PROJECT ID: 2 **PANEL ROOM: ELB 013**

JOMO KENYATTA UNIVERSITY OF AGRICULTURE

AND TECHNOLOGY

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

BSc Electronic and Computer Engineering

PROJECT REPORT

PROJECT TITLE:

SMART GATE CONTROLLER

Submitted by:

JEFF TARIMO KINYANJUI - EN272-3960/2015

PROJECT SUPERVISOR

MR. MATHENGE

A Final Year Project Report submitted to the Department of Electrical and Electronic Engineering in partial fulfillment of the requirements for the award of a Bachelor of Science Degree in Electronic and computer Engineering.

NOV 2021

DECLARATION

This project report is my original work, except where due acknowledgement is made in the text, and to the best of our knowledge has not been previously submitted to Jomo Kenyatta University of Agriculture and Technology or any other institution for the Award of a degree or diploma.

SIGNATURE..... DATE .....

NAME: REGISTRATION NUMBER:

JEFF TARIMO EN272-3960/2015

SUPERVISOR CONFIRMATION

This project proposal has been submitted to the Department of Electrical and

Electronic Engineering, Jomo Kenyatta University of Agriculture and Technology, with my approval as the University supervisor:

SIGNATURE..... DATE.....

NAME: MR. MATHENGE

ABSTRACT

Security and ease of access have been viewed as entities that cannot co-exist with many believing one has to be sacrificed for the other. This project aims to provide both functionalities allowing institutions, companies, schools and organizations to provide security features into their facilities through facial recognition and a smart gate controller.

The project seeks to implement two of the previously stated entities which are security and ease of access into the aforementioned facilities. The circuitry aims to recognize native individuals at the entrances using facial and number plate recognition while also providing ways to facilitate the entry of guests via id card scanning. It also allows tracking of individuals within the facility leading to better accountability of individuals preventing incidents such as theft. The project aims to provide easy access to facilities and reduce time wasted during security checks. It also digitizes the whole process of checking and confirming of credentials and giving access to individuals smooth-lining the whole process.

This involves the use of several cameras to provide facial recognition for individuals and number plate recognition for cars at the gate as well as id scanning for guests. A database will be incorporated to handle data checking and link individuals to their vehicles as well as the time stamp. More cameras would also be required for tracking individuals within the facility. A microcontroller will handle data processing of the entire process.

The solution is implemented by the help of several cameras, a screen(optional) and a microcontroller along with the embedded code for the process.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OFACRONYMS AND ABBREVIATIONS

AI – Artificial Intelligence.

CCTV – Closed Circuit Television.

CSI – Camera Serial Interface.

GPIO – General Purpose Input Output.

GUI – Graphical User Interface.

HDMI – High Definition Multimedia Interface.

ICAO – International Civil Aviation Organization.

LiDAR – Light Detection and Ranging.

ML – Machine Language.

MRZ – Marked Readable Zones.

OCR – Optical Character Recognition.

POE – Power Over Ethernet.

USB – Universal Serial Bus.

Chapter 1 : INTRODUCTION

1.1 Background Information

The Cambridge Dictionary defines security as protection of a person, building, organization, or country against threats such as crime or attacks by foreign countries. In the context of this project we look at security at the institution level. Identification allows systems and/or people to differentiate between people whose information is previously unknown and people whose information is known. Identification can be achieved in several ways such as:

1. Biometric data, that is, fingerprint scan or retina scan
2. Documented data, that is, identity cards, passports and birth certificates
3. Facial recognition

The latter will be heavily involved to apply the identification process by the use of computer vision. Computer vision, as per Ben Dickson, is the field of computer science that focuses on replicating parts of the complexity of the human vision system and enabling computers to identify and process objects in images and videos in the same way that humans do [1]. This functionality helps the computer understand not only different types of objects but also similar objects with different characteristics. Therefore, with this in mind, computer vision plays a great role in identifying facial profiles hence differentiating between individuals.

Facial recognition is the process of identifying or verifying the identity of a person using their face. It captures, analyzes, and compares patterns based on the person's facial details.

1. The **face detection** process is an essential step in detecting and locating human faces in images and videos. With some additional code manipulation in conjunction with computer vision we can draw a rectangle at the face borders to identify the face in real time.
2. The **face capture** process transforms analog information (a face) into a set of digital information (data or vectors) based on the person's facial features. This involves converting the image pixels into a 4 value array representing equivalent Red, Green and Blue (RGB channel) integer values. This conversion allows the processor to compute the images before comparison is done.

3. The **face match** process verifies if two faces belong to the same person. This is done with the help of an underlying database of images to compare to.

In addition to facial recognition, documented data plays a great role in security especially if a comprehensive database is not available. Data can be scanned from either identity cards or passports. With the aid of the codes at the bottom of identity cards and passports we can decipher an individual's records; this code is known as machine readable zones. These zones contain details of the individual and are usually separated by double arrow characters (<<). These zones can be captured and converted to readable text using a method called Object Character Retention. This technology aids in converting images to strings hence allowing us to obtain any relevant information from the marked readable zones.

Furthermore, Object Character Retention will also be heavily involved in number plate recognition allowing us to link a specific vehicle to a specific individual regardless whether they are a guest or a native of the facility. Using an identical method to the scanning of identity cards, the difference between the two lies in the marked readable zones. These zones are not within the number plate however, with simple code manipulations, we can achieve this easily then perform the image to string conversion. This further enhances security especially for members accessing the facility.

1.2 Problem Statement

Institutions, organizations, companies and residential areas are currently devoid of proper systems that are able to account for guests and members who seek to access their services. As it stands, these facilities rely on physical records to log in visitors with an emphasis of book keeping and more often than not rely on the trust of their employees or residents to maintain some form of responsibility when it comes to security. This approach has clear and concise limitations making the system not only ineffective but also outdated. The following are challenges associated with the status quo of this very system:

1. **Time Management:** The book-keeping system presents a challenge in the use of time resources. A high influx of people accessing the facility simultaneous may experience

queues during the book-keeping process and may even lead to frustrations regarding to how the facility is being run.

2. **Human Error:** Book-keepers may end up making mistakes during the logging process compromising the validity of the data. Errors may also arise during interpretation since the legibility of the data is purely dependent on the book-keepers who feed the data in.
3. **Data Redundancy:** Physical copies of data provide a challenge in backup options putting the data at risk of loss without recovery.
4. **Data Analytics:** Analysis of data consumes more time when physical data is used as reference for the analysis. This will increase necessary hours to analyze the data and may even lead to inconsistencies in the analysis done.
5. **Physical contact concerns:** With the recent outbreak of the recent pandemic the corona virus, the World Health Organization has promoted avoiding physical contact. Access to the continually stated facilities is usually left to the guests who will end up sharing writing material encouraging the spread of the virus.
6. **Storage of data:** Accumulation of data over a long period of time may lead to space concerns due to their physical nature. This issue presents itself as a filing nightmare and access to specific data may be tedious.
7. **Identity theft:** Logging of numerous data sets in books is a hub for con artists and scammers who may get hold of personal information to defraud individuals in different way raising a security concern.

This system aims to solve the aforementioned issues affecting the process of physical data capturing and storage. Using several cameras and a microcontroller along with its associated code we can develop a system to provide digital and contactless capturing of data using the two core technologies of Object Character Retention and Computer Vision. Facial-recognition will also be incorporated which will interact with the data to provide access.

1.3 Problem Justification

The existing system of book-keeping underpins the need for a new system to **provide security at a much higher level** compared to the book-keeping process. Due to the facial recognition feature, the probability of providing fraudulent information is very low. The feature also allows us to set cameras at different points within the facility and provide a log of members accessing certain areas of the facility. This reduces the likelihood of theft occurring especially in sensitive areas which have expensive or important equipment. It also allows for a real time monitoring option from a centralized location streamlining the entire process.

This proposed system allows the **reduction of logging time** which may be pivotal depending on the facility. Hospitals and Cold rooms are an example of time-sensitive facilities whereby slow access may mean life or death in the case of a hospital and loss of valuable resources in the latter case. With this in mind, it is important the new system is implemented to sustain better use of time resources rather than wasting time in queues.

Furthermore, the system aims to **reduce physical contact in the logging process** in accordance with the health and safety rules provided by the Ministry of Health. This greatly reduces the risk factor of spreading the coronavirus. The system shall not require any handling of stationery or potential surfaces that may spread the virus. This measure shall allow a high influx of guests without having concerns of super-spreaders at the entrance of the facility.

Other secondary motivations to this project present themselves in the following form:

1. A database shall provide data redundancy to the system improving reliability and backup options for entire logging process.
2. Data captured shall be more accurate and standardized compared to data captured via physical means.
3. Space management shall be achieved as physical documents shall not be stored.

1.4 Objectives

1.4.1 Main Objective

- To design and implement a smart gate controller to provide access via identity management of individuals during entry into a facility.

1.4.2 Specific Objectives

- To design and implement a module capable of facial recognition of individuals from an underlying image database.
- To develop a module capable of retrieving relevant information and id scanning of a Kenyan id card of the individual into the system.
- To design and implement a front-end interface which allows interaction with user for any necessary analysis of the data.
- To develop a model responsible for number plate scanning and matching the associated vehicle to a specific member in the database.
- To design and implement a network of cameras to demonstrate real time detection and identification of individuals within the facility.

Chapter 2 : LITERATURE REVIEW

2.1 COMPUTER VISION

Computer vision is a technology that allows processors to interpret images using machine learning techniques. Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it to learn for themselves. [2]. Computer vision in this digital age is a very important tool to software engineers using it for various applications.

2.1.1 How Computer Vision works

Computer visions reads an image through a camera, image file, video stream or any relevant image reference and converts this image into a 4 value array known as a numpy array. The first 3 values have a range of values from 0 to 255 representing the intensity of Red, Green and Blue within the image. The final values return the number of channels in the image which is three for colored images and one for black and white images. Red, Green and Blue are the channels for colored images and with the correct combination can mathematically represent the image to allow the processor to decode the image. This in turn assists the machine to interpret the image hence apply different manipulations required for the programmer's needs.

Humans have a very fluid sensory system allowing interpretation of images very quickly. This allows us to differentiate between two different objects quite easily. A bicycle is easily differentiable from a car and we need not perform any procedures to find the difference. For computers, this feature cannot work in a similar way. The processor which is the brain of the computer can only perform logic and mathematical calculations hence the process of identifying objects must be applied differently. By converting images into an array of integer values allows for easier comparison.

Machine learning allows the processor to differentiate between the bicycle and the car by simple calculations. Before this however, we need to train a computer vision model to recognize and differentiate between objects. This involves the use of an image database to act as reference for the object in question. To allow the model to differentiate between a bicycle and a car we

provide different images showing this is a bicycle and that is a car. It is important to note that this database has a greater efficiency when multiple images are used as references.

Figure 2. 1 Difference between a bicycle and a car

2.1.2 Applications of Computer Vision

This impressive piece of technology has vast applications in today's world and has led to a growth of the scope of a programmer within the software field. These advancements include:

1. **Medical Advancements**: Computer vision has proved an important tool to aid doctors in diagnosis of several ailments by using images from a patient's organs. A group of researchers has trained an AI system to analyze CT scans of oncology patients. The algorithm showed 95% accuracy, while humans – only 65%. [3].
2. **Self-driving vehicles**: Using a set of sensors and cameras, car manufacturers are building vehicles that are able to navigate without a driver. The sensors and cameras are able to notify a centralized computer of the car's surrounding which in turns makes informed decisions on the data. Companies such as Tesla and Google are making major investments into this technology as it is the future of the automotive industry.
3. **Animal conservation**: Ecologists benefit from the use of computer vision to get data about the wildlife, including tracking the movements of rare species, their patterns of behavior, etc., without troubling the animals. [3]. This aids greatly in a greater understanding of the eco-system and record any shifts within it.
4. **AI-Driven Drone Software**: AI drone software is a robust and powerful technology with wide-scale application in various industries from aerial mapping, to modeling, and analytics. AI drones have quickly made inroads into various industries, automating the

legacy systems for better efficiency and precision. The computer vision technology powered by robust machine learning algorithms makes it possible for the software to observe, process, analyze, and interpret drone imagery in real-time to identify and extract the required information. [4].

5. **Factory management.** It is important to detect defects in the manufacture with maximum accuracy, but this is challenging because it often requires monitoring on a micro-scale. For example, when you need to check the threading of hundreds of thousands of screws. A computer vision system uses real-time data from cameras and applies ML algorithms to analyze the data streams. This way it is easy to find low-quality items. [3].
6. **Terrain Mapping**: AI drones have made significant inroads into civil engineering projects. Today, AI drones are excessively used across various civil engineering projects for faster, and precise terrain mapping, which is a prerequisite for projects AI drones are equipped with powerful sensors (LiDAR) and navigation systems to surveil the desired terrain and collect required data. The data is then processed using computer vision and machine learning models to create 3D models. [4].
7. **AI Image Processing**: Image processing has received a great boost from computer vision allowing which involves manipulation to increase image quality or capture important data from an image. This in turn has helped improve quality of some fields such as photography, forensics and data analytics.
8. **Security**: Computer vision also plays an important role in facial recognition applications, the technology that enables computers to match images of people's faces to their identities. Computer vision algorithms detect facial features in images and compare them with databases of face profiles. Consumer devices use facial recognition to authenticate the identities of their owners. Social media apps use facial recognition to detect and tag users. Law enforcement agencies also rely on facial recognition technology to identify criminals in video feeds. [1]. Facial recognition will be covered independently in this chapter.

