**Project Report**

***on***

**SYSTEM SECURITY USING FACE DETECTION**

***In partial fulfillment of requirements for the degree***

***of***

**BACHELOR OF TECHNOLOGY**

**IN**

**INFORMATION TECHNOLOGY**

***Submitted by:***

ANCHAL MEHTA (1710DMTIT01478)

AMAN SONI (1710DMTIT01484)

ANJALI MEHTA (1710DMTIT01485)

ASTHA BHARGAVA (1601DMBIT01017)

***Under the guidance of***

PROF.SCHIN YELE

PROF. ARUNA PATIDAR



**SHRI VAISHNAV VIDYAPEETH VISHWAVIDYALAYA, INDORE**

**SHRI VAISHNAV INSTITUTE OF INFORMATIN TECHNOLOGY**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**JULY-DEC 2020**

**SHRI VAISHNAV VIDYAPEETH VISHWAVIDYALAYA, INDORE**

**SHRI VAISHNAV INSTITUTE OF INFORMATION TECHNOLOGY**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**DECLARATION**

We here declare that work which is being presented in the project entitled “**SYSTEM SECURITY USING FACE DETECTION**” in partial fulfillment of degree of **Bachelor of Technology in Information Technology** is an authentic record of our work carried out under the supervision and guidance of **Mr.** **Sachin Yele** Asst. Professor of Information Technology. The matter embodied in this projecthas not been submitted for the award of any other degree.

Aanchal Mehta

Aman Soni

Date: Anjali Mehta

Astha Bhargava

I

**SHRI VAISHNAV VIDYAPEETH VISHWAVIDYALAYA, INDORE**

**SHRI VAISHNAV INSTITUTE OF INFORMATION TECHNOLOGY**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**PROJECT APPROVAL SHEEET**

Following team has done the appropriate work related to the “**System Security Using Face Detection**” in partial fulfillment for the award of **Bachelor of Technology in Information** **Technology** of “SHRI VAISHNAV INSTITUTE OF INFORMATION TECHNOLOGY” andis being submitted to SHRI VAISHNAV VIDYAPEETH VISHWAVIDYALAYA, INDORE.

**Team:**

1. **Aanchal Mehta**
2. **Aman Soni**
3. **Anjali Mehta**
4. **Astha Bhargava**

**Internal Examiner** **External Examiner**

Date

II

**SHRI VAISHNAV VIDYAPEETH VISHWAVIDYALAYA, INDORE**

**SHRI VAISHNAV INSTITUTE OF INFORMATION TECHNOLOGY**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**CERTIFICATE**

This is to certify that **Ms. Aanchal Mehta**, **Mr. Aman Soni, Ms. Anjali Mehta** and **Ms. Astha Bhargava** working in a team have satisfactorily completed the project entitled “**System Security Using Face Detection**” under the guidance of **Prof.. Sachin Yele** and **Prof. Aruna Patidar** in the partial fulfillment of the degree of **Bachelor of Technology in Information Technology** awarded by SHRI VAISHNAV INSTITUTE OF INFORMATION TECHNOLOGY affiliated to SHRI VAISHNAV VIDYAPEETH VISHWAVIDYALAYA, INDORE during the academic year **July** **2020-Dec 2020**.

|  |  |
| --- | --- |
| (Prof. Sachin Yele) |  |
| (Prof. Aruna Patidar) | Prof. Manish Kumar |
| **Project Guide** | **Project Coordinator** |

Dr. Jigyasu Dubey

**Head, Department of Information Technology**

III

**ACKNOWLEDGEMENT**

We are grateful to a number of persons for their advice and support during the time of complete our project work. First and foremost our thanks goes to **Dr. Jigyasu Dubey** Head of the Department of Information Technology and Prof**. Sachin Yele and Prof. Aruna Patidar** the mentor of our project for providing us valuable support and necessary help whenever required and also helping us explore new technologies by the help of their technical expertise. His direction, supervision and constructive criticism were indeed the source of inspiration for us.

We would also like to express our sincere gratitude towards our Director **Dr. Anand Rajavat**  for providing us valuable support.

We are really indebted to **Prof. Manish Kumar,** project coordinator for helping us in each aspect of our academics activities. We also owe our sincere thanks to all the **faculty members** of Information Technology Department who have always been helpful.

We forward our sincere thanks to all **teaching and non-teaching staff** of Information Technology department, SVVV Indore for providing necessary information and there kind co-operation.

We would like to thanks our parents and family members, our classmates and our friends for their motivation and there valuable suggestion during the project. Last, but not the least, we thank all those people, who have helped us directly or indirectly in accomplishing this work. It has been a privilege to study at SHRI VAISHNAV VIDYAPEETH VISHWAVIDYALAYA, INDORE.

IV

**ABSTRACT**

System Security Using Face Detection is designed for motion detection and webcam monitoring. It can be used to enhance the automated security systems. It can be used to detect any fraudulent activity by sensing motion around the areas where surveillance is needed. This project will be designed with the help of image processing and OpenCV. Webcams are video cameras used for the purpose of observing an area. They are often connected to a recording device or IP network, and may be watched by a security guard or law enforcement officer. In case of location we have less percentage of movement (like home courtyard during night); then we need to check the whole recorded video to show where and when that motion occurs which are wasting in time. So, we aim at processing the real time video captured by a Webcam to detect motion in the Scene using image detection and open CV, with keeping in mind that camera still recorded which means real time detection. The results show accuracy and efficiency in detecting motion. In the application, There can be a web camera attached to the computer. These webcams can be used as an eye to find motion. Whenever a movement occurs in front of the Webcam that frame is stored in a specific location. The project also focuses on reducing the storage capacity needed for these types of applications. The main purpose of this project is to design and develop a prototype of a human motion detection system. Here, we present some basic ways to perform a human motion detection algorithm and also a new way to consider for background updating using spatial information instead of temporal. The experiments carried out to evaluate the performance of the prototype system are attempt.

V

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Diagram Number** | **Diagram Name** | **Page Number** |
|  |  |  |
| Fig. 4.1.3 | Use Case | 12 |
| Fig. 4.2.1 | DFD 0 | 13 |
| Fig. 4.2.2 | DFD 1 | 14 |
| Fig. 4.2.3 | DFD 2 | 14 |
| Fig. 5.1 | Class Diagram | 16 |
| Fig. 5.2.1 | Sequence Diagram | 17 |
| Fig. 5.2.2 | Collaboration Diagram | 18 |
| Fig. 5.3 | State Diagram | 19 |
| Fig. 5.4 | Activity Diagram | 20 |
| Fig. 5.5 | Object Diagram | 21 |

VI

TABLE OF CONTENT

|  |  |
| --- | --- |
| Declaration | I |
| Project Approval Sheet | II |
| Certificate | III |
| Acknowledgement | IV |
| Abstract | V |
| List of Figures | VI |
| CHAPTER 1 – **INTRODUCTION** | **1-3** |
| 1.1 Introduction | 1 |
| 1.2 Problem Statement | 2 |
| 1.3 Need for the proper system | 2 |
| 1.4 Objective | 2 |
| 1.5 Module of the system | 3 |
| 1.6 Scope | 3 |
| CHAPTER 2 - **LITERATURE SURVEY** | **4-6** |
| 2.1 Existing System | 4 |
| 2.2 Proposed System | 4 |
| 2.3 Feasibility Study | 4 |
| 2.3.1 Technical Feasibility | 5 |
| 2.3.2 Economical Feasibility | 5 |
| 2.3.3 Operational Feasibility | 6 |
| CHAPTER 3 - **REQUIREMENTS ANALYSIS** | **7-9** |
| 3.1 Method used for Requirement analysis | 7 |
| 3.2 Data Requirements | 7 |
| 3.3 Functional Requirements | 7 |
| 3.4 Non Functional Requirements | 8 |
| 3.5 System Specification | 9 |
| 3.5.1 Hardware specification | 9 |
| 3.5.2 Software Specification | 9 |
| CHAPTER 4 – **DESIGN** | **10-14** |
| 4.1 Software Requirements Specification | 10 |
| 4.1.1 Glossary | 10 |
| 4.1.2 Supplementary | 10 |
| 4.1.3 Use Case Model | 11 |
| 4.2 Data flow Diagram | 11 |
|  |  |

