VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI

Project Report on

TRACKING PERSON USING INTELLIGENT SURVEILLANCE

Submitted by

Abubakkar Momin	2JR18CS001
Shreya Hukkerikar	2JR19CS072
Xavier Dias	2JR19CS098
Yunus Ilkal	2JR20CS428

In partial fulfillment of the requirement for the award of the

Bachelor of Engineering

In

Computer Science & Engineering

Under the Guidance of

Dr. Pritam Dhumale

HOD & Associate Professor

Subject Code: 18CSP83

Jain College of Engineering & Research,Belagavi

Department of Computer Science & Engineering

Academic Year 2022 - 2023

Jain College of Engineering & Research, Belagavi

(Approved by AICTE , New Delhi ,Affiliated to VTU Belagavi & Recognized by Govt. of Karnataka)

Department of Computer Science & Engineering

CERTIFICATE

This is to certify that the report on **“TRACKING PERSON USING INTELLIGENT SURVEILLANCE”** is a bonafied work carried out by **Abubakkar Momin (2JR18CS001), Shreya Hukkerikar (2JR19CS072), Xavier Dias (2JR19CS098) and Yunus Ilkal (2JR20CS428)** in partial fulfillment of VIII semester, to award the degree in Computer Science & Engineering of the Visvesvaraya Technological University, it is witnessed that all corrections/suggestions indicated have been incorporated in the report.

The report has been approved as it satisfies all the academic requirements in respect of report as prescribed for the degree in engineering

Guide	**HOD**	**Principal**
(Dr. Pritam Dhumale)	**(Dr. Pritam Dhumale)**	**(Dr. S. V Gorabal)**

Name of External examiners	**Signature with date**

1.

2.

Declaration

We the members of the project team, studying in the VIII semester of Computer Science & Engineering, Jain College of Engineering and Research, hereby declare that the entire project entitled **"Tracking person using Intelligent Surveillance"** has been carried out by us independently under the guidance of **Dr. Pritam Dhumale** Department of Computer Science & Engineering, Jain College of Engineering and Research. This Project work is submitted to the Visvesvaraya Technological University, Belagavi, in partial fulfillment of the requirement for the award of the degree of **Bachelor of Engineering** in **Computer Science & Engineering.** This dissertation has not been submitted previously for the award of any other degree to any other institution or university.

Date:
Place: Belagavi

Name of the Students	USN	Signature
1. Abubakkar Momin	2JR18CS001	
2. Shreya Hukkerikar	2JR19CS072	
3. Xavier Dias	2JR19CS098	
4. Yunus Ilkal	2JR20CS428	

Acknowledgement

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people who made it possible and whose constant guidance and encouragement crowned our efforts with success. We consider our privilege to express the voice of gratitude and respect to all those who guided us and inspired us in completion of this project.

First and foremost, we ought to pay due regards to our renowned institution, which provided us a platform and an opportunity for carrying out this project work.

Our sincere gratitude to our beloved Principal **Dr. S. V. Gorabal,** for providing an ideal atmosphere to pursue my objectives under his able administration

We are also thankful to **Dr. Nandeesh Mathad,** Dean Academics and HOD, Mechanical Engineering Department, who has given valuable suggestions during the work along with his moral support and encouragement.

We are thankful to **Dr. Pritam Dhumale**, HOD, Dept. of Computer Science & Engineering, for motivating us and also allowing us to use the facilities of the department to complete this Project successfully.

We express our sincere thanks to **Prof. Veena B.M**. and **Prof. Abhilasha J**., Project Coordinator for having supported the work related to this Project. Their contributions and technical support in preparing this report are greatly acknowledged.

We express our sincere thanks to our guide **Dr. Pritam Dhumale**, HOD & Associate Professor, Dept. of Computer Science & Engineering for their constant co-operation, support, and invaluable suggestions

We would like to thank our parents for their constant moral support throughout the completion of this project. Finally, last but not the least, we would like to extend our deep sense of gratitude to our friends who always inspired us and encouraged us throughout the completion of this project.

Abubakkar Momin	2JR18CS001
Shreya Hukkerikar	2JR19CS072
Xavier Dias	2JR19CS098
Yunus Ilkal	2JR20CS428

ABSTRACT

Nowadays to find a missing person is very hard task, although we all are updated by social media, it requires numerous paper work to be done and it is time consuming process and also after doing this hard work, there are not much chances of proper result. This project gives out a system, which helps both police department and public by speeding up the process of searching using face recognition. Therefore, how this system works is that when the person goes missing the respective guardian of that person can upload the image, which then will get store in our database. Next is, the face recognition model in our system will find a match of that person in the database. If a match is found, it will be notified to the police and the guardian of that person.

Keywords: Surveillance, Accuracy, Location, Notification.

CONTENTS

Abstract		**i**
Declaration		**ii**
Acknowledgement		**iii**
List of Figures		**v**
Chapter 1	**Introduction**	**1-3**
1.1	Overview	1
1.2	Motivation	2
1.3	Yolo	2-3
Chapter -2	**Literature survey**	**4-9**
Chapter -3	**Problem statement**	**10-13**
3.1	Existing system	10
3.2	Disadvantages of existing system	10-11
3.3	Proposed system	11
3.4	Objective	12
Chapter -4	**System design**	**13-16**
Chapter -5	**Requirements**	**17-21**
5.1	Dependencies Required	17-20
5.2	Functional Requirements	20
5.3	Non Functional Requirements	20
5.4	Hardware Requirements	21
5.5	Software Requirements	21
Chapter -6	**Implementation**	**22-26**
6.1	Implementation Details	22
6.2	Algorithms	23
6.3	Face recognition function	24
6.4	Location function	25
6.5	WhatsApp function	25-26
6.6	Mail function	26
Chapter -7	**Test cases**	**27-28**
7.1	System test	27
7.2	Types of tests	27
7.2.1	Unit testing	27

7.2.2	Integration testing	27
7.2.3	Functional test	28
7.2.4	System Test	28
7.2.5	White Box Testing	28
Chapter- 8	Results	**29-31**
Conclusion		**32**
Future scope		**33**
References		**34**

List of Figures

Figure No.	Caption	Page No.
4.1	System Diagram	13
4.2	Basic layout of typical ML system having several stages	14
4.3	Elementary constituents of CNN	16
6.3	Face recognition function	24
6.4	Location function	25
6.5	WhatsApp function	25
6.6	Mail function	26
8.1	Home interface	29
8.2	Registration interface	29
8.3	Face recognition interface	30
8.4	Mail interface	30
8.5	Location CSV interface	31
8.6	Surveillance images zip	31
8.7	WhatsApp message interface	31

Chapter 1

INTRODUCTION

1.1 Overview

With roughly 100,000 individuals going missing every year and 25,000 remaining lost forever, the issue of missing persons has become increasingly prevalent in recent years, with a significant rise in the number of cases reported globally. In 2020, there were approximately 450,000 reported cases of missing persons worldwide, a figure that has now increased to over 1 million this year. This increase in missing persons is in part due to the growing population worldwide, and the numbers suggest that more individuals are becoming victims of this unfortunate circumstance each day.

India and Kenya are among the countries most affected by this issue, with a staggering 2,300 Indians and at least 20,000 Kenyans reported missing every day and year, respectively. Unfortunately, a significant portion of these cases remain unresolved, with only 30% of the reported victims being found within a reasonably short period of up to 3 months. In India alone, roughly 100,000 individuals go missing every year, with 25,000 remaining lost forever due to the country's vast expanse of rural terrain and the lack of infrastructure in terms of databases and tools for finding missing people.

In contrast, the United States reports over 600,000 cases of missing persons annually, with only 80,000 cases remaining unresolved within a year. However, the issue is a global one, and police and non-governmental organizations working with missing people worldwide have reviewed their policies and are planning to improve coordination of their work.

The reasons why people go missing are complex and varied, and the circumstances leading to adults or children becoming missing persons can be multi-layered. As such, it is crucial to develop innovative solutions that can help find missing persons quickly and safely.

The prototype solution developed for this purpose leverages the use of readily available mobile phone devices and the internet. This approach has shown success in the past, and the goal is to provide a fast and efficient solution to find missing persons, ensuring their safe and sound return to their families.

1.2 MOTIVATION

The issue of missing persons is a significant concern, not just in India but across the globe. The emotional and psychological impact on the family and friends of the missing person is immense, and the longer the person remains missing, the more devastating the consequences. In addition, there are often legal implications, such as the need to file an FIR, which can be a lengthy and cumbersome process. The lack of manpower and resources dedicated to finding missing persons in many countries further exacerbates the problem.

In India, the problem is particularly acute. There are several reasons for this, including the vast size of the country, the lack of proper infrastructure, and the absence of a centralized database for tracking missing persons. The absence of a national repository for missing children means that information is not shared between different states, making it more challenging to locate missing children. The lack of budgetary allocation for tracking missing people is another issue that needs to be addressed.

One solution to this problem is the use of technology, specifically mobile phone devices and the internet. By leveraging these readily available tools, it is possible to create a platform that can help in the search for missing persons. The development of such a platform can significantly reduce the time taken to locate missing persons and increase the chances of their safe return.

Overall, the need for an effective and efficient system for finding missing persons is clear. With the right tools and resources, it is possible to improve the chances of locating missing persons and provide some relief to the families and friends who are anxiously waiting for their loved ones' safe return.

1.3 YOLO

YOLO (You Only Look Once) is a real-time object detection system that can be used for face recognition. YOLO is an algorithm that uses convolutional neural networks (CNNs) to identify objects and their location in an image or video frame. It was originally designed for object detection, but has been adapted for face recognition by some researchers.

