is the starting point for all facial analysis methods, including face recognition, face alignment, face modelling, face verification, and facial tracking. The computer must first detect the face before proceeding with the analysis. Face detection is a technique for identifying faces in a picture by separating the patterns that make up the faces from the dimensions of each face. If face modelling and segmentation are done correctly, this is conceivable. This also necessitates taking into account factors such as pose, resolution, focus, shadow, and occlusion. Face recognition in general is demonstrated. Face detection has been the subject of significant research in recent decades, a very simple task for computers.

Basic algorithms of object detection are:

1. Viola Jones

2.SMQT Feature & SNOW Classifier

3.Neural Network

4. Support Vector Machines

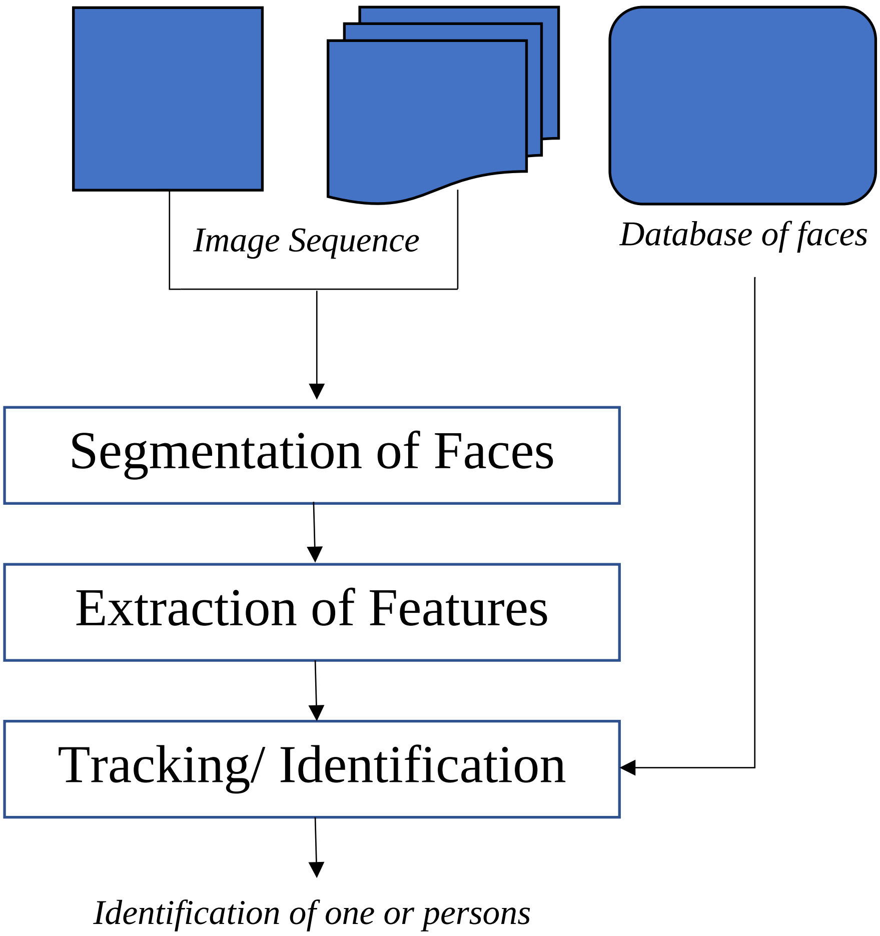


Figure 2.1:The process of face detection

Following the detection with various methods, precision and recall of each approach is compared. The face locations are contained in the first XML, which contains the truth dimensions value. Performance of XMLs is calculated. The recall and precision of object detection algorithms have been calculated using a single file

$$R_{OB} = \frac{\text{No. of correctly detected rectangles}}{\text{No. of rectangles in the database}} \quad \text{....(2.3.1)}$$

$$P_{OB} = \frac{\text{No. of correctly detected rectangles}}{\text{Total No. of detected rectangles}} \quad \text{....(2.3.2)}$$

Precision & recall is metrics values to assess the efficiency of algorithm. Precision, is also called positive predictive value, is a number that expresses the percentage of obtained values that are necessary which provides info on false alarm. No. of objects discover is represented by recall. The closer the perfect system is to 1, the more accurate the algorithm's results.

2.3.1 Face Detection Algorithm:

- Viola Jones
- SMQT features and SNOW Classifier
- Neural Network
- Support Vector Machines

On the basis of the precision and recall values, it can be inferred that Viola-Jones has the highest value, followed by SMQT Features and SNOW C, then Neural, and finally SVM. For face detection, Viola-Jones is best of these algorithms. Approach can be used to compare different additional object algorithms in the future.

2.4 Three Triangle Method for Face Recognition using Dlib and OpenCV

AUTHORS: earlie kit m sanches, noel b linsangen, randy e angelia

To date, facial recognition has been one of the most popular biometric identification techniques. Since Woodrow Wilson Bledsoe's first use of it in the 1960s, improvements and other algorithms and approaches have been explored. Governments, organisations, and IT corporations are all begging for a good strategy and policies around it these days, with some even demanding that it be appropriately constrained. Everyone is still racing to develop a facial recognition system that is quick, accurate, and non-racist. Face recognition is predicted to evolve more than ever as a result of the pandemic's tremendous digital transition. Despite technological developments, the traditional approach of facial recognition still has to be improved.

The method of facial detection is complicated by variances in people's faces such as expressions, complexion, the usage of eyeglasses, the presence of facial hairs, lighting differences, and camera quality. As a result, researchers are still working to address these challenges. Paul V. and Michael J. discovered a 15x faster and more reliable facial identification system, with a stunning 95 percent accuracy at around 17 frames per second, compared to other systems. One study offered a new dataset containing 106 points for facial landmark detection, which includes a variety of facial angles, skin colours, and lighting situations in the hopes of improving facial detection technologies. Face form and features like eye, nose and mouth is used for traditional facial recognition systems. It is critical to deal with landmark detection.

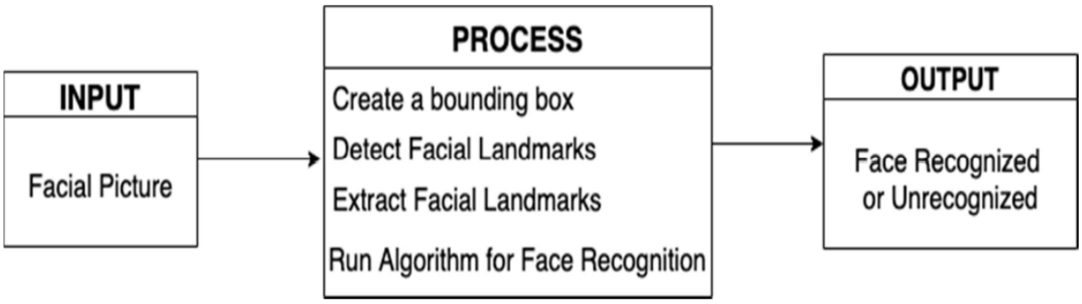


Figure 2.2 : Conceptual Network

Figure. 2.1 shows this paper's Conceptual framework diagram. A live video feed will serve as the input. OpenCV's Haar cascades detects a face on video feed which will then make a bounding box around said face. Once the bounding box for the