DFD level 0 12

* + 1. DFD level 1 13
    2. DFD level 2 1
  1. CHAPTER 5 - **SYSTEM MODELING 15-33**
  2. 5.1 Detailed Class Diagram 16
  3. 5.2 Interaction Diagram 17
  4. 5.2.1Sequence Diagram 17
  5. 5.2.3Collaboration Diagram 18
  6. 5.3State Diagram 18
  7. 5.4Activity Diagram 19
  8. 5.5Object Diagram 20
  9. Test Plans and Implementation Images 21

CHAPTER 6 - **CONCLUSION & FUTURE WORK 34-35**

* 1. 6.1Limitation of Project 34
  2. 6.2Future Enhancement 35

CHAPTER 7 - **BIBLIOGRAPHY & REFERENCES 36**

* 1. 7.1Reference Books 36
  2. 7.2Other Documentations & Resources 36

**CHAPTER 1 INTRODUCTION**

**1.1 INTRODUCTION**

This project would be focused on the Webcam Facial object Detection module where we would perform research on the techniques and methodology to detect motion and to develop a module for a technique that we prefer to use in this project. This module would record down facial objects and pass it into the next module that would be on object classification where it classifies human objects. Thus, this project is to come up with a solution that detects facial or human motion effectively and record it down with one or more objects that are moving and causing motions in front of the screen to detect. The purpose of this project is to help new security researchers on behalf of Artificial Intelligence, which in this case is the human motion detection system. The question to be addressed here in this module is, given a sequence of images, how do we detect motion or track a moving object? The project is to mainly answer this particular question addressed by providing a prototype to emulate or prove the algorithms or techniques that are available to perform motion detection by an input of images in a number of frames.

This Motion Detection System can be used in surveillance and security systems. The system that this project came up with will be useful for security in a fixed restriction area. Therefore, the background of the targeted area is assumed to be non-moving and considerations of sudden change in lighting are ignored as well. However, the considerations of other factors are taken into consideration. Basically, the initial plan was to use a technique called image segmentation to abstract the foreground image from the source image obtained and later processed to filter out noises or small images disturbance. To perform this, we would use Open Source Computer Vision Libraries from Intel to obtain contours from the foreground image subtracted. We will map these contours’ pixels with the original images’ to send raw data into the other module of the project performed by our partner on classifying the image frame obtained on whether it’s a human motion or not. This module would return a percentage of recognition rates on whether the motion belongs to humans or not. Based on a level of acceptable percentage that it is sure it’s a human motion, the program would detect and display the motion with a

bounding box on the human which is in a different colour to other moving objects that caused motion as well since all moving objects are bounded by the rectangles. The program will record down the scenes when the motion event occurs.

**1.2 PROBLEM STATEMENT**

The Problem domain regarding object detection is for security purposes. In the era of less understanding of Smart Surveillance systems, there are very few choices for security softwares and terms for securing the data with technologies. So, the use of the Object Detection term is very helpful for security purposes using Computer Vision. Basically, it provides the surveillance during contact in front of the screen. So, it will solve the system security problem by detecting the object in front of the Camera and if any specific object detects in the screen then it will make a sound or buzzer to know something is happening in front of the screen.

It will also solve one more problem before the normal cameras record the video and save the video on database and according to the vendor, they can install cameras wherever they like. This system’s code can be used as the driver software which can be directly installed on the camera devices.

**1.3 NEED FOR THE PROPER SYSTEM**

This proper system is needed to help new researchers learn and further research on the human motion detection system. The subject to be addressed here in this module is, given a sequence of images, how do we detect motion or track a moving object? This project is to mainly answer this particular question addressed by providing a prototype to emulate or prove the algorithms or techniques that are available to perform motion detection by an input of images in a number of frames.

**1.4 OBJECTIVES**

As proposed earlier, this project is to be linked with another project to come up with the final system called Human Motion Detection System. This project would be focused on the Video Motion Detection module where we would perform research on the techniques and methodology to detect motion and to develop a module for a technique that we prefer to use in this project. This module would record down motion and pass it into the next module that would be on object classification where it classifies human and non-human objects. Thus, this project is to come up with a solution that detects motion effectively and record it down with one or more objects that

are moving and causing motions. The purpose of this project is to help new researchers learn and further research on their topic of interest, which in this case is the human motion detection system. The question to be addressed here in this module is, given a sequence of images, how do we detect motion or track a moving object? The project is to mainly answer this particular question addressed by providing a prototype to emulate or prove the algorithms or techniques that are available to perform motion detection by an input of images in a number of frames.

**1.5 MODULES OF THE SYSTEM**

There are three main modules in the system:

**Create Dataset** : Using this user can create a dataset This option will appear in the center sidebar of the GUI which is 1st Option.

**Train Dataset:** Using this user can train a dataset. This Option will appear in the 2nd option .

**Face Detection:** Here users can detect and if the user is Unauthentic then a message is sent to Authentic Person.

This module used by three types of user:

* Authentic Person
* Unauthentic Person
* Admin

**1.6 SCOPE**

The WebCam Motion Detection System provides an infinite scope to improvise, because the technology is always going to update. This means that the security system will always need to upscale itself in order to be immune from any illegitimate attempts to gain access to the systems.

**CHAPTER 2**

**LITERATURE SURVEY**

**2.1 EXISTING SYSTEM**

The algorithms that were used in old systems were quite complex and unoptimized, for which the computational power of the systems seemed incapable. They only detect the object/human but no other functionalities are provided by that. The algorithms and procedures used in the existing systems are pretty easy and optimized.

**2.2 PROPOSED SYSTEM**

The purpose of this project is to help new researchers learn and further research on their topic of interest, which in this case is the human motion detection system. The question to be addressed here in this module is, given a sequence of images, how do we detect motion or track a moving object? The project is to mainly answer this particular question addressed by providing a prototype to emulate or prove the algorithms or techniques that are available to perform motion detection by an input of images in a number of frames.

**2.3 FEASIBILITY STUDY**

**Feasibility study** is the practical extent to which a project can be performed successfully. To evaluate feasibility, a feasibility study determines whether the solution considered to accomplish the requirements is practical and workable in the software. Information such as resource availability, cost estimation for software development, benefits of the software to the organization after it is developed and cost to be incurred on its maintenance are considered during the feasibility study.

Study of requirement analysis is done through different feasibility studies. Feasibility study is undertaken whenever there is a possibility of probability of improving the existing system or designing a new system. Feasibility study helps to meet user requirements.

It enables us to determine the potential of an existing system and improve it. It helps to develop a technically and economically feasible system. It helps to know what should be embedded in the system. It also helps to develop a cost-effective system. We can make better utilization of available resources.

The project concept is feasible because of the following:

2.1 Technical Feasibility

2.2 Economical Feasibility

2.3 Operational Feasibility

**2.3.1 TECHNICAL FEASIBILITY**

Technical feasibility assesses the current resources (such as hardware and software) and technology, which are required to accomplish user requirements in the software within the allocated time and budget. For this, the software development team ascertains whether the current resources and technology can be upgraded or added in the software to accomplish specific user requirements. Technical feasibility also performs the following tasks.

* Analyzes the technical skills and capabilities of the software development team members
* Determines whether the relevant technology is stable and established
* Ascertains that the technology chosen for software development has a large number of users so that they can be consulted when problems arise or improvements are required.

We have used python 3.8 as a programming language for our web application. We have used jupyter text editor.Once software is installed there is no need for maintaining the software.

**2.3.2 ECONOMICAL FEASIBILITY**

Economic feasibility measures the required software is capable of generating financial gains for an organization. It involves the cost incurred on the software development team, estimated cost of hardware and software, cost of performing feasibility study, and so on. For this, it is essential to consider expenses made on purchases (such as hardware purchase) and activities required to carry out software development. In addition, it is necessary to consider the benefits that can be achieved by developing the software. Software is said to be economically feasible if it focuses on the issues listed below.

* Cost incurred on software development to produce long-term gains for an organization
* Cost required to conduct full software investigation (such as requirements elicitation and requirements analysis)
* Cost of hardware, software, development team, and training.

As economic feasibility involves the estimated cost of hardware and software, so in our project we don’t need any specific hardware or software ,we just need a good computing system with webcam.For developing this system there are no higher initial costs. Only good understanding of technology and terms.