In face recognition using YOLO, the algorithm is trained to detect faces in images or video frames. It works by dividing the image into a grid of smaller regions, and each region is then analyzed to detect the presence of a face. YOLO uses a single neural network to predict bounding boxes and class probabilities directly from full images in one evaluation, which is why it is so fast and efficient.

YOLO has been found to be effective in recognizing faces in real-time, even in complex and cluttered environments. However, it may not be as accurate as other, more complex face recognition algorithms, as it relies heavily on the quality of the input image and the training data used to train the algorithm. Nonetheless, YOLO has shown promise in certain face recognition applications and continues to be an area of active research.

In addition to its efficiency, one of the advantages of using YOLO for face recognition is its ability to detect multiple faces in a single image or video frame, making it ideal for crowd surveillance applications. Furthermore, YOLO has been successfully used in various real-world applications, such as tracking pedestrians in traffic scenes, detecting and tracking vehicles in surveillance footage, and even detecting and tracking animals in wildlife monitoring scenarios.

However, like any machine learning algorithm, YOLO has its limitations. One of the biggest challenges faced in using YOLO for face recognition is the need for a large and diverse training dataset to ensure accurate detection of faces in different conditions, such as varying lighting, facial expressions, and occlusions. Additionally, YOLO's speed and efficiency come at the cost of some loss of accuracy compared to more complex algorithms, which may be a concern in certain applications that require high precision face recognition. None the less, YOLO remains a popular choice for face recognition tasks and is continually being improved and refined by researchers in the field.

Chapter 2

LITERATURE SURVEY

[1] “Finding missing persons using ML”, by Neha Gholape, Ashish Gour, Shivam Mourya, 2023.

The system utilizes a combination of machine learning (ML), deep learning (DL), and artificial intelligence (AI) technologies to create a dynamic and robust solution for finding missing persons. The system is designed to have a high prediction outcome probability, even when faced with changes in the image or lack of proper information. This is achieved by training the system on a large dataset of facial images and using a deep neural network (DNN) to learn the features that are essential for identifying faces. The system is also able to adapt to changes in the facial appearance of the missing person, such as changes in hairstyle or facial hair. The proposed system has the potential to significantly contribute to the field of missing persons' search and rescue efforts. By providing a reliable and efficient way to identify and locate missing individuals, the system could help to reduce the number of unresolved missing person’s cases. Additionally, the system could provide a valuable tool for law enforcement agencies and non-governmental organizations working on missing person’s cases. Overall, the proposed system represents a promising application of ML, DL, and AI technologies in the field of missing persons' search and rescue.

[2] “Real Time Face Detection using CCTV Camera", by Nirmal Kumar Saraswat, Arvinda Kushwaha, 2022.

The system described in the paper "Real Time Face Detection using CCTV Camera" is designed to operate using CCTV cameras, which are widely used for security purposes. The system uses smart CCTV technology to incorporate facial recognition capabilities, which allows it to filter out Persons of Interest (POI) from human face images in real-time. This system has the potential to improve security measures by providing real-time monitoring and identification of individuals through facial recognition, making it an invaluable tool in surveillance and security applications.

The system operates in real-time, allowing for quick and efficient identification of individuals of interest. It is capable of detecting and tracking multiple faces simultaneously, making it a useful tool in busy environments such as airports, train stations, and shopping malls. The system is designed to recognize faces in various lighting conditions, making it effective in low-light and night-time environments. One potential concern with the system is that it could potentially infringe on privacy rights if not used responsibly. However, if implemented correctly and within the bounds of local laws and regulations, the system has the potential to greatly enhance security and surveillance capabilities in various settings. Overall, the system described in the paper has the potential to make significant contributions to the field of security and surveillance.

[3] "Crime Reporter and Missing Person Finder", by Pranay Thali, Dhiraj Mujmule, Ankit Bhingarde, Prof. Yogesh Salve, 2021.

The focus of their project was on designing an intuitive and user-friendly interface for adding new complaints from users. They employed Agile software development methodologies to ensure a streamlined and iterative approach to application development. The team prioritized user experience and ease of use in the design of the complaint submission process, aiming to make it simple and efficient for users to report crimes or missing persons. By leveraging Agile development practices, the team was able to continuously improve and refine the application's interface based on user feedback, resulting in a user-friendly and efficient system for reporting complaints related to crime and missing persons. In addition to the user-friendly interface for adding new complaints, the project also aimed to streamline the process of handling and managing complaints related to crime and missing persons. The team used a centralized database to store and manage all complaints, allowing law enforcement agencies to easily access and track each case. Overall, the project aimed to provide a comprehensive and efficient solution for reporting and managing complaints related to crime and missing persons, with a focus on user experience and continuous improvement.

[4] "CCTV-Based Surveillance System with Face Recognition Feature" by Michael Bryan P. Lumaban, Giged T. Battung, 2021.

The study focused on exploring the application of three face recognition algorithms: Local Binary Pattern Histogram (LBPH), Eigenface, and Fisherface. The results revealed that the LBPH algorithm achieved the highest face recognition rate, with an accuracy rate of 95.92%. This indicates that the LBPH algorithm is highly effective in accurately recognizing faces in the CCTV-based surveillance system. The findings of this study contribute to the field of face recognition technology and highlight the potential of the LBPH algorithm for real-world applications, such as in surveillance systems for improved security and identification purposes. The research conducted by Michael Bryan P. Lumaban and Giged T. Battung provides valuable insights for the development and implementation of effective face recognition algorithms in CCTV-based surveillance systems.

[5] “Face Modeling Process Based on Dlib” by Junhang ding, jinna sun, 2021.

The researchers were focused on developing a face modeling process using the Dlib library and Random Forest algorithm. The goal was to accurately model facial features using this algorithm and create a stable and efficient system. However, they found that the Random Forest algorithm was not suitable for this task, as it was too time-consuming and unstable. This instability was due to the fact that even small changes in data could le ad to changes in the structure of the regression tree, making it difficult to maintain consistency in the model.The limitations of the Random Forest algorithm in face modeling based on Dlib highlight the importance of choosing the right algorithm for the task at hand. In this case, the researchers identified the need for further research and optimization to find a more efficient and robust algorithm for face modeling applications. This could involve exploring other machine learning algorithms or optimizing the parameters of the Random Forest algorithm to improve its stability and performance. Overall, this study provides important insights into the challenges of using machine learning algorithms for face modeling and the need for continued research and development to improve their effectiveness and efficiency.

[6] “Missing Infant Finder Android Application” is a mobile application developed by Vivek C Shet and Mr. Srinivasulu M in 2021.

The application is designed to assist in locating missing infants by using an Android mobile device to capture and submit pictures of infants to the application. The application uses face detection technology, specifically the SWF-SIFT technique, to analyze the submitted pictures and provide results on whether the infant in the picture matches any missing infant profiles in the database. SWF-SIFT (Scale-Invariant Feature Transform with Spatial Weighted Filtering) is an image processing technique that is used to detect and match features in images, such as faces. The technique is based on identifying and extracting key points in an image and then using these points to generate a unique signature for the image. This signature can then be used to match the image against a database of known images. The SWF-SIFT face detection technology, the Missing Infant Finder Android Application also allows users to register missing infants in the application's database. This feature helps to increase the chances of locating missing infants by providing a centralized location for information on missing infants. The application also provides a search feature that allows users to search for missing infants by various criteria, such as name, location, or age. Furthermore, the Missing Infant Finder Android Application incorporates a user-friendly interface, making it easy for anyone to use. The application's interface is designed to be simple and intuitive, making it accessible to a wide range of users. The developers also aimed to make the application lightweight and efficient, allowing it to be run on low-end Android devices. Overall, the Missing Infant Finder Android Application provides a valuable tool for locating missing infants and demonstrates the potential of using mobile technology to aid in missing persons' search and rescue efforts.

[7] “Android Based Application - Missing Person Finder” is a mobile application developed by Birari Hetal, Sanyashiv Rakesh, Porje Rohan, and Salve Harish in 2020.

The application is designed to help locate missing persons using an Android mobile device.The application uses various technologies such as GPS, Google Maps, and facial recognition to help locate missing persons. The GPS and Google Maps functionality are used to track the user's location and the location of the missing person, while facial recognition technology is used to match the missing person's photo with photos of individuals in the area. Additionally, facial recognition technology is used to match the missing person's photo with photos of individuals in the area. The application's use of facial recognition technology enables the user to capture photos of people in the vicinity and compare them with the missing person's photo to find potential matches. This mobile application is designed to be user-friendly and efficient, making it a valuable tool in missing person search and rescue efforts. The development team comprised of Birari Hetal, Sanyashiv Rakesh, Porje Rohan, and Salve Harish worked together to create a mobile application that could be easily used by anyone with an Android device.

[8] “Notification System Based on Face Detection and Recognition” is a study conducted by Abdullah Ahmad Basuhail and Ahmed AbdulQadir Al-bakeri in 2020.

The study aimed to develop a notification system using face detection and recognition to alert house owners of visitors to their home. The Notification System Based on Face Detection and Recognition is a security system designed to provide homeowners with real-time notification of visitors to their homes. The system uses advanced face detection and recognition technology to identify the faces of visitors and send an email notification to the homeowner. The technology uses a camera to capture an image of the visitor's face, which is then analyzed by the system to determine whether the visitor is recognized or unknown. If the visitor is recognized, the system sends an email notification to the homeowner, providing details about the visitor, such as their name and phone number. This allows the homeowner to quickly and easily identify who is at their door and decide whether to allow them entry. The use of SMTP protocol for email notification ensures that the notifications are delivered in a timely and efficient manner. This system can enhance the security of homes and provide homeowners with peace of mind.

[9] "Face recognition by fusion of local and global matching scores using DS theory" is a study conducted by Kisku D R in 2019.