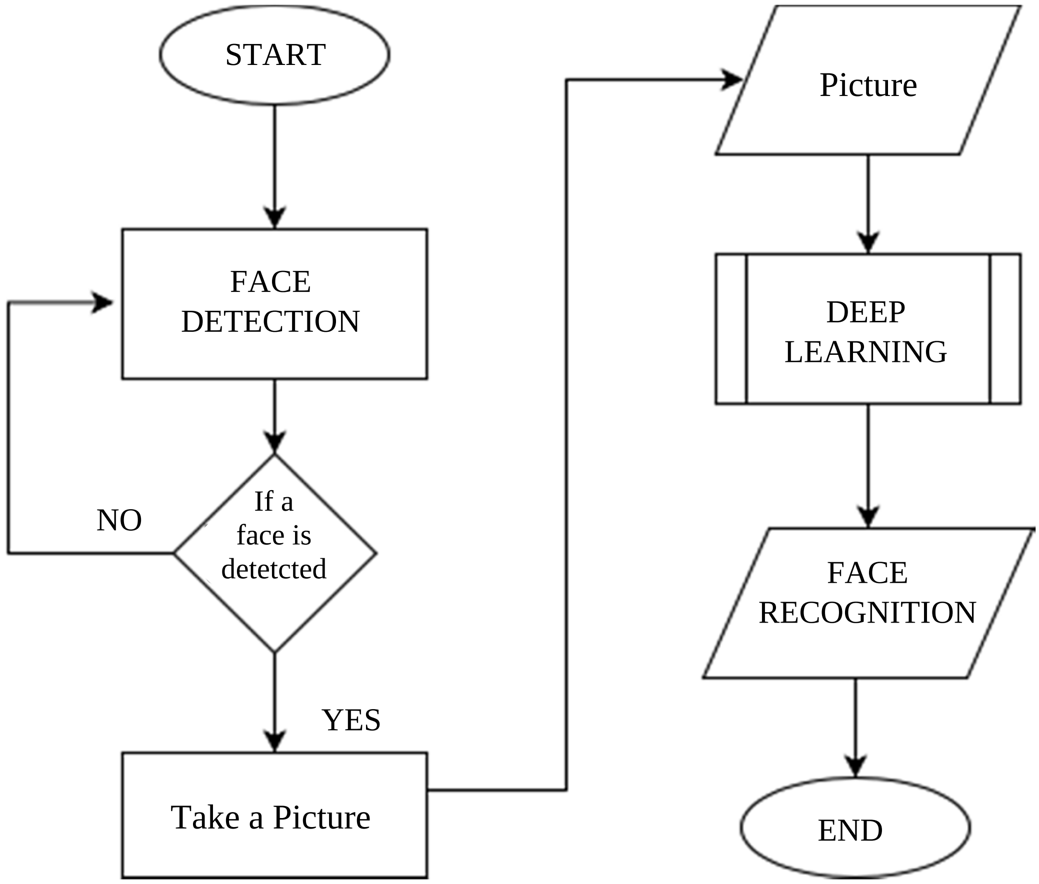


Figure 2.3 : Main program flowchart

The application will initially look for a face in order to accurately recognise it. Face detection is aided by using the OpenCV Library's Haar Cascade face detector. It's a well-known machine learning object detection technique that can recognise faces and body components in an image or video. It recognises a face by using Haar Features, which are described as an adjacent rectangular region in a given area of the image.

The Dlib 68-point feature extraction is used to properly extract the landmarks. It employs the 68 points that round the face. It's a 68-point face model that can quickly detect facial landmarks. This 68-point model shows how easy it can access face features like the eyes, nose, and mouth. Utilizing the Dlib 68-points feature extraction, the researcher created a Custom 4-point Landmark model for the feature extraction.

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}} \qquad \ldots\ldots\ldots (2.4.1)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}} \qquad \ldots\ldots\ldots (2.4.2)$$

$$\text{F1} - \text{score} = \frac{2*\text{recall}*\text{precision}}{\text{recall}+\text{precision}} \qquad \ldots\ldots\ldots (2.4.3)$$

2.5 Smart Home Security using IoT and Face Recognition

AUTHOR: Suraj Pawar, Vipul Kithani, Sagar Ahuja, Sunita Sahu

House security system keeps burglars and thieves out of your home and valuables, as well as your family. Property crimes account for 77% of all crimes, with guns accounting for 38% of robberies. In the United States and the United Kingdom, identity theft is fastest growing offense. Using IoT and face recognition, our proposed system provides a low-cost and energy-efficient option for home security. Because the face detection module performs well in some conditions, the person does not wear accessories that cover part of the face, IoT will enable sensing and triggering system on motion detection using sensors such as Pir for detecting motion and Ultrasonic for measuring distance to calculate the position of person because of face detection module. The LBP achieved an accuracy of 80% when tested using real-time photos. Someone should stand in frame of camera will recognise the face and compare it to the faces saved in the Raspberry Pi's home member db. If match of face is found, then the door will automatically be unlocked else it will remains lock. In the future, a more precise facial recognition algorithm may be presented, and more accurate and less expensive sensors may be used to improve the system's precision. The Android app will be updated to make it easier to use, Artificial intelligence will be used to generate digital assistance for home monitoring.

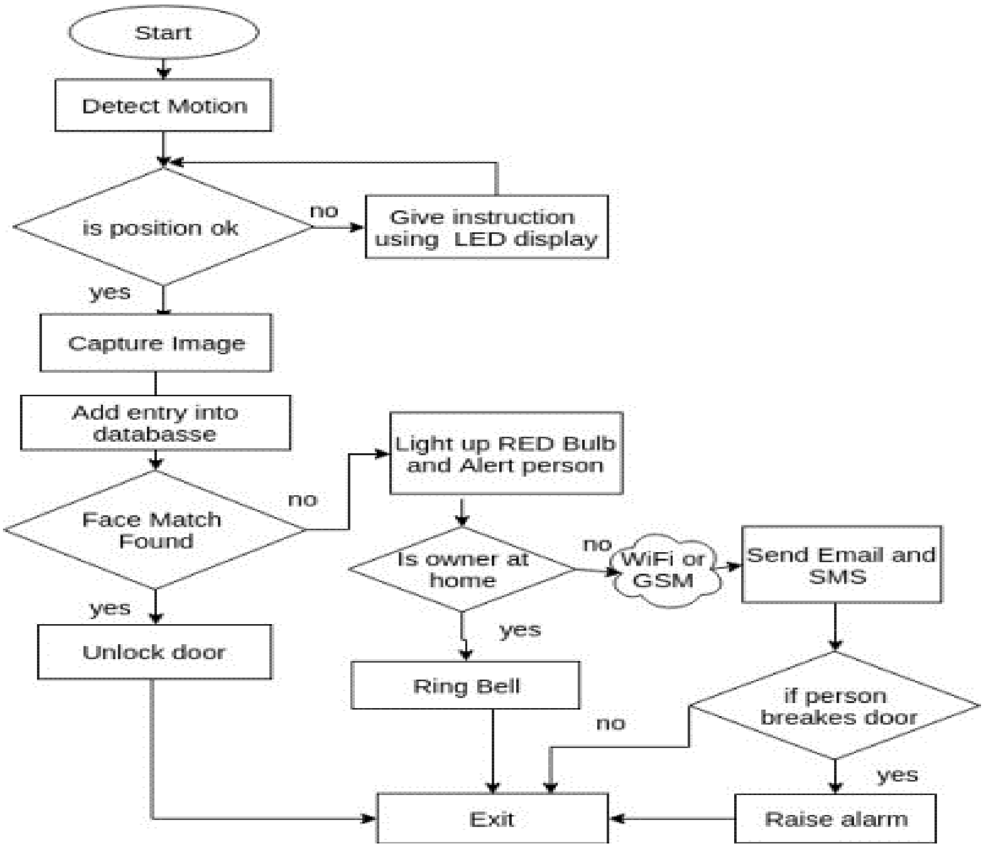


Figure 2.4:System flow diagram

2.6 Design of Face Detection and Recognition System for Smart Home Security Application

AUTHOR: Dwi Ana Ratna Wati, Dika Abadianto

Home security is extremely crucial for the community as well as individuals and for a long time they have been addressing the problems of traditional home security which is the common type of home security where user mechanically uses a key to open and close the doors such as the key duplication, losing key or changing hands. Therefore it can be said that, the level of security provided by the same is extremely poor. This method is also ineffective and inefficient because a huge amount of time is lost in the process of opening the door where the user has to inert the key into the lock and turn the key inside the lock in a particular direction. Compared to this, the smart home security system is incredibly functional, productive and gives maximum security to homes and the community as a whole. Face recognition technology replaces the traditional key with camera which recognises the face of the owner as soon as they approach the door making it more efficient and effective. The problems of duplication or misplacing the key is therefore can be avoided.