**2.3.3 OPERATIONAL FEASIBILITY**

Operational feasibility assesses the extent to which the required software performs a series of steps to solve business problems and user requirements. This feasibility is dependent on human resources (software development team) and involves visualizing whether the software will operate after it is developed and be operative once it is installed. Operational feasibility also performs the following tasks.

* Determines whether the problems anticipated in user requirements are of high priority
* Determines whether the solution suggested by the software development team is acceptable
* Analyzes whether users will adapt to a new software
* Determines whether the organization is satisfied by the alternative solutions proposed by the software development team.

Our software will operate after it is developed and be operative once it is installed. This system uses threshold methods for detecting the object on the video or to moving object. It is surely satisfies the moving object detection requirements by surveillance buzzer sound. It makes objects detected.

**CHAPTER 3**

**REQUIREMENTS ANALYSIS**

**3.1 METHOD USED OF REQUIREMENTS ANALYSIS**

**• Eliciting requirements**

The process of gathering requirements by communicating with the customers is known as eliciting requirements.

• **Analyzing requirements**

This step helps to determine the quality of the requirements. It involves identifying whether the requirements are unclear, incomplete, ambiguous, and contradictory. These issues were resolved before moving to the next step.

• **Requirements modeling**

In Requirements modeling, the requirements are usually documented in different formats such as use cases, user stories, natural-language documents, or process specification.

• **Review and retrospective**

This step is conducted to reflect on the previous iterations of requirements gathering in a

bid to make improvements in the process going forward.

**3.2 DATA REQUIREMENTS**

When the client approaches the organization for getting the desired product developed, it comes up with rough ideas about what all functions the software must perform and which all features are expected from the software. Referencing this information, the analysts do a detailed study about whether the desired system and its functionality are feasible to develop. This feasibility study is focused towards the goal of the organization. This study analyses whether the software product can be practically materialized in terms of implementation, contribution of project to organization, cost constraints and as per values and objectives of the organization. It explores technical aspects of the project and product such as usability, maintainability, productivity and integration ability.

**3.3 FUNCTIONAL REQUIREMENTS**

Functional requirements are statements of services the system should provide, how the system should react to particular inputs and how the system should behave in particular situations.

The Functional Requirements of the project are as follows :

1. The system is designed and programmed in such a way that whenever there is an unauthorized attempt to gain access to the system, the system will lock by itself and the Administrator will get notified via text message about the malicious and fraudulent attempt to enter into the system.
2. There will be no such requirement as to start/run/initialize the software, the software will start implicitly during the boot process. This functionality will help the Administrator in a way that they will not have to worry about the initialization of the software

**3.4 NON-FUNCTIONAL REQUIREMENTS**

Non functional requirements define system properties and constraints that arise through user needs,because of budget constraints or organizational policies, or due to the external factors such as safety regulations, privacy registration and so on.

Non functional requirements are:

• Reliability

• Maintainability

• Efficiency

• Extensibility

• Accuracy

These are basically the quality constraints that the system must satisfy according to the project contract. So, these are the :-

* The system provides portability to install the system in any desired pc to provide its functionalities.
* This is a security proposed system to authenticate unknown one.
* The maintainability of the system is the just install and run and some advancements are to be done in future updates.
* The performance of the system depends on the webcam, what moving object is on the screen.
* The processing of the authentication takes some time to analyze the object.

**3.5 SYSTEM SPECIFICATIONS**

**3.5.1 HARDWARE REQUIREMENTS**

* WebCam
* Monitor
* USB adapters
* 4 GB RAM
* External camera

**3.5.2 SOFTWARE REQUIREMENTS**

* Python3
* Open CV
* Pandas
* Numpy
* Jupyter notebook
* classifier

**CHAPTER 4 DESIGN**

**4.1 SOFTWARE REQUIREMENTS SPECIFICATIONS**

This is a document that captures a complete description about how the system is expected to perform. It is usually signed off at the end of requirements engineeringphase. A Software Requirements Specification - a requirements specification for a software system - is a complete description of the behaviour of a system to be developed. It includes a set of use cases that describe all the interactions the users will have with the software. Use cases are also known as functional requirements. In addition to use cases, the SRS also contains non-functional (or supplementary) requirements. Non-functional requirements are requirements which impose constraints on the design or implementation such as performance engineering requirements, quality standards, or design constraints

**4.1.1 GLOSSARY**

* **OpenCV** : OpenCV is an open-source library for computer vision. It provides the facility to the machine to recognize the faces or objects. In this tutorial we will learn the concept of OpenCV using the Python programming language.

# **Jupyter Notebook** : Jupyter Notebook is an open-source, web-based interactive environment, which allows you to create and share documents that contain live code, mathematical equations, graphics, maps, plots, visualizations, and narrative text.

**4.1.2 SUPPLEMENTARY SPECIFICATIONS**

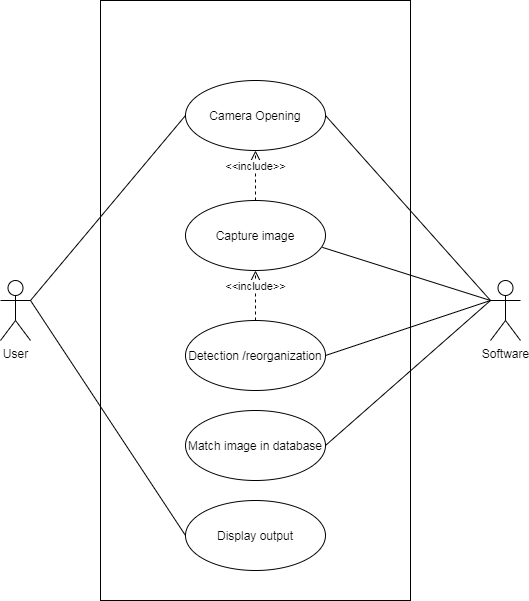
Supplementary specifications define the Requirement that are not easily defined in the use case Model. Requirements such as legal standards, quality aspects, supportability and execution criteria of the system. The supplementary specifications cover all the non-functional requirements of the system. The scope of the supplementary specifications is limited to all the non-functional requirements. The Supplementary Specifications and the use-case model together capture a complete set of requirements on the system.

Following points should be well considered:

* Documents and reports that must be provided by the new system: there can also be few reports, which can help management in decision-making and cost controlling
* Details of the information needed for each document and paper
* The required frequency and distribution for each document
* Portable source of information for each document and report
* With the help of a computerized system, the task of keeping records in an organized manner will be solved . The greatest of all is the retrieval of information, which will be at the click of the mouse.
* So, the proposed system helps in saving time in different operations and making information flow easy giving value

**4.1.3 USE CASE MODEL**

Use case models are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. Hence, when a system is analysed to gather its functionalities, use cases are prepared and actors are identified.

****

*Fig 4.1.3 Use Case Diagram*

**4.2 DATA FLOW DIAGRAM**

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both.

It shows how data enters and leaves the system, what changes the information, and where data is stored.

The objective of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts as a starting point for redesigning a system. The DFD is also called as a data flow graph or bubble chart.

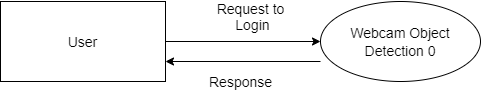
A **data flow diagram** (**DFD**) is a graphical representation of the "**flow**" of **data** through an information system, modelling its process aspects. A**DFD** is often used as a preliminary step to create an overview of the system, which can later be elaborated. In computers, the path of **data** from source document to **data** entry to processing to final reports. **Data** changes format and sequence (within a file) as it moves from program to program.

Standard symbols for DFDs are derived from the electric circuit diagram analysis are:

* **Circle:** A circle (bubble) shows a process that transforms data inputs into data outputs
* **Data Flow:** A curved line shows the flow of data into or out of a process or data store.
* **Data Store:** A set of parallel lines shows a place for the collection of data items. A data store indicates that the data is stored which can be used at a later stage or by the other processes in a different order. The data store can have an element or group of elements.
* **Source or Sink:** Source or Sink is an external entity and acts as a source of system inputs or sink of system outputs.