In the study, the researcher proposed a new method for face recognition by combining both local and global features using the Dempster-Shafer (DS) theory. The DS theory is a mathematical framework for reasoning with uncertainty and has been used in various applications such as image recognition and decision-making systems.The proposed method aims to improve the accuracy of face recognition by capturing both local and global features of a face. The local features are extracted from specific facial landmarks such as eyes, nose, and mouth, while the global features capture the overall structure and shape of the face.The study conducted experiments on several publicly available face databases to evaluate the performance of the proposed method. The results showed that the proposed method outperformed existing methods that only used either local or global features. The use of DS theory to combine local and global features helped to overcome the limitations of individual feature-based approaches and provided a more accurate and robust face recognition system.

[10] "Property Based testing of JSON Based Web Service" is a study conducted by Fredlund Benac Earle in 2019.

The study aimed to use property-based testing to test the properties of a JSON-based web service. Property-based testing is a software testing technique where the properties of a program are tested against a set of inputs, with the goal of ensuring that the program behaves correctly under a variety of conditions.In the case of the JSON-based web service, the study focused on testing the properties of the service's input and output data. Specifically, the study used the Hypothesis library, a Python library for property-based testing, to generate a large number of input and output JSON data samples, and then tested the service against those samples to ensure that it correctly handled a range of input and output scenarios. The study found that property-based testing was an effective way to test the JSON-based web service, and was able to uncover a number of bugs and issues that would have been difficult to find using traditional testing methods. The study suggests that property-based testing may be a useful technique for testing other types of web services and software systems.

Chapter 3

PROBLEM STATEMENT

3.1 EXISTING SYSTEM

Traditionally, surveillance systems for tracking persons have relied on video cameras and human operators to manually monitor the footage in real-time. This method can be time-consuming and prone to human error, as operators may miss important details or become fatigued over long periods of monitoring.

To address these issues, some traditional systems have incorporated motion detection technology to trigger alerts when movement is detected within the camera's field of view. However, this method may generate false alarms if the camera is positioned in an area with high levels of activity, such as a crowded street or public transportation hub.

Another traditional approach is the use of RFID (Radio Frequency Identification) technology, which involves attaching RFID tags to individuals and tracking their movements using RFID readers installed in strategic locations. However, this method requires individuals to wear or carry the RFID tags, which may not be practical or desirable in all situations.

Overall, traditional surveillance systems for tracking persons have limitations in terms of accuracy, efficiency, and privacy concerns. They may also require significant investment in hardware and infrastructure to be effective.

3.2 DISADVANTAGES OF EXISTING SYSTEM

There are several disadvantages to existing systems for tracking persons using intelligent surveillance, including:

1. Privacy concerns: The use of intelligent surveillance for tracking persons raises serious privacy concerns, as it involves the collection and storage of personal data.

2. High cost: The installation and maintenance of intelligent surveillance systems can be expensive, particularly for large areas.

3. Limited accuracy: Existing systems for tracking persons using intelligent surveillance may not be accurate in all situations, particularly in crowded areas where there are many people.

4. Vulnerability to hacking: Intelligent surveillance systems are vulnerable to hacking, which could compromise the security and privacy of the system and its users.

5. Dependence on lighting conditions: Existing systems may be affected by changes in lighting conditions, such as changes in the intensity and direction of sunlight.

6. Limitations in identifying persons: Existing systems may have difficulty identifying persons based on their physical appearance alone, particularly if they are wearing masks or other items that obscure their faces.

7. Invasive nature: The use of intelligent surveillance systems for tracking persons can be perceived as invasive and may lead to feelings of discomfort or distrust among members of the public.

3.3 PROPOSED SYSTEM

The proposed system for tracking a person using intelligent surveillance involves collecting data from various sensors and cameras while ensuring the security and privacy of individuals. The collected data undergoes feature extraction using robust algorithms to account for environmental factors, and non-facial features are also extracted to improve identification accuracy. A CNN classifier is trained on a large dataset of known individuals to improve its accuracy and matching capabilities. Real-time location tracking is used to track the person's movements, and predictive modeling techniques are employed to anticipate where they might go next. The database of known individuals is updated regularly with relevant information such as criminal history and medical records. In case of a positive match, the relevant authorities or relatives are notified promptly while maintaining privacy concerns and legal implications. This proposed system addresses the limitations of traditional systems by improving identification accuracy, ensuring security and privacy, and enabling effective response in emergency situations.

3.4 OBJECTIVE

- **Display Information about missing person**

  Display Information about missing person: This refers to displaying information about a missing person to the public. It could include details such as the person's name, age, gender, height, weight, last known location, and any distinguishing features. This information is typically displayed on public notice boards, social media platforms, and other mediums to help in the search for the missing person.

- **Adding new complaint**

  This refers to the process of adding a new complaint or report about a missing person. This could be done by a concerned individual, a family member or a law enforcement agency. The complaint typically includes details such as the person's name, age, last known location, and any other relevant information. This helps to create a record of the missing person and assists in their search.

- **Removing Complaints**

  This refers to the process of removing a complaint or report about a missing person. This could be done if the person has been found or if there is a mistake in the information provided. Removing the complaint helps to avoid confusion and ensures that the information displayed is accurate.

- **Searching person by particular attribute such as name, location**

  This refers to the ability to search for a missing person based on specific attributes such as their name, location, age, or any other relevant information. This feature is useful for law enforcement agencies and concerned individuals who are searching for a missing person.

- **Notification Portal**

  This refers to a platform that provides notifications and updates about missing persons. It could include information such as the latest sightings, updates on the search, and any other relevant information. This helps to keep the public informed and engaged in the search for the missing person.

Chapter 4

SYSTEM DESIGN

The system design for the proposed intelligent surveillance system involves the integration of various components such as data collection, feature extraction, CNN classifier, location tracking, database management, and notification protocols. Data collection involves identifying the sources and locations of sensors and cameras and ensuring that the collected data is of high quality and resolution.

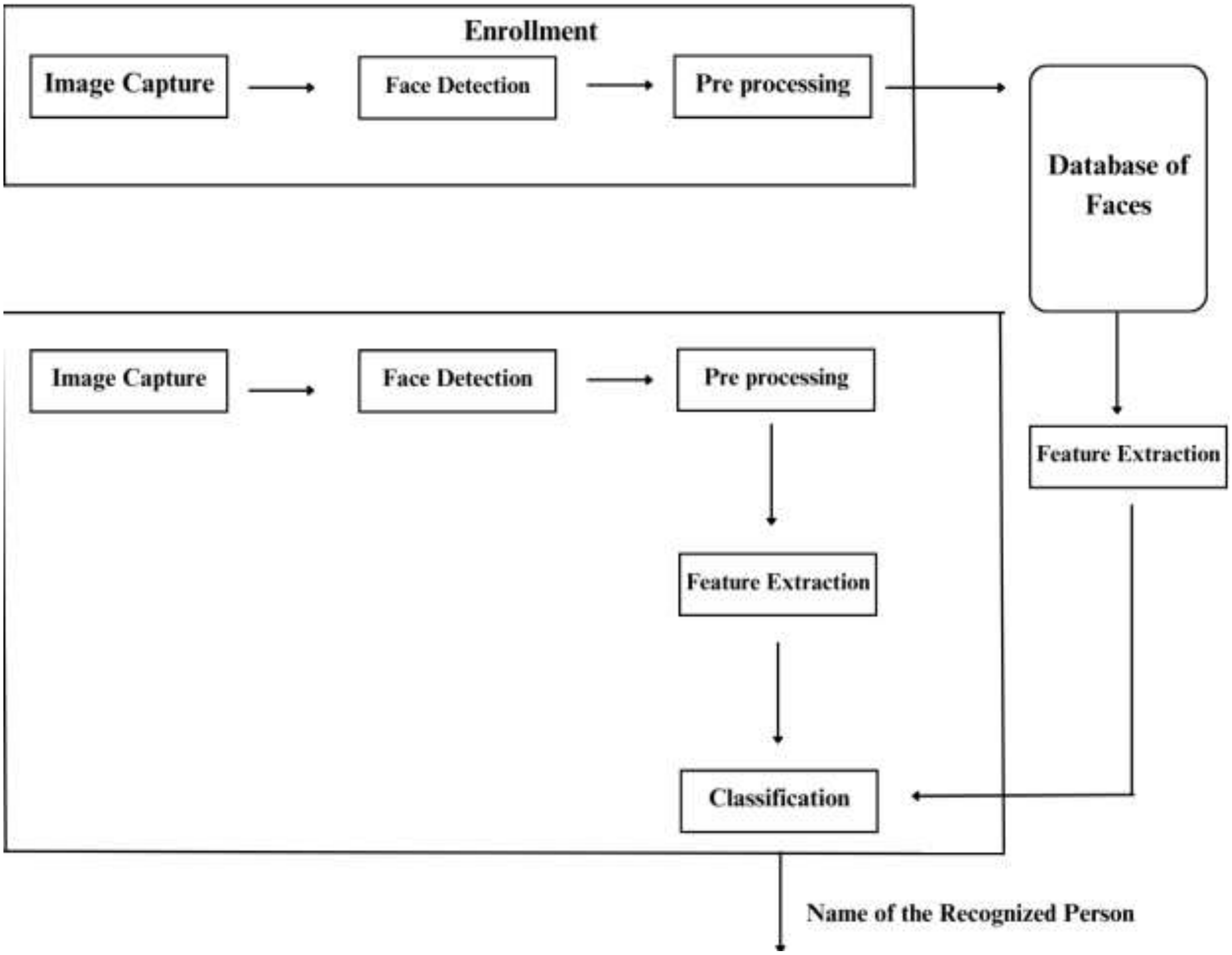


Fig 4.1 System Diagram for Tracking Person

Feature extraction is performed using algorithms such as PCA or LBP to extract relevant features from the data, including non-facial features such as body posture and clothing. The CNN classifier is trained on a large dataset of known individuals to improve its accuracy, and real-time location tracking is used to track the person's movements. The database of known individuals is updated regularly, and relevant information is extracted from it, such as criminal history and medical records. In case of a positive match, the relevant authorities or relatives are notified promptly while maintaining privacy concerns and legal implications. Overall, the system design is aimed at improving the accuracy and reliability of person tracking using intelligent surveillance while considering privacy concerns and ethical implications.