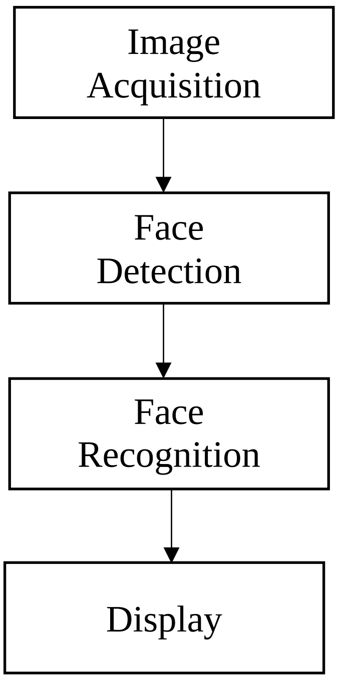


Figure 2.5: Steps in face recognition system for smart home security application

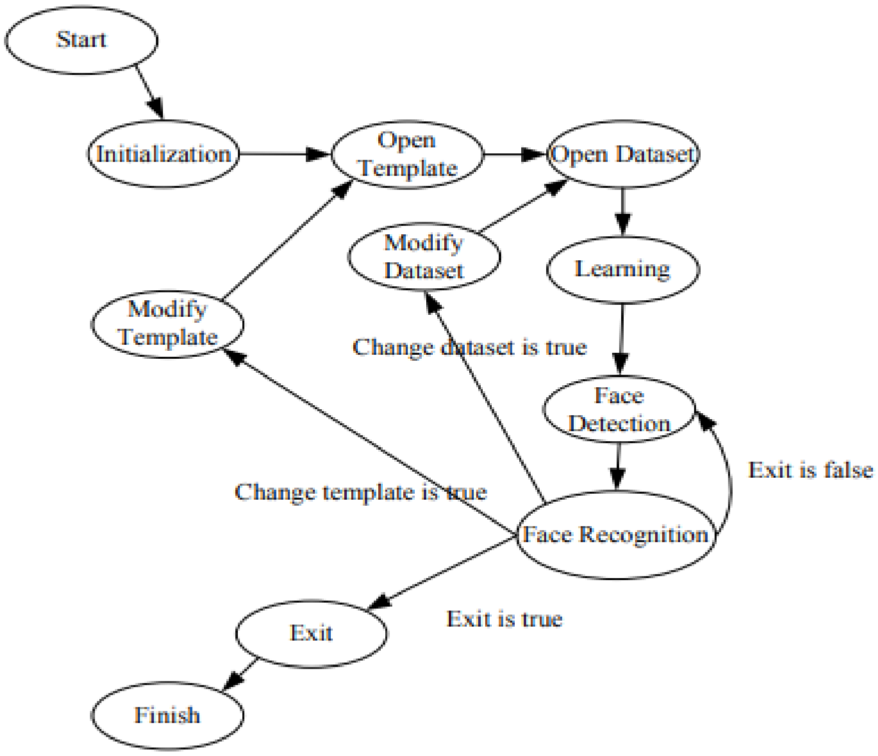


Figure 2.6: State diagram of face recognition software

The study aimed to develop a model of smart home security system using face recognition.

The findings of the testing include

1. The face recognition model works well during the following circumstances.

A) The distance between the camera and face must be under 240 cm.

B) The whole face of the individual must be clearly visible.

C) Individual must avoid wearing shirt matching their skin tone.

D) Background colour must be distinct from the skin tone of the individual.

2. The performance of the model will be determined by the minimum Euclidean distance threshold and an accuracy of 80% will be obtained using the criteria of 0.14x1014 as the minimum Euclidean distance.

2.7 Development of Application and Face Recognition for Smart Home

AUTHOR : Sere Khunchai, Chaiyapon Thongchaisuratkrul

In Thailand, the need for a smart house has grown in recent years. The Internet of Things plays a major part in everyday. The smart home eliminates the issue to turn off equipment, which can result in excessively high electricity bills or even a house fire. It is critical to have a home security for youngsters and the elders. In the not-too-distant future, the smart home will play a larger role in easing daily life. The system applies to all house occupants, not only youngsters, the elderly, or those with impairments. The smart home system, on the other hand, is pricey. To control the system, multiple programmes or applications are required for each device and brand. Price is the most important consideration for 42% of non-smart home consumers. The smart home, which is controlled by a sensor, a computer, and an Android app, is the subject of this research. It may be controlled by a remote or other smart home devices. The author wants to make Android and iOS applications for smart homes on the Internet that can be managed with Thai voice commands and include a low-cost facial recognition system for security. The presentation "Developing an Android Application-Controlled Smart Home System" served as the inspiration for this work. The author wants to design a smart home app for Android and iOS that can be accessible via the internet. The voice control system supports the Thai language. Face recognition for smart homes is possible thanks to an intelligent system. With an intelligent system, face detection for smart home provide security. However, it is less expensive and simpler to use.

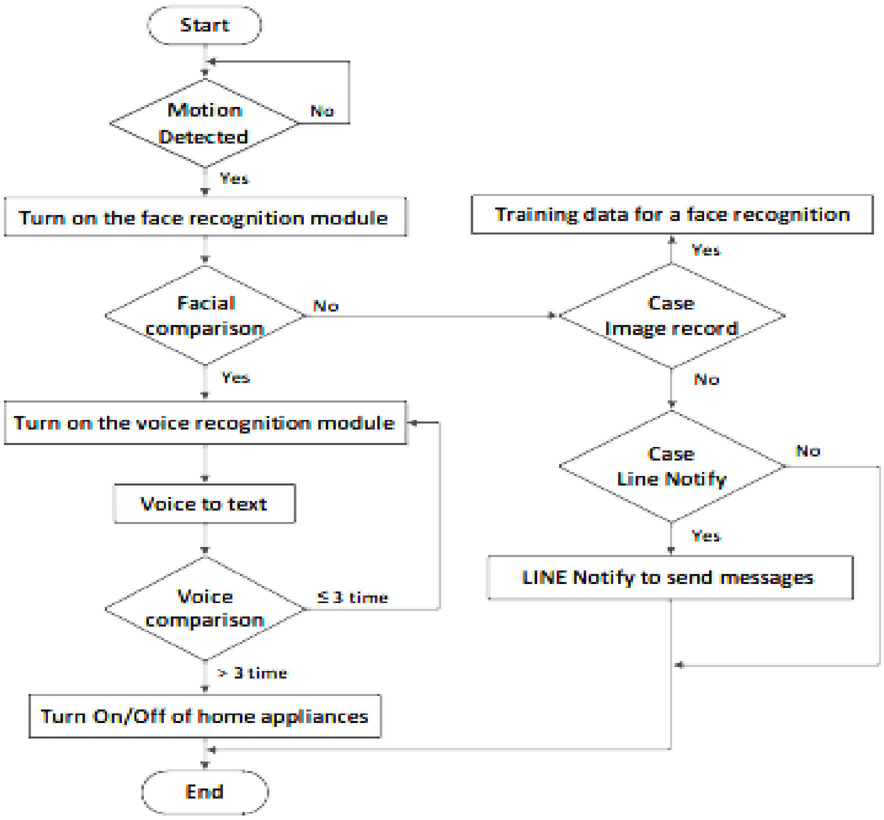


Figure 2.7:Flowchart of face recognition program for smart home

The system will be slower and more prone to mistakes if the user has a familiar face or wears the same glasses. The number of faces used for facial recognition, as well as the distance between the camera and the users, affects the speed. The accuracy of voice control suffers as the distance between the microphone and the user increases. The voice control system, on the other hand, will operate at 85 decibels and a distance of no more than 4 metres. For manual control of the smartphone, which is dependable, and can also appropriately regulate the electricity through the smartphone in both Android and iOS apps. A time delay of 1 to 3 seconds, which fluctuates depending on the speed of data flow, is the system's weakness. This research provides as a template for project-based learning advancement: A smart house case study that may be used as a blueprint for creating low-cost, easy-to-implement smart home innovation. To learn more, the author feels that residents would benefit from the development of facial recognition and voice control, as well as low-cost extension and enhanced intruder detection distance.

2.8 Human Detection using HOG-SVM, Mixture of Gaussian and Background Contours Subtraction

Author: Abdourahman HOUSSEIN AHMED, Kidiyo KPALMA1 , Abdoulkader OSMAN GUEDI

Object detection in video is first stage in applications that handle video data automatically. Many methods have been proposed in this field in recent years, including the background subtraction from background, the Gaussian mixture method, and the systems HOG and Haar. However, the task's difficulty stems primarily from the challenge of accurately modelling the objects/Humans because to their wide range of physical appearance, position, lighting, and other factors. A human detection on histograms of gradients, svm classifiers, Gaussian mixture.

A human detection method is provided that uses a HOG, a svm classifier, a Gaussian mixture, and background subtraction. In the general instance, the extraction of important primitives is a problem without a solution because it appears that there are no qualities that can describe a basis from all perspectives. To increase Human detection, current systems aim to mix multiple primitives. By integrating a Gaussian approach and the elimination of background contours, this method enhances segmentation compared to the HOG-SVM Human detection system. The results show that this technology may be utilised in intelligent video-protection systems due to its efficiency.

2.9 Pedestrian Detection Based on Motion Compensation and HOG/SVM Classifier

Author: Fen xu,Feng xu

Detecting moving human bodies in live videos, particularly those captured by a moving camera, is not an easy process. In most situations, detection is split into two stages: moving object segmentation and human body categorization. In the segmentation step, optical flow and difference approaches are widely used. Template matching, feature characterization, cluster analysis, and machine learning approaches employing techniques like support vector machines, Adaboost, neural networks, and others are common classification methods. In the last decade, numerous studies have been conducted in this field. Segen and Pingali use curve to determine the match zone between frames after extracting the edge feature. Oren M uses a wavelet to extract the pedestrian template. Heisele B detects pedestrians using a colour characteristic. Ronfard segmented the human body into numerous components, retrieved the gradients for each region, and built a classifier for the human body using SVM. Searches videos for suitable candidates based on the contour information of the human body. To detect the potential component of human bodies, uses a combination of an orientation histogram and a gradient histogram. Viola P proposes a method for detecting pedestrians that combines a contour model with leg motion patterns.