**4.2.1 DFD LEVEL 0**

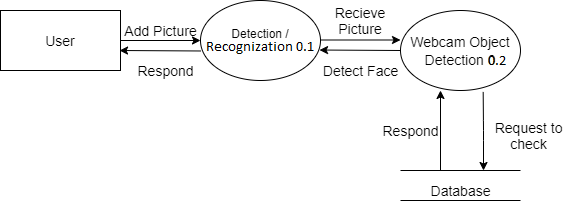
It is also known as a fundamental system model, or context diagram represents the entire software requirement as a single bubble with input and output data denoted by incoming and outgoing arrows. Then the system is decomposed and described as a DFD with multiple bubbles. Parts of the system represented by each of these bubbles are then decomposed and documented as more and more detailed DFDs. This process may be repeated at as many levels as necessary until the program at hand is well understood.

****

*Fig 4.2.1 DFD LEVEL 0*

**4.2.2 DFD LEVEL 1**

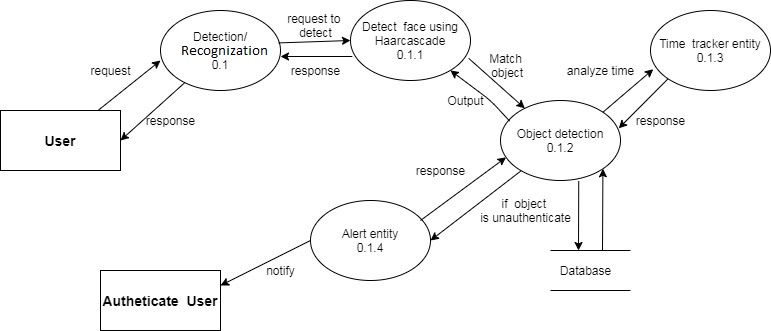
In 1-level DFD, a context diagram is decomposed into multiple bubbles/processes. In this level, we highlight the main objectives of the system and breakdown the high-level process of 0-level DFD into sub processes**.** DFD Level 1 provides a more detailed breakout of pieces of the Context Level Diagram.

****

*Fig 4.2.2 DFD LEVEL 1*

**4.4.3 DFD LEVEL 2**

DFD Level 2 then goes one step deeper into parts of Level 1. It may require more text to reach the necessary level of detail about the system’s functioning A level 2 data flow diagram (DFD) offers a more detailed look at the processes that make up an information system than a level 1 DFD does. It can be used to plan or record the specific makeup of a system.

****

*Fig 4.2.3 DFD LEVEL 2*

**CHAPTER 5 SYSTEM MODELING**

System modelling is the process of developing abstract models of a system, with each model presenting a different view or perspective of that system. It is about representing a system using some kind of graphical notation, which is now almost always based on notations in the Unified Modelling Language (UML). Models help the analyst to understand the functionality of the system; they are used to communicate with customers.

**Models can explain the system from different perspectives:**

An external perspective, where you model the context or environment of the system.

An interaction perspective, where you model the interactions between a system and its environment, or between the components of a system.

A structural perspective, where you model the organization of a system or the structure of the data that is processed by the system.

A behavioural perspective, where you model the dynamic behaviour of the system and how it responds to events.

**Five types of UML diagrams that are the most useful for system modelling:**

**Activity diagrams**, which show the activities involved in a process or in data processing.

**Use case diagrams**, which show the interactions between a system and its environment.

**Sequence diagrams**, which show interactions between actors and the system and between system components.

**Class diagrams**, which show the object classes in the system and the associations between these classes.

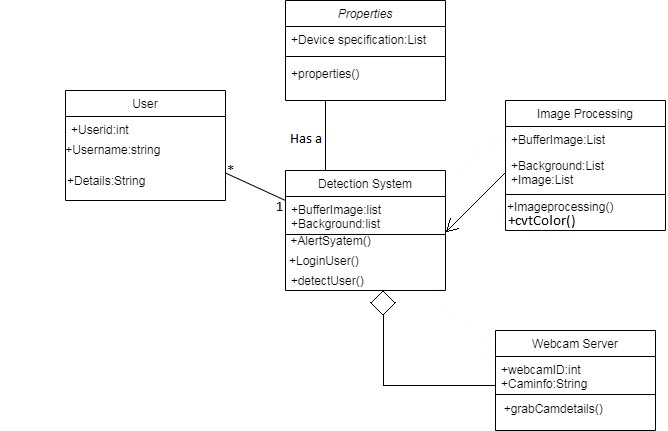
**State diagrams**, which show how the system reacts to internal and external events.

Models of both new and existing system are used during requirements engineering. Models of the existing systems help clarify what the existing system does and can be used as a basis for discussing its strengths and weaknesses. These then lead to requirements for the new system. Models of the new system are used during requirements engineering to help explain the proposed requirements to other system stakeholders. Engineers use these models to discuss design proposals and to document the system for implementation.

**5.1 DETAILED CLASS DIAGRAM**

In software engineering, a class diagram in the Unified Modelling Language is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations, and the relationships among objects.

The purpose of class diagrams is to model the static view of an application. Class diagrams are the only diagrams which can be directly mapped with object-oriented languages and thus widely used at the time of construction.

****

*Fig 5.1 Class Diagram*

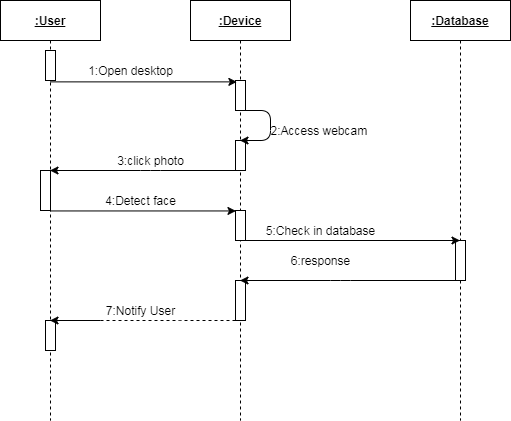
**5.2 INTERACTION DIAGRAM**

Interaction diagrams are used in UML to establish communication between objects. It does not manipulate the data associated with the particular communication path. Interaction diagrams mostly focus on message passing and how these messages make up one functionality of a system. Interaction diagrams are designed to display how the objects will realize the particular requirements of a system. The critical component in an interaction diagram is lifeline and messages.

There are two types of interaction diagrams -- **sequence diagrams** and **collaboration diagrams**. Each diagram is a graphical view of the scenario.

**5.2.1 SEQUENCE DIAGRAM**

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.

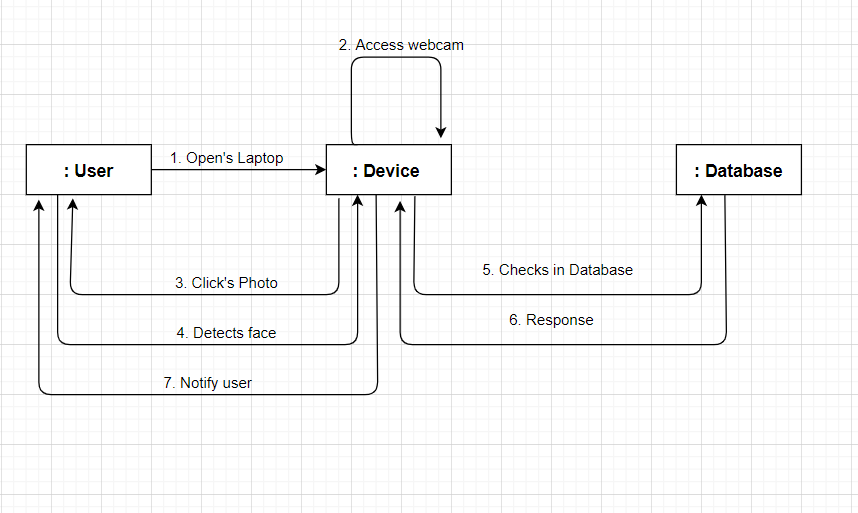
****

*Fig 5.2.1 Sequence Diagram*

**5.2.2 COLLABORATION DIAGRAM**

Collaboration Diagram represents the interaction of the objects to perform the behaviour of a particular use case or a part of use case. The designers use the Collaboration Diagrams to define and clarify the roles of the objects that perform a particular flow of events.