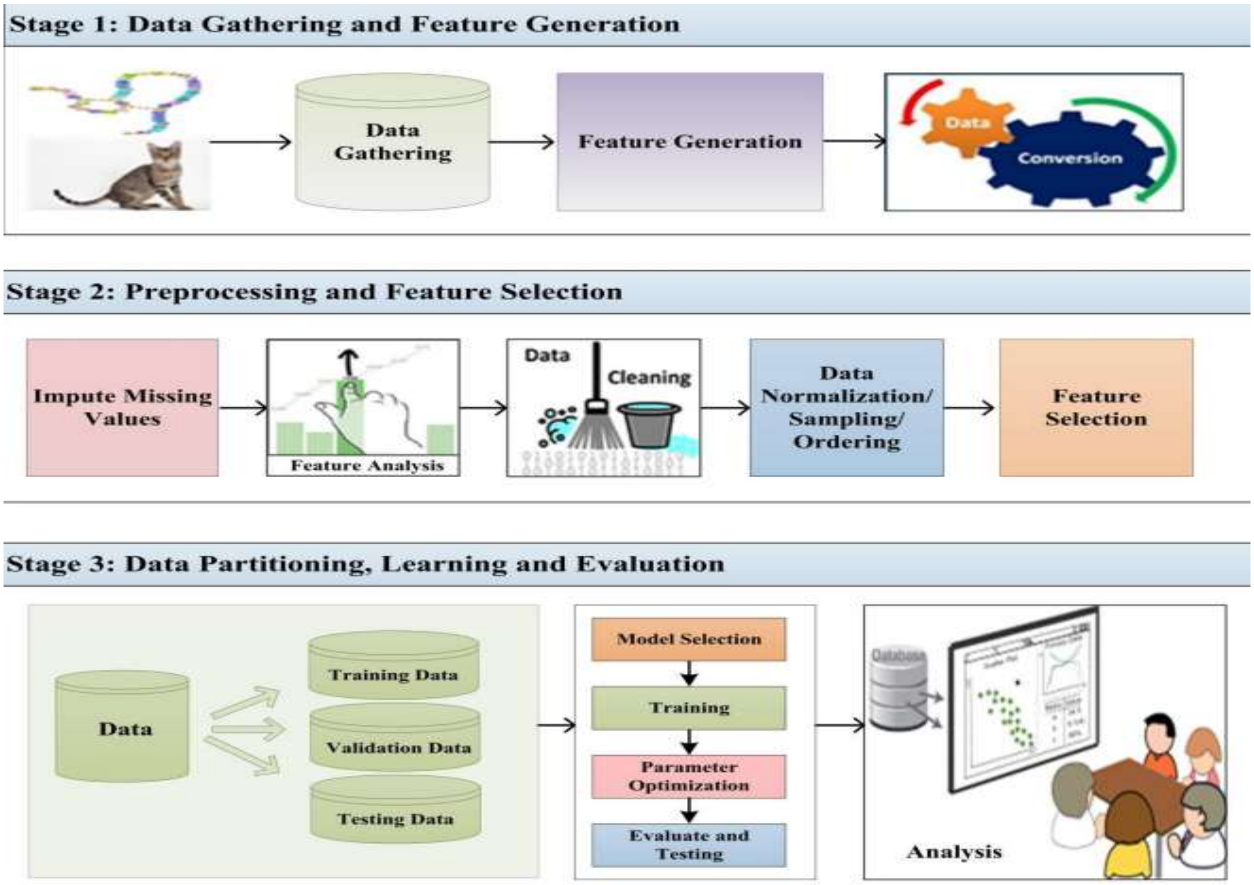


Fig 4.2 Basic layout of a typical ML system having several stages.

- **Data collection**

  Identify the sources and locations of sensors and cameras that will be used for data collection. Ensure that the data collected is of high quality and resolution to enable accurate tracking and identification. Consider the privacy concerns of individuals who may be captured by the sensors, and take measures to ensure data is collected and stored securely.

- **Feature extraction**

  Use algorithms such as Principal Component Analysis (PCA) or Local Binary Patterns (LBP) to extract relevant features from the data. Ensure that feature extraction is robust to changes in lighting, pose, and other environmental factors. Extract non-facial features such as body posture, gait, or clothing to improve identification accuracy.

- **CNN classifier**

  Train the CNN classifier on a large dataset of known individuals to improve its accuracy and ability to match features with a database. Consider using transfer learning techniques to improve classifier performance and reduce training time. Evaluate the classifier's performance using metrics such as accuracy, precision, and recall.

- **Track location**

  Use GPS coordinates or other location-based data to track the person's movements in real-time. Consider using predictive modeling techniques to anticipate where the person is likely to go next based on their previous movements and patterns. Ensure that the location tracking is accurate and reliable to enable effective response in case of emergency situations.

- **Extract details from database**

  Maintain a database of known individuals that can be used to match against identified features. Ensure that the database is up-to-date and accurate to improve the identification process. Extract relevant information from the database such as criminal history, medical records, and other identifying information.

- **Notify authorities or relatives**

  Establish protocols and procedures for notifying relevant authorities or relatives in case of a positive match. Ensure that the notification process is fast and efficient to enable timely response and action. Consider privacy concerns and legal implications when sharing information about identified individuals.

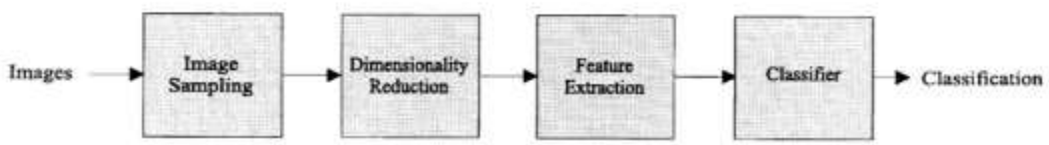


Fig 4.3 Elementary Constituents of CNN

The elementary constituents of a Convolutional Neural Network (CNN) include convolutional layers, pooling layers, activation functions, fully connected layers, and loss functions. Convolutional layers are the main building blocks of a CNN and consist of a set of filters that slide over the input image and perform convolution operations to extract relevant features. Pooling layers down sample the feature maps generated by convolutional layers to reduce the dimensionality of the data and enable the network to focus on more salient features. Activation functions such as ReLU (Rectified Linear Unit) and sigmoid are used to introduce non-linearity into the network and enable it to learn complex relationships between the input and output. Fully connected layers connect every neuron in one layer to every neuron in the next layer and are used to perform the final classification or regression task. Loss functions measure the difference between the predicted output and the true output and are used to optimize the network during training. Together, these elementary constituents allow CNNs to perform image classification, object detection, segmentation, and other computer vision tasks with high accuracy and efficiency.

Chapter 5

REQUIREMENTS

5.1 Dependencies Required

CSV

CSV stands for "Comma-Separated Values". It is a module in Python that allows reading and writing data in CSV format, which is a common file format for storing tabular data. The CSV module provides methods to parse CSV files into Python data structures, such as lists or dictionaries, and to write data from Python to CSV files**.**

Zipfile

The Zipfile module in Python allows you to work with ZIP files, which are compressed archives that can contain one or more files. It provides methods to create, extract, and manipulate ZIP files, as well as to read metadata and contents of the files inside the ZIP archive.

BeautifulSoup

BeautifulSoup is a popular Python library for web scraping, which is the process of extracting data from websites. It provides methods to parse HTML or XML documents and navigate their structure, allowing you to extract data from specific HTML tags, attributes, or classes.

Cv2

Cv2 is a Python library for computer vision tasks, which is the process of analyzing and interpreting visual information from the world, usually from images or videos. Cv2 provides a wide range of functions for tasks such as image processing, object detection, facial recognition, and more.

Idna

The Idna library is used for handling Internationalized Domain Names (IDNAs) in Python. It provides methods to convert domain names containing non-ASCII characters to ASCII, which is the standard format for domain names, and vice versa.

Numpy

Numpy is a powerful numerical computing library for Python. It provides support for working with large, multi-dimensional arrays and matrices, along with a collection of mathematical functions for performing operations on these arrays efficiently. Numpy is widely used in scientific computing, data analysis, and machine learning.

Face_recognition

Face_recognition is a Python library for facial recognition tasks. It provides methods for face detection, facial feature extraction, and face recognition using deep learning algorithms. It allows you to recognize faces in images or videos and perform tasks such as face detection, face verification, and face identification.

Os

The Os module is a Python module that provides a way to interact with the operating system, such as reading and writing files, creating and deleting directories, and executing system commands. It allows you to perform various file and directory operations in a cross-platform way, making it useful for tasks such as file management, system administration, and automation.

Datetime

The Datetime module in Python provides classes for working with dates, times, and time intervals. It allows you to create, manipulate, and format dates and times, and perform operations such as calculating time differences, formatting dates and times, and converting between different time zones.

Mime

The Mime module in Python provides functions for working with Multipurpose Internet Mail Extensions (MIME) types, which are a standard way of specifying the format of email messages and attachments. It provides methods for parsing MIME types, handling headers, encoding and decoding attachments, and more.