They created a system that used motion correction with HOG&SVM to recognise pedestrian in changeable backdrop. This paper explains how to detect pedestrians using a strategy which combines motion correction and HOG&SVM. The camera projection model is used to produce the transformation matrix. The results of the tests indicate that this method is successful at detecting pedestrian targets in a changing environment.

2.10: A review of the comparative studies on traditional and intelligent face recognition methods

Authors: Enjie Jiang

Face recognition becomes sophisticated as biometric identification technology advances. Face recognition are employed in everyday life in a variety of domains, including politics, the military, the business, and entertainment. For example, we may use our phones to "recognise our faces" to pay for groceries and travel to the station, as well as detect criminals in crowds using face recognition. As a result, face recognition technology became incorporated into every part of people's everyday lives, ushering in the "facial recognition" age. Face recognition is the process of confirming a face image in a database by comparing it to a static or dynamic image from one of the samples. Human faces, like human fingerprints and iris, are self-contained, and they can perform identity identification tasks such as modern fingerprint, DNA recognition, and so on. Face recognition technology has various advantages over fingerprint and DNA recognition, therefore it increases identity identification efficiency and provides additional development and application options. Face recognition, on the other hand, would constantly be hampered in real-world applications by external factors such as lighting, expression, and colour if viewed from a different angle. In the realm of face recognition, the algorithm's capacity to eliminate interference and effectively realise face identification is an essential study subject.

2.10.1.Traditional face recognition technology:

Traditional facial recognition techniques come in a variety of forms. Face recognition using a geometric technique & face recognition using PCA approach both bought in this study. They were then analysed and compared to AI system below in order to determine its benefits, drawbacks.

Face recognition is a traditional technical procedure

(1)Detection of faces:

The system must decide whether or not there is a face in the image before it can detect it. If this is the case, find position with coordinates

(2) Extract features:

Detecting and identifying the feature points of face organs can be used to determine the position of facial feature points. Algorithm can determine the precise description information of facial shape and organs, and then acquire the description of facial features. An overview of geometric feature-based face recognition algorithms. Face recognition research was mostly undertaken in the early stages of the field (1964-1990) by analysing the aspects of face geometric structure. This algorithm has improved throughout time as science and technology have progressed. The eyes, nose, ears, and mouth make up the human face. Varied people's facial features have different shapes. Because of the regularity of human genetic and physiological attributes, its structure can be used as a face recognition indication. In the early phases of this method, characteristics such as ratio distance of facial feature points were utilised as features for face identification, based on manually identifying the feature points, in order to achieve face recognition in a semi-automatic manner.

Face recognition has gone through four stages: theoretical, semi-automatic, automated, and intelligent recognition. New and more efficient algorithms arise one after the other as time passes. We may conclude from the prior examination of face recognition technology in this work that traditional face recognition is more vulnerable to external malignant conditions and has poorer recognition accuracy when compared to intelligent algorithms. Combining multiple strategies can increase the accuracy of face recognition. Deep learning and neural network-based face identification systems are more resilient, automated, and accurate than traditional face recognition algorithms. Face recognition systems, such as the Facenet algorithm, can even outperform the human eye. However, intelligent face recognition algorithms, such as the above-mentioned enhancement of neural network structure and the loss algorithm, still have a lot of room for improvement. With theoretical studies of deep learning algorithms, research into semi-supervised and unsupervised learning, and so on, face recognition still has a long way to go in terms of progress.

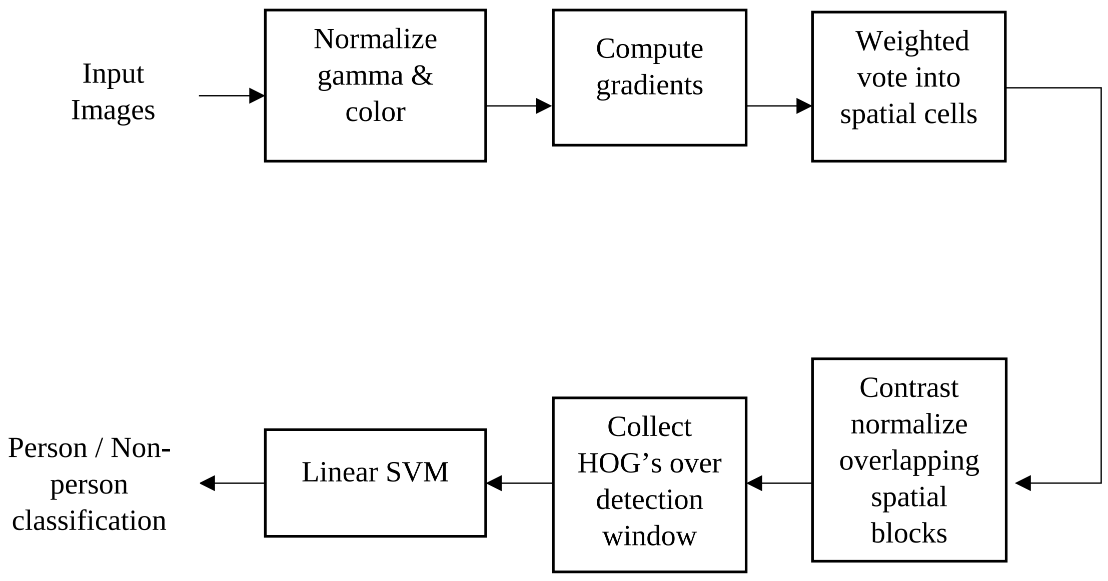


Figure 2.8: HOG & SVM algorithm flow chart

CHAPTER 3
PROPOSED SYSTEM

Using a camera and facial recognition, our suggested system provides a cost and energy efficient alternative for home protection. When the face of the person appears in the camera the frame gets encoded and compared with the encoded value of the trained images.

The app provides security by detecting the faces which appear on the camera using HOG model. The face from the live feed is encoded, compared to the trained photos' encodings, and then classified as belonging to a known or unknowable person.

If a known individual is detected in the live feed, the person's record is kept; otherwise, if a stranger or intruder is seen in the camera, the owner or user will receive a text message alert. The Twilio API is used to send the alert. Along with the alarm, the unknown person's picture is also taken, saved in the folder for unknown persons, and the log is updated.

3.1 ALGORITHM USED

We have utilized face_recognition algorithm which is a python package created using dlib's state-of-the-art face recognition built with deep learning. The HOG + Linear SVM face detection from dlib is quick and effective. For the dlib facial recognition network, the output feature vector is 128-d (i.e., a list of 128 real-valued numbers) that is used to quantify the face. Training the network is done using triplets:

HOG model is used to detect the faces in the image. After creating the image's 128-dimensional encoding. This 128-dimensional vector computes the Euclidean distance between it and each of the 128-dimensional vectors in the faces array that we previously computed. Then the value of the encodings are sorted, the smallest number indicates that these 128-d vectors are closest to the input image in Euclidian space. This suggests that the input image and these pictures are the most similar.

3.1.1 STEPS PERFORMED

1. We pass the person’s picture to the model and their name.
2. The model takes every picture, converts them into some numerical encoding, and stores them in a list and all the labels (names of persons) in another list.
3. In the Prediction Phase when we pass a picture of an unknown person recognition model converts the unfamiliar person’s Image into encoding.
4. After converting an unknown person’s Image into encoding, it tries to find the most similar encoding based on the distance parameter. The store encoding with the least distance from the encoding of an unknown person will be the closest match.
5. After getting the closest match encoding, we take the index of that encoding from that list and use indexing. We find the detected person’s name.

3.1.2 HISTOGRAM OF ORIENTED GRADIENTS (HOG) WORKING

HOG, or Histogram of Oriented Gradients, is a feature descriptor. For the goal of object detection, it is utilised in computer vision and image processing. In the confined area of an image, the technique counts instances of gradient orientation. The HOG description emphasises an object's structure or shape. Since it computes the features using both the magnitude and the angle of the gradient, it is superior to other edge descriptors. It creates histograms for the areas of the image based on the gradient's magnitude and directions.

To find faces in an image, we’ll start by making our image black and white because we don’t need color data to find faces. Then, we'll examine each individual pixel in our image one by one. We wish to examine the pixels immediately surrounding each individual pixel. To determine how dark a pixel is in comparison to its immediate surroundings is our objective. Then, we want to depict an arrow indicating the way that the image is darkening. Every pixel in the image will eventually be replaced by an arrow if you carry out that process for every single pixel in the image. These arrows, which are known as gradients, depict the transition from light to dark throughout the entire image.