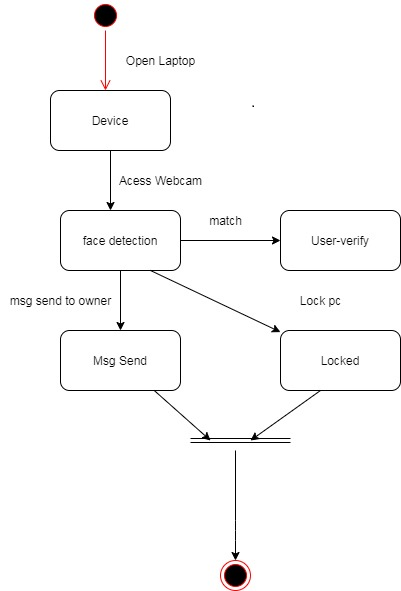
Collaboration Diagram depicts the relationships and interactions among software objects. They are used to understand the object architecture within a system rather than the flow of a message as in a sequence diagram. They are also known as “Communication Diagrams.”

****

*Fig 5.2.2 Collaboration Diagram*

**5.3 STATE DIAGRAM**

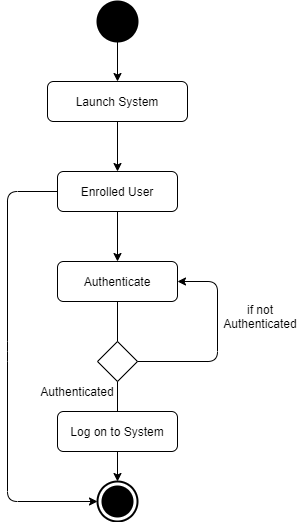
A state diagram is used to represent the condition of the system or part of the system at finite instances of time. It’s a behavioural diagram and it represents the behaviour using finite state transitions. State diagrams are also referred to as State machines and State-chart Diagrams. These terms are often used interchangeably. So simply, a state diagram is used to model the dynamic behaviour of a class in response to time and changing external stimuli.



*Fig 5.3 State Diagram*

**5.4 ACTIVITY DIAGRAM**

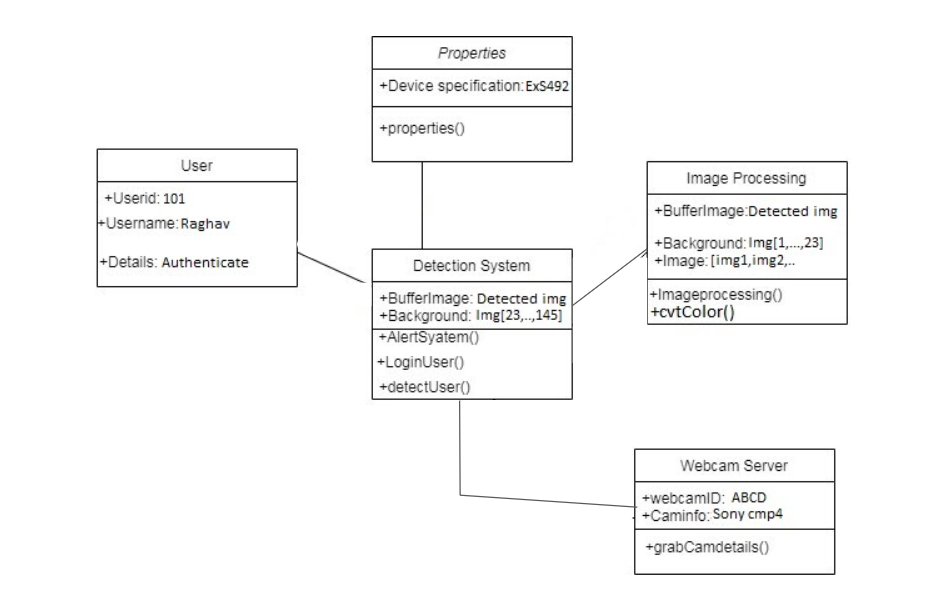
An activity diagram is a behavioural diagram i.e. it depicts the behaviour of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed. We can depict both sequential processing and concurrent processing of activities using an activity diagram. They are used in business and process modelling where their primary use is to depict the dynamic aspects of a system.

****

*Fig 5.4 Activity Diagram*

**5.5 OBJECT DIAGRAM**

Object is an instance of a particular moment in runtime, including objects and data values. A static UML object diagram is an instance of a class diagram; it shows a snapshot of the detailed state of a system at a point in time, thus an object diagram encompasses objects and their relationships at a point in time. It may be considered a special case of a class diagram or a communication diagram.

****

*Fig 5.5 Object Diagram*

**5.6 TEST PLANS AND IMPLEMENTATION IMAGES**

**Test Plans:**

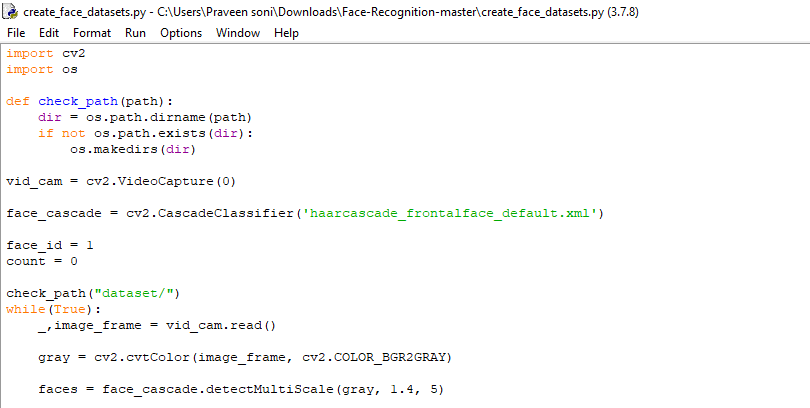
|  |  |  |
| --- | --- | --- |
| **S. No** | **Parameter** | **Description** |
| 1. | Introduction | This moving objects system is basically used for security purposes. It detects the object in the field of view of the webcam through applying an object detection algorithm. |
| 2. | Feature to be tested | The feature of a real time object detected needs to be tested. |
| 3. | Test schedule | It includes some varieties of phases for ex. requirement understanding, test plan creation, test case, test execution, in different environments.   * Firstly, team understands the requirements for implementation of the project * Then, create the schedule for every phase or functionality. * After the test case system will check on every platform or device for the environment testing. * If all the testing will complete successfully then ready for run. |
| 4. | Test type/ Category | In this team will test the system on the basis of test type which are feature testing, branch and data flow testing. |
| 5. | Environmental needs | We need some environmental requirements such as hardware, software, so the system should have at least 4 gb ram, 500 gb harddisk, windows 7,8,10. |
| 6. | Open risk / issue | In implementation, we are suffering from driver errors and also have some bugs during testing. |
| 7. | Exit / Criteria | When any user authenticates successfully it will exit the system and login to screen. If without any error. |

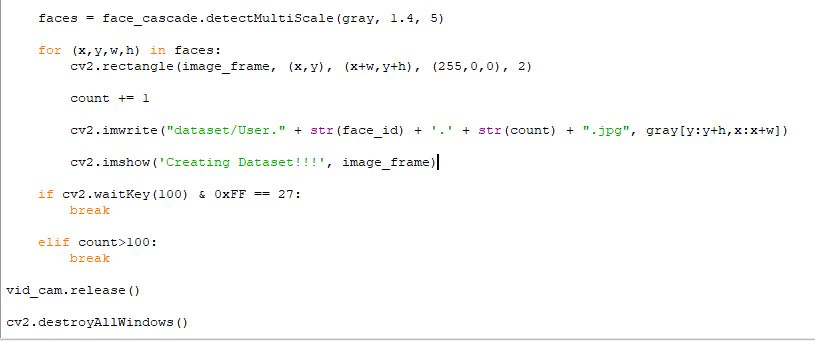
**Test Cases**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test case title** | **Test case**  **description** | **Test steps** | **precondition** | **Test data** | **Expected Results** | **Actual result** | **Status** |
| Login Functionality | Verify login functionality with authenticated users. | Navigate to system  (Allow system to click picture) | Valid user | Webcam clicks the image | Able to access the system/ screen | User logged in successfully | Pass |
| Login Functionality | Verify login functionality with unauthenticated users. | Navigate to system  (Allow system to click picture) | Invalid user | Webcam clicks the image | Unable to access the system/ screen | User failed to login | Fail |

