Multiple Face Recognition

A major project report submitted in partial fulfilment of the requirement for the award of degree of

Bachelor of Technology

in

Computer Science & Engineering / Information Technology

Submitted by

Karna K Chaudhary (201172)

Vishu K Banerjee (201386)

Under the guidance & supervision of

Dr. Aman Sharma

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology, Waknaghat, Solan - 173234 (India)

CERTIFICATE

This is to certify that the work which is being presented in the project report titled **"Multiple Face Recognition"** in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science And Engineering and submitted to the Department of Computer Science And Engineering, Jaypee University of Information Technology, Waknaghat is an authentic record of work carried out by "Karna K Chaudhary (201172) and Vishu K Banerjee (201386)" during the period from August 2023 to May 2024 under the supervision of Dr. Aman Sharma, Department of Computer Science and Engineering, Jaypee University of Information Technology, Waknaghat.

Karna K Chaudhary (201172)

Vishu K Banerjee (201386)

The above statement made is correct to the best of my knowledge.

Dr. Aman Sharma
Assistant Professor (SG)
Computer Science & Engineering and Information Technology
Jaypee University of Information Technology, Waknaghat

Candidate's Declaration

We hereby declare that the work presented in this report entitled **'Multiple Face Recognition'** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science & Engineering / Information Technology** submitted in the Department of Computer Science & Engineering and Information Technology**,** Jaypee University of Information Technology, Waknaghat is an authentic record of my own work carried out over a period from August 2023 to May 2024 under the supervision of **Dr. Aman Sharma** (Assistant Professor (SG), Department of Computer Science & Engineering and Information Technology).

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

(Student Signature with Date)
Student Name: Karna K Chaudhary
Roll No.: 201172

(Student Signature with Date)
Student Name: Vishu K Banerjee
Roll No.: 201386

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

(Supervisor Signature with Date)
Supervisor Name: Dr. Aman Sharma
Designation: Assistant Professor
Department: Computer Science & Information Technology
Dated:

Acknowledgement

Firstly, I express my heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the project work successfully. I am grateful and wish my profound indebtedness to Supervisor Dr. Aman Sharma, Assistant Professor (SG) Department of CSE Jaypee University of Information Technology, Waknaghat. Deep Knowledge & keen interest of my supervisor in the field of Deep Learning to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project.

I would also generously welcome each one of those individuals who have helped me straight forwardly or in a roundabout way in making this project a win. In this unique situation, I might want to thank the various staff individuals, both educating and non-instructing, who have developed their convenient help and facilitated my undertaking. Finally, I must acknowledge with due respect the constant support and patience of our parents.

Karna K Chaudhary (201172)
Vishu K Banerjee (201386)

TABLE OF CONTENTS

Contents

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Crime rates are increasing day by day and Identifications of criminals are one of the primary issues for investigation teams or police personnel, since property, life property and giving safety to the persons are the basic concerns of police but to stop the crime, the availability of police is limited, and it is one of the major concerns. To stop this, trying to reduce the crime rates (as crimes are committed by criminals), We made this project. The goal of this project is to identify the criminals with the help of CCTV footage, image and video with higher accuracy and better response rate. With the help of this technology, they (Police officers) can catch the criminals before doing another crime. This project is built on Python 3.11.x with the use of **OpenCV** along with algorithms like **MultiTask Cascaded Convolutional Neural Network (MTCNN), FaceNet** and **Face_recognition** etc.

In summary, this initiative aims to enhance our ability to do identification as easily and fast as possible in a society that is becoming more and more reliant on technology. One word at a time, it aims to close the divide between technology and mankind.

List of Abbreviations, Symbols, or Nomenclature

1. MFR: - Multiple Face Recognition
2. CIS: - Criminal Identification System
3. CNN: - Convolution Neural Network
4. LBP: - Local Binary Pattern
5. mtCNN: - Multi-Task Cascade Neural Network

Chapter 1: INTRODUCTION

1.1. Introduction

Criminal record of a person includes personal information, a photo, and other details concerning them. We require identification about any history sheeter in order to identify them. One of the ways is face identification. The investigation of criminals using face recognition technology has revolutionized the way law enforcement organizations fight crime. With the use of cutting-edge technology and sophisticated algorithms, criminal suspects can be located and apprehended by analyzing and matching face features in pictures and videos. Since its inception, the accuracy and speed of suspect identification have been substantially improved, ultimately leading to a society that is safer and more secure. The main benefit of face recognition is its ability to quickly and precisely identify prospective suspects from large amounts of visual data, including surveillance film, photos, and web images. Due to the efficiency, criminal investigations are completed much more quickly, allowing law enforcement to concentrate their attention on leads with higher probabilities of success. In situations where conventional identification techniques may fail, face recognition performs best. Thanks to the technology's powerful capabilities, suspects trying to change their look through disguises, facial hair, or other ways frequently find themselves easily identified. By spotting possible risks in real- time, it has been crucial in resolving cold cases, finding runaways, and preventing criminal activity. In addition to helping to solve crimes. Face recognition has demonstrated value in deterring criminal activities. Law enforcement authorities can prevent crimes from happening or escalating by swiftly identifying known criminals or persons of interest, which promotes public safety and crime deterrence.

1.2. Problem Statement

With the continuous development of computer technology, human dependence on technology has increased, which leads to the importance of security issues. Ensuring user authentication is crucial to prevent attacks and address security weaknesses. There are Various authentication methods are available, including fingerprint scanning, voice recognition, SMS one-time passcodes, and face recognition. Face recognition, an essential application of image processing in both still images and video, presents a significant challenge in creating an automated system that matches the human capacity for recognizing faces.

1.3. Objectives

1. Develop a Criminal Identification System (CIS): The primary objective of this project is to create an accurate and efficient CIS system capable of accurately identifying criminals from video of CCTV cameras.
2. Analyze the importance of CNN & how we can improve the model from the previous model with the help of CNN, and different datasets used in the face recognition system and discuss the different models of CNN. The deep-learned CNN can be used for face recognition to provide more security in authentication purposes.
3. Evaluation and Performance Metrics: Assess the model's ability to recognize criminals in different scenarios such as background lighting, resolution of videos and images, and indoor-outdoor conditions.

By achieving these objectives, the project aims to contribute to the advancement of Criminal Identification system and its practical utilization in various security concerning areas, ultimately improving human-computer interaction and automate and task.

1.4. Significance and Motivation of the Project Work

Since there is no dedicated Criminal Identification System, it can assist the police department to identify criminals in their premises rather than have to go through the whole footage and comparing different criminals one by one. This leads to watching a single video multiple times to identify criminals i.e., if there is ‘N’ numbers of criminals and ‘M’ numbers of videos then they have to watch the footage and verify criminals ‘M*N’ times.