2.2 FACIAL RECOGNITION

Facial recognition is a way of identifying or confirming an individual's identity using their face. Facial recognition systems can be used to identify people in photos, videos, or even in real-time. Of all the biometric measurements, facial recognition is considered the most natural. Intuitively, this makes sense, since we typically recognize ourselves and others by looking at faces, rather than thumbprints and irises. It is estimated that over half of the world's population is touched by facial recognition technology regularly [5].

2.2.1 How Facial Recognition works

Facial recognition involves multiple steps to successfully recognize a face profile matched to an individual. The steps are as follows:

1. **Face Detection:** This is greatly assisted by ML that allows a programmer to train a system to identify the boundaries of any given face. The result is a snippet of code that is able to draw boundaries around the given face profile hence distinguishing it from any other parts of an individual. This gives the system a chance to restrict its recognition to the face only and disregard any other frames not around the area of the face.
2. **Face Analysis:** The image is then captured and analysis begins. The facial recognition system then reads the geometry of the individual's face with factors such as distance between the eyes, distance from forehead to chin, contour of the lips, shape of cheekbones and depth of eye sockets are recorded. The aim of this is to identify distinct facial landmarks of the individual to build what is known as a faceprint as a unique identifier. This process also allows the image to be converted to digital data which can be easily processed.
3. **Finding a match**: The faceprint is then compared against a database of other known faces. For example, the FBI has access to up to 650 million photos, drawn from various state databases. On Facebook, any photo tagged with a person's name becomes a part of Facebook's database, which may also be used for facial recognition. If your faceprint matches an image in a facial recognition database, then a determination is made. [5].

2.2.2 Applications of Facial Recognition

Beyond just unlocking phones or laptops, the biometric software behind facial recognition applications can accurately identify faces today, better than other people can. While this makes the technology an obvious choice for security and identification purposes, it can also be used creatively and repurposed to serve different industries. [6]. These applications include:

1. **Fleet Management**: Trucking companies use facial recognition to keep a track record of who is driving a certain vehicle. It helps avoid any loss of goods on transit and promotes accountability of the driver. They could also be used to send alerts when unauthorized personnel are trying to gain access to a vehicle. The system could also be trained using ML to detect drowsiness in drivers hence preventing accidents due tired drivers.
2. **Law Enforcement:** Huge law enforcement agencies utilize facial recognition to track and identify criminals. The abundance of a large database by these agencies allows the system to be able to identify millions of individuals. Using several cameras this can be achieved in real time to aid in apprehending potential suspects or persons of interest.
3. **Military applications:** Militaries around the world have seen an increase in trend of using this piece of technology against their adversaries. Drones are fitted with facial recognition systems to help identify assailants and have seen a lot of combat hours in locating and neutralizing terrorist assets in the fight against terrorism.
4. **Border Control:** Border controls today, sync with information database such as INTERPOL's 'Facial Identification' method, to identify individuals against an accuracy scale. Processing facial information over the cloud also gives ample opportunities to run predictive algorithms over the footage to factor in things beyond just typical quality-of-image-enhancements, but factors like aging, plastic surgery, cosmetics and even the effects of narcotics. [6]. This practice is mostly used within the European Union.
5. **Ride-sharing companies**: Ride sharing companies such as Uber and Grab are using facial recognition features to ensure the right driver picks up the right customer to improve security. This has helped to quell fears from commuters entering strange cars in the name of ride sharing companies. Drivers are required to scan their face on their mobile device, to verify their identity—and the corresponding driver and safety

credentials. This information is relayed to their passengers to once again, give them ease of mind. [6]. It also protects the drivers from hijack situations and keeps them safe from fraudulent customers.

6. **Retail Industry:** Facial recognition could allow retailers to capture what shoppers are looking at in physical shops, turning what was long-known as "offline" shopping habits, into online ones as well. This essentially means greater insights and analytics into the purchasing habits of their customers. Physical retailers can also glean more information about the state of mind of their different shoppers by literally, the look on their faces. Facial recognition technologies can identify a whole range of emotions—happy, sad, anxious, angry—allowing actionable reports to on-ground retail operators. [6].
7. **Security:** Facial recognition is used in very secure building where security is of the utmost priority. Institutions such as large banks and mints, government facilities, embassies, large corporations and many more use this system to protect against unauthorized entry which may end up to be detrimental to the company or facility.

2.3 OPTICAL CHARACTER RECOGNITION

OCR, short for optical character recognition, refers to the technology used to convert printed, written, or typed characters into a digital format. The process allows text to be read by a computer which makes the characters able to be edited and searched [7]. OCR has a major role in today's market ranging from simply scanning documents to aiding the visually impaired.

2.2.1 How OCR works

The first step of the process involved is the capturing of the image using a camera or an optical scanner. After this process two different approaches can be used known as pattern recognition and feature detection with the latter being used for this proposed project.

1. **Pattern Recognition:** Pattern recognition, also known as matrix matching and pattern matching, works by comparing a character that it scans to a backlog of character images that the program has on file. This only works when characters are an exact pixel match to the characters that the OCR uses as a reference. In the 1960s, the OCR-A font was created to be printed on checks in order to standardize the font for bank's OCR

programs. Eventually, pattern recognition expanded to recognizing multiple common fonts like Arial and Times New Roman, but the program was still restricted to only converting characters that were in a select few fonts. [7].

2. **Feature Detection:** Feature detection, also referred to as feature extraction, allows OCR to read characters in almost any font. Instead of looking for a character s exact replica, feature detection works by distinguishing specific features that a character will have no matter the font. For example, if the program scans a character that has two slanting lines that meet at the top to make a point, it can tell that the letter is a capital A. Feature detection expands the uses and capabilities of OCR programs, and is the most common type of OCR used today. [7].

The software typically has a preprocessing phase that attempts to make the text in the document clearer and easier to read. No scanner is perfect, so with most modern, commercial scanners, there are bound to be imperfections in the scanned image. It does this by cleaning up the image and isolating the characters from everything else. It makes sure that the lines of text are properly aligned and the pixels are smoothed out. [8].

2.2.2 Applications of OCR

Optical Character Recognition, which might seem like a new piece of technology, was first developed in 1929 by an Austrian engineer Gustav Tauschek eventually obtaining a patent for a device that uses the principle. However, many believe that it was first developed during the first world war when a machine that converted character texts into telegraphs by physicist Emanuel Goldberg. This technology is used till this day and has many applications such as:

1. **Data entry**: OCR is currently used in different institutions and organizations for data entry systems to reduce workload that comes with manual input of data. Programs are created using the principle and collect data and feed it into a database or any other relevant system that requires the data to run.
2. **Digitizing books**: It is also used by authors, writer and publishers around the world to digitize written or typed reading material. It aims to save time during the transfer of the content to online users and has proved a capable tool for this task.
3. **Aiding the visually impaired**: The visually impaired have benefitted from the growth of OCR helping them into their day to day lives in interpreting important texts around

them or even allow them to understand written content not in the form of braille. The system developed involves scanning a piece of text and outputting it as audio feed.

4. **Identity Card and Passport scanning**: Identity cards and passports usually have special zones that recognized internationally usually at the bottom of the document known as Marked Readable Zones. These zones contain the information of the holder in a special format with each detail separated by a double left arrow (<<). This helps extraction of details easier for the system designed to collect the information.

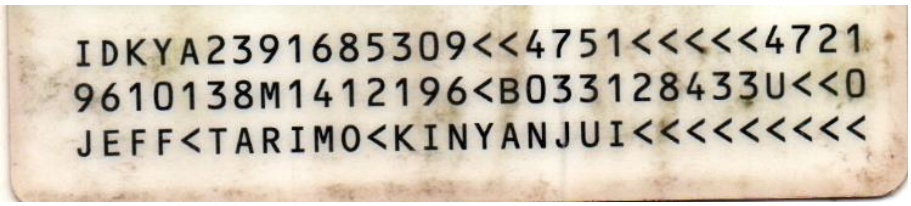


Figure 2. 2 Marked Readable Zone in a Kenyan Identity Card

5. **License Plate Recognition**: License plates can be recognized using OCR and with an existing database be linked to an individual for safety reasons. This technology has been implemented in parking lots, by security agencies to track certain vehicles linked to cases or persons of interest.

2.4 EXISTING TECHNOLOGIES IN SECURITY SYSTEMS

2.4.1 Book-keeping

Most institutions and organizations use this method to ensure the safety of their premises. This process involves the manual entry of data of guests coming in and out of the facility. Details captured include the name of the guest, phone number, id card number along with time of entry and exit. More often than not the guest is obliged to leave a piece of documentation at the security office as an extra layer of security. The documentation is returned to the guest when he or she exits the facility and a probable check of their belongings to ensure nothing has been taken but this is not common. Workers, students or residents of the facility use the facility

identity card and their details are not logged within the security book since they are taken as natives of the facility.

Advantages of this system

1. It is a cheap method to implement and not many components required to run it.
2. It does not require any skilled labor to perform the data entry.

Disadvantages of this system

1. It consumes a lot of time especially in instances where large numbers are queued awaiting the data capturing.
2. May lead to error during input due to illegible handwriting or mistakes when transferring the data to the security book.
3. Does not provide the ability to analyze the traffic in and out of the facility in real time.
4. Storage room for the books becomes difficult after a long history of data capturing making analysis even harder.
5. This method is a hub of spreading of the new coronavirus as exchange of documents or stationery may lead to an outbreak of the virus within the facility.

2.4.2 CCTV Cameras

Closed Circuit Television (CCTV) cameras have been integrated into security systems due to their availability to give residents of a certain institution or organization a sense of security while also protecting the institution from any cases of theft or destruction of property. These systems are widely used and consist of using a network of connected cameras to view in real time current events around a certain camera or a history of the same.

Advantages of this system

1. Allows real time viewing of events occurring at specific area of any given facility.
2. Readily available in the market.
3. Keeps a history of video streams to allow for review during a given incident to provide transparency.

Disadvantages of this system

1. As constructed CCTV cameras do not perform facial recognition to recognize individuals.
2. They are also unable to pull data from key documents such as license plates.
3. Cannot read identification documents.

2.4.3 Automated Border Control Systems

This is a system which allows for an automated border passage by authenticating an electronic machine readable travel document, establishing that the passenger is the rightful holder of the document, querying border control records and automatically determining eligibility for border crossing according to pre-defined rules and which is composed of a self-service system and an e-gate. [9]. After the identification process is complete, a physical barrier such as a gate or turnstile opens to permit passage. E-gates came about in the mid-2000s as an automated method of reading the newly ICAO mandated e-passports. Airports worldwide are struggling with security issues because of the growing number of passengers traveling globally. In recent years, the number of travelers in airports has increased considerably. Installation of airport e-gates has become a necessity to ensure security of passengers. Airport e-gates allow automated entry to passengers at security checkpoints. E-gates identify individuals using either iris scanning, fingerprint scanning, or facial scanning. Also, e-gates at airports reduce the check-in time of passengers. [10].

Figure 2. 3 Automated Border Control Systems

Advantages of this system

1. Uses biometric details to provide accurate results.
2. Robust and efficient.

Disadvantages of this system

1. These systems are very expensive to acquire and commission.
2. Automated Border Control Systems cannot scan license plates belonging to certain individuals.
3. They require a large database to operate with due to their locations.

2.4.4 RESEARCH GAP AND PROPOSED APPROACH

Having understood the current technologies applied currently, I noticed a great gap in these systems and hence this project aims to address the limitations of the existing technology to provide a fully integrated system that can perform the following:

1. **Facial Recognition**: The proposed system will use an image database to perform facial recognition improving accuracy of authorized entry. This entry of the individual is logged for future reference within the system. This reduces time spent during verification and improves efficiency and accuracy of the entire process.

2. **Identity card or passport scanning**: The proposed system will be able to scan identity cards or passports when entering the facility to retrieve necessary information from a guest. This data is logged in the system along with the time in the guest has accessed the facility.
3. **Number Plate Recognition**: The proposed system looks to captured vehicle registration plates and then linked to the driver to provide a layer of security within the facility. The license plate is linked to an individual via their scanned identity card or passport.
4. **Real-time viewing of specific zones**: The proposed system will also integrate a network of cameras at specific areas in the facility to allow real time viewing of these zones. Furthermore, working alongside the facial recognition feature they will be able to detect certain individuals and log their appearances in these areas in the system.
5. **Analysis of data**: The proposed system will log any relevant information and display the logged data in a user friendly dashboard to allow smooth access of movements within the facility and logged visitors within the facility.