Email

The Email module is a Python module for handling email messages. It provides classes and methods for creating, sending, and parsing email messages, including support for common email protocols such as SMTP and POP3. It allows you to send and receive emails programmatically and perform tasks such as email parsing, handling attachments, and managing email headers.

Pywhatkit

Pywhatkit is a Python library that provides a range of utilities for automating various tasks, such as sending WhatsApp messages, performing web searches, sending emails, converting text to speech, and more. It provides simple APIs for performing these tasks using Python code, making it useful for automation and scripting purposes.

Requests

The Requests library is a popular Python library for making HTTP requests. It provides a simple and convenient way to send HTTP requests and handle responses, allowing you to interact with web services, fetch data from APIs, and perform tasks such as downloading files, handling cookies, and handling authentication.

Time

The Time module in Python provides functions for working with time-related operations, such as measuring time intervals, formatting time values, and converting between different time representations. It is useful for tasks that involve time-based operations, such as scheduling, benchmarking, and performance measurement.

Thinker

It seems that "Thinker" might refer to a specific library or module that you are referring to. However, as of my knowledge cutoff date in September 2021, I am not aware of any widely used Python library or module with that name. If you have more specific information about the "Thinker" library or module, please provide more details so I can provide accurate information.

Shutil

The Shutil module in Python provides functions for file and directory operations, such as copying, moving, and deleting files and directories. It also provides functions for working with file permissions, archiving files, and more. It is useful for tasks that involve file and directory manipulation, such as file management, backup and restore, and data organization.

PIL

PIL stands for "Python Imaging Library", which is a deprecated Python library for working with images. It has been superseded by the Pillow library, which is a more actively maintained fork of PIL. PIL provides functions for image processing tasks such as opening, manipulating, and saving images in various formats, as well as performing operations such as resizing, filtering, and color manipulation.

5.2 Functional Requirements

1. The system should be able to detect and track people using intelligent surveillance techniques.
2. The system should be able to capture images or videos of the person being tracked and store them for future reference.
3. The system should be able to identify and distinguish between different individuals in a crowded environment.
4. The system should be able to provide real-time updates on the location and movements of the person being tracked.
5. The system should be able to send alerts or notifications when the person being tracked moves outside a defined area.

5.3 Non-Functional Requirements

1. The system should be highly accurate in detecting and tracking individuals.
2. The system should be able to operate in real-time with minimal latency.
3. The system should be able to handle large volumes of data and process it efficiently.
4. The system should be able to work in different lighting conditions and weather conditions.
5. The system should be user-friendly and easy to use for the intended audience.

5.4 Hardware Requirements

1. Laptop / Computer with Core i5 8th gen
2. 2GB Hard Disk Storage
3. RAM above 512 Mb
4. ROM 4Gb

5.5 Software Requirements

1. Operating System :   Windows 8+
2. Server side Script: Python 3.7+
3. IDE   :   PyCharm/ visual studio code

Chapter 6
IMPLEMENTATION

6.1 Implementation Details

The implementation of a person tracking system using intelligent surveillance involves several components and processes. First, data is collected from various sensors and cameras located in different areas. The collected data undergoes feature extraction using robust algorithms to account for environmental factors such as lighting and pose. Non-facial features are also extracted, such as body posture and clothing, to improve identification accuracy. The extracted features are then used to train a Convolutional Neural Network (CNN) classifier on a large dataset of known individuals to improve its accuracy and matching capabilities. In addition, real-time location tracking using GPS coordinates or other location-based data is used to track the person's movements, and predictive modeling techniques are employed to anticipate where they might go next. The system also includes a database of known individuals that is regularly updated, and relevant information is extracted from it, such as criminal history and medical records. In case of a positive match, the relevant authorities or relatives are notified promptly while maintaining privacy concerns and legal implications. The system design includes various components such as the sensors and cameras, the feature extraction algorithms, the CNN classifier, the real-time location tracking system, and the database management system. The system must be designed to ensure the security and privacy of individuals captured by the sensors and cameras, and appropriate measures must be taken to ensure data is collected and stored securely. Overall, the implementation of a person tracking system using intelligent surveillance involves a combination of advanced technologies such as machine learning, computer vision, and real-time location tracking. It has the potential to significantly improve public safety and security, but it must be designed and implemented carefully to address privacy concerns and legal implications. The CNNs used in your project are deep learning models that have been trained on a large dataset of images to learn various features of the human face. These features include the position and shape of the eyes, nose, mouth, and other facial characteristics. The CNNs are capable of identifying these features even in images that are taken from different angles, under different lighting conditions, and with variations in pose and expression.

6.2 Algorithms

Convolutional Neural Networks (CNNs)

CNN is a type of neural network that are commonly used for image recognition tasks, such as face recognition. CNNs are designed to process images in a way that mimics the visual cortex of the brain. They use multiple layers of interconnected nodes to progressively extract and identify features from an image. The first layers of the network identify basic features such as edges and lines, while the deeper layers identify more complex features such as facial features like eyes, nose, and mouth. Once the features have been extracted, the CNN uses them to classify the image as a particular person or not.

Face recognition is the process of identifying or verifying the identity of a person using their facial features. It involves several steps such as face detection, feature extraction, and matching. The face recognition algorithm takes an input image and first detects the location of the face in the image using a face detection algorithm. Once the face has been detected, feature extraction algorithms such as Principal Component Analysis (PCA), Local Binary Patterns (LBP), or Histogram of Oriented Gradients (HOG) are used to extract the unique facial features from the image. These features are then compared with the features stored in a database to find a match.

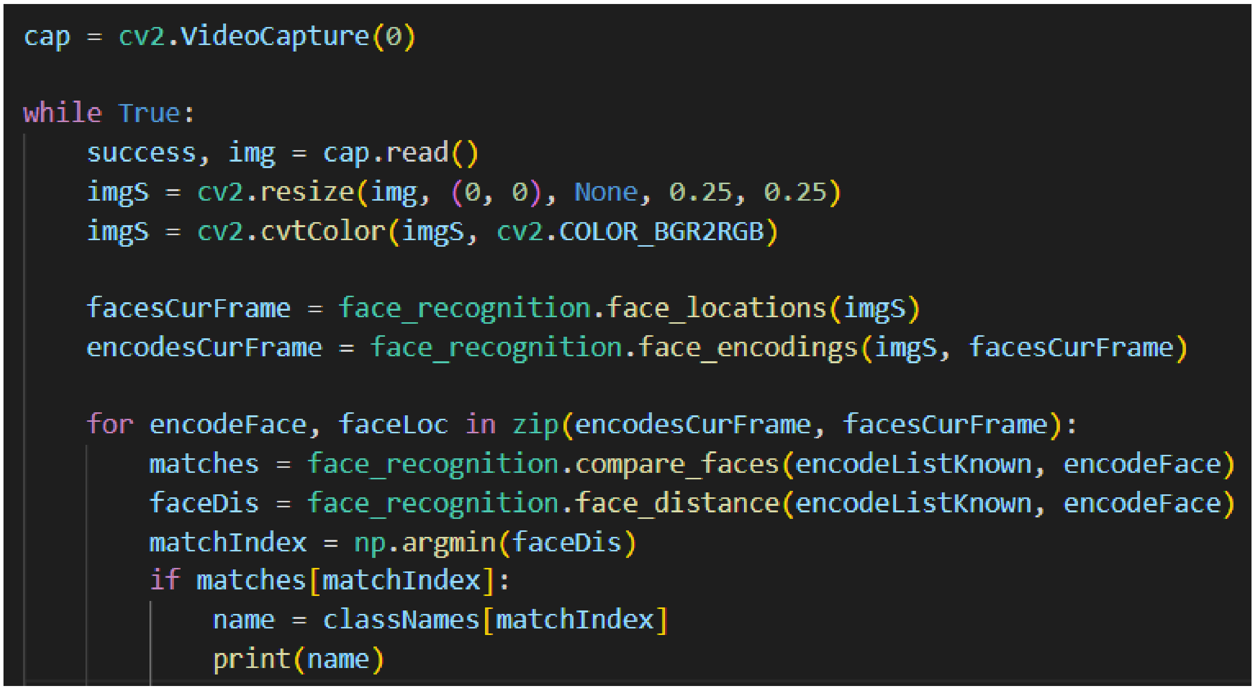
Python has several face recognition libraries that can be used to implement the face recognition algorithm. One of the most popular libraries is the "face_recognition" library, which is built on top of OpenCV and dlib. The library uses deep learning algorithms to extract and identify facial features from images and videos. It has a simple API that can be used to train the classifier on a dataset of known faces, and then use it to recognize faces in real-time.

In summary, CNNs are a powerful tool for image recognition tasks like face recognition, while face recognition algorithms involve several steps such as face detection, feature extraction, and matching. Python libraries like "face_recognition" provide an easy-to-use API for implementing these algorithms in practice.