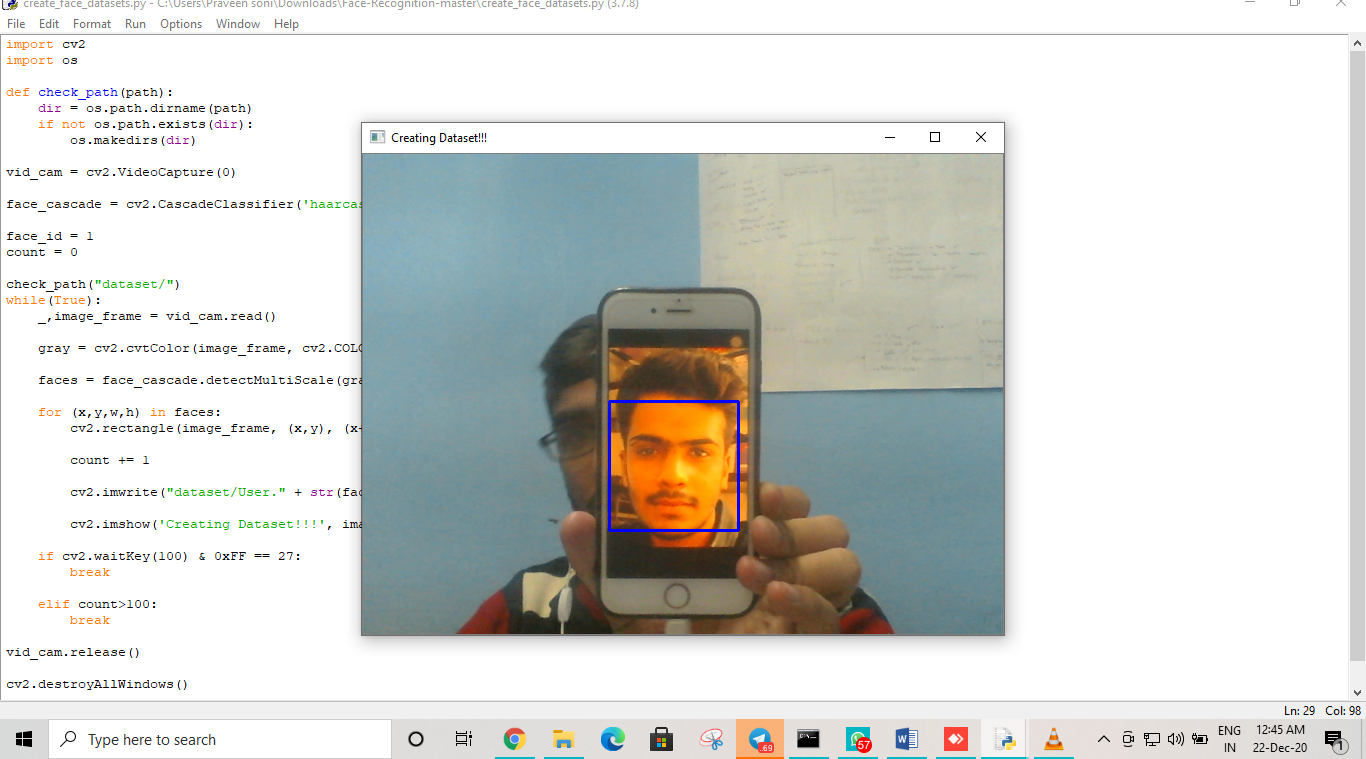
**Implementation Images :**

Step 1 : First part is for creating face dataset for matching during runtime. the first user need to enroll his/her face in the system for further processing.

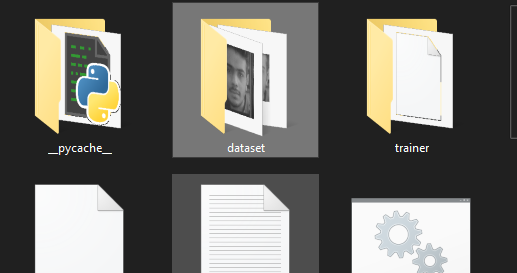
****

****

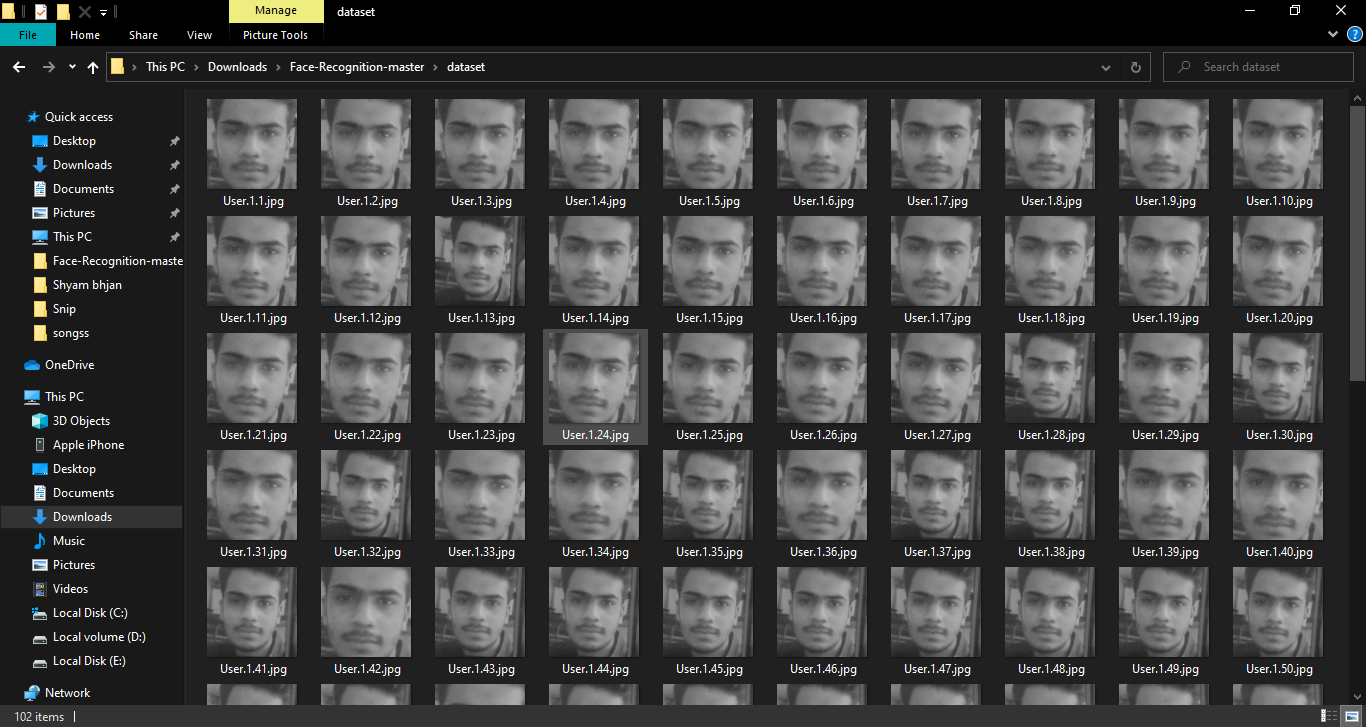
Output of this file is :

****

All the 100 faces are stored in the folder called ‘Dataset’ :