Scope of the System: The scope of CIS to identify criminals within allotted time frame. The system can be used with specified user requirements. It can be used in Police stations, Bank, Hotels. A thorough study found that numerous ways and combinations of these methods can be used to construct a new Criminal Identification system. Among the different types of methods, We have chosen to implement a combination of knowledge-based techniques for face detection and a neural network approach for face identification. The accuracy of the system depends on the camera viewpoint, background illumination, facial components, quality of the footage, and finally on quality of the image that is available to train the model. Some features have to be included in the system in order to process live photos at a high rate with great precision and accuracy. Another aspect of the research is the creation of a model that can identify a criminal when trained on specific facial sketches from a criminal dataset.

1.5. Organization of Project Report

1.5.1. Tools & Technologies

Using Python for a project like Criminal identification System is a practical choice due to its versatility, extensive libraries, and a growing community of developers in the field of machine learning and image and video classification and processing. Here's how Python can be utilized for an CIS project:

1. Data Collection and Preprocessing: Python can be used to collect image and video data and preprocess it for analysis. Libraries like ‘Bing API’, useful for collecting image data, libraries such as ‘OpenCV’, ‘CNN are used for extracting features from image and video files.

2. Machine Learning and Deep Learning: Python offers a wide range of ‘machine learning’ and ‘deep learning’ libraries that are essential for building

3. CIS models. Some popular libraries include ‘openCV’ for traditional machine learning algorithms and deep learning frameworks like ‘mtCNN’ and ‘FaceNet’ for building neural networks.

4. Feature Extraction: Python can be used to extract relevant features from images and video datasets. Libraries like ‘openCV’ are commonly used for feature extraction.

5. Model Training: Python allows you to train machine learning and deep learning models using audio features as input data. You can implement and train models like Convolutional Neural Networks (CNNs), MultiTask Cascaded Neural Networks (mtCNNs), or hybrid models designed for face recognition tasks.

6. Real-time Prediction: If a project involves real-time criminal identification from live CCTV cameras as direct input, Python can be used to capture and process video input in real-time. Libraries like ‘openCV’ can help with this.

7. Visualization: Python offers various libraries for data visualization, which can be helpful for visualizing images and videos data, model training progress, and results. Popular libraries for visualization include ‘matplotlib’ and ‘seaborn’.

1.5.2. Technical Requirements

1. Hardware Requirements:

   a. Processing Power: Depending on the complexity of the machine learning models, a computer with a high-end GPU or multiple set of GPUs are required for extraction of frames from the videos, can speed up the model training and produce the output efficiently.

   a. Memory: Sufficient RAM is essential for loading and processing large datasets, especially for using deep learning models.

   b. Storage: Storage space for images and video datasets, model checkpoints, trained mode and other project-related files.

   c. Camera: A good quality CCTV camera is required in case of live criminal detection

2. Software Requirements:

   d. Python: Most CIS projects are implemented in Python due to its vast availability of libraries for machine learning, image processing and analysis.

   e. Development Environment: VScode editor with python and jupyter notebook for writing and debugging of code.

   f. Machine Learning Frameworks: Install the necessary machine learning and deep learning libraries, such as OpenCV, MTCNN, scikit-learn, numpy, FaceNet etc.

   g. Image Processing Libraries: Libraries like dlib, face_recognition are used for image data preprocessing.

1.5.3. Deliverables/ Outcomes

The deliverables and outcomes of a Criminal Identification System (CIS) project are following: -

1. Trained CIS Model: The primary deliverable is a trained machine learning or deep learning model capable of recognizing faces in video and live stream. This model should be able to take video input and predict the find and mark criminals present in the video based on database of existing criminals.

2. Accuracy Metrics: A report or documentation detailing the performance CIS model, including accuracy, precision, recall, F1-score, and confusion matrices. This provides an evaluation of how well the model performs in identification of criminals.

3. Codebase: The project's codebase, including data preprocessing scripts, model training, testing and real-time prediction implementations. Proper documentation and code comments included for easier understanding and future development.

4. Demo/Prototype: A working demo or prototype showcasing the functionality of CIS system, especially real-time criminal identification from live video inputs.

Chapter 2: Literature Survey

Table 1: Literature Survey

S. No.	Paper Title	Journal/ Conference (Year)	Tools/ Techniques/ Dataset	Results	Limitations
1.	Real time face recognition system based on YOLO and InsightFace	N/a (2023)	YOLO- V7	Achieved accuracy of 93% with better response rate by generating highly discriminative face embeddings.	Recognizing faces by generating highly discriminati ve face embeddings.
2.	A Systematic Literature Review on the Accuracy of Face	Vol. 8 No. 30 (2022): EAI Endorsed Transactions on Internet of Things (2022)	CNN, Viola-Jones Algorithm	Achieving high accuracy and quick response time.	Tuning Overhead
3.	Research and application of deep learning in image recognition	2nd International Conference on Power, Electronics and Computer Applications (ICPECA) (2022)	Haar Cascade, OpenCV	Achieving accuracy of 87%	Requiremen t of high computation al power
4.	Criminal Identification System using Facial Recognition [2021]	International Conference on Innovative Computing & Communicatio n (ICICC) (2021)	OpenCV, Image Processing	System matching more than 80% of the captured images with database images.	Was only able to capture images, not video
5.	Criminal Face Identification System	International Journal for Research in Applied Science & Engineering	Haar Cascade, OpenCV	Achieving accuracy of 86%.	Images are divided into several parts of face and then detect the suspect

		(2020)			through their database.
6.	Face Recognition System Using Machine Learning Algorithm	2020 5th International Conference on Communication and Electronics Systems (ICCES) (2020)	PCA, Linear Discriminant Analysis, SVM	Achieved high accuracy (more than 90%)	Only ORL datasets were used for performing experiments.
7.	Criminal Face Detection System Using Python	IJIRT Volume 7 Issue 2 ISSN: 2349-6002 (2020)	CNN	Achieved accuracy of 96.2% using a hybrid approach.	Performed on the basis of Thumbprint and matching information with FIR records.
8.	Criminals and Missing Children Identification Using Face recognition And Web Scraping	International Conference on System, Computation, Automation and Networking (ICSCAN) (2020)	Haar Cascade Classifier, OpenCV, Python, Faceplib	Performs well with both images and videos. Achieved accuracy of 90%.	Multilingual Database not used
9.	Face Recognition using Haar-Cascade Classifier for Criminal Identification.	N/a (2020)	Haar Cascade Classifier, OpenCV, Python, Face_recognition	Good performance on low resolution images	Achieved low accuracy
10.	Criminal Identification System using Facial Recognition	International journal of Recent Technology and Engineering, vol-7 (2019)	Haar Cascade, OpenCV, LBPH	Recognized faces automatically in real time. Achieved accuracy of 95%.	Slow Conversion compared to other deep learning model
11.	Automated Criminal Identification System using Face Detection and Recognition	International Research Journal of Engineering and Technology p-ISSN: 2395-0072 (2019)	Haar Classifier, Viola-Jones, OpenCV	Achieving high accuracy	Captured video converted into frames