Images of the same individual that are extremely dark or extremely light will have entirely different pixel values if we evaluate the pixels directly. However, if the brightness is only taken into account in one direction, both extremely dark and extremely bright images will represent the same thing. However, keeping the gradient for each every pixel gives us far too much information. In the end, we fail to see the big picture. To better understand the fundamental pattern of the image, it would be preferable if we could just perceive the basic flow of lighting and darkness at a higher level. We'll do this by cutting the image into squares that are each 16x16 pixels in size. To find faces in this HOG image, all we have to do is find the part of our image that looks the most similar to a known HOG pattern that was extracted from a bunch of other training faces:

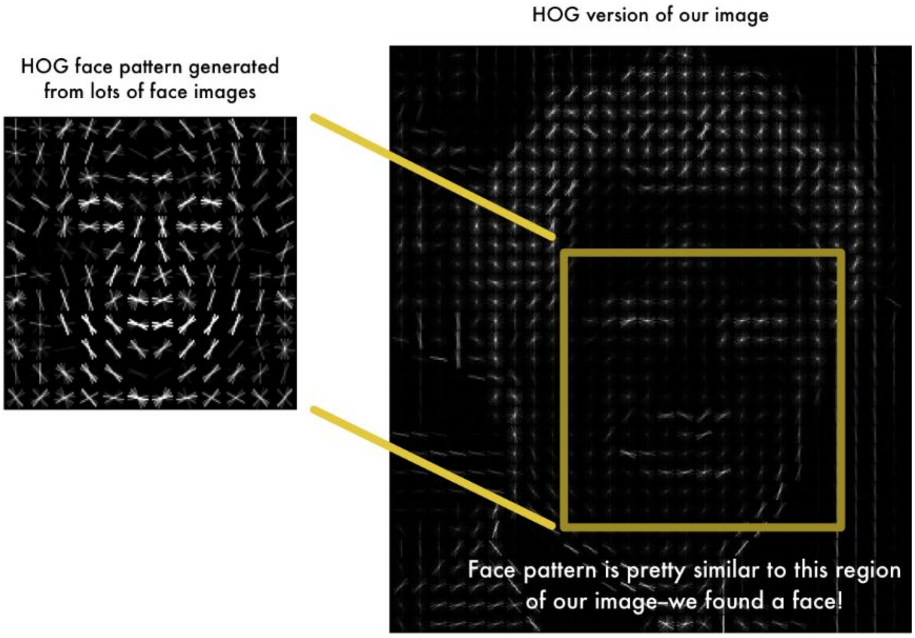


Figure 3.1: HOG pattern generation

3.1.3 ENCODING FACES

The training process works by looking at 3 face images at a time:

- Load a training face image of a known person
- Load another picture of the same known person
- Load a picture of a totally different person

The neural network learns to accurately create 128 measurements for each person after doing this step millions of times for millions of photos of thousands of distinct people. The measurements should be nearly the same for any ten images of the same person. Run our face images through pre-trained face encoding network to get the 128 measurements for each faces.

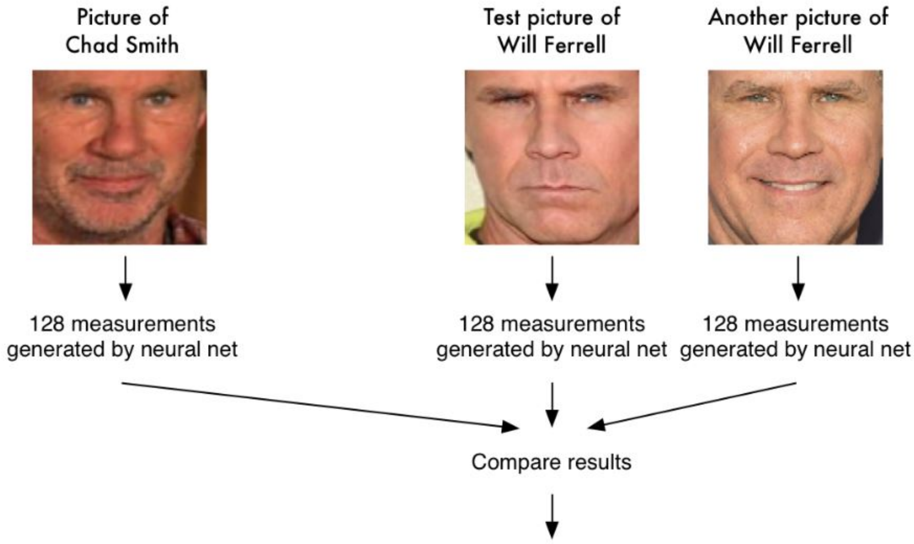


Figure 3.2: Encoding technique

3.1.4 CLASSIFYING IMAGE

All we need to do is train a classifier that can take in the measurements from a new test image and tells which known person is the closest match. Running this classifier takes milliseconds. The result of the classifier is the name of the person. We could use any classification methods. Here we have used Linear SVM model to classify

3.2 SYSTEM REQUIREMENTS

The most common set of requirements defined by o or software application is physical resources, also known as hardware. Frequently, a list of hardware requirements is accompanied by a list of hardware compatibilities. The following components are required for the system:

(a) Processor : i5 processor
2.5Ghz or faster processor is required for proper functioning and fast execution of the program

(b) 4GB RAM
Ram is required for functioning of system without lag

CHAPTER 4

IMPLEMENTATION

The system design of our project is as follows:

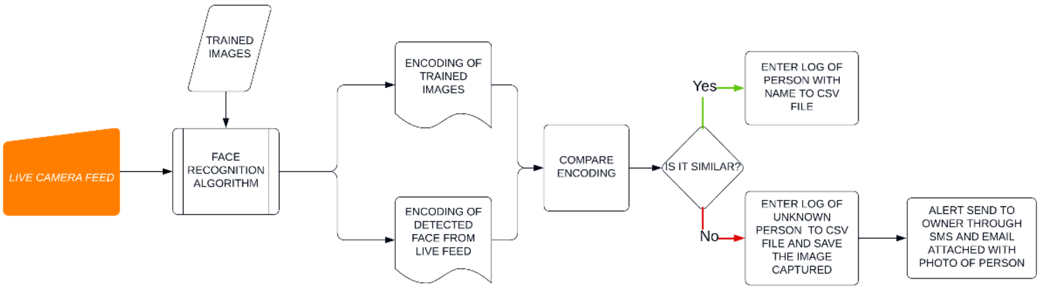


Fig 4.1: Implementation flow chart

4.1 MODULE DESCRIPTION

4.1.1. FACE ENCODING:

Face encoding is a method of representing the face with 128 computer-generated measures. Two photographs of the same person would have similar encoding, whereas two photographs of two distinct persons would have completely different encoding.

Given an image, return the 128-dimension face encoding for each face in the image.

face_recognition.api.face_encodings(face_image,known_face_locations=None, num_jitters=1, model='small')[source]

Parameters:

- face_image – The image that contains one or more faces
- known_face_locations – Optional - the bounding boxes of each face if you already know them.
- num_jitters – How many times to re-sample the face when calculating encoding. Higher is more accurate, but slower (i.e. 100 is 100x slower)

- model – Optional - which model to use. “large” or “small” (default) which only returns 5 points but is faster.

  Returns:

  A list of 128-dimensional face encodings (one for each face in the image)

4.1.2. LIVE FEED CAPTURING AND COMPARING WITH TRAINED IMAGES:

The live feed from webcam is taken and the face is live frame is encoded with the help of face locations and the live image frame.

The live frame encoding value is compared with the trained image encoding values list and gets the Euclidean distance for each comparison face

Compare a list of face encodings against a candidate encoding to see if they match.

Syntax:

face_recognition.api.compare_faces(known_face_encodings,
face_encoding_to_check, tolerance=0.6)

Parameters:

- known_face_encodings – A list of known face encodings
- face_encoding_to_check – A single face encoding to compare against the list
- tolerance – How much distance between faces to consider it a match. Lower is more strict. 0.6 is typical best performance.

Returns:

A list of True/False values indicating which known_face_encodings match the face encoding to check

4.1.3. DECIDING KNOWN OR UNKNOWN PERSON:

If the argmin value of the live frame is matched with nearest value of trained images then the corresponding name of the detected person is displayed and the function for marking log for known person is executed. Else it is classified as unknown person and the corresponding function gets executed

4.1.4.KNOWN PERSON LOG ENTRY:

The name of the detected people who are known person are entered into the "KnownList" csv file which contains all the log datas of known people

The images of the person are taken and saved into another folder as jpg files

4.1.5. UNKNOWN PERSON LOG ENTRY:

The detected people who are unknown and not entered into dataset are entered into the "UnknownList" csv file which contains all the log datas of unknown people detected.