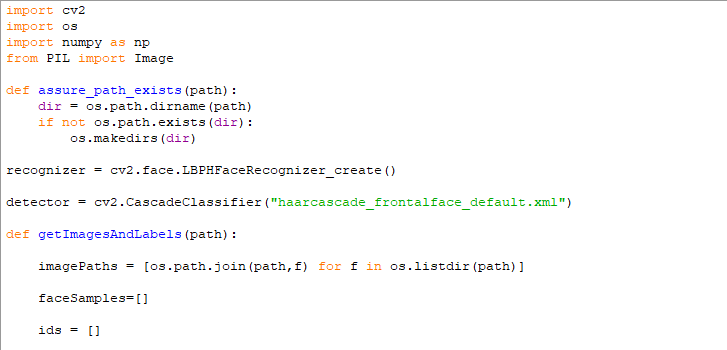
****

**In this File has 100 Face images for further processing**

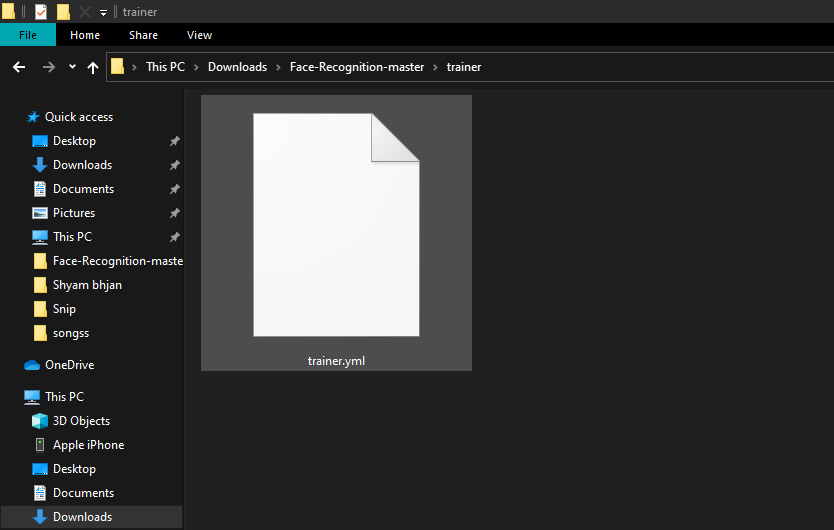
****

Step 2 : After creating the image dataset in the dataset folder. We need to train one model which indicates all the image files which are in the dataset folder. The classifiers make the (.yml) file which has the ability to match the real time image during authentication.

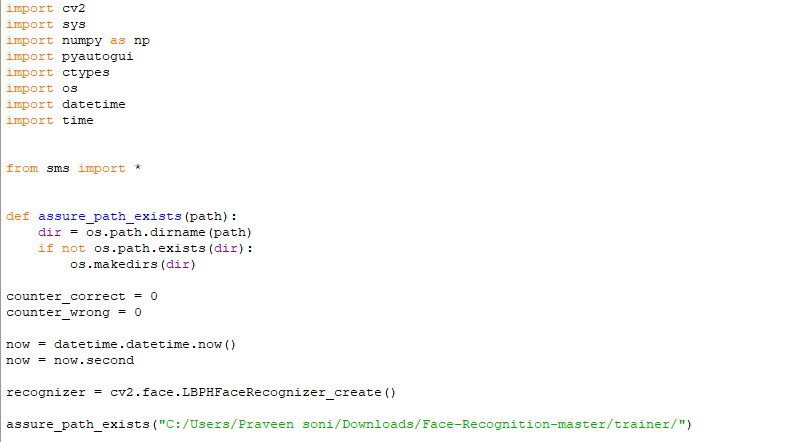
The (.yml) file will save in the ‘trainer’ folder.

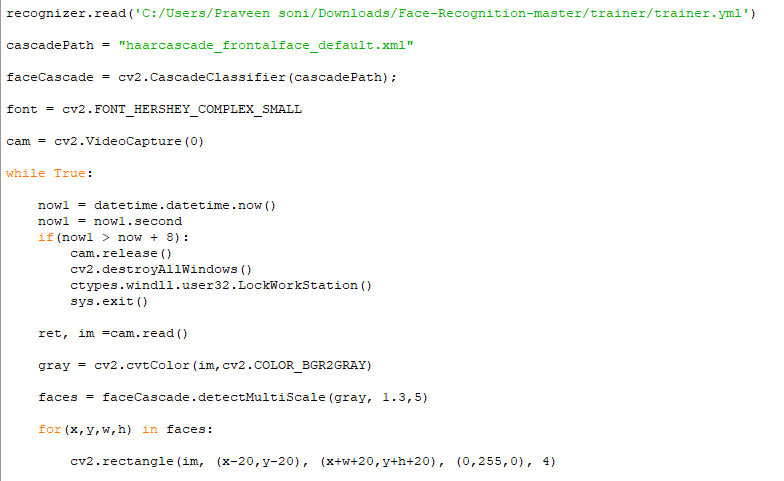
****

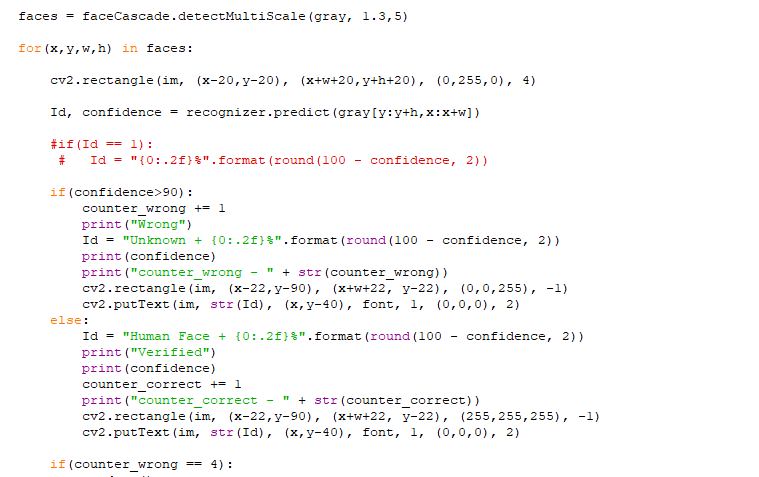
****

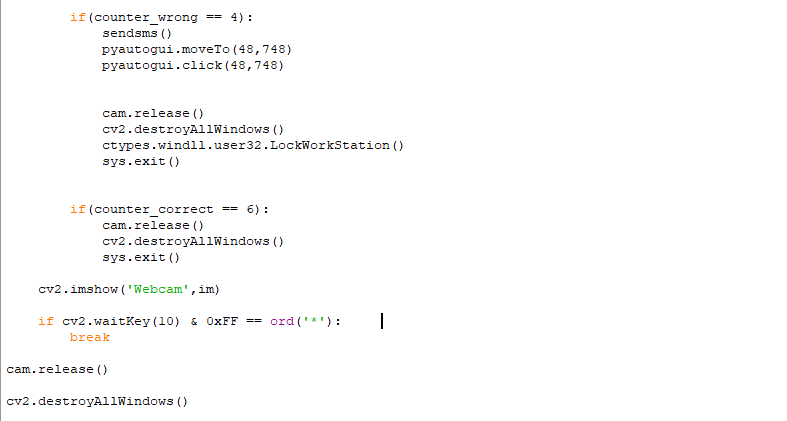
****

Step 3 : In this we check our code for the final run which opens the camera and tries to check if the user is authenticated or not. If not it will lock the workstation.