12.	Automated criminal identification by face recognition using open computer vision classifiers	3rd International Conference on Computing Methodologies and Communication (ICCMC) (2019)	Haar Cascade	Quick processing and better response time.	The system can effectively identify faces even if the stored set of images of the person in the database differ from the input image.
13.	CIS: An Automated Criminal Identification System [2018]	Third International Conference on Computing Methodologies and Communication (ICCMC) (2019)	HOG, Deep Learning Networks	The author has achieved accuracy of 80%	Distinctive identity used in neural networks as input value.
14.	Criminal Face Recognition System	International Journal of Engineering & Technology	Viola-Jones, Linear Discriminant Analysis	Using its two degrees of freedom, the system allows two modes of operation.	Image-based face detection and recognition
15.	Intelligent Criminal Identification System	2013 8th International Conference on Computer Science & Education, Colombo, Sri Lanka (2013)	Naive Bayesian Classification	It could detect suspicious human behavior	The system uses an explicit clustering mechanism based on the available evidence.

2.1. Overview of Literature Survey

This section discusses many research articles to show how you can apply different algorithms/methods to make this technology better. Which algorithms give higher accuracy to which type of dataset.

Anjeana et. al [1], performing this model with a new real time unique face recognition network called YOLO-InsightFace that combines YOLO_V7, a cutting-edge deep learning model and InsightFace, one-of-a-kind 2D & 3D face analysis python module as YOLO-V7 is highly accurate and fast while InsightFace can recognize faces by generating highly discriminative face embeddings. Abdullah et al. [2], Analyse the performance based on CCTV footage as they took help of the CCTV footage and compared the images from the footage with a criminal database if they were not able to find the biometric from the crime scene.

In the third stage they defined system design and its workflow. To identify the criminals, The authors used Principal Component Analysis (PCA) algorithm to detect similar features between images stored in a database and the captured footage. The facial recognition system is totally dependent on a database that contains personal information for individuals. If the system detects a face, it will definitely be in their database. The proposed model achieved an accuracy of 80%. Apporva et. al [4], proposed a robust face detection methodology which was suitable for real-time environments. The Haar cascade classifier algorithm was used in that model as the Haar cascade classifier can effectively identify faces even when the input image does not perfectly match the images stored in the database. The suggested system has been put into practice and examined. An investigation developed by Sharma et al. [5] is used to automatically identify a person's face. Tests are conducted using the ORL dataset. The system yields more than 90% accuracy when tested on a database containing over 1000 photos, despite a few false positives. It made use of several techniques, such as PCA with linear discriminant analysis, which yields 97% accuracy, and related methods, which produced an average accuracy of 91%. Tamrakar et. al [6], proposed a methodology for criminal identification using face recognition as traditional methods such as thumbprint or matching information with FIR records which is not always effective. Three different methods were proposed for this face recognition model such as feature-based, holistic and hybrid methods. The holistic approach uses the entire face as input for detection and recognition, whereas the feature-based approach divides up local features like the nose and eyes for use as input for the face detection system. The feature-based and holistic approaches are combined in the hybrid approach. The authors' suggested model yielded 96.2% accuracy. Chevewalla et al. [7], proposed a model where they found that in computer vision, face recognition is a highly challenging task and the more crucial task is to achieve high accuracy and better response rate. Countries like Germany and Australia have implemented face recognition technology at borders and propose a comprehensive solution for image-based face detection and recognition with improved accuracy. Shahriar et.al [8], provided a study on facial recognition where a biometric technology that creates a mathematical map of a person's facial features and stores information as a face print. It creates a feature vector that maps the object with a set of numbers by using machine-learning images. This technology is used by various organizations such as Google and Facebook to create digital profiles for their users. Facial recognition technology will be used in the proposed project to identify criminals who are on the run and have a criminal history. According to a report by NCRB (National Crime Records Bureau), most of the crimes are committed by the same criminals repeatedly. Face recognition technology is a useful technique to identify criminals by capturing pictures or video images through cameras installed in various locations. It can also be used to identify missing children and many more. The drawback is that the images frequently captured are less sharp, blurry, and challenging for the human eye to distinguish. The system proposal's short calculation time makes it particularly helpful for identifying suspicious individuals and capable of efficiently recognizing multiple faces. For every face in the dataset, the system generates a unique template that it then compares to the other images. The corresponding details are shown when the input face matches an existing image. A system like this can help an organization's security procedures and reduce crime. Rasanayagam et. al [9], performs a model based on the analysis of faces, emotions, ages and genders to identify the criminal. CNN techniques that are based on deep learning are used to implement face recognition, emotion, age, and gender identifications. The LeNet architecture is used for the identification of suits. Although Haar Cascade is a useful technique for object detection but its ability to detect features in objects is limited because of its age. The main purpose of the HOG (Histogram of Oriented Gradients) descriptor is features extraction, although it can take some time. Networks of Neurals Deep Learning is a unique identity that is used as input values in neural networks to compute and analyse similarity in recognition. After a month of training, the author's accuracy was 80%. Adikari et. al [10], The researchers used Naive Bayesian classification to determine the most likely suspects of criminal episodes after grouping crime data into subsets depending on the evidence that was available. Technology implemented user interfaces using Prime Faces and JSF (Java Server Faces). Prime faces 3.2 was utilised in the development of all user interface widgets, including buttons, menus, and input fields. The built-in Ajax functionality of Prime faces, which is based on the standard JSF 2.0 Ajax API, is used to update the components when the algorithms have completed processing. The Oracle Database 10g Express Edition was utilised by the publisher to implement the database. Kumar et. al [11] offered a real-time criminal identification system based on face recognition using a fully automatic facial recognition system. The system uses Haar Feature-based cascade classifiers and OpenCV LBPH (Local Binary Pattern Histograms) algorithms for face detection and recognition. Many researchers have employed the Viola-Jones framework to locate faces and objects in an image, but face detection remains a challenging task. Classifiers for face detection are freely accessible, like OpenCV. The authors reported achieving 95% accuracy in their experiments.

2.2.Key Gaps of Literature Survey

1. It highlights the primary research focus in crime prediction, in terms of crime type and category, study time, and which type of crime has been addressed by most researchers.

2. Firstly, clarifies the frequency of algorithms which were used while training models and considered the factors/features while choosing the algorithm for our model.

3. Analyzes model estimation accuracy by focusing on four factors: performance metric, evaluation value, dataset used, and model validation approaches.

4. Demonstrates the tools applied in the selected research papers, as well as the strengths and weaknesses of the prediction models, as well as the limitations and future direction of crime prediction.

Closing these gaps could contribute to a more comprehensive and robust understanding of Criminals Identification System model, making them more effective and applicable in real-world scenarios. Researchers may consider addressing these gaps in future work to advance the field.