The images of the unknown person are taken and saved into folder as jpg files

4.1.6. ALERT SYSTEM FOR UNKNOWN PEOPLE:

If an unknown person is detected the photo of the person along with an alert message will be sent to the registered number of the house owner or user

The alert system is connected through an SMS sending API "Twilio"

CHAPTER 5

EVALUATION

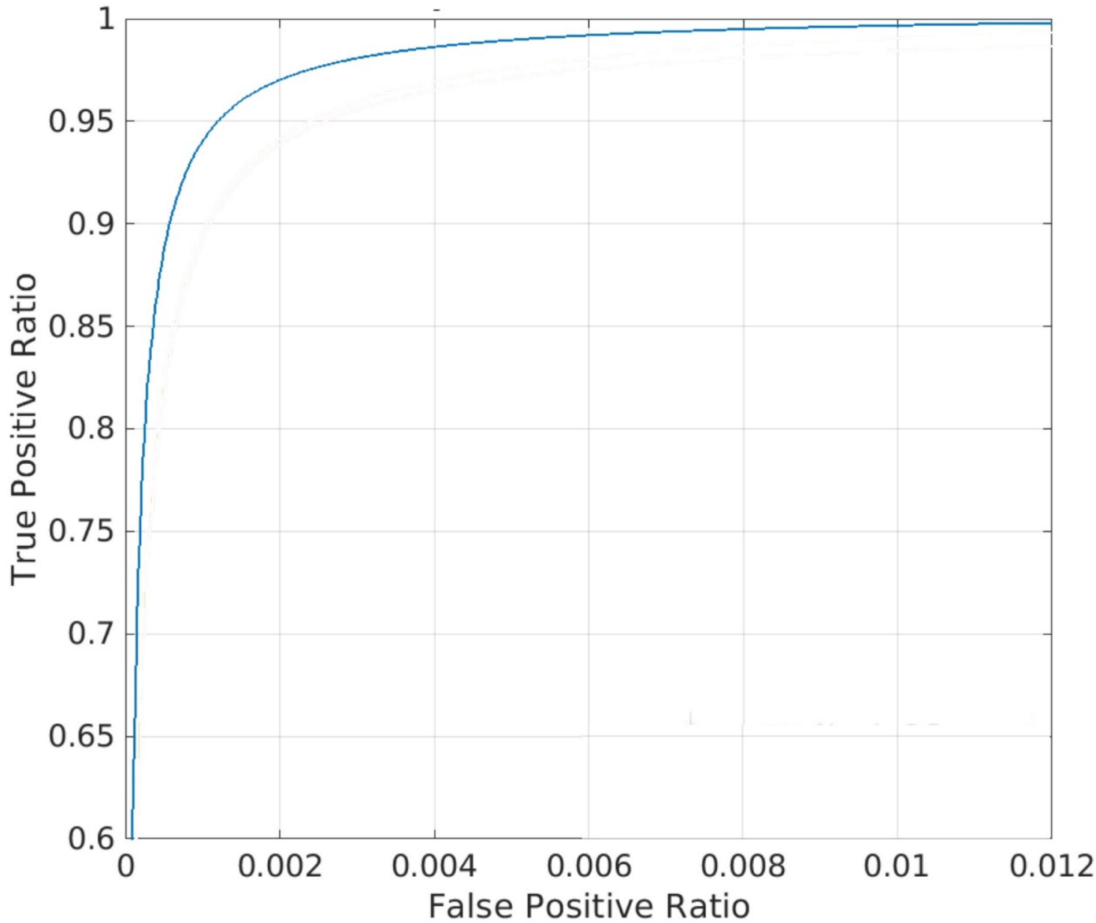
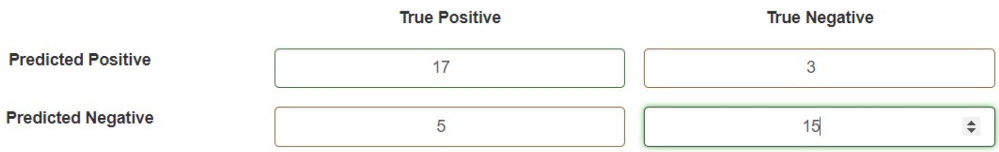


Figure 5.1 : ROC curve



	True Positive	True Negative
Predicted Positive	17	3
Predicted Negative	5	15

Figure 5.2 : Confusion matrix

Measure	Value	Derivations
Sensitivity	0.7727	TPR = TP / (TP + FN)
Specificity	0.8333	SPC = TN / (FP + TN)
Precision	0.8500	PPV = TP / (TP + FP)
Negative Predictive Value	0.7500	NPV = TN / (TN + FN)
False Positive Rate	0.1667	FPR = FP / (FP + TN)
False Discovery Rate	0.1500	FDR = FP / (FP + TP)
False Negative Rate	0.2273	FNR = FN / (FN + TP)
Accuracy	0.8000	ACC = (TP + TN) / (P + N)
F1 Score	0.8095	F1 = 2TP / (2TP + FP + FN)

Figure 5.3 : Evaluated values

Sensitivity (Recall):

Correct predictions which are positive divided with positive gives sensitivity.

Also known as Recall or TPR

High level is 1.0 and 0.0 is low

$$\mathrm{SN} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} = \frac{\mathrm{TP}}{\mathrm{P}} \qquad \text{……… (5.1)}$$

Specificity (True negative rate):

Negative prediction divided with negative gives the specificity

Also called TNR

High level is 1.0 and low is 0.0

$$\mathrm{SP} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} = \frac{\mathrm{TN}}{\mathrm{N}} \qquad \text{……… (5.2)}$$

Precision (Positive predictive value):

Correct positive prediction divided with positive predictions gives Precision

Also called PPV

High level is 1.0 and low level is 0.0

$$\mathrm{PREC} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \qquad \text{………} (5.3)$$

False positive rate:

Inaccurate positive predictions divided with negative prediction gives FPR

High value is 0.0 and low value is 1.0

Calculated as 1-specificity

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{TN} + \mathrm{FP}} = 1 - \mathrm{SP} \qquad \text{………} (5.4)$$

Negative predictive value:

Ratio of correctly predicted as negative with actual negative

False negative rate:

A false negative rate, also known as a false negative

It is a test result that shows that a condition does not exist when it does not.

OUTPUT:

The output obtained from our project is shown below:

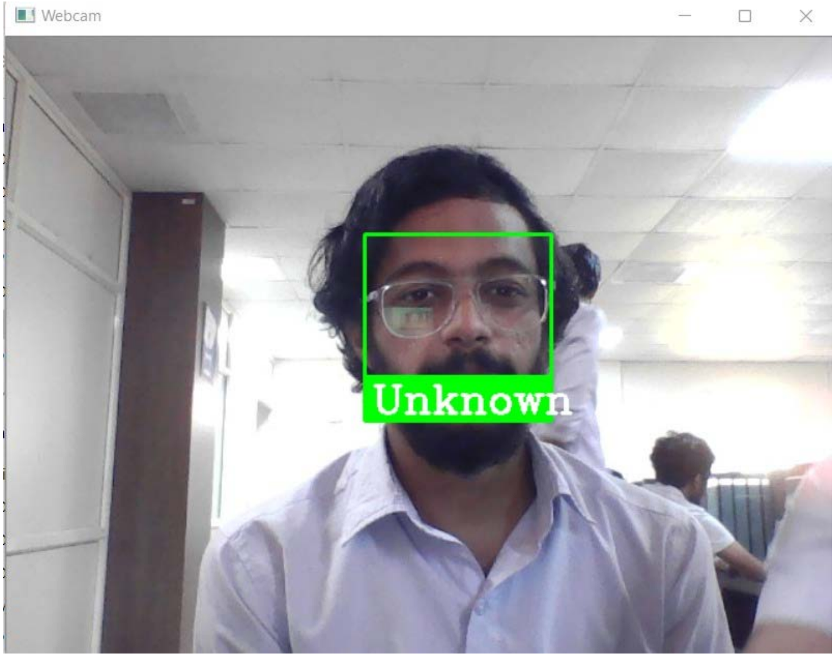


Figure 5.4: The detection of unknown person

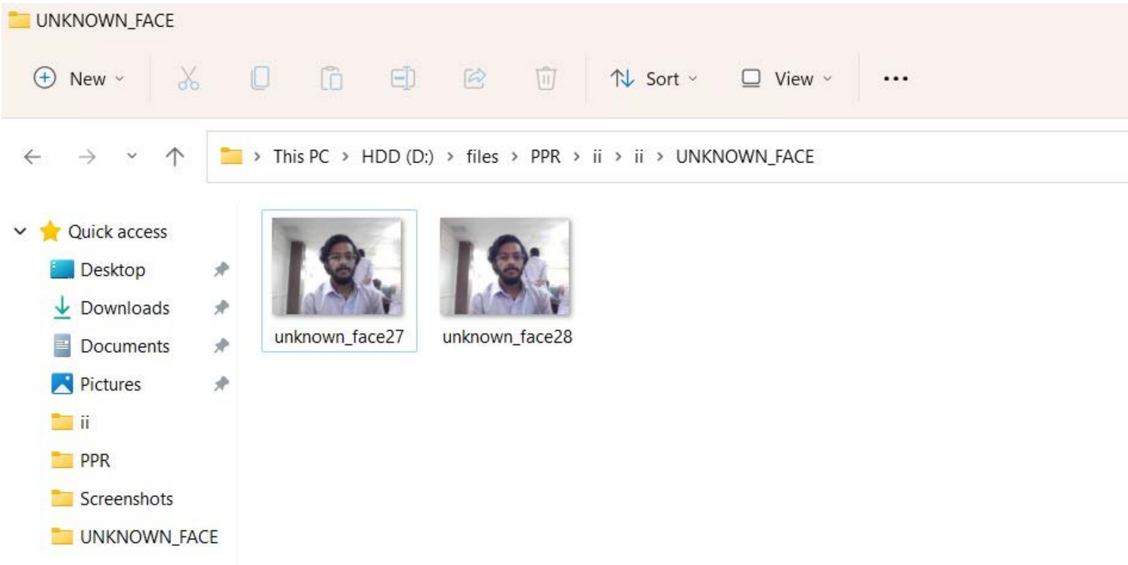
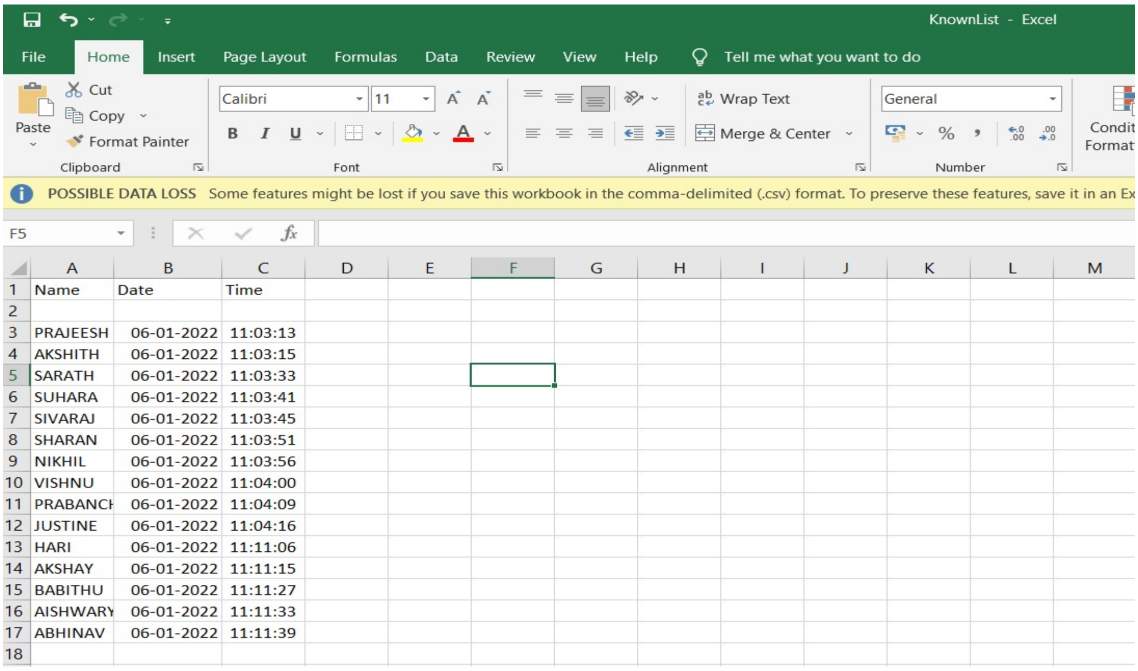


Figure 5.5: Unknown person folder



	Name	Date	Time
1	Name	Date	Time
2			
3	PRAJEESH	06-01-2022	11:03:13
4	AKSHITH	06-01-2022	11:03:15
5	SARATH	06-01-2022	11:03:33
6	SUHARA	06-01-2022	11:03:41
7	SIVARAJ	06-01-2022	11:03:45
8	SHARAN	06-01-2022	11:03:51
9	NIKHIL	06-01-2022	11:03:56
10	VISHNU	06-01-2022	11:04:00
11	PRABANCH	06-01-2022	11:04:09
12	JUSTINE	06-01-2022	11:04:16
13	HARI	06-01-2022	11:11:06
14	AKSHAY	06-01-2022	11:11:15
15	BABITHU	06-01-2022	11:11:27
16	AISHWARY	06-01-2022	11:11:33
17	ABHINAV	06-01-2022	11:11:39
18			

Figure 5.6: Known person log

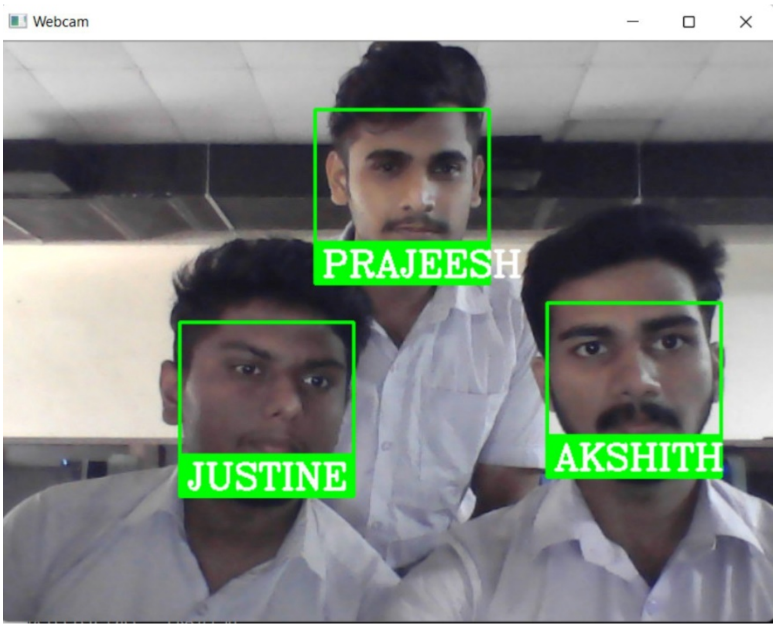


Figure 5.7: The detection of known person

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

The house monitoring system we developed has detected known and unknown people with the help of face recognition module. The face recognition module is a python module which is pre-trained using dlib and can be classified using 2 algorithms: CNN or HOG + Linear SVM model. This helps in detecting the faces more efficiently

This module is simple to use and comparatively good accuracy. So the detection and classification can be done easily and fast for monitoring house. It uses simple techniques to compare faces. It basically calculates the encodings of the trained faces and compares with the encoding of face from live feed. The strictness or accuracy of detection can be adjusted according to the need of the user by changing the tolerance value.

If the known person is detected, then the name of the person is entered into csv file and picture to folder of known people. Whereas, if unknown person is detected then the time is entered in csv file and an alert is sent to the owner through sms and photo of the person through email.

Apart from home surveillance, this project can be used in restricted areas or locations which can improve the security and safety. It can be tested and implemented using CCTV.

CHAPTER 7

REFERENCES

[1] Paul-Marc BOUGHARIOS, Charbel HABIB, Semaan GEORGES and Mustapha HAMAD, “*Managing and Monitoring House Visits with Home Visitor Logger*” in *ECCE Department – Faculty of Engineering, Notre Dame University, P.O. Box: 72, Zouk Mikael*.

[2] Xiujuan Ren, Junhang Ding, Jinna Sun, Qingmei Sui, “*Face Modeling Process Based on Dlib*” in *College of Control Science and Engineering, Shandong University, China, 250100*, 2020,

[3] Kirti Dang, Shanu Sharma, “*Review and Comparison of Face Detection Algorithms*” in 978-1-5090-3519-9/17/$31.00 c 2017 IEE

[4] earlie kit m sanches,noel b linsangen,randy e angelia,"*Three Triangle Method for Face Recognition using Dlib and OpenCV*" in 970-0-7381-1177-3/20/$31.00@2020 IEEE

[5] Suraj Pawar, Vipul Kithani, Sagar Ahuja, Sunita Sahu, “*Smart Home Security using IoT and Face Recognition"*, 978-1-5386-5257-2/18/$31.00 ©2018 IEEE

[6] Dwi Ana Ratna Wati, Dika Abadianto, “*Design of Face Detection and Recognition System for Smart Home Security Application*” in *IEEE*, 978-1-5386-0658-2/17/$31.00 ©2017 IEEE

[7] Seree Khunchai, Chaiyapon Thongchaisuratkrul, “Development of Application and Face Recognition for Smart Home”978-1-7281-7240-8/20/$31.00 ©2020 IEEE

[8] Abdourahman HOUSSEIN AHMED, Kidiyo KPALMA1 , Abdoulkader OSMAN GUEDI, “*Human Detection using HOG-SVM, Mixture of Gaussian and Background Contours Subtraction*” 978-1-5386-4283-2/17 $31.00 © 2017 IEEE DOI 10.1109/SITIS.2017.62

[9] Fen xu,Feng xu,” *Pedestrian Detection Based on Motion Compensation and HOG/SVM Classifier”* 978-0-7695-5011-4/13 $26.00 © 2013 IEEE DOI 10.1109/IHMSC.2013.227

[10] Enjie Jiang,” *A review of the comparative studies on traditional and intelligent face recognition methods”*, June 27,2021 at 18:07:35 UTC from IEEE Xplore

[11] Seree, K. and Chaiyapon, T., “Development of Smart Home System Controlled by Android Application.” The 6th International Conference on Technical Education (ICTE2019), Proceedings of the IEEE , (19 March 2019) : pp. 87-90.