****

****

****

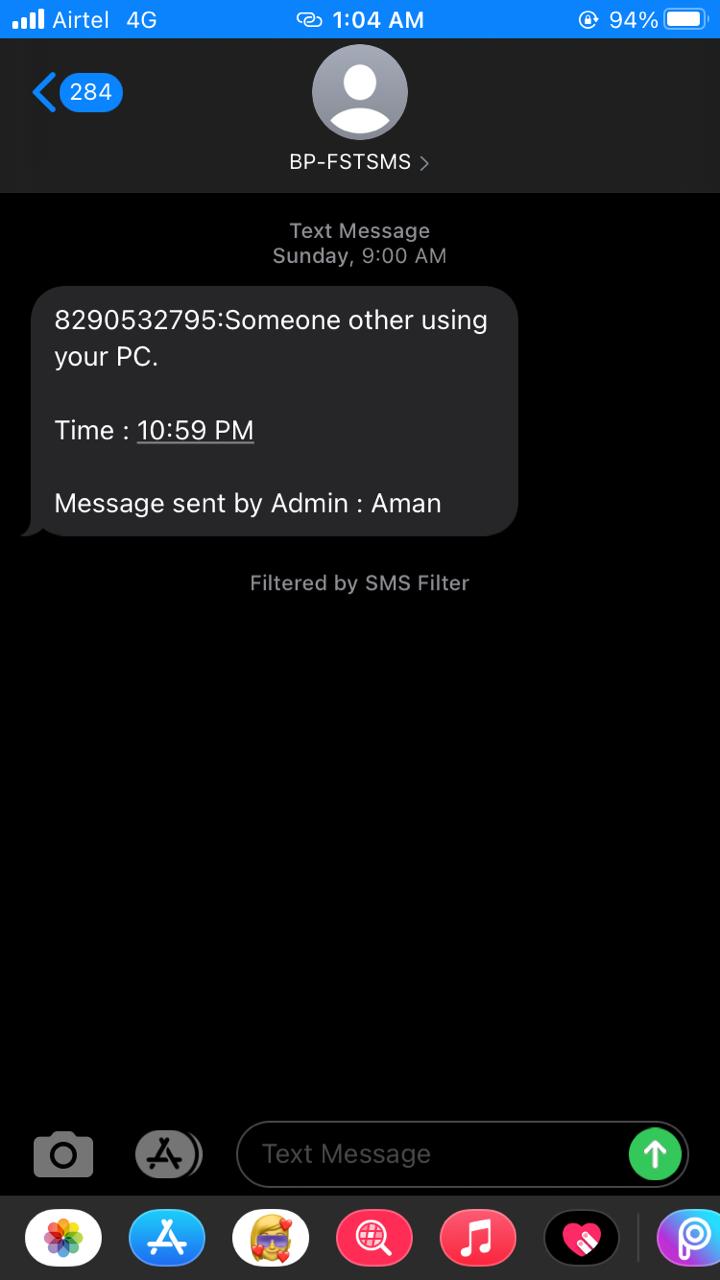
****

SMS integration code using FAST2SMS API :

Send sms code in the sms.py file which has a separate function to send a message to the added number. The function can be accessed in any other file by importing.

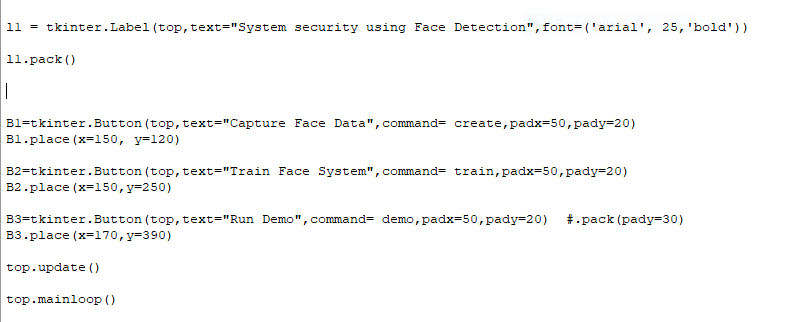
****

Text Message on registered mobile number :

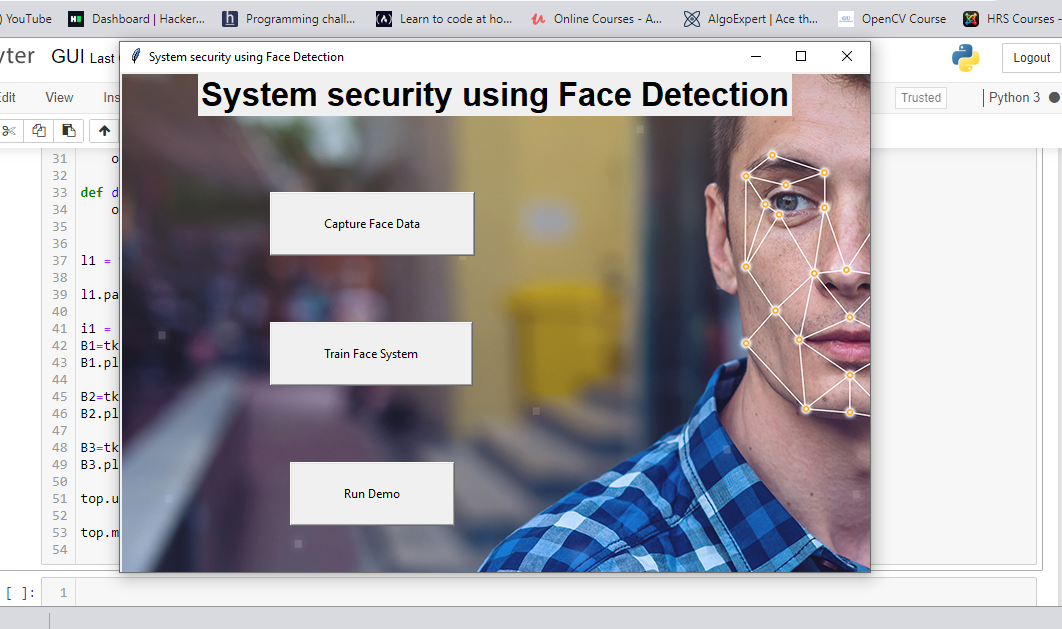
****

GUI(Graphical User Interface) to run these major files :





Output of Tkinter :



**CHAPTER 6**

**CONCLUSION AND FUTURE WORK**

**6.1 LIMITATIONS OF PROJECT**

First of all we have difficulty with Face recognition System:

* Identify similar faces (inter-class similarity)
* Accommodate intra-class variability due to:
* head pose
* illumination condition
* Expressions
* facial accessory
* aging effects
* Cartoon faces

As a conclusion, both algorithms have their pros and cons. The final decision on which algorithm is to be chosen to be implemented in the application software system depends on the requirements of the application system. If it is for example a security surveillance system or such, the presence of a human there is much more important compared to the detection of motion or movements. If this is the case, then the latter human motion detection prototype system’s algorithm developed is much more suitable to be implemented.

The drawbacks of algorithm1, spatial update which u

ses spatial data to update background is as follows:

* Small movements or changes in background such as a jerking screen is not sufficient to update the background however is causing a lot of noise to be detected and spoiling the recognition performance. In this case, many false alarms of rectangles are being drawn.
* When a non-perfect subtraction result is obtained, the algorithm of binary image processing to enhance the image will result in a worse representation of a human shape.
* The aspect of the percentage value to be set for the background update is difficult since the ratio of having high updates rates and low updates rates can be both causing problem to the algorithm. When high update rates, the human will have a higher possibility of being in the background image thus giving an extra 44 humans movement since he had left his initial position. On the contrary, having low update rates gives many noises to the result caused by small changes in the real-world background scene.
* It requires very high illumination to be able to detect movements in the subtraction stage since information is being threshold by the Otsu function.

Now we discussed about the drawbacks of algorithm 2, temporal update + edge detection where the background is updated periodically,

* A non moving human is not being detected.
* Only portions of moving body parts are returned. (partially solved by using bounding rectangle calculations)
* Fails to get a perfect shape of the object since the subtraction results gives both background and current frame’s difference pixels resulting in an object with shadows. The distance of the object and its shadow cast by the background is very much dependent on the speed of the computer or processor implementing the system and also the speed of the camera’s capture measured in frames per second.
* The region may return human shapes being focused in the left side, centre or right side simply because the obtained result of the region is through different processes from the result of obtaining the bounding region.

**6.2 FUTURE ENHANCEMENT**

The technology is expected to grow and will create massive revenues in the coming years. Surveillance and security are the major industries that will be intensely influenced by technology. Schools and universities and even healthcare are also planning to implement the facial recognition technology on their premises for better management. Complicated technology used in facial technology is also making its way to the robotics industry

**CHAPTER 7**

**BIBLIOGRAPHY AND REFERENCES**

**7.1 REFERENCE WEBSITES**

* International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 10, October 2015
* G. Bradski and, A. Kaehler, “Learning OpenCV”, OReilly Publications, 2008
* P. Viola, M. Jones, “Robust Real-Time Face Detection”, International Journal of Computer Vision 57(2), 137154, 2004
* Geeksforgeeks:https://www.geeksforgeeks.org/detect-an-object-with-opencv-python

**7.2 OTHER DOCUMENTATIONS AND RESOURCES**

Open CV: <https://opencv.org/>

Tkinter : <https://docs.python.org/3/library/tkinter.html>

Fast2sms (SMS) : <https://docs.python.org/3/library/tkinter.html>

Youtube (Codemy.com) : <https://www.youtube.com/watch?v=75jbNpc8vN4&t=298s>