Chapter 3: SYSTEM DEVELOPMENT

3.1. Requirement and Analysis:

3.1.1. Requirement

Criminal Identification System (CIS) is the task of automatically detecting and identifying the criminals in CCTV footage. Human dependence on network technology has grown as computer technology has advanced, highlighting the relevance of security concerns. User authentication is critical for avoiding threats and security flaws. Various authentication methods exist, including fingerprint scanning, voice recognition, SMS one-time passcodes, and facial recognition. Face recognition is a critical application of image processing in both still photos and video. It's a real challenge to create an automated system that can detect faces.

1. Data Availability and Quality: Availability of images and video with different background illumination, facial features and scenarios to make a diverse dataset can be a challenge for training and testing CIS models. The quality of the data is crucial, as it should encompass different scenarios to ensure model generalization.
2. Real-Time Processing: Implementing CIS for real-time identification of criminals using CCTV live stream as direct input to the system. This also includes reducing the complexity of the system.
3. Face Recognition and Identification: Detecting faces in CCTV footage as well as in images even if there is inconsistent background illumination, different background, low resolution images or video.
4. Scalability: It should be able to handle a huge number of videos or images.
5. Performance: The system should provide a fast and responsive user experience, ensuring quick face recognition and criminal identification.
6. Privacy: Users' privacy and data confidentiality should be maintained, and explicit consent for data collection and processing should be obtained.

3.1.2. Analysis

When analyzing the problem of Criminal Identification System (CIS), it was essential to consider various aspects and challenges associated with this field. Solving the challenges associated with Criminal Identification System (CIS) requires a multidisciplinary approach involving data collection, feature engineering, model development, and ethical considerations.

1. Data Augmentation and Diverse Datasets:
    a. We have created a larger and more diverse criminal datasets to improve model generalization.
    b. Collected facial images of people from images and videos as well as from live footage, ensuring that the dataset has a wide range of facial characteristics, races, and ages.
    c. Collected photographs taken in a variety of lighting, environments, and angles. This will help our model's recognition of faces in real-world conditions, which may differ significantly from controlled laboratory environments.
2. Feature Engineering: We Explored different advanced feature extraction techniques, such as Multi-Task Cascaded Convolutional Neural Networks (mtCNN), Local Binary Patterns (LBP), Haar-Cascade Classifier.
3. Model Selection: While experimenting with different ML & DL models, we found that mtCNN along with face_recognition using dlib and FaceNet will result higher accuracy and precision.
4. Regularization Techniques: Applied regularization techniques, including dropout and L1/L2 regularization, to prevent overfitting and enhance the model's generalization capabilities.

3.2. Project Design and Architecture

3.2.1. Design

Project design of our system include various stages from data input and preprocessing, frame extraction from CCTV footage and facial feature extraction to real-time processing and displaying the output.

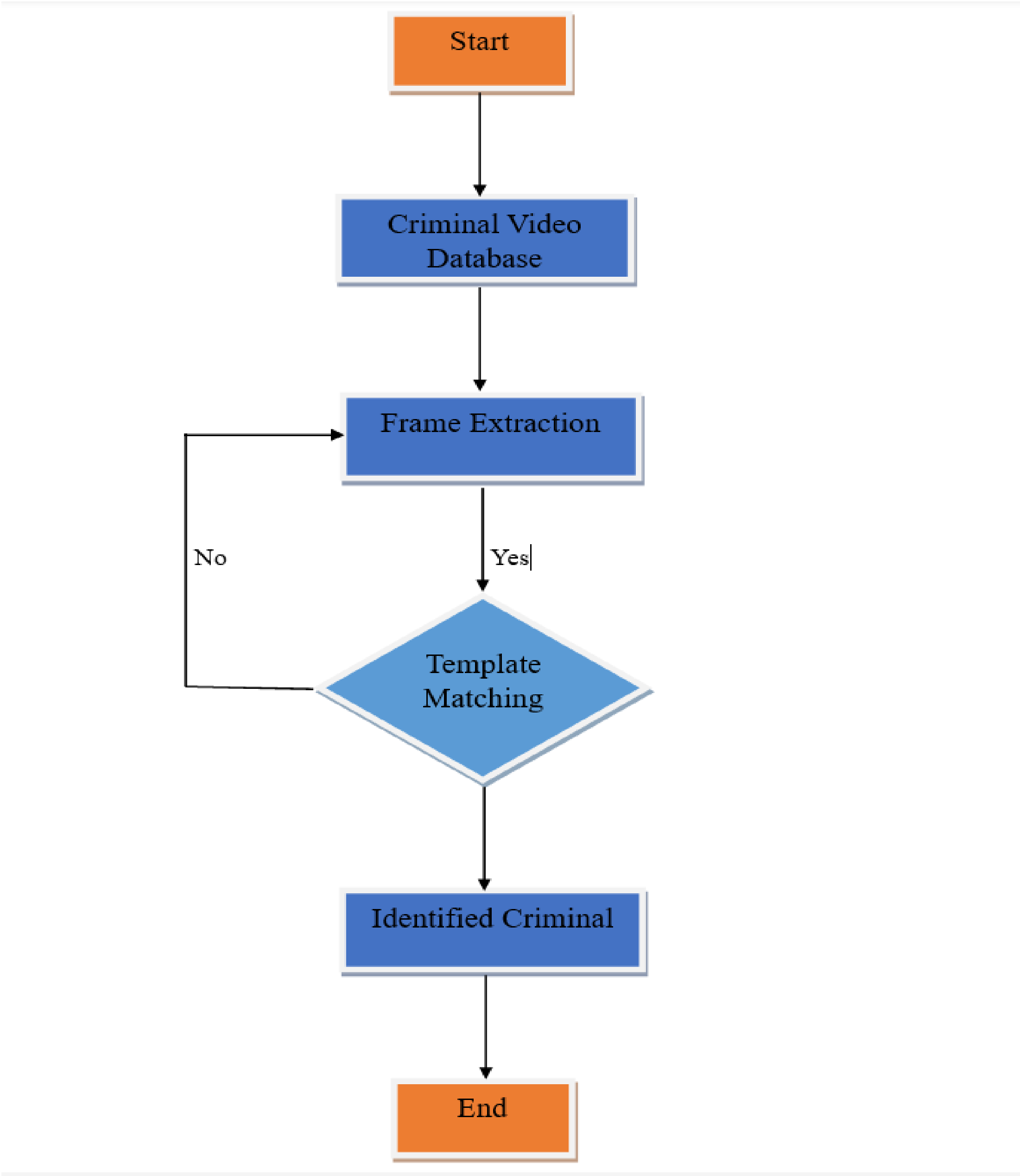


Figure 1: Flow Graph of Major Project Problem

Level 0 DFD: It shows high-level overview of the entire system, illustrating the primary process, Criminal Identification System (CIS), the external entities, such as the Existing Criminal Dataset, that interact with the system, and the data flow between them. This serves as the starting point for creating more detailed DFDs at lower levels, where the main process is broken down into more specific subprocesses.