Integrating computer vision, facial recognition and OCR the proposed system will perform these functionalities to provide a security system combining the best technological practices to achieve secure systems in any given institution or organization. The system shall be dubbed the **smart gate controller security system.**

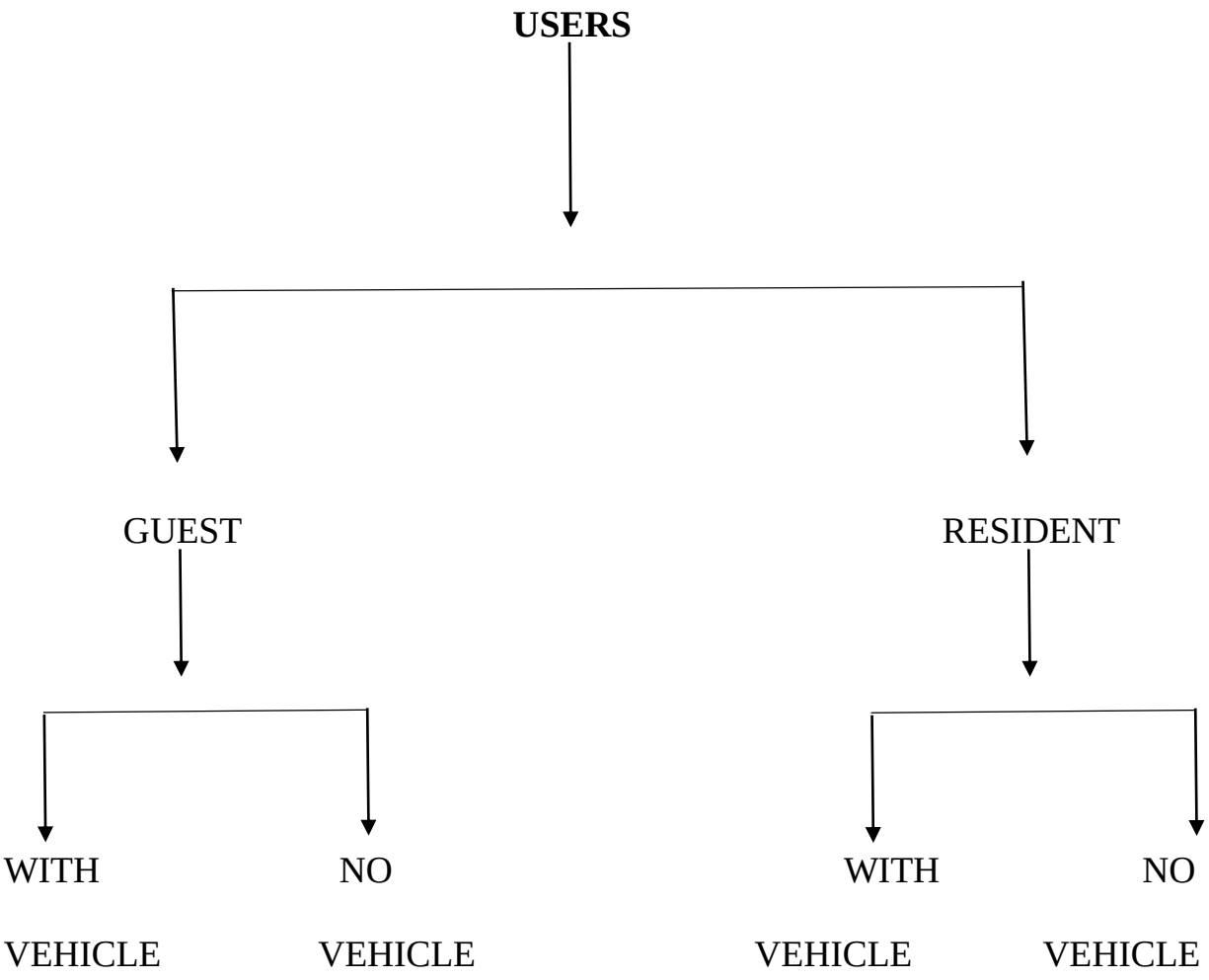
Benefits of this system

1. Data captured is accurate and reliable and easily analyzed and viewed in real time within the proposed system.
2. Time wasted is greatly reduced during data capturing process.
3. Security is enhanced in specified zones within the facility using the proposed system.
4. Identity cards or passports need not be left at the security office reducing the risk of losing detrimental documents.
5. The proposed system will not require special skills to operate and will be simple to operate.

Chapter 3 : METHODOLOGY

3.1 CLASSIFICATION OF USERS

To better understand the proposed project, we need to first find a way to classify our users. Users of the proposed project shall be categorized as shown below:



With this in mind we can now design an algorithm for the proposed system allowing us to understand the program flow.

3.2 BLOCK DIAGRAM

Below shows the block diagram for the proposed project.

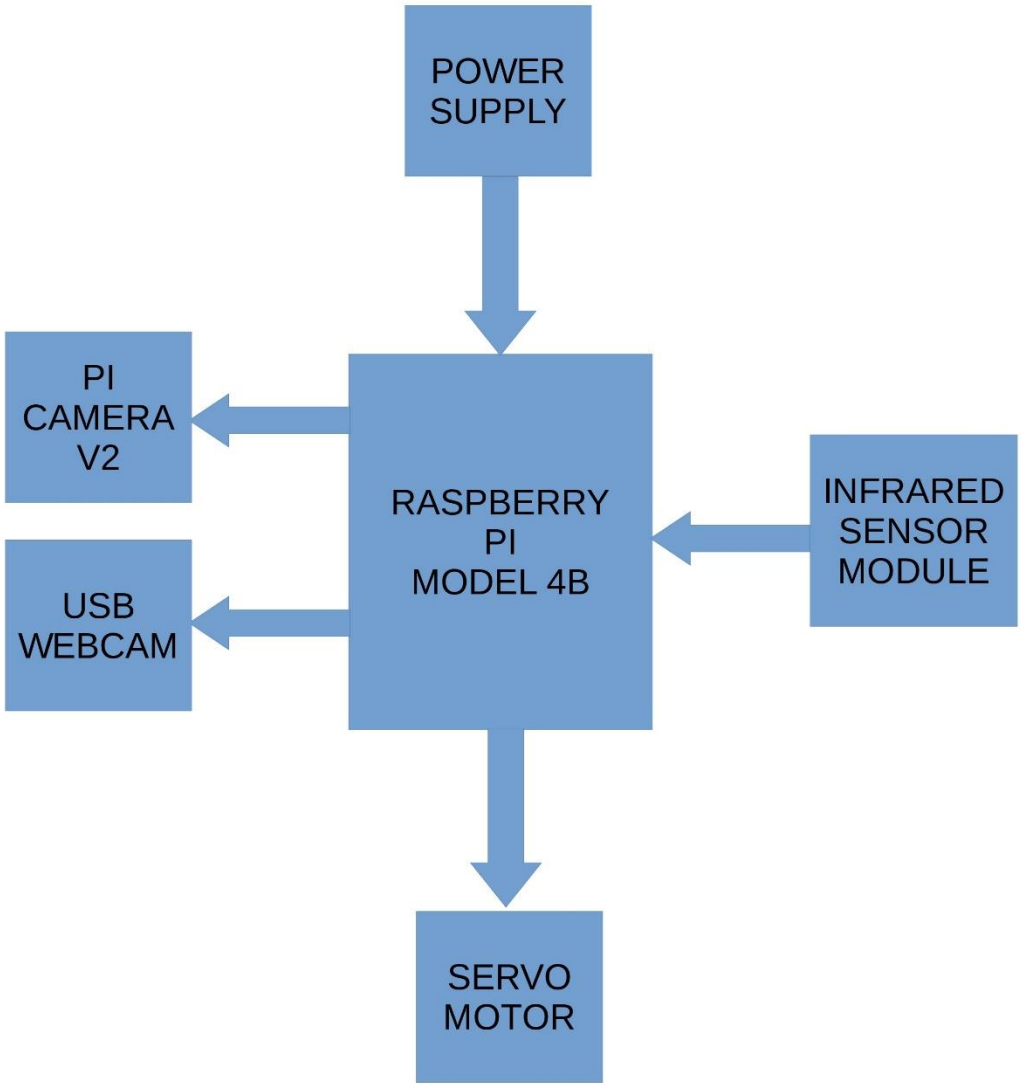


Figure 3. 1 Block Diagram of the system

3.3 ALGORITHMS

The following shows a detailed algorithm for the proposed system which seeks to address the main objective:

1. Prompt the user if he/she is a resident or a guest.
2. If the user is a resident go to 4.
3. If the user is a guest go to 7.
4. Prompt resident if he/she has a vehicle or not.
5. If resident has a vehicle go to 10.
6. If resident has no vehicle go to 13.
7. Prompt guest if he/she has a vehicle or not.
8. If guest has a vehicle go to 16.
9. If guest has no vehicle go to 18.
10. Provide access to facial recognition camera and license plate camera scanners.
11. If face is recognized allow access and log entry into system along with license plate number and time.
12. If face is not recognized deny access.
13. Provide access to facial recognition camera only.
14. If face is recognized allow access and log entry into system along with time.
15. If face is not recognized deny access.
16. Provide access to identity card scanning camera and license plate camera scanners.
17. Log relevant data into system along with time and license plate number and allow access.
18. Provide access to identity card scanning camera only.
19. Log relevant data into system along with time and allow access.

3.3.1 FACIAL RECOGNITION ALGORITHM

The following shows a detailed algorithm for the facial recognition software. The algorithm looks to address the first specific objective as laid out in chapter one:

1. Read reference images (image database) from certain path and put them in a list.
2. Covert image from BGR (blue-green-red) scheme to RGB (red-green-blue) scheme.
3. Perform face encodings for images and save the encodings in a list.
4. Capture video from the camera.
5. Run an infinite loop to read camera images.
6. Find face locations in image to prepare for comparison.
8. Find the difference and if the difference is small, faces match and access granted.

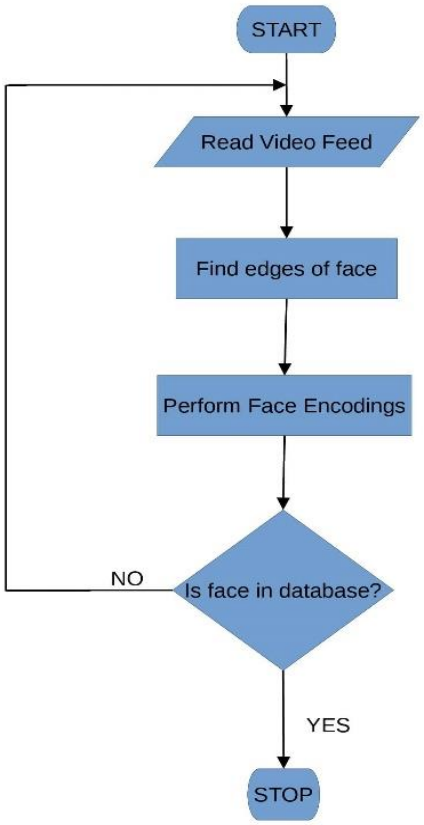


Figure 3. 2 Facial Recognition Algorithm

3.3.2 IDENTITY CARD SCANNING ALGORITHM

The following shows a detailed algorithm for the identity card scanning software. This will address the second specific objective as detailed in the first chapter:

1. Capture identity card image using camera.
2. Crop image to find borders of MRZ zones.
3. Convert image to string using OCR (Tesseract OCR).
4. Split text from the double arrows (<<) in the MRZ zones.
5. Save identity with name, id number and time as parameters saved.

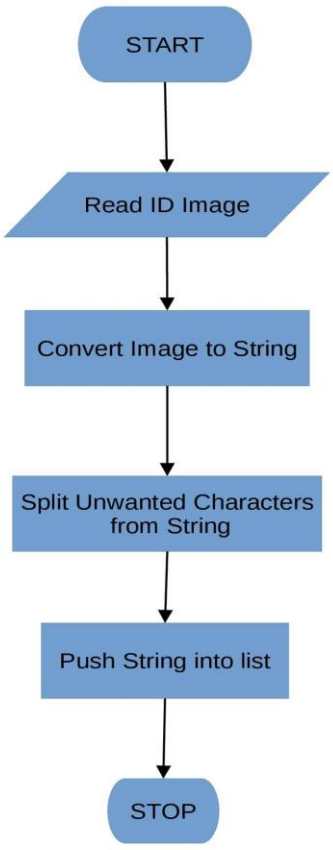


Figure 3. 3 ID Card Scanning Algorithm

3.3.3 LICENSE PLATE SCANNING ALGORITHM

The following shows a detailed algorithm for the license plate scanning software. The algorithm looks to address the fourth specific objective as laid out in chapter one:

1. Capture license plate using camera.
2. Crop image to find borders of the license plate.
3. Convert image to string using OCR (Tesseract OCR).
4. Save license plate number.