The project is implemented in 4 parts:

- Face_recognition.py
- Location.py
- Whatsapp.py
- Mail.py

6.3 Face_recognition Function

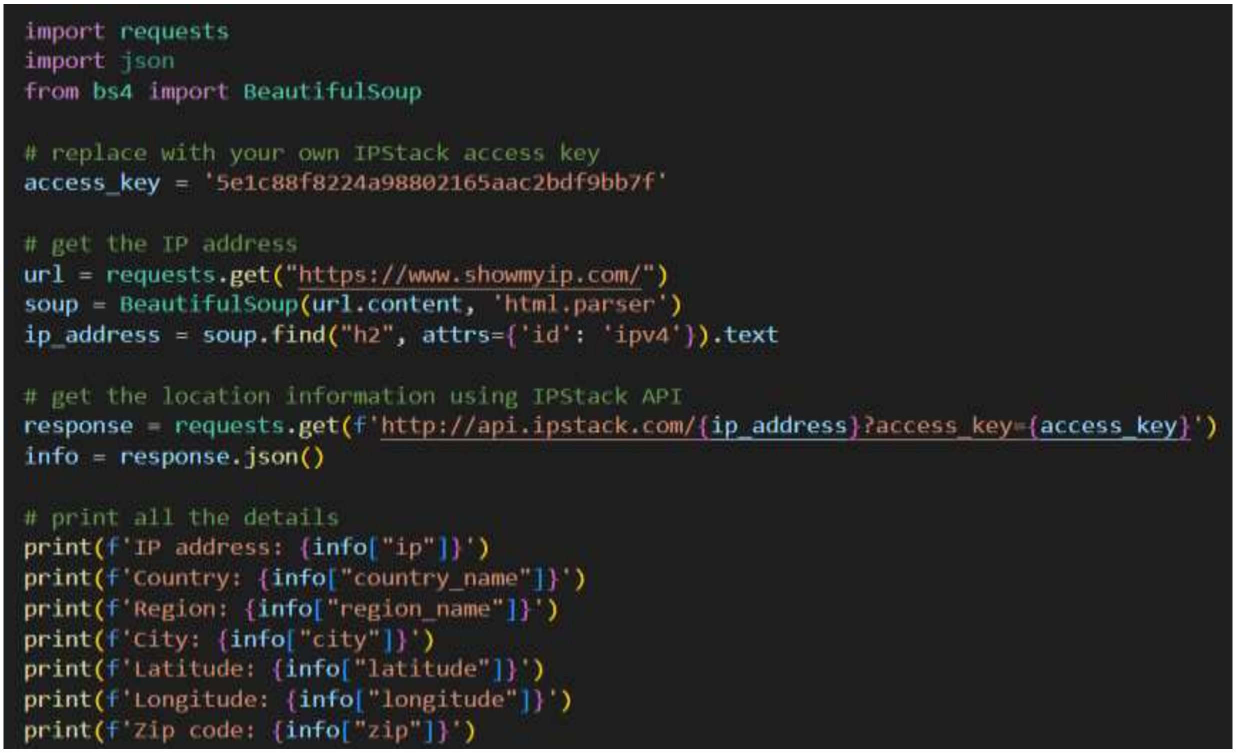


```
cap = cv2.VideoCapture(0)

while True:
    success, img = cap.read()
    imgS = cv2.resize(img, (0, 0), None, 0.25, 0.25)
    imgS = cv2.cvtColor(imgS, cv2.COLOR_BGR2RGB)

    facesCurFrame = face_recognition.face_locations(imgS)
    encodesCurFrame = face_recognition.face_encodings(imgS, facesCurFrame)

    for encodeFace, faceLoc in zip(encodesCurFrame, facesCurFrame):
        matches = face_recognition.compare_faces(encodeListKnown, encodeFace)
        faceDis = face_recognition.face_distance(encodeListKnown, encodeFace)
        matchIndex = np.argmin(faceDis)
        if matches[matchIndex]:
            name = classNames[matchIndex]
            print(name)
```

Fig 6.3 Face_recognition Function

The code is using Python and the OpenCV and face_recognition libraries to perform real-time face recognition on video input from the user's camera. It captures video frames using the cv2.VideoCapture function, resizes and converts the frames to RGB format. The face_recognition library is then used to detect faces and extract encodings from the faces in each frame. The program then compares these encodings with encodings for known faces in a database, using the compare_faces function. If a match is found, the program prints the name of the person. The code can be useful for various applications such as surveillance, access control, and attendance management.

6.4 Location Function

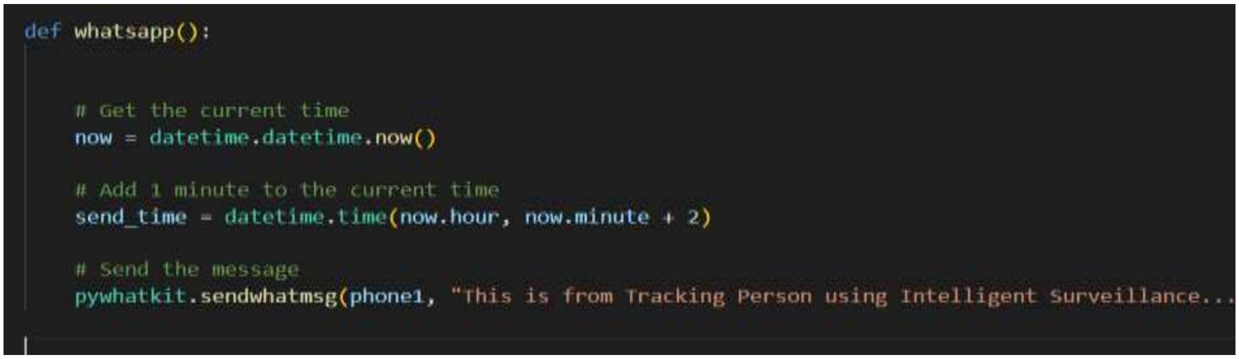


```
import requests
import json
from bs4 import BeautifulSoup

# replace with your own IPStack access key
access_key = '5e1c88f8224a98802165aac2bdf9bb7f'

# get the IP address
url = requests.get("https://www.showmyip.com/")
soup = BeautifulSoup(url.content, 'html.parser')
ip_address = soup.find("h2", attrs={'id': 'ipv4'}).text

# get the location information using IPStack API
response = requests.get(f'http://api.ipstack.com/{ip_address}?access_key={access_key}')
info = response.json()

# print all the details
print(f'IP address: {info["ip"]}')
print(f'Country: {info["country_name"]}')
print(f'Region: {info["region_name"]}')
print(f'City: {info["city"]}')
print(f'Latitude: {info["latitude"]}')
print(f'Longitude: {info["longitude"]}')
print(f'Zip code: {info["zip"]}')
```

Fig 6.4 Location Function

This code retrieves the IP address of the user by scraping the ShowMyIP website. Then, it uses the IPStack API to get the location information of that IP address based on the access key provided. The location information includes the country, region, city, latitude, longitude, and zip code, which are then printed out. This code can be useful for geolocation-based services or applications that need to customize content or features based on the user's location.

6.5 Whatsapp Function

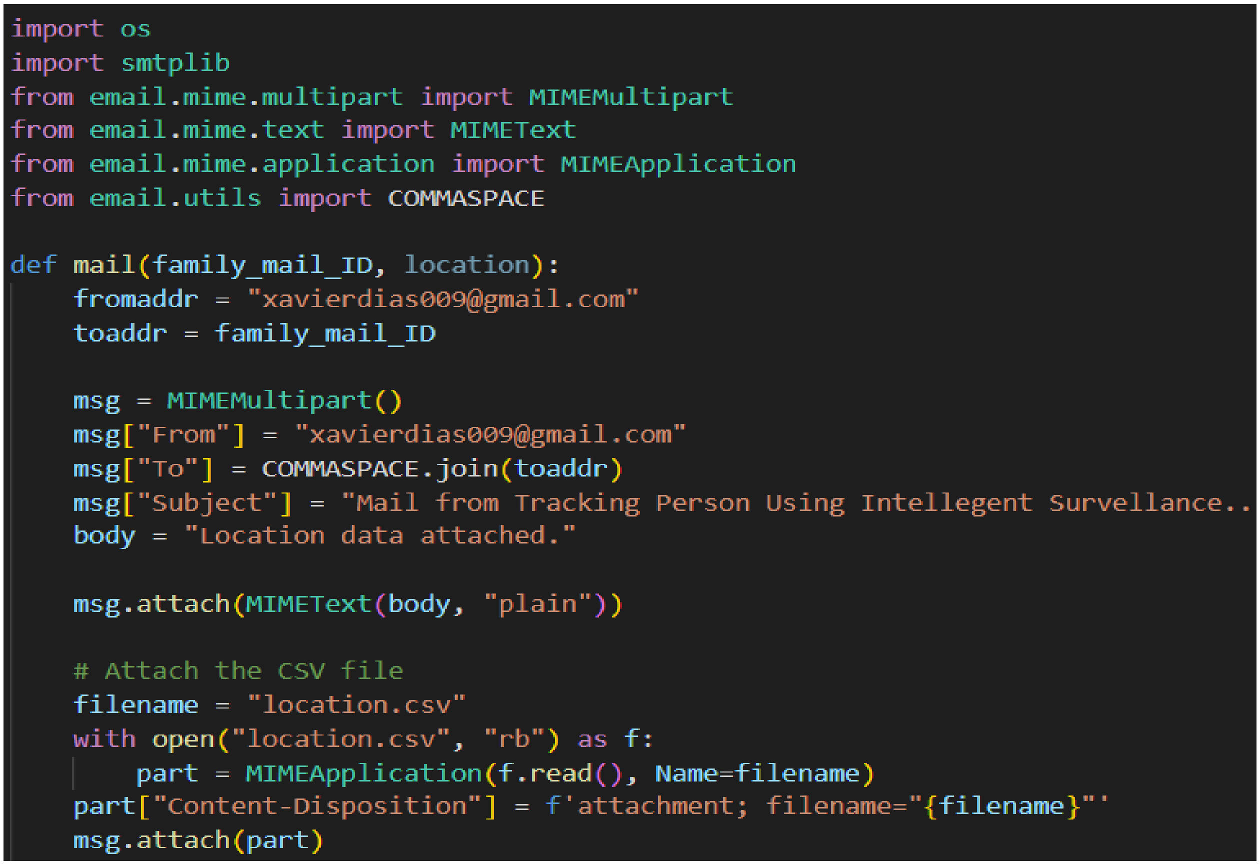


```
def whatsapp():


    # Get the current time
    now = datetime.datetime.now()

    # Add 1 minute to the current time
    send_time = datetime.time(now.hour, now.minute + 2)

    # Send the message
    pywhatkit.sendwhatmsg(phone1, "This is from Tracking Person using Intelligent Surveillance...
```

Fig 6.5 Whatsapp Function

This is a Python function that uses the **pywhatkit** library to send a WhatsApp message to a given phone number. The message content is hardcoded in the function and includes a message about a missing person being located. The function also sets a specific time for the message to be sent, which is 2 minutes after the current time. The **phone** parameter should be a string representing the phone number in international format (e.g. "+1234567890").