Figure 2: Level 0 DFD

Level 1 DFD: It shows more detailed view of the system by decomposing the primary process from the Level 0 DFD into subprocesses, such as Face Recognition. This helps in understanding the specific functions of the system and their interactions. Embedded data is then compared with embedding data from video frames. This decomposition process can be continued to create even more detailed DFDs at subsequent levels.

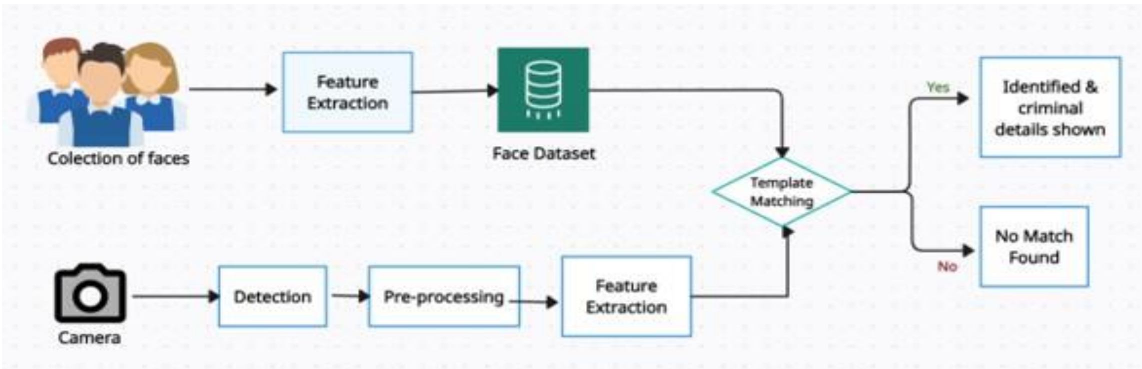


Figure 3: Level 1 DFD

Level 2 DFD: It shows more detailed view of the system by further decomposing subprocesses from the Level 1 DFD into sub-subprocesses, such as feature extraction, and training and testing of models. This hierarchical decomposition continues until an adequate level of detail is achieved, facilitating a thorough understanding of the system's processes and their interactions.

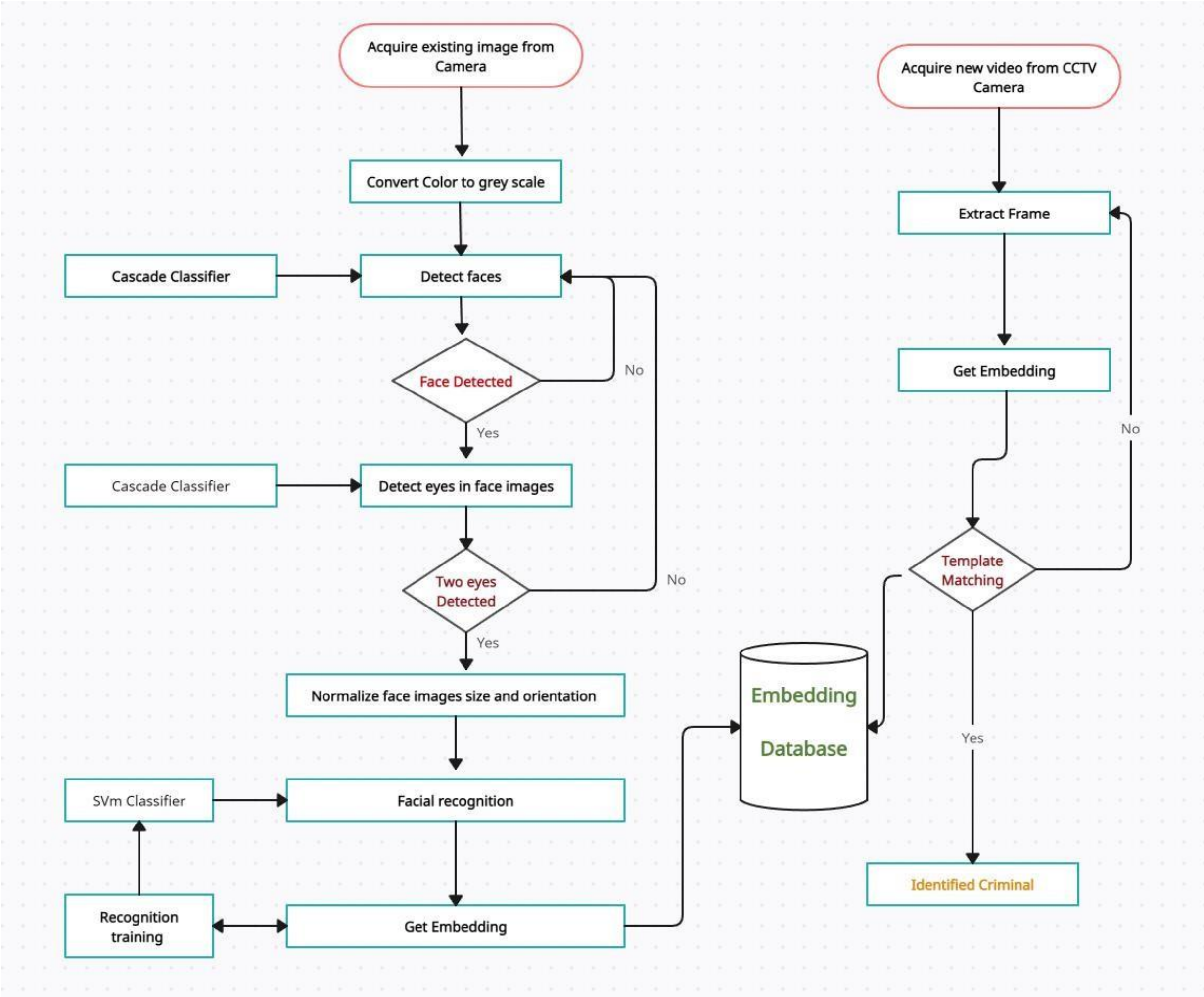


Figure 4: Level 2 DFD

3.2.2. Architecture of Algorithms used

1. OpenCV

OpenCV is a commonly used computer vision library developed by intel. It provides a framework for face recognition applications. It uses a number of techniques including deep learning, Support Vector Machines (SVMs), and Haar cascades, to achieve accurate and efficient face recognition.