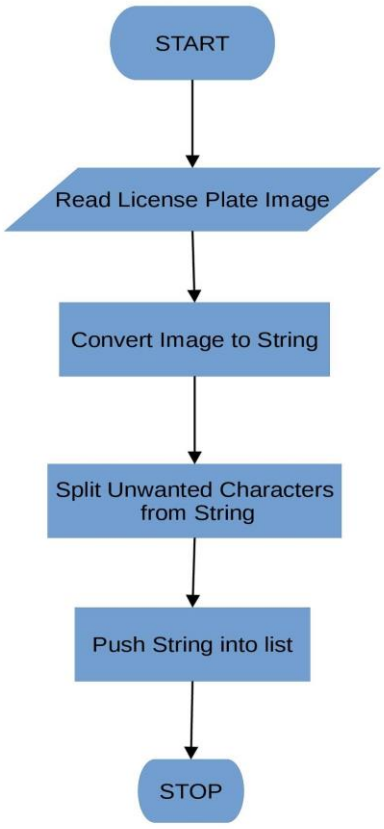


Figure 3. 4 License Plate Scanning Algorithm

3.4 OVERALL FLOWCHART

Below shows a flowchart describing the processes involved in the proposed system:

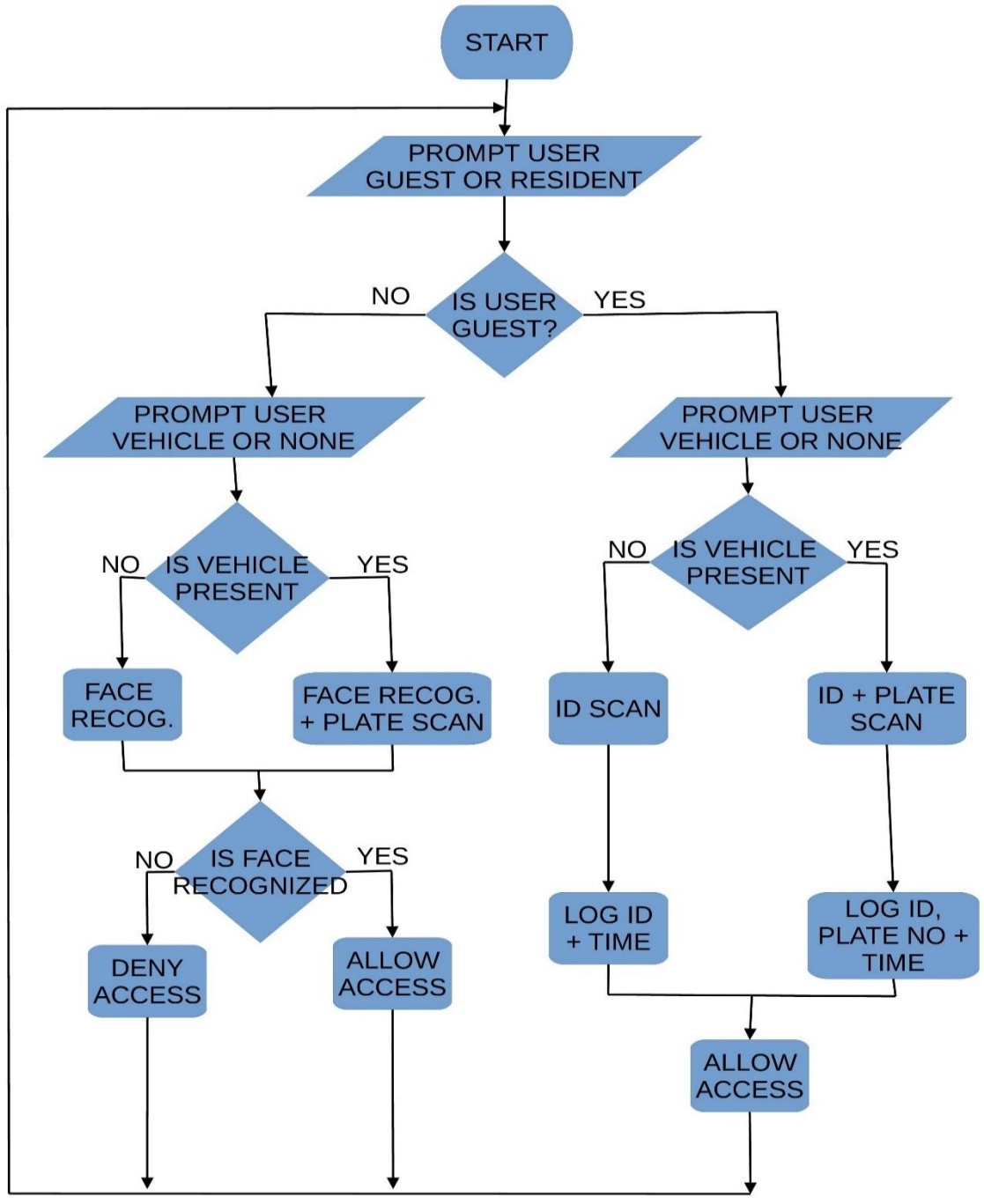


Figure 3. 5 Overall Flowchart of the system

3.5 FRONT END DEVELOPMENT

Data collected from the proposed system will need to be viewed and this will be done via a user-friendly web application using anvil. Anvil is a powerful tool that helps build web applications using the python programming language. Several advantages arise from the use of this tool:

1. Using one programming language will avoid unnecessary translation between programming languages saving computing resources.
2. It will allow the user to connect remotely to the microcontroller and control the controller's services.
3. Very robust and requires less code to develop.
4. Inbuilt database capabilities offer data logging functionality to the application and to the service as a whole.

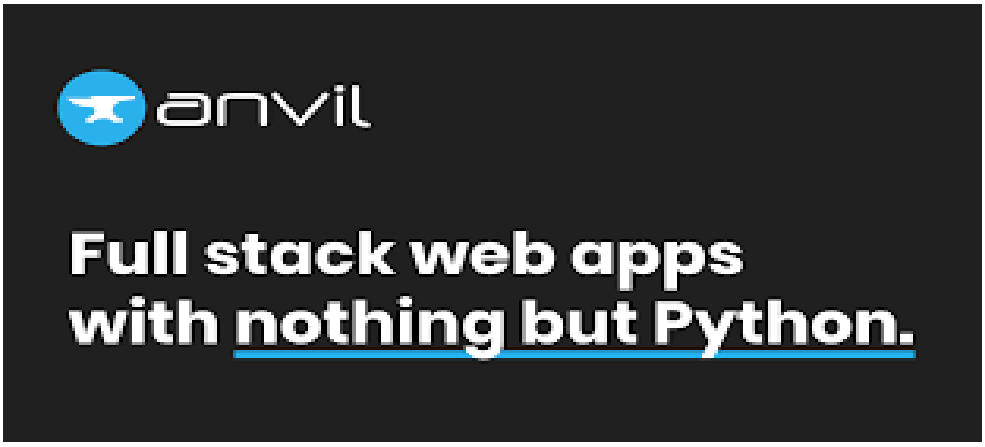


Figure 3. 6 Web Applications with Anvil

The interface will look to address the third specific objective as laid out in chapter one.

3.6 IMPLEMENTATION

3.6.1 HARDWARE IMPLEMENTATION

Below shows the hardware layout of the entire system:

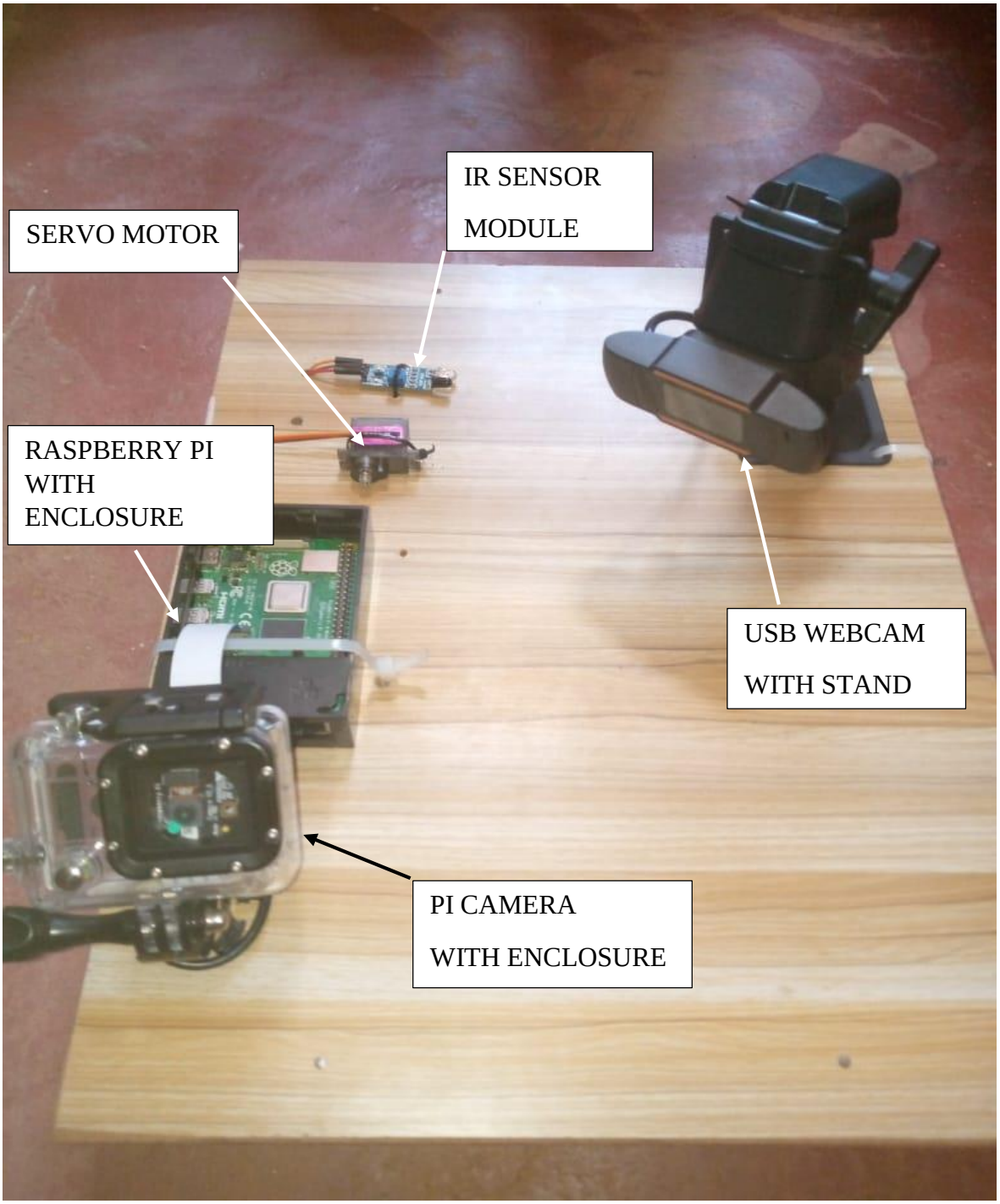


Figure 3. 7 Hardware setup of the system

3.6.1.1 RASPBERRY PI

A raspberry pi is a digital microcontroller with 40 GPIO pins and several peripheral interfaces to support several devices. These peripherals include:

1. 2 USB 2.0 ports
2. 2 USB 3.0 ports
3. 1 Ethernet port
4. 2 micro HDMI ports
5. 1 USB-C power delivery port
6. 1 POE port
7. 1 3.5mm Audio jack
8. 1 Raspberry pi camera CSI port

The raspberry pi will be used to control all processes of the project and perform any timing necessary in the delivery of the services.