6.6 Mail Function



```python
import os
import smtplib
from email.mime.multipart import MIMEMultipart
from email.mime.text import MIMEText
from email.mime.application import MIMEApplication
from email.utils import COMMASPACE

def mail(family_mail_ID, location):
    fromaddr = "xavierdias009@gmail.com"
    toaddr = family_mail_ID

    msg = MIMEMultipart()
    msg["From"] = "xavierdias009@gmail.com"
    msg["To"] = COMMASPACE.join(toaddr)
    msg["Subject"] = "Mail from Tracking Person Using Intellegent Survellance..
    body = "Location data attached."

    msg.attach(MIMEText(body, "plain"))

    # Attach the CSV file
    filename = "location.csv"
    with open("location.csv", "rb") as f:
        part = MIMEApplication(f.read(), Name=filename)
    part["Content-Disposition"] = f'attachment; filename="{filename}"'
    msg.attach(part)
```

Fig 6.6 Mail Function

This is a Python function that sends an email with an attached CSV file containing location data. The function takes two arguments: **family_mail_ID**, which is the email address of the recipient, and **location**, which is the location data to be included in the CSV file. The function uses the **smtplib** library to send the email and the **MIME** library to attach the CSV file to the email. The email's subject is "Mail from Tracking Person Using Intelligent Surveillance.." and the body of the email simply says "Location data attached."

Chapter 7

TEST CASES

7.1 SYSTEM TEST

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of an0onents, sub-assemblies, assemblies and/or a finished product It is the process of exercising cookware with the intent of ensuring that the Software system meets its requirements and user annexations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement

7.2 TYPES OF TESTS

7.2.1 Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

7.2.2 Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event riven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

7.2.3 Functional test

Functional tests provide systematic demonstrations that functions tested are available as functional testing is centered on the following items:

Valid Input identified classes of valid input must be accepted.

Invalid Input identified classes of invalid input must be rejected

Functions identified functions must be exercised.

Output identified classes of application outputs must be exercised.

Systems/Procedures interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

7.2.4 System Test

System testing ensures that the entire integrated software system meets requirements. It configuration tests to ensure known and predictable results. An example of system testing is the oriented system integration test. System testing is based on process descriptions and emphasizing pre-driven process links and flows, integration points.

7.2.5 White Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box written from tests, as most other kinds of tests. must be definitive source document, such as specification or as requirements document, such specification or requirements document. It is a testing in which the software under test is treated, as a black box.

Chapter 8

RESULTS

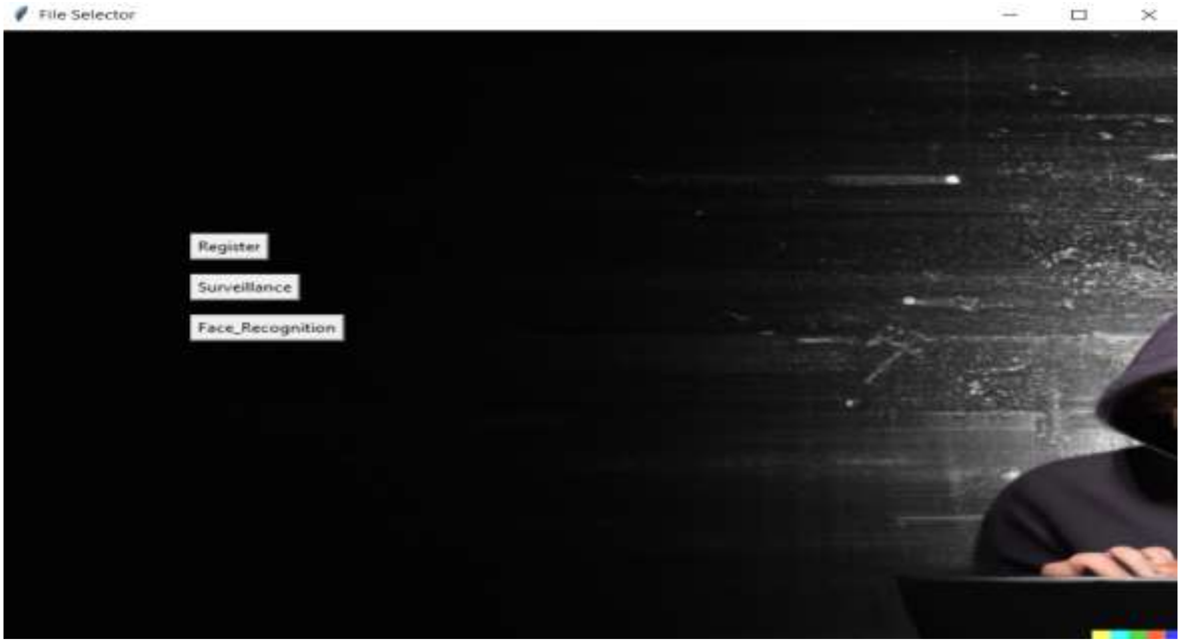


Fig 8.1 Home Interface

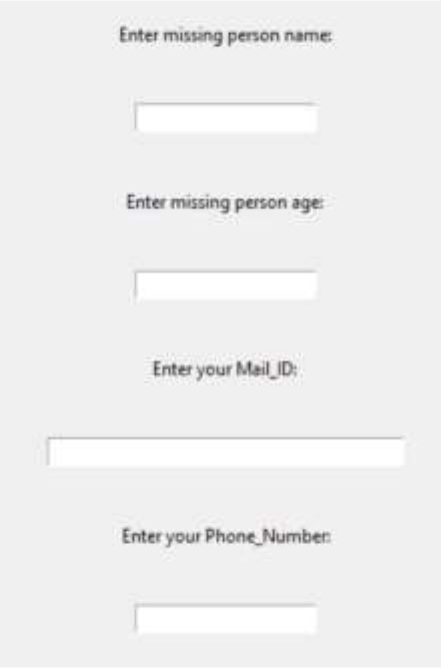
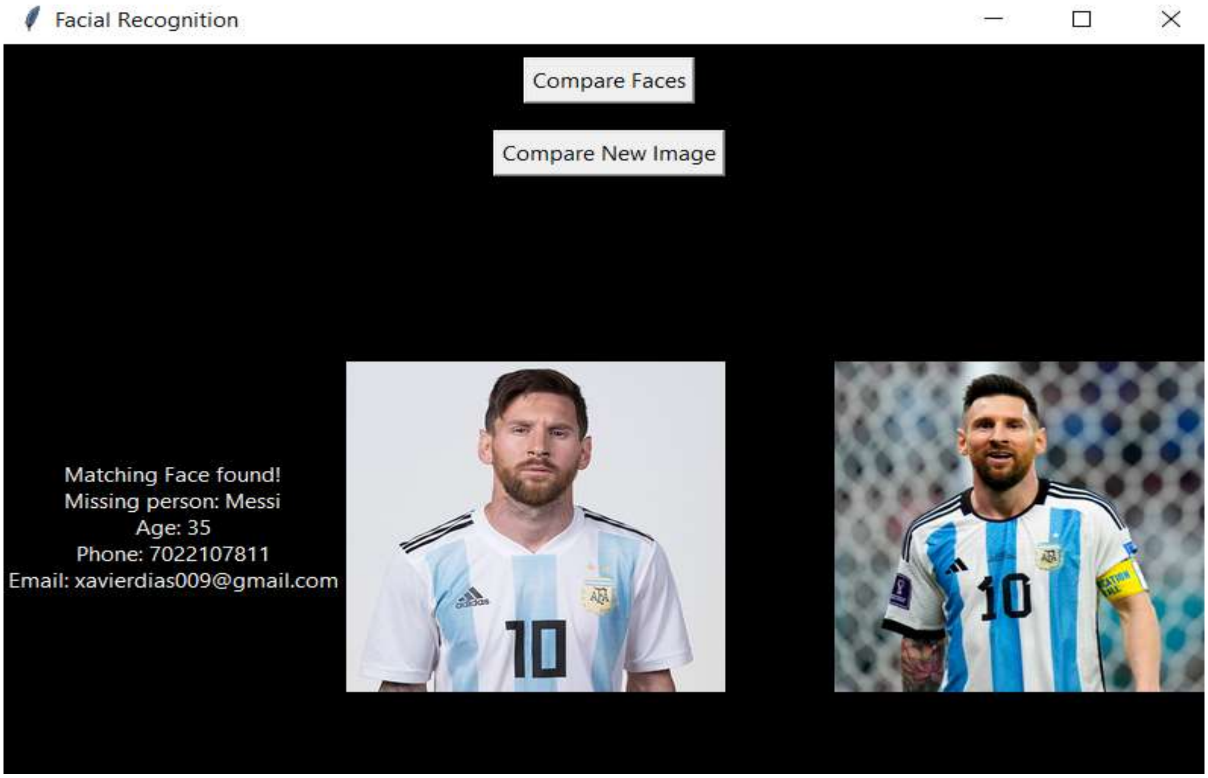
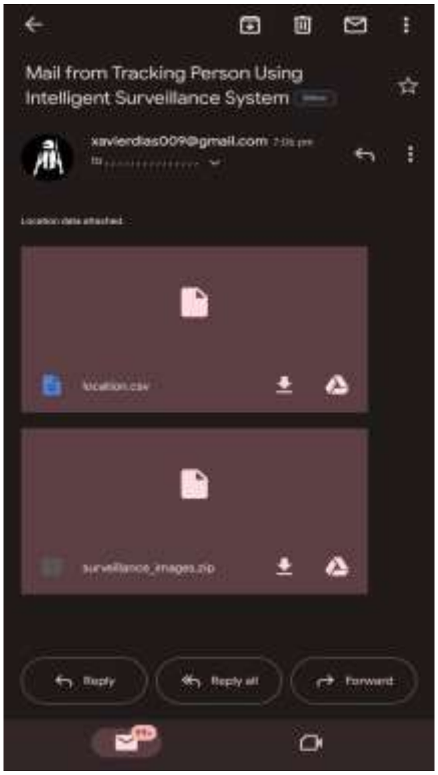