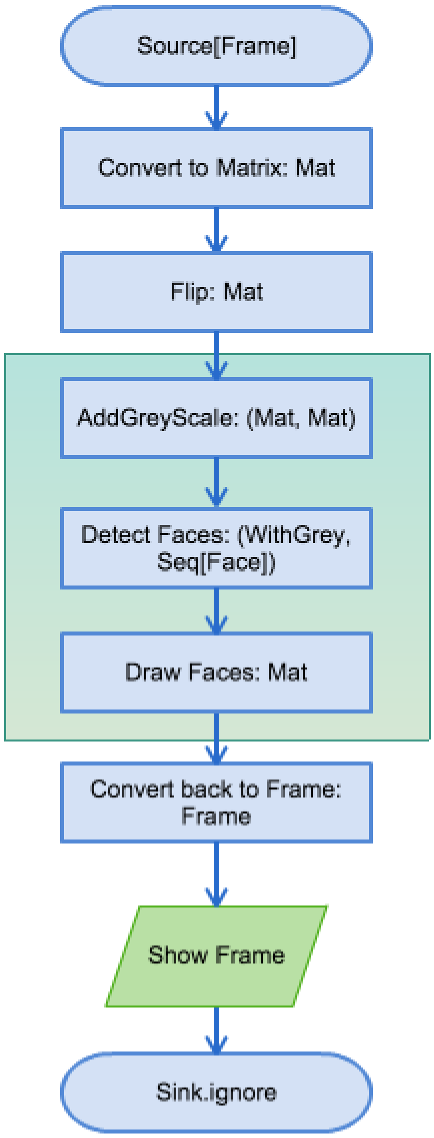


Figure 5: OpenCV Face detection Architecture [reddit]

2. Haar-Cascade Classifier

The Haar-Cascade classifier is very useful for object detection in images and videos. It is developed by Paul Viola and Michael Jones. It is based on a machine learning technique that trains the classifier by using a large number of images, both positive and negative.

- **Positive Images:** Features that we want our classifier should identify from data.
- **Negative Images:** Features that we do not want our classifier should identify from the data. I.e., data does not contain the features that we want to identify.

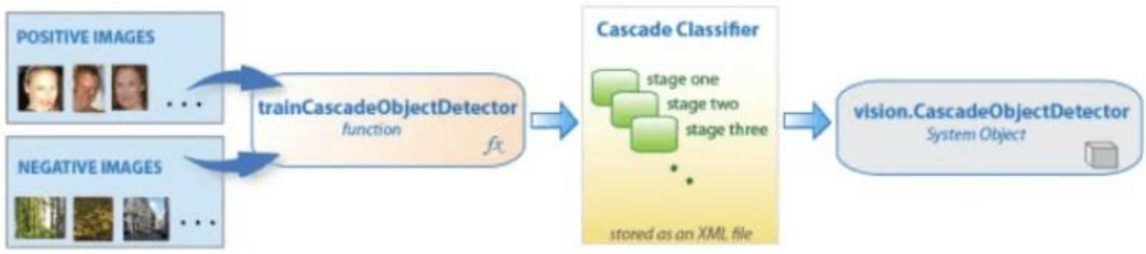


Figure 6. Haar Cascade Classifier Architecture

3. FaceNet

FaceNet is a neural network-based system employing deep learning algorithms for facial recognition. It utilizes a convolutional neural network (CNN) to map facial features into a Euclidean space, where distances represent the similarity between faces. This space enables the system to compare faces and determine if they belong to the same individual.

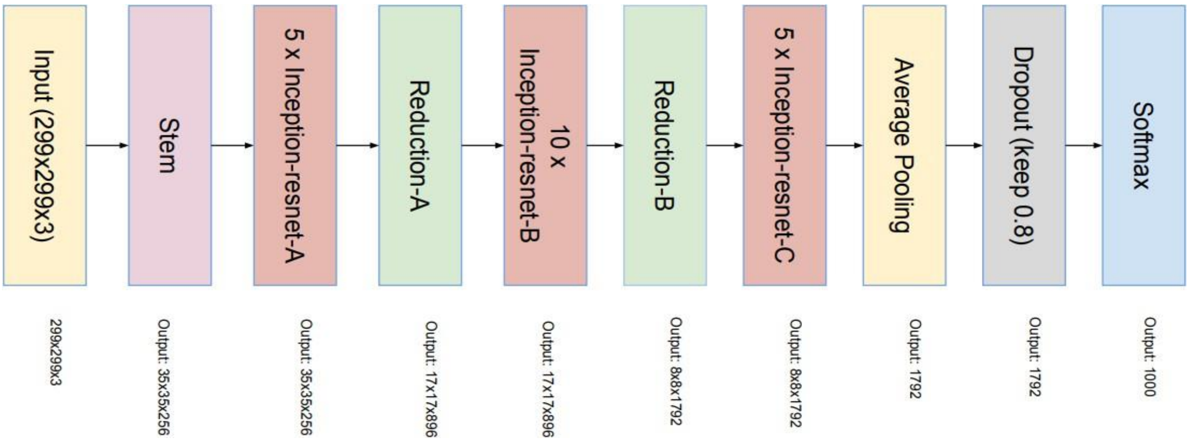


Figure 7: FaceNet Size Reduction [towards datascience]

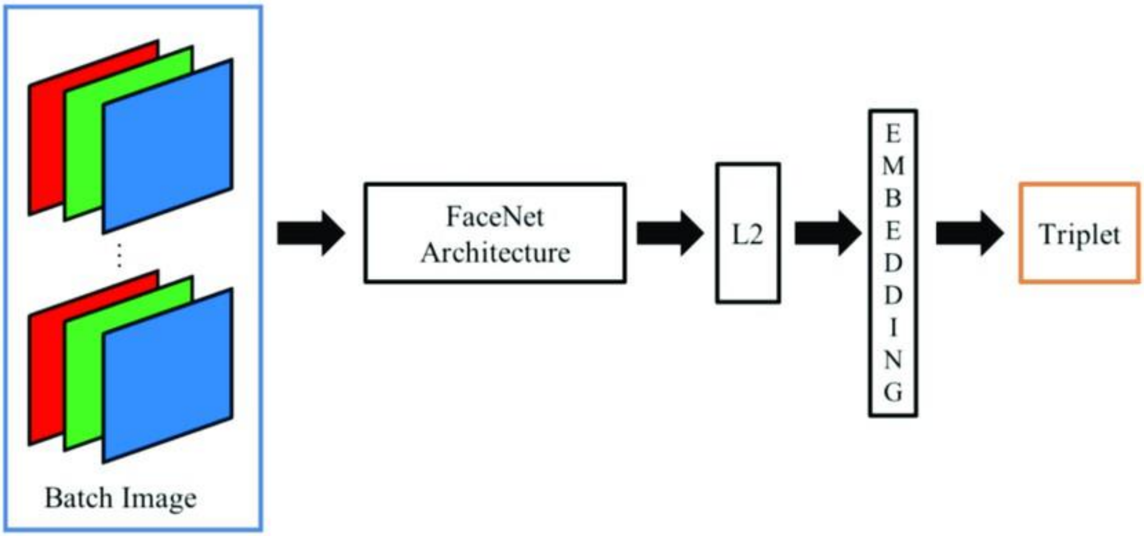


Figure 8: FaceNet Architecture [gfg]

4. MultiTask Cascaded Convolutional Neural Network

MultiTask Cascade Convolutional Neural Network is a 3-stage cascade network. The process involves three stages of convolutional networks that can identify faces and locate landmarks such as the eyes, nose, and mouth. The MTCNN algorithm employs a cascading sequence of neural networks to detect, align, and extract facial features from digital images with high accuracy and speed.