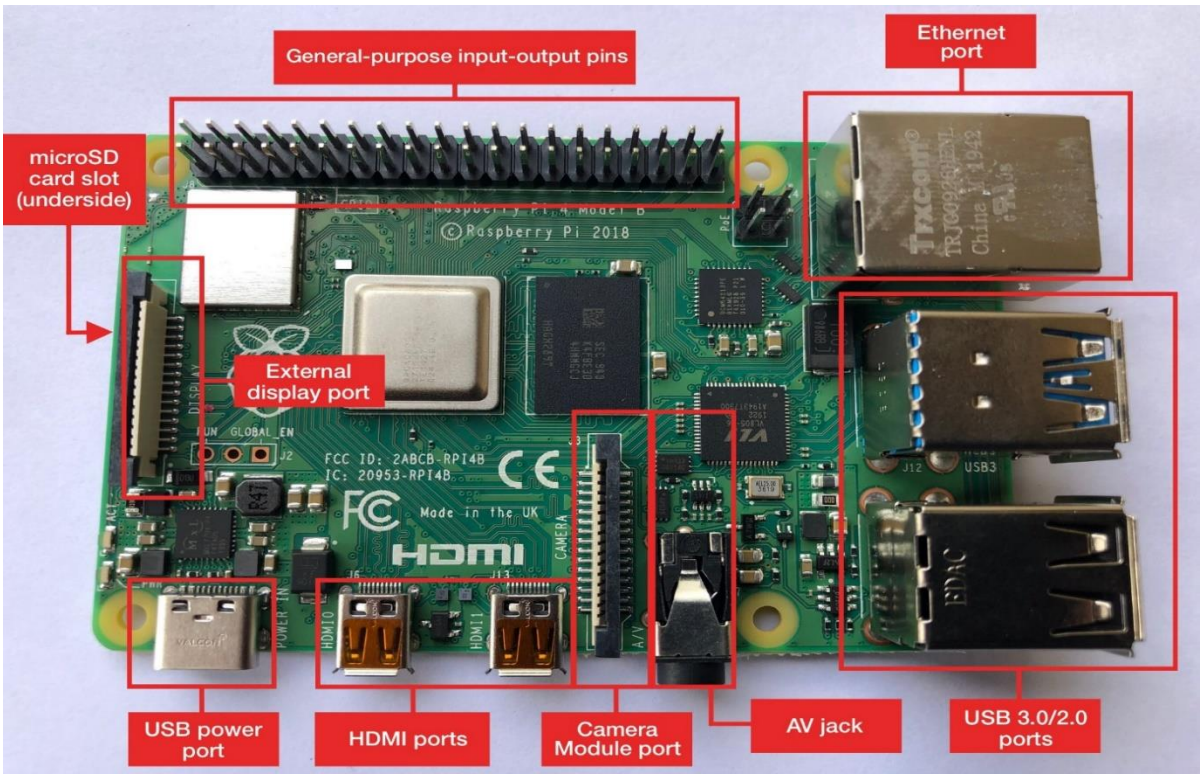


Figure 3. 8 Raspberry Pi Components

3.6.1.2 SERVO MOTOR

A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration. The servo motor was used to control the gate to allow entry of individuals and was interfaced to the microcontroller as indicated below:

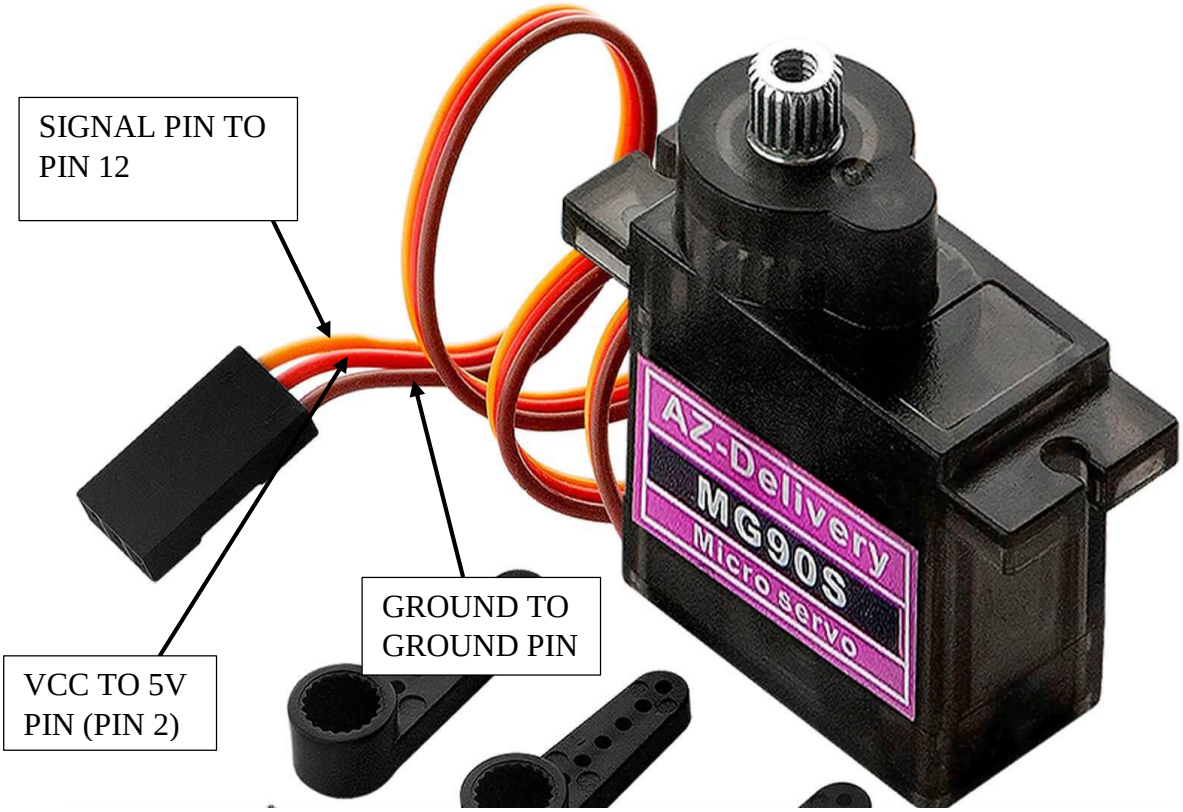


Figure 3. 9 Servo Motor Connections

3.6.1.3 INFRARED SENSOR MODULE

The IR Infrared Obstacle Avoidance Sensor Module has a pair of infrared transmitting and receiving tubes. When the transmitted light waves are reflected back, the reflected IR waves will be received by the receiver tube. The onboard comparator circuitry does the processing and the green indicator LED comes to life.

The infrared sensor will be used to send a signal to the microcontroller that processing and entry of an individual has been completed. Once this signal is received the microcontroller

sends a signal to the servo motor to close the gate and prepare for the next individual. The sensor outputs a HIGH if an obstacle is sensed and LOW if otherwise.

The connection from the infrared sensor to the microcontroller is shown below:

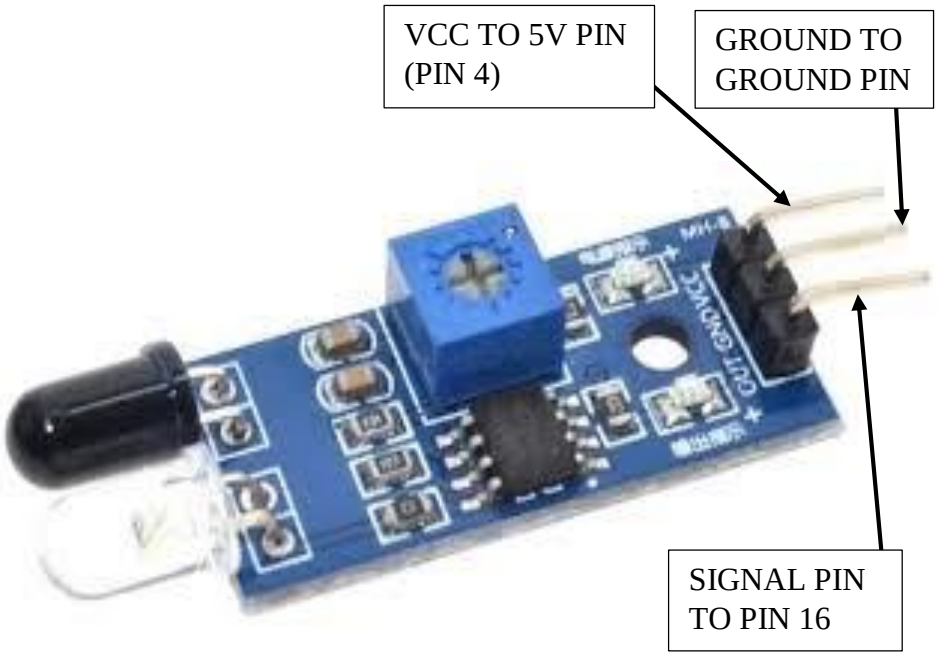


Figure 3. 10 Infrared Sensor Module Connections

3.6.1.4 RASPBERRY PI CAMERA VERSION 2

The Raspberry Pi Camera Module v2 is a high quality 8 megapixel Sony IMX219 image sensor custom designed add-on board for Raspberry Pi, featuring a fixed focus lens. It's capable of 3280 x 2464-pixel static images, and also supports 1080p30, 720p60 and 640x480p60/90 video. It attaches to Pi by way of one of the small sockets on the board upper surface and uses the dedicated CSI interface, designed especially for interfacing to cameras.

The raspberry pi camera will be used for the facial recognition where it will be tasked with collecting a video stream, provide facial detection as per the underlying code and recognized individuals based on the dataset (image database) in the raspberry pi.

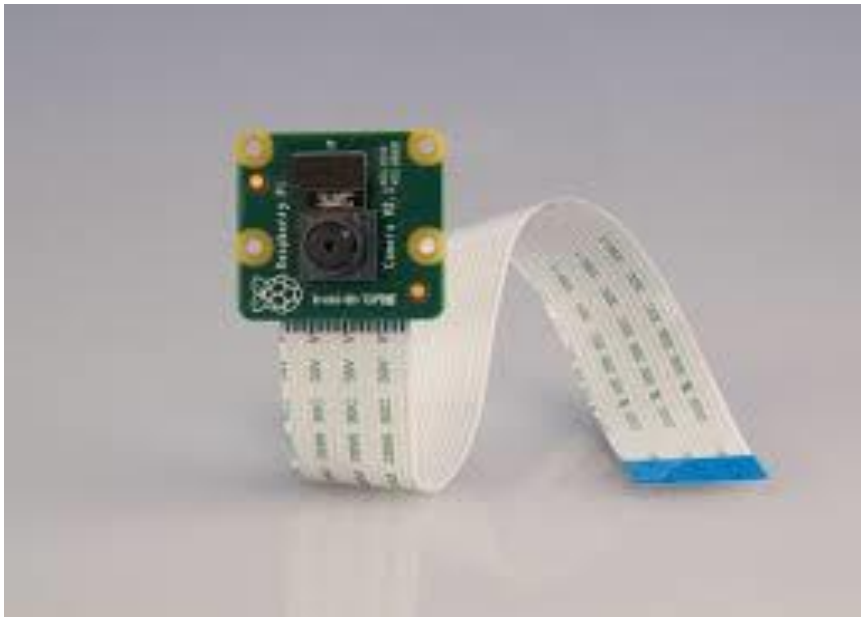


Figure 3. 11 Raspberry Pi Camera Module Version 2

3.6.1.5 USB WEBCAM

The USB Webcam will offer imaging services to our microcontroller to enable the user to capture identity card information as well as license plate information. The microcontroller will receive the images and will be required to process them appropriately and convert them to text to allow us to push this data to our database.

Figure 3. 12 USB Webcam by Suncity Technologies

3.6.2 SOFTWARE IMPLEMENTATION

3.6.2.1 FACIAL RECOGNITION SOFTWARE

Before performing facial recognition, we will require to first train our model based on the dataset (image database) within our local directory. The following was done during this process:

1. A folder named dataset was saved in the same directory as all the code required to be run during the entire process. This folder contains subfolders named after individuals who will be recognized by the system. The subfolders contain the images used to train our system.
2. We will then require several libraries to help our scripts run. Libraries include: open-cv, facial-recognition, imutils, numpy, datetime, time, pickle, picamera, RPI.GPIO and anvil. These libraries are imported into our scripts to provide complete functionality to our system. To install our libraries, we type the following command in our terminal: "sudo pip3 install <name_of_library>"
3. A python script is then run to serialize the photos within the dataset folder into encodings to allow our microcontroller to compare these images against the photos obtained during facial recognition. The python script is saved as "train_model.py".
4. This script will save the resulting encodings in a file named "encodings. pickle". These encodings will be accessed during the facial recognition process.
5. We will then require an XML script which will be responsible for identifying key face features such as depth of eye sockets, length of face, shape of cheekbones etc. This helps our facial recognition software to identify these parameters within the images. This is also saved in a similar directory and named "haarcascade_frontalface_detection.xml".