Fig 8.2 Registration Interface



Matching Face found!
Missing person: Messi
Age: 35
Phone: 7022107811
Email: xavierdias009@gmail.com

Fig 8.3 Face_recognition Interface


Mail from Tracking Person Using Intelligent Surveillance System

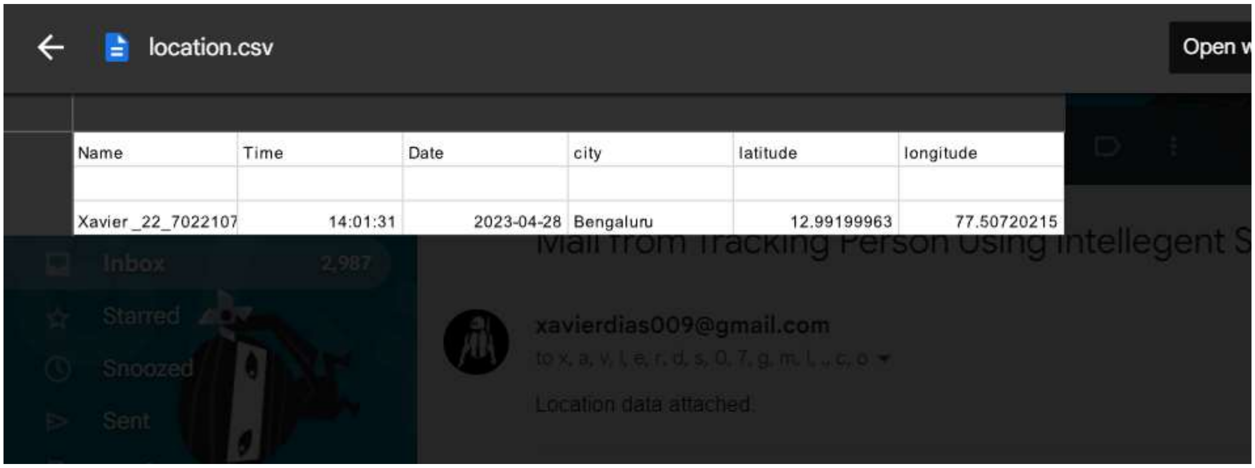
xavierdias009@gmail.com 7:06 pm

Location data attached.

location.csv

surveillance_images.zip

Fig 8.4 Mail Interface



Name	Time	Date	city	latitude	longitude
Xavier_22_7022107	14:01:31	2023-04-28	Bengaluru	12.99199963	77.50720215

Mail from Tracking Person Using Intellegent S

xavierdias009@gmail.com

to x, a, v, i, e, r, d, s, 0, 7, g, m, l, ., c, o

Location data attached.

Fig 8.5 Location CSV Interface

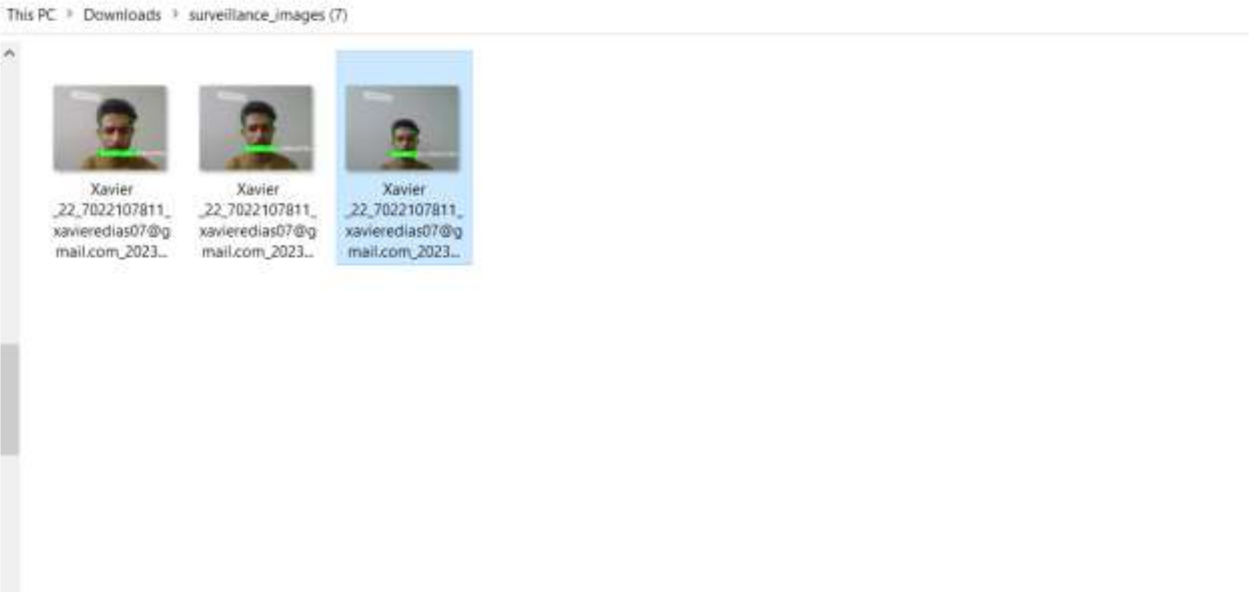
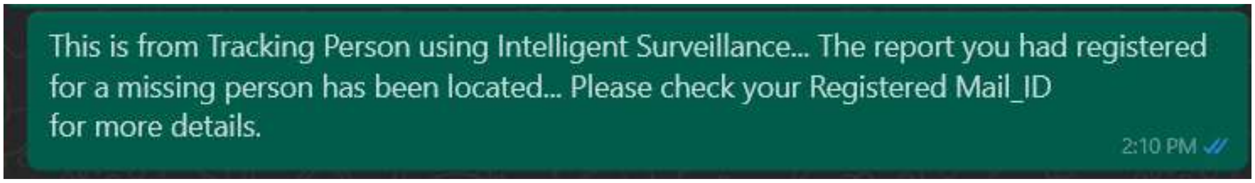

This PC › Downloads › surveillance_images (7)
Xavier _22_7022107811_ xavieredias07@g mail.com_2023...
Xavier _22_7022107811_ xavieredias07@g mail.com_2023...
Xavier _22_7022107811_ xavieredias07@g mail.com_2023...

Fig 8.6 Survellance Images Zip



This is from Tracking Person using Intelligent Surveillance... The report you had registered for a missing person has been located... Please check your Registered Mail_ID for more details.

2:10 PM

Fig 8.7 Whatsapp message Interface

CONCLUSION

The system works by searching for a match in the database, and if one is found, it redirects the user to that person's profile. Additionally, the system integrates with Google Maps to provide the exact location of the person, which can be extremely helpful for law enforcement in locating the missing person.

Compared to traditional manual scanning processes that require going through databases and images one by one, the face recognition method used in the system is much faster and efficient. This means that identifying missing persons can be done much more quickly, potentially increasing the chances of finding them before it's too late. Overall, the project aims to provide a useful tool for law enforcement agencies to help find missing persons and reunite them with their loved ones.

FUTURE SCOPE

Looking forward, we are working on developing an interface that can accept pre-recorded video footage and search for matches within the database. This would be a major step forward in improving the speed and accuracy of the system. By using video footage, the system could analyze multiple frames of a person's face to identify them, potentially increasing the chances of finding missing persons. With this new development, we hope to provide even more advanced tools to help law enforcement agencies in their crucial work.

REFERENCES

[1] **“**Finding missing persons using ML”, by Neha Gholape, Ashish Gour, Shivam Mourya, 2023.

[2] “Real Time Face Detection using CCTV Camera", by Nirmal Kumar Saraswat, Arvinda Kushwaha, 2022.

[3] “Crime Reporter and Missing Person Finder”, by Pranay Thali, Dhiraj Mujmule, Ankit Bhingarde, Prof. Yogesh Salve, 2021.

[4] “CCTV-Based Surveillance System with Face Recognition Feature”, by Michael Bryan P. Lumaban, Giged T. Battung,2021.

[5] “Face Modeling Process Based on Dlib” , by Junhang ding ,jinna sun,2021.

[6] “Missing Infant Finder Android Application” is a mobile application developed by Vivek C Shet and Mr. Srinivasulu M in 2021.

[7] “Android Based Application - Missing Person Finder” is a mobile application developed by Birari Hetal, Sanyashiv Rakesh, Porje Rohan, and Salve Harish in 2020.

[8] “Notification System Based on Face Detection and Recognition” is a study conducted by Abdullah Ahmad Basuhail and Ahmed AbdulQadir Al-bakeri in 2020.

[9] "Face recognition by fusion of local and global matching scores using DS theory" is a study conducted by Kisku D R in 2019.

[10] "Property Based testing of JSON Based Web Service" is a study conducted by Fredlund Benac Earle in 2019.