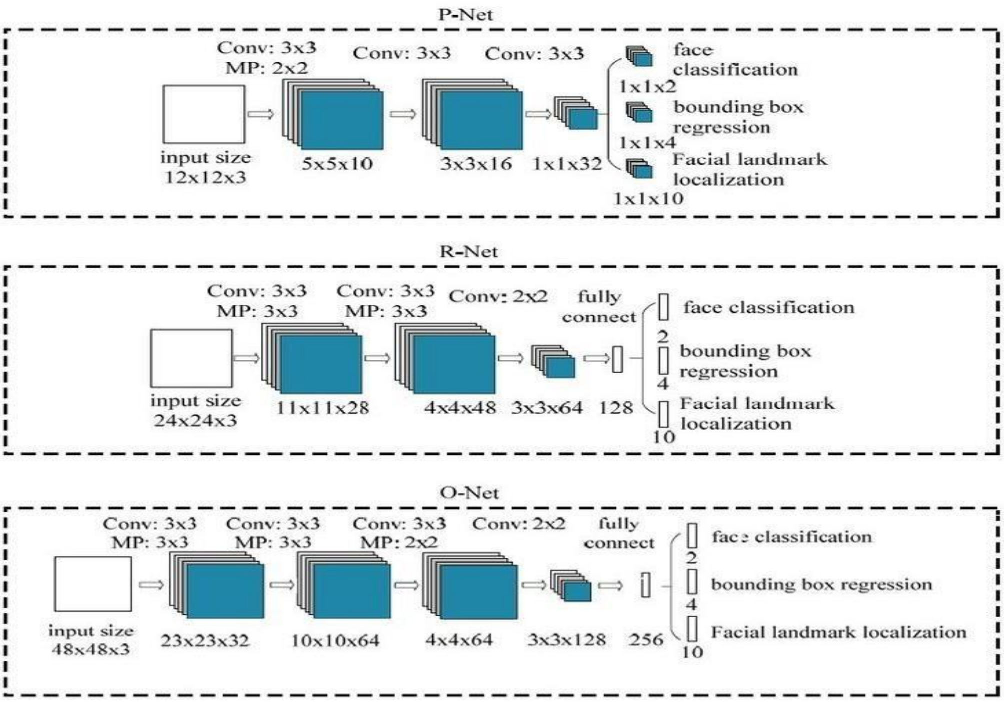


Figure 9: mtCNN Model Architecture

5. CONVOLUTIONAL NUERAL NETWORK

It is a type of neural network helpful in processing data with grid-like architecture, such as images. A digital image is a binary representation of a scene. It consists a matrix arrangement of pixels. Every pixel has some DN number that indicates how bright and what colour image contains.

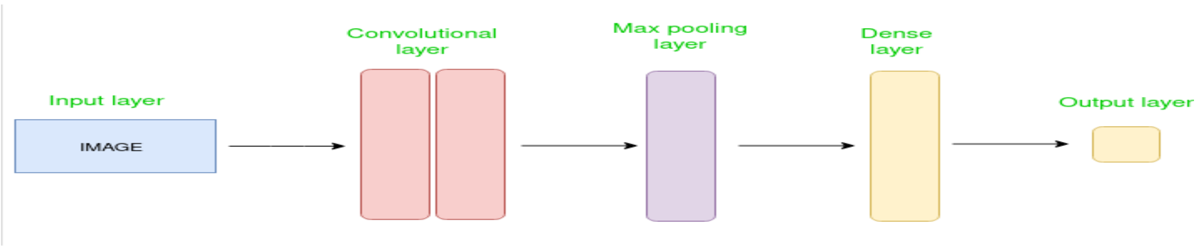


Figure 10. CNN Architecture [gfg]

3.3.Data Preparation

1. Data Collection: We collect image dataset from various open-source platforms that provide CCTV footage and image data with different types of background illumination, facial features, background and resolution. We use 'Microsoft DigiFace' and '14 Celebrity Faces Dataset' image dataset for model training.

2. The dataset we use in this project is a combination of actual and synthetic faces to train the model. FaceNet is a Google AI-developed deep convolutional neural network that extracts facial features from photos. It is trained using a vast dataset of over 200 million photos of faces, including images from the LFW dataset, the CASIA Web Face dataset, and the VGG Face dataset. The actual size of the training dataset is not known; however, it is likely to be in the exabyte range.

3. Number of Attributes, fields, description of the data set:

    a. 14 photos per identity in a subset of 720K images with 10K IDs.
    b. Around 100 hours of CCTV footage.

c. Additionally, we added some of our own actual photos for testing the system in real-time.

4. The training dataset has a diverse variety of face characteristics, races, and ages, which contributes to FaceNet being a robust and accurate facial recognition system. FaceNet has shown cutting-edge performance on a number of facial recognition benchmarks, including the LFW dataset.

5. Image Preprocessing: At image preprocessing stage use ‘PIL’ and ‘numpy’ module. It is a popular python module for image preprocessing. It uses different types of interpolation techniques. we resize the image into 160 by 160 pixels and then convert it into array using ‘numpy’ module.

6. Feature Extraction: We use ‘MultiTask Cascaded Convolutional Networks (mtCNN)’. It extractss feature in two stages, firstly bonding box coordinates after that ‘Inception ResNet V1’ is used for facial feature extraction.

3.4.Implementation

3.4.1. Algorithm / Pseudo code of the Project Problem

1. Gather a dataset of image and CCTV footage for training and testing of the model.
2. Extract frames from the CCTV footage using openCV.
3. Extract relevant features from each image. Common features include facial features using mtCNN.
4. Convert facial features into embedding and store it in database.
5. Now extract Facial features from CCTV footage frame and get the embedding of the same.
6. Compare embedding using facenet.
7. Utilize the validation set to fine-tune the model's hyperparameters for enhanced performance.
8. Evaluate the model's performance on the test set using metrics such as accuracy, precision, recall, and F1 score.