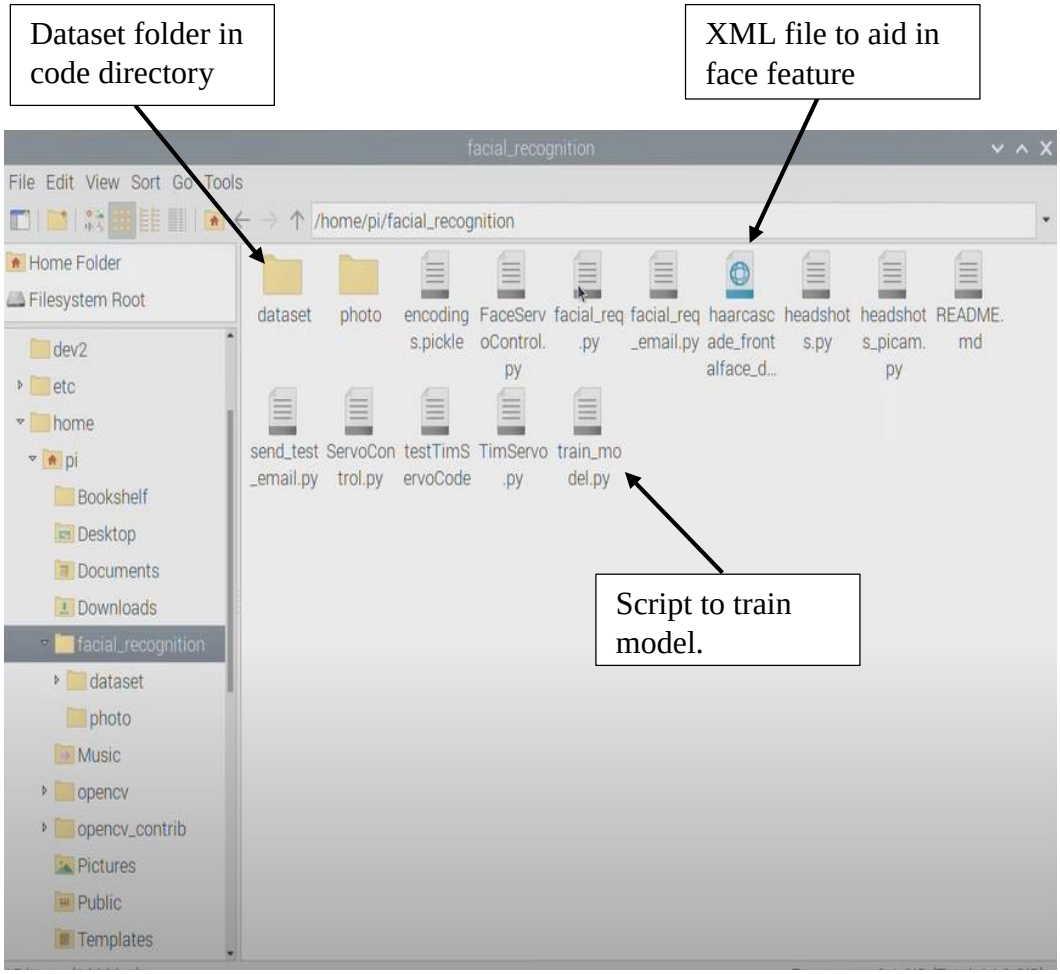


Figure 3. 13 Python Files and prerequisites for facial recognition

```
#! /usr/bin/python

# import the necessary packages
from imutils.video import VideoStream
from imutils.video import FPS
import face_recognition
import imutils
import pickle
import time
import cv2
import datetime
import RPi.GPIO as GPIO
import anvil.server
import anvil.media
from picamera.array import PiRGBArray
from picamera import PiCamera
import pytesseract
import numpy as np
```

Figure 3. 14 Imported libraries needed in our python scripts

Facial recognition will go through the following steps from start to finish:

1. All libraries will be imported to aid our program to run.

2. Source of the video stream will be initialized to the raspberry pi camera as shown below. The previously stated encoding file is also referenced along with xml file responsible for detecting face features.

```
#Initialize 'currentname' to trigger only when a new person is identified.
currentname = "unknown"
#Determine faces from encodings.pickle file model created from train_model.py
encodingsP = "encodings.pickle"
#use this xml file
#https://github.com/opencv/opencv/blob/master/data/haarcascades/haarcascade_frontalface_default.xml
cascade = "haarcascade_frontalface_default.xml"

# load the known faces and embeddings along with OpenCV's Haar
# cascade for face detection
print("[INFO] loading encodings + face detector…")
data = pickle.loads(open(encodingsP, "rb").read())
detector = cv2.CascadeClassifier(cascade)

# initialize the video stream and allow the camera sensor to warm up
print("[INFO] starting video stream…")
vs = VideoStream(usePiCamera=True).start()
#vs = VideoStream(usePiCamera=True).start()
time.sleep(2.0)

# start the FPS counter
fps = FPS().start()
```

Figure 3. 15 Initialization of program variables

3. The camera will then be run in an infinite loop to capture the frames as video and to loop over the frames. The frames will then be resized and converted into a gray image for processing. The encodings from the file “encodings. pickle” will then be loaded to compare the frames against the saved encodings.

```
while True:
    # grab the frame from the threaded video stream and resize it
    # to 500px (to speedup processing)
    frame = vs.read()
    frame = imutils.resize(frame, width=500)

    # convert the input frame from (1) BGR to grayscale (for face
    # detection) and (2) from BGR to RGB (for face recognition)
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)

    # detect faces in the grayscale frame
    rects = detector.detectMultiScale(gray, scaleFactor=1.1,
        minNeighbors=5, minSize=(30, 30),
        flags=cv2.CASCADE_SCALE_IMAGE)

    # OpenCV returns bounding box coordinates in (x, y, w, h) order
    # but we need them in (top, right, bottom, left) order, so we
    # need to do a bit of reordering
    boxes = [(y, x + w, y + h, x) for (x, y, w, h) in rects]

    # compute the facial embeddings for each face bounding box
    encodings = face_recognition.face_encodings(rgb, boxes)
    names = []
```

Figure 3. 16 Looping of the frames and loading of encodings

4. A loop is then run to go through encodings and match the resulting frames to a specific individual whose dataset has been saved locally. If a match is detected, we print the name of recognized individual as well as the current time.

```
for encoding in encodings:
    # attempt to match each face in the input image to our known
    # encodings
    matches = face_recognition.compare_faces(data["encodings"],
        encoding)
    name = "Unknown" #if face is not recognized, then print Unknown

    # check to see if we have found a match
    if True in matches:
        # find the indexes of all matched faces then initialize a
        # dictionary to count the total number of times each face
        # was matched
        matchedIdxs = [i for (i, b) in enumerate(matches) if b]
        counts = {}

        # loop over the matched indexes and maintain a count for
        # each recognized face face
        for i in matchedIdxs:
            name = data["names"][i]
            counts[name] = counts.get(name, 0) + 1

        # determine the recognized face with the largest number
        # of votes (note: in the event of an unlikely tie Python
        # will select first entry in the dictionary)
        name = max(counts, key=counts.get)

        #If someone in your dataset is identified, print their name on the screen
        if currentname != name:
            currentname = name
            now = datetime.datetime.now()
            dt_string = now.strftime("%m/%d/%Y, %H:%M:%S")
            p.start(7.5)
            new_dt_string = currentname + " " + dt_string
            print(new_dt_string)
            p.ChangeDutyCycle(7.5)
            time.sleep(1)
            p.ChangeDutyCycle(2.5)
            time.sleep(1)
```

Figure 3. 17 Looping of the encodings and finding matches

3.6.6.2 ID SCANNING SOFTWARE

The ID scanning software will undergo through the following steps to obtain the necessary information:

1. The USB webcam will be initialized and an image of the ID card taken.
2. Cropping of the image will be done to ensure the correct data will be collected as well as to correct for possible errors.
3. The cropped image will then be converted to text and the required data collected.

```
camera = PiCamera()
camera.resolution = (640, 480)
camera.framerate = 32
camera.rotation = 180
camera.sharpness = 60
rawCapture = PiRGBArray(camera, size=(640, 480))
    # allow the camera to warmup
time.sleep(0.1)

    # capture frames from the camera
camera.capture(rawCapture, format = 'bgr')
image = rawCapture.array

text = pytesseract.image_to_string(image)

cv2.imshow("IDENTITY CARD", image)
```

Figure 3. 18 ID Scanning Script

3.6.6.3 LICENSE PLATE SCANNING SOFTWARE

The License plate scanning software will undergo through the following steps to obtain the necessary information:

1. The USB webcam will be initialized and an image of the license plate taken.
2. Cropping of the image will be done to ensure the correct data will be collected as well as to correct for possible errors.
3. The cropped image will then be converted to text and the required data collected.

```
camera = PiCamera()
camera.resolution = (640, 480)
camera.framerate = 32
camera.rotation = 180
camera.sharpness = 60
rawCapture = PiRGBArray(camera, size=(640, 480))
    # allow the camera to warmup
time.sleep(0.1)

    # capture frames from the camera
camera.capture(rawCapture, format = 'bgr')
image = rawCapture.array

text = pytesseract.image_to_string(image)

cv2.imshow("IDENTITY CARD", image)
```

Figure 3. 19 License Plate Scanning Script

3.6.6.4 CCTV MODE

The CCTV Mode depends on two scripts that run simultaneously to offer a live stream of the two cameras which are configured to perform facial recognition. The scripts are similar to the facial recognition scripts with each accessing each camera as a source to allow the user to

view both sources at the same time. Once an individual has been recognized, their name is sent to a remote database that has been powered by anvil. Each camera has been given a designated location to report to the database with the raspberry pi camera set to “AREA B” while the USB webcam set to “AREA A”. The scripts will be run at the backend and video stream will be shown and the following command will be used to run the two scripts simultaneously: “sudo python3 script1.py & sudo python3 script2.py “.

3.6.6.5 WEB APPLICATION

The resulting web application using anvil contained 11 forms for data entry and viewing. Four of the forms were used to connect to the microcontroller and log the details required at the gate based on the individual’s classification. The four classifications were outlined as “RESIDENTS”, “RESIDENTS WITH VEHICLE”, “GUESTS” and “GUESTS WITH VEHICLES”. A log in form is also present to give authorized access to the application. The index page will allow the logging in of users based on the aforementioned classifications as shown below:

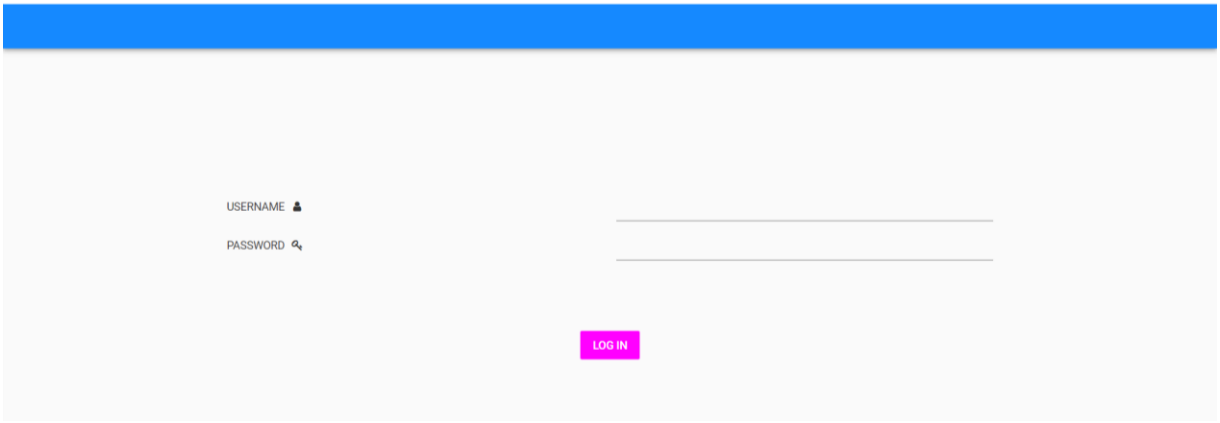


Figure 3. 20 Web App Log In Form

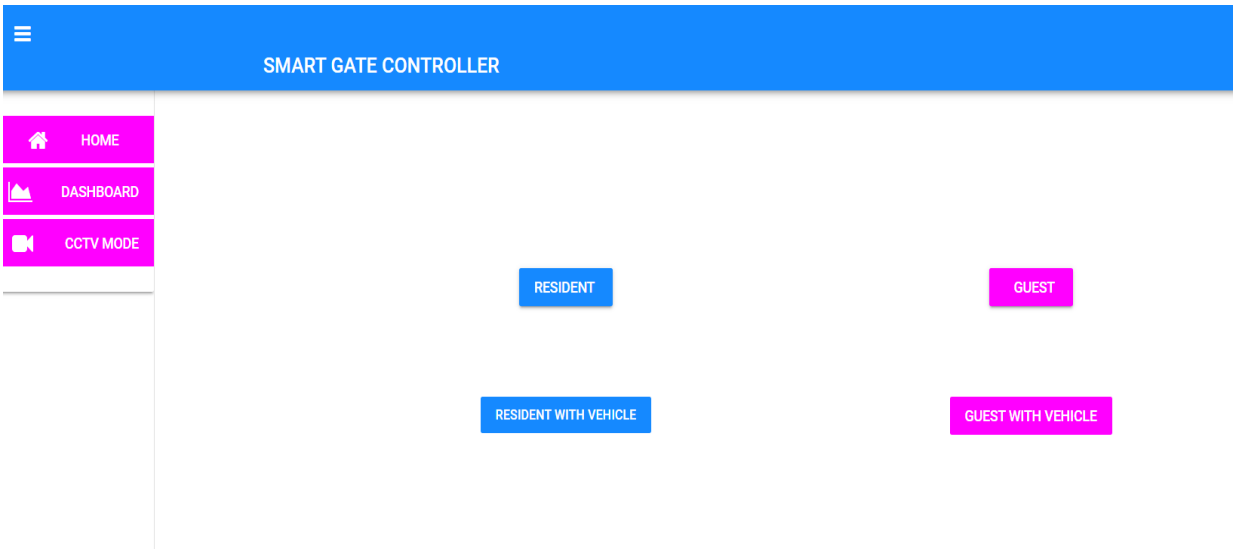


Figure 3. 21 Web App Index Form

Each button when clicked goes to a different form to allow registration of the individual based on their classification. The remaining four forms will be used to view the data that has been logged. More information is provided in the next section.