[12] N. King. “Smart home. A definition” Intertek Research and Testing Center, 2003, pp. 1-6.

[13] ITU, “Internet of Things Global StandardsInitiative,”[Online]. Available:http:// /www.itu.int [Accessed 20 10 2019].

[14] Zheng Yuanpan, Li Guangyang, Li Ye. Research Overview of Application of Deep Learning in Image Recognition [J]. Computer Engineering and Application,2019,55(12):20-36

[15] Brunelli R, Poggio T. Face recognition through geometrical features[J]. Lecture Notes in Computer Science, 1995, 588(12):792-800.

[16] Journal. Overview of Face Recognition Technology[J]. Computer Knowledge and Technology, 2019,15(02):171-172.

[17] Oren M, Papageorgion C, Sinba P, et al. Pedestrian detection using wavelet templates. IEEE Proc on Computer Vision and Pattern Recognition, 1997: 193-199.

[18] Heisele B, Woehler C. Motion-based recognition of pedestrian. Patern Recognition, 1998: 1325-1330.

[19] Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors . Proceedings of 8th European Conference on Computer Vision, 2004: 69-82.

[20] C. Stauffer and W. E. L. Grimson, “Adaptive background mixture models for real-time tracking” Proceedings.] S. R. K, T. C. Sarma, and K. S. Prasad, “Face Recognition Office Security System Using LabVIEW 8.6,” Int. J. Electron. Commun. Instrum. Eng. Res. Dev., vol. 3, no. 2, p. 8, 2013 No PR00149), pp. 246-252

[21] Canny, J., “A Computational Approach to Edge Detection”, IEEE Trans. Pattern Analysis and Machine Intelligence, 8(6):679–698, 1986

[22] S. R. K, T. C. Sarma, and K. S. Prasad, “Face Recognition Office Security

System Using LabVIEW 8.6," Int. J. Electron. Commun. Instrum. Eng. Res. Dev., vol. 3, no. 2, p. 8, 2013

[23] P. Rajiv, R. Raj, and M. Chandra, "Email based remote access and surveillance system for smart home infrastructure ☆," Perspect. Sci., pp. 3–5, 2016.

[24] Dwi Ana Ratna, Wati Dika Abadianto, "Design of Face Detection and Recognition System for Smart Home Security Application", 2017 2nd ICITISEE.

[25] Sandesh Kulkarni, Minakshee Bagul, Akansha Dukare, Prof. Archana Gaikwad, "Face Recognition System Using IoT", IJARCET Volume 6, Issue 11, November 2017, ISSN: 2278 – 1323.

[26] Ravi Kishore Kodali, Vishal Jain, Suvadeep Bose and Lakshmi Boppana, "IoT Based Smart Security and Home Automation System", IEEE 2016.

[27] EFRATY B, HuANG C, sHAH s K, et al. Facial landmark detection in uncontrolled conditions[C]. 2011 Intemational Joint Conference on Biometrics(UCB). IEEE, 2011: 1ł8.

[28] REN S, CAO X, WEI Y, et al. Face alignment at 3000 fps via regressing local binary features[C]. Computer Vision and Pattem Recognition(CVPR). IEEE, 2014: 1685-1692.

[29] T. F. Cootes, M. Ionita, C. Lindner, and P. Sauer. Robust and accurate shape model fitting using random forest regression voting. In ECCV, 2012.

[30] Md. Abdur Rahim, Md. Najmul Hossain, Tanzillah Wahid & Md. Shafiul Azam, "Face Recognition using Local Binary Patterns (LBP)", Global Journal of Computer Science and Technology Graphics & Vision Volume 13 Issue 4 Version 1.0 Year 2013.

ANNEXURE

1. LIVE FEED CAPTURING AND COMPARING WITH TRAINED IMAGES

```
imag = cap.read()

imagS = cv2.resize(imag, (0, 0), None, 0.25, 0.25)

imagS = cv2.cvtColor(imagS, cv2.COLOR_BGR2RGB)


CurFrameFace = face_recognition.face_locations(imagS,model="hog")

CurFrameEncode = face_recognition.face_encodings(imagS, CurFrameFace)


for encodeFace, faceLoc in zip(CurFrameEncode, CurFrameFace):

    facematches = face_recognition.compare_faces(encodeListKnown,
    encodeFace,tolerance=0.55)

    DistanceFace = face_recognition.face_distance(encodeListKnown,
    encodeFace)

    print('Detected Face distance: ',DistanceFace)

    IndexMatching = np.argmin(DistanceFace)
```

2. DECIDING KNOWN OR UNKNOWN PERSON:

```
if facematches[IndexMatching]:

    name = classNames[IndexMatching].upper()

    print("Detected Face: ",name)

    y1, x2, y2, x1 = faceLoc

    y1, x2, y2, x1 = y1 * 4, x2 * 4, y2 * 4, x1 * 4

    cv2.rectangle(imag, (x1, y1), (x2, y2), (0, 255, 0), 2)

    cv2.rectangle(imag, (x1, y2 - 35), (x2, y2), (0, 255, 0), cv2.FILLED)
```

```
        cv2.putText(imag, name, (x1 + 6, y2 - 6),
        cv2.FONT_HERSHEY_COMPLEX, 1, (255, 255, 255), 2)

        markKnown(name,increment)

        increment = increment + 1

else:

        name = "Unknown Person"

        y1, x2, y2, x1 = faceLoc

        y1, x2, y2, x1 = y1 * 4, x2 * 4, y2 * 4, x1 * 4

        cv2.rectangle(imag, (x1, y1), (x2, y2), (0, 255, 0), 2)

        cv2.rectangle(imag, (x1, y2 - 35), (x2, y2), (0, 255, 0), cv2.FILLED)

        cv2.putText(imag, name, (x1 + 6, y2 - 6),
        cv2.FONT_HERSHEY_COMPLEX, 1, (255, 255, 255), 2)

        markUnknown(name,increment)

        increment = increment + 1

        cv2.putText(imag, name, (x1 + 6, y2 - 6),
        cv2.FONT_HERSHEY_COMPLEX, 1, (255, 255, 255), 2)

        markKnown(name,increment)

        increment = increment + 1
```

3. KNOWN PERSON LOG ENTRY:

```
def markKnown(name,increment):

    with open('KnownList.csv', 'r+') as f:

    ListData = f.readlines()

    ListName = []

    for line in ListData:
```

```
            entry = line.split(',')
            ListName.append(entry[0])
            if name not in ListName:
                ret,frame = cap.read()
                cv2.imwrite('DETECTED_KNOWN/face' + str(increment) + '.jpg',frame)
                now = datetime.now()
                timedatestring = now.strftime('%H:%M:%S')
                f.writelines(f'\n{name},{timedatestring}')
```

4. UNKNOWN PERSON LOG ENTRY:

```
def markUnknown(name,increment):
    with open('UnknownList.csv', 'r+') as f:
        ListData = f.readlines()
        ListName = []
        for line in ListData:
            entry = line.split(',')
            ListName.append(entry[0])
            if name not in ListName:
                ret,frame = cap.read()
                cv2.imwrite('UNKNOWN_FACE/unknown_face' + str(increment) +
                '.jpg',frame)
                now = datetime.now()
                timedatestring = now.strftime('%H:%M:%S')
                f.writelines(f'\n{name},{timedatestring}')
```

5. ALERT SYSTEM FOR UNKNOWN PEOPLE:

```
import os

from twilio.rest import Client

def sendAlertTwilio(name):

    account_sid = os.environ['xxxxxxxxxxxxxxxxxxx']

    auth_token = os.environ['xxxxxxxxxxxxxxxxxxx']

    target_number = 'xxxxxxxxxx'

    source_number = 'xxxxxxxxxx'

    client = Client(account_sid, auth_token)

  message = client.messages.create(body=name+' has entered frame.', from_=source_number, to=target_number)

    print(message.sid)
```