3.6.6.6 DATABASE

Data that has been collected will be stored in a remote database thanks to anvil's database capabilities. A total of 6 data tables have been used to support the application's capabilities. Four of these tables are to store data from our four classifications. The classification "RESIDENT" stores its data in a table called "RESIDENTS". It contains two rows namely "name" and "time" indicating the name of a logged individual as well as the time they were logged in. The classification "RESIDENTS WITH VEHICLE" stores its data in a table called "RESIDENTS WITH CARZ". It contains three rows namely "name", "plate" and "time" indicating the name of a logged individual, the plate number of their vehicle as well as the time they were logged in. The classification "GUESTS" stores its data in a table called "GUESTZ". It contains three rows namely "name", "identification" and "time" indicating the name of a logged individual, their identification number as well as the time they were logged in. The classification "GUEST WITH VEHICLE" stores its data in a table called "GUESTS WITH CARS". It contains four rows namely "name", "identification", "plate" and "time" indicating the name of a logged individual, their identification number, the plate number of their vehicle as well as the time they were logged in. The other data table contains the user

credentials to provide easy addition of users to the system and another table is used to log individuals recognized in CCTV mode. The tables are shown below:

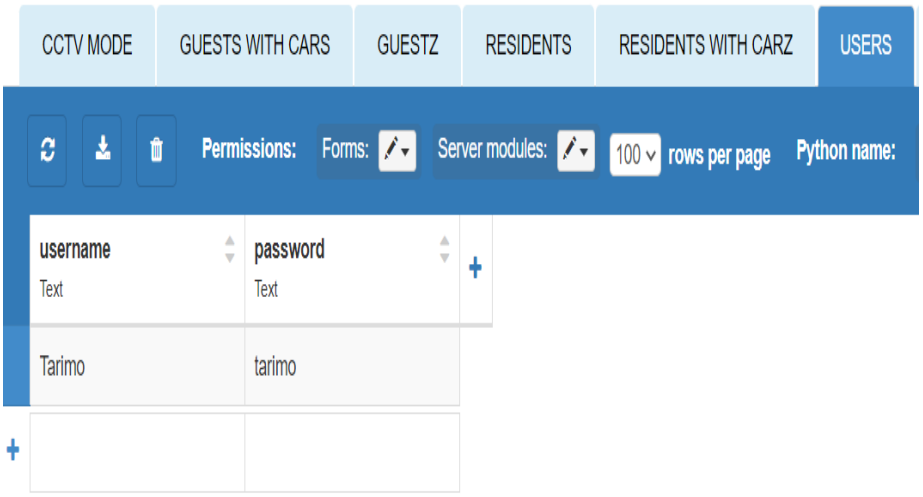


Figure 3. 22 Users data table

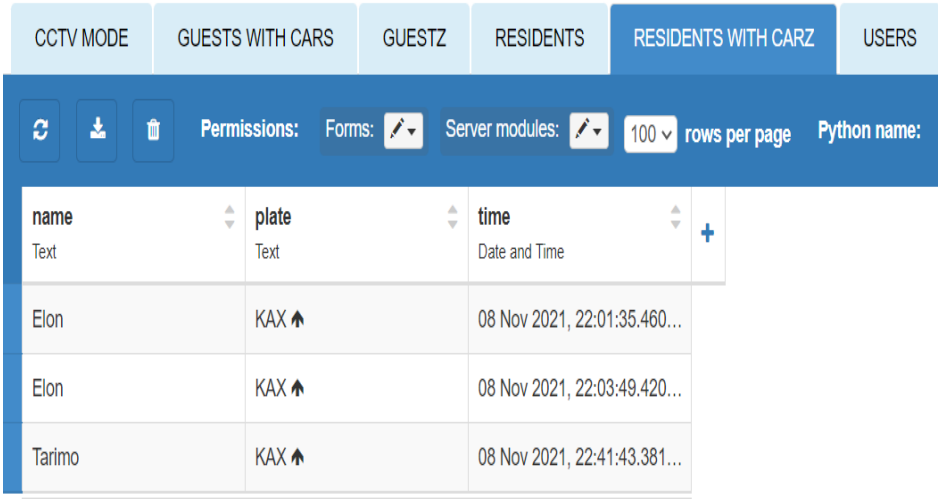


Figure 3. 23 Residents with cars data table

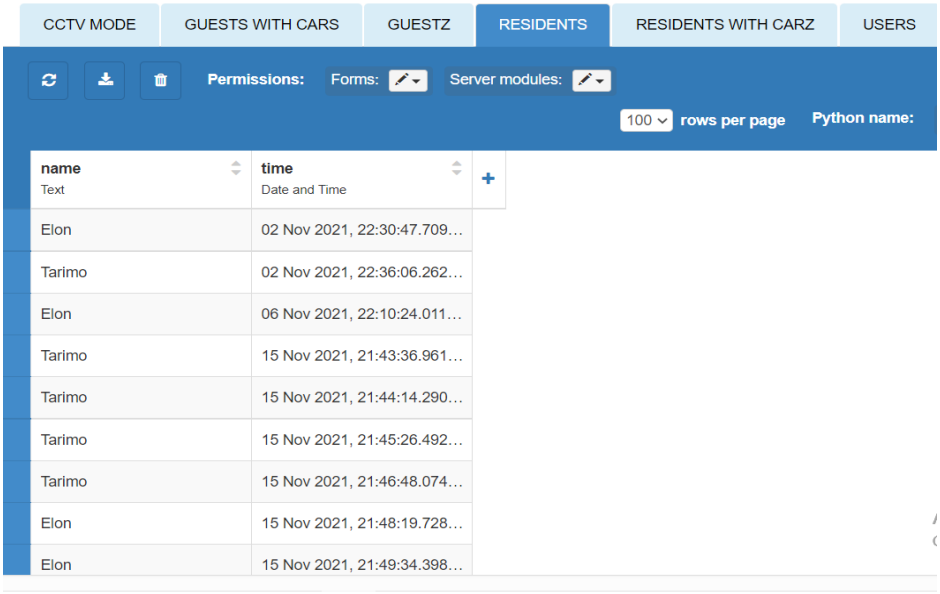


Figure 3. 24 Residents data table

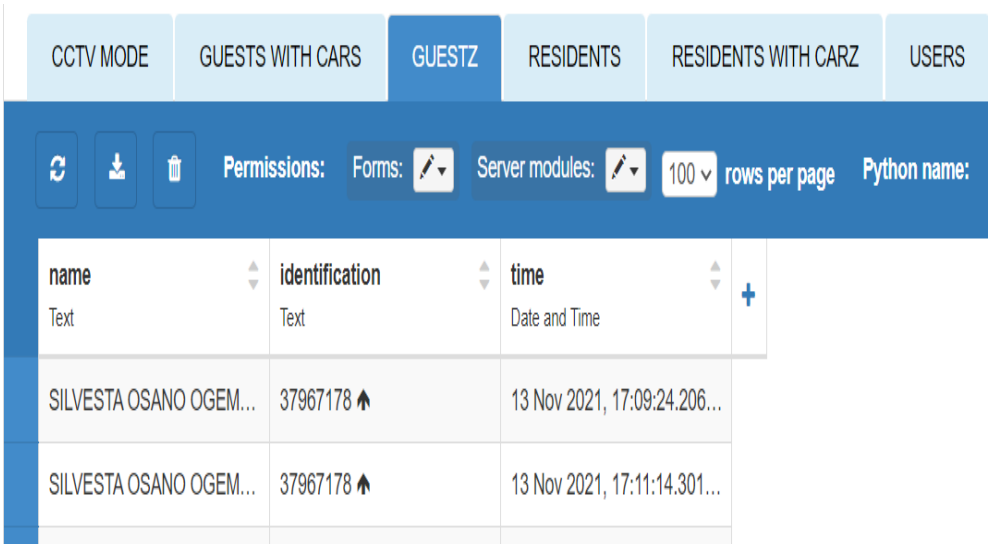


Figure 3. 25 Guests data table

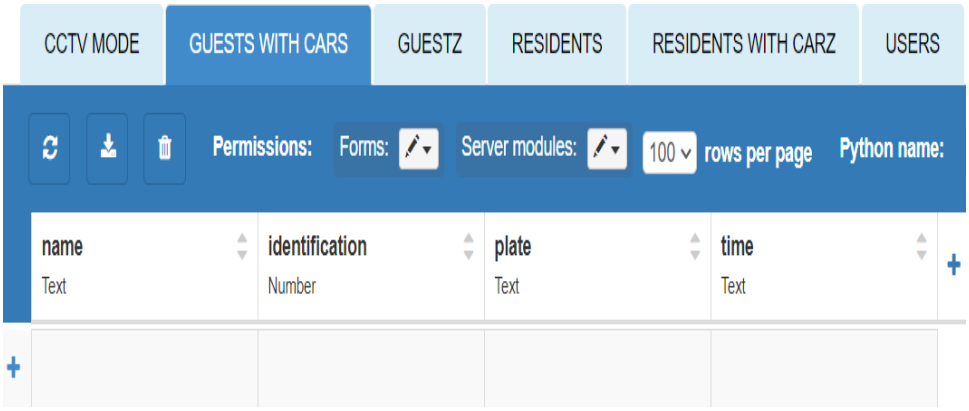


Figure 3. 26 Guests with cars data table

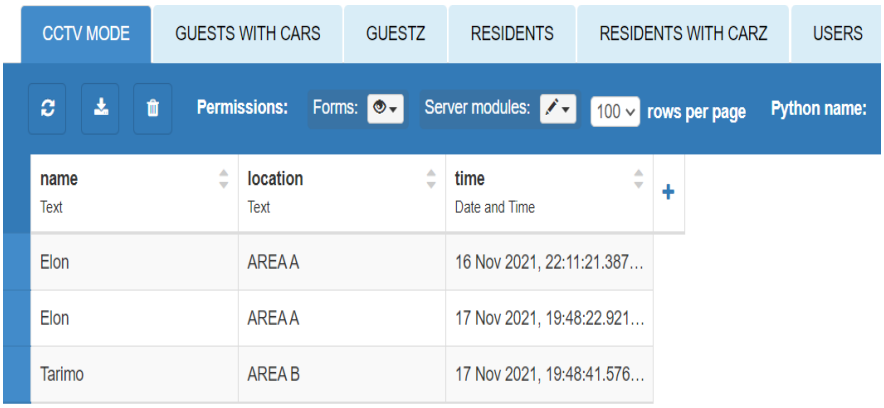


Figure 3. 27 CCTV Mode data table

Chapter 4 : RESULTS

4.1 FACIAL RECOGNITION

The facial recognition module was able to successfully interact with the web application to log in the user and store the resulting data in our database on the front end. The data was not only accurate but also easily queried at every level to return to precise data and filters added to search all parameters in our table. The image below shows the data as viewed by the user.

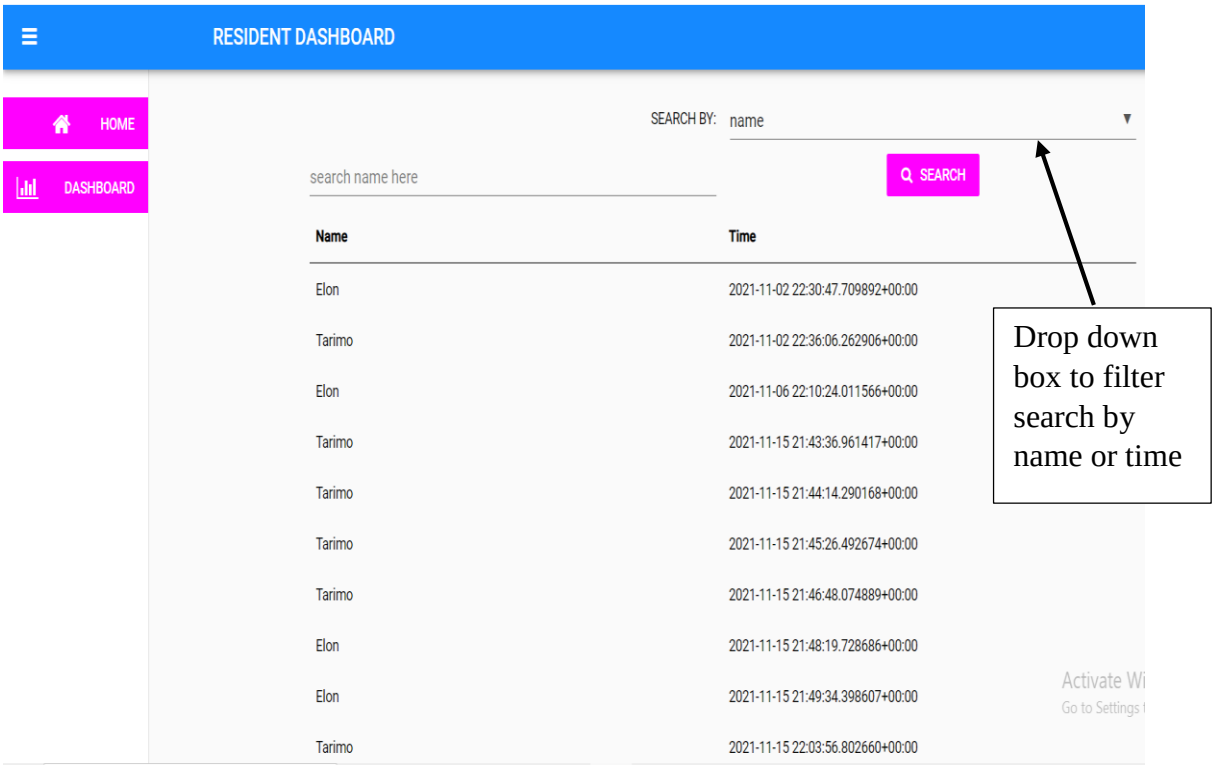


Figure 4. 1 Resulting Resident Data

The drop down box at the top right allows us to search data as per the chosen parameter and the user can type the value of the parameter in the text box provided at the top left. Links on the left panel can be used to change from viewing the data to logging new users in which aids in efficient navigation.

4.2 ID SCANNING

The ID scanning module was able to extract key details from an identity card, that is; the name of the individual as well as their identification number. These parameters as well as time

of entry were sent to the remote database and data captured was accurate and easily viewed via the “DASHBOARD” feature. The web application is able to filter data based on a search and all criteria, that is; name, identification number and time are able to be searched efficiently.

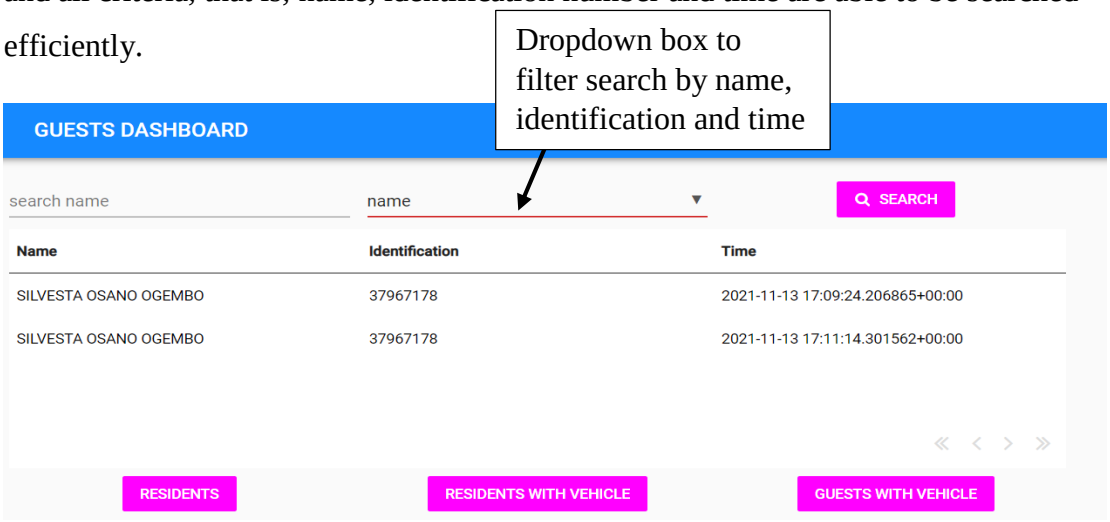


Figure 4. 2 Resulting Guest Dashboard

4.3 LICENSE PLATE SCANNING

The License plate recognition feature was successful in its design to identify and export the resulting text to our remote database and link the plate to an individual whether the individual is a guest or resident. A sample plate of “KAX” was used to test the module and the resulting data has been shown in the image below.

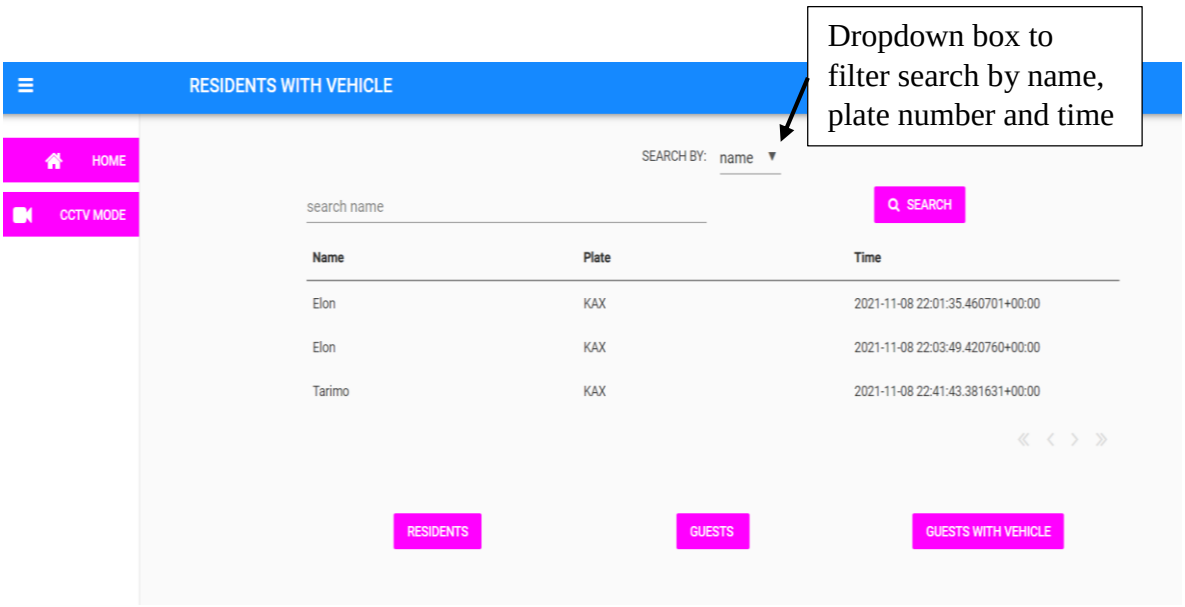


Figure 4. 3 Resulting Residents With Vehicle Dashboard

The similar concept of dropdown boxes to aid in data filtering has also been implemented allowing the user to interact with the data in a more controlled form allowing easy searches within the dashboard.

4.4 CCTV MODE

The resulting CCTV mode was implemented successfully and the two script designed were able to send the data remotely to our database indicating where the individual was recognized as well as the time they were recognized within that area. The data was easily viewed using one of the numerous dashboards and the resulting data has been shown below:

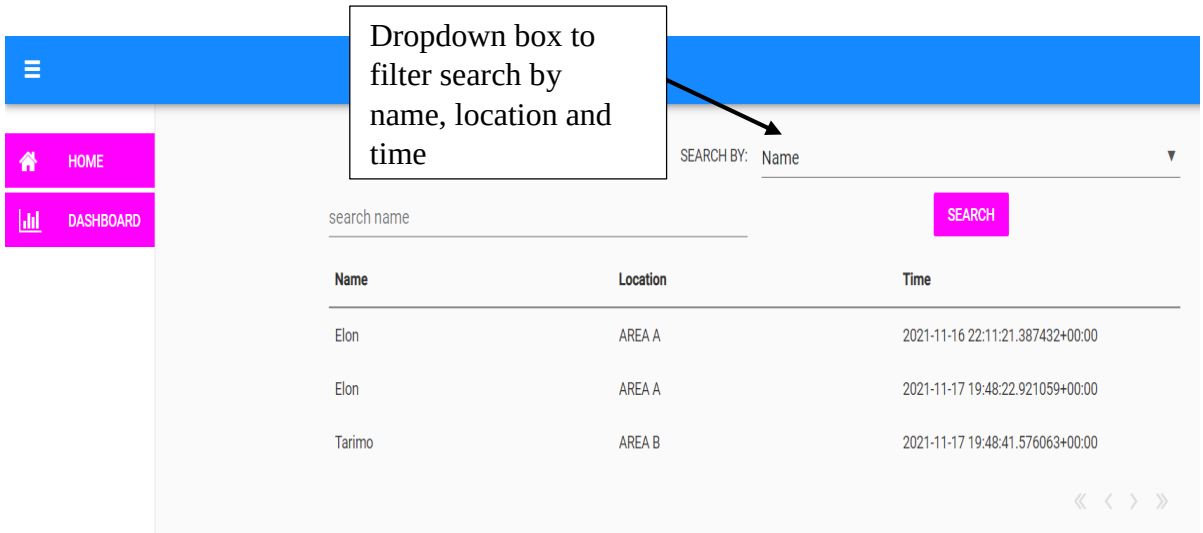


Figure 4. 4 Resulting CCTV Mode Dashboard

4.5 WEB APPLICATION

The web application successfully interacted with the services offered by the microcontroller and return values and data from the microcontroller to our front-end web application. The resulting web application has been shown above with the ability to collect and display our data effectively.

Chapter 5 : CONCLUSION

The smart gate controller was eventually a success with the laid specific objectives met in the time that was provided. The controller aimed to log users entering a facility using facial recognition for residents of that facility with the aid of a camera, identity card scanning for guests entering the facility and provide license plate recognition capabilities for both sets of users. Additionally, the system as a whole was able to offer CCTV capabilities embedded with a facial recognition feature to detect individuals in certain areas. Finally, the data collected from all these processes was able to be viewed through the web application and the functionality of the project controlled remotely using the same web application.

5.1 CHALLENGES FACED

1. Under low or bad lighting, the cameras were not able to perform efficiently and some errors were noted during testing.
2. The CCTV mode cannot run at the same time as the facial recognition, id scanning and license plate scanning due to limited number of cameras.
3. The cameras were only able to operate in 30 frames per second due to their quality affecting the speed of the cameras to catch frames.

5.2 RECOMMENDATIONS

1. The use of higher resolution cameras will reduce the errors in the system however this will come at a cost.
2. The increase of the number of cameras would allow all the modules to run concurrently.
3. Good lighting conditions must be imposed areas where cameras are accessible.

Chapter 6 : TIMEPLAN

ACTIVITIES							
Documentation							
Proposal Writing							
Proposal Presentation							
Design and Coding							
Hardware configuration and testing							
Final Report Writing							
Final Presentation							

Table 6. 1 Projected Time Plan

Chapter 7 : BUDGET

DESCRIPTION	QUANTITY	UNIT PRICE	TOTAL
Raspberry Pi 4-8GB RAM	1.00	12,900.00	12,900.00
USB Webcam	1.00	2,000.00	2,000.00
Connecting wires and zip ties	1.00	150.00	150.00
Raspberry Pi camera V2	1.00	3,800.00	3,800.00
Wooden Enclosure	1.00	300.00	300.00
Servo Motor	1.00	300.00	300.00
Infrared Sensor Module	1.00	200.00	200.00
		TOTAL	19,350

Table 7. 1 Estimated Budget

Chapter 8 : References

[1]. Ben Dickson, "What is computer vision", [Online]. Available: https://bdtechtalks.com/2019/01/14/what-is-computer-vision/. [Accessed: 02-June-2021]

[2]. Expert AI team, "What is Machine Learning", [Online]. Available: https://www.expert.ai/blog/machine-learning-definition/. [Accessed: 02-June-2021]

[3]. Yulia Gavrilova, "What is computer vision and how it works", [Online]. Available: https://serokell.io/blog/guide-to-computer-vision. [Accessed: 03-June-2021]

[4]. Muhammad Imran, "Computer Vision Vs Machine Learning", [Online]. Available: https://www.folio3.ai/blog/computer-vision-vs-machine-learning/. [Accessed: 03-June-2021]

[5]. Kaspersky Lab Team, "What is Facial Recognition – Definition and Explanation", [Online]. Available: https://www.kaspersky.com/resource-center/definitions/what-is-facial-recognition. [Accessed: 04-June-2021]

[6]. AnyConnect Academy Team, "Don't Take Facial Recognition at Face Value: Various Uses & Applications of Face Recognition Across Different Industries", [Online]. Available: https://anyconnect.com/blog/facial-recognition-applications/. [Accessed: 03-June-2021]

[7]. Vangie Beal, "OCR", [Online]. Available: https://www.webopedia.com/definitions/ocr/ . [Accessed: 05-June-2021]

[8]. Andreas Rivera, "How Does OCR work", [Online]. Available: https://www.efilecabinet.com/how-does-ocr-work/. [Accessed: 04-June-2021]

[9]. The European Commission, Official Website of the European Union, [Online]. Available: https://ec.europa.eu/home-affairs/what-we-do/networks/european_migration_network/glossary_search/automated-border-control-abc_en. [Accessed: 05-June-2021]

[10]. Technavio, “Three Pros and Cons of Airport e-Gates”, [Online]. Available: https://blog.technavio.com/blog/three-pros-and-cons-of-airport-e-gates. [Accessed 07-June-2021]

[11]. Singh, Rajesh & Gehlot, Anita & Prajapat, Mahesh & Singh, Bhupendra, (2021), “Computer Vision”. 10.1201/9781003245759-6.

[12]. Pan, Yue & Wu, Xianguo & Skibniewski, Miroslaw. (2021), “Computer Vision.”, 10.1007/978-981-16-2842-9_10.

[13]. Somogyi, Zoltán. (2021), “Face Recognition.”, 10.1007/978-3-030-60032-7_6.