3.4.2. Screen shots of the various stages of the Project

Step 1: Importing necessary libraries and data collection

```
from mtcnn import MTCNN
from cv2 import imread
from PIL import Image
import numpy as np
import os
import matplotlib.pyplot as plt
import pickle

import warnings

warnings.filterwarnings("ignore")
```

```
import cv2
import numpy as np
from keras_facenet import FaceNet
```

```
from sklearn.metrics import accuracy_score, precision_score, f1_score
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import Normalizer
from sklearn.svm import SVC
```

```
import cv2
from matplotlib import pyplot as plt
from deepface import DeepFace
import numpy as np
from moviepy.editor import *
import math
```

Step 2: FaceNet Model Summary

```
Model: "inception_resnet_v1"
__________________________________________________________________________________________________
 Layer (type)                Output Shape                 Param #   Connected to
==================================================================================================
 input_1 (InputLayer)        [(None, None, None, 3)]      0         []

 Conv2d_1a_3x3 (Conv2D)      (None, None, None, 32)       864       ['input_1[0][0]']

 Conv2d_1a_3x3_BatchNorm (B  (None, None, None, 32)       96        ['Conv2d_1a_3x3[0][0]']
 atchNormalization)

 Conv2d_1a_3x3_Activation (  (None, None, None, 32)       0         ['Conv2d_1a_3x3_BatchNorm[0][0
 Activation)                                                        ]']

 Conv2d_2a_3x3 (Conv2D)      (None, None, None, 32)       9216      ['Conv2d_1a_3x3_Activation[0][
                                                                    0]']

 Conv2d_2a_3x3_BatchNorm (B  (None, None, None, 32)       96        ['Conv2d_2a_3x3[0][0]']
 atchNormalization)

 Conv2d_2a_3x3_Activation (  (None, None, None, 32)       0         ['Conv2d_2a_3x3_BatchNorm[0][0
 Activation)                                                        ]']

 Conv2d_2b_3x3 (Conv2D)      (None, None, None, 64)       18432     ['Conv2d_2a_3x3_Activation[0][
                                                                    0]']
...
Trainable params: 23467824 (89.52 MB)
Non-trainable params: 29600 (115.62 KB)
```

Figure 11: Summary of FaceNet model

Step 3: Feature Extraction

Feature extraction using MultiTask cascaded Neural Network is a common method in Face Recognition. The resulting extracted faces serve as feature vectors representing ndarray embedding. In CIS, these features are then used as input to Deep learning model for Criminal Identification. mtCNN-based features are robust to different type facial characteristics and environmental conditions, making them well-suited for Face Recognition.

```
[[-1.14524513e-02  6.93037361e-02  5.02286851e-02  4.98742098e-03
   5.97312152e-02  2.89968140e-02  6.15635701e-02 -3.47620472e-02
   6.98064570e-04 -8.34265053e-02  3.71230058e-02 -7.93191977e-03
  -3.41656408e-03  8.28322470e-02  5.92603199e-02 -7.74029642e-02
  -2.95162592e-02  7.76858032e-02 -2.84852460e-02  4.52459827e-02
  -2.86860783e-02 -4.82577160e-02  4.24610078e-02 -3.66229191e-03
  -9.27957008e-04 -3.49441655e-02  1.24201337e-02  5.28465696e-02
   3.00897867e-03 -3.21040899e-02  3.79555561e-02 -4.03206684e-02
  -5.49265444e-02  1.00202262e-02  8.07551220e-02  6.17043115e-03
   4.31719013e-02  1.97351780e-02  3.18917050e-03 -2.76855733e-02
```

```
1/1 [==============================] - 1s 973ms/step
1/1 [==============================] - 0s 414ms/step
1/1 [==============================] - 0s 129ms/step
1/1 [==============================] - 0s 78ms/step
1/1 [==============================] - 0s 47ms/step
1/1 [==============================] - 0s 32ms/step
1/1 [==============================] - 0s 30ms/step
1/1 [==============================] - 0s 28ms/step
1/1 [==============================] - 0s 30ms/step
1/1 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 27ms/step
1/1 [==============================] - 0s 29ms/step
1/1 [==============================] - 0s 27ms/step
1/1 [==============================] - 0s 26ms/step
166/166 [==============================] - 1s 5ms/step
2/2 [==============================] - 0s 20ms/step
Loaded 1 examples for class: karna
```

Figure 12: Feature Extraction

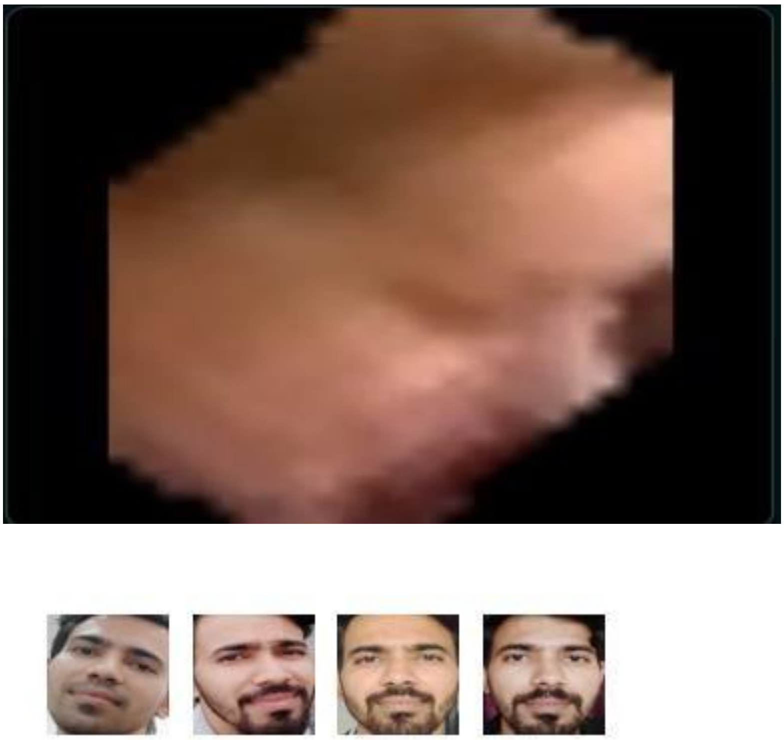


Figure 13: Face Extraction

Step 4: Model Training

Criminal Identification System (CIS) was performed using mtCNN, FaceNet and SVM models. The pre-trained FaceNet model is trained using a vast dataset of over 200 million photos of faces, including images from the LFW dataset. After Than we train our mtCNN model on our custom dataset.

```
1/1 [==============================] - 0s 98ms/step
1/1 [==============================] - 0s 100ms/step
1/1 [==============================] - 0s 161ms/step
1/1 [==============================] - 0s 133ms/step
1/1 [==============================] - 0s 68ms/step
1/1 [==============================] - 0s 69ms/step
1/1 [==============================] - 0s 79ms/step
1/1 [==============================] - 0s 70ms/step
1/1 [==============================] - 0s 64ms/step
1/1 [==============================] - 0s 71ms/step
1/1 [==============================] - 0s 72ms/step
1/1 [==============================] - 0s 74ms/step
1/1 [==============================] - 0s 77ms/step
1/1 [==============================] - 0s 366ms/step
1/1 [==============================] - 0s 75ms/step
1/1 [==============================] - 0s 94ms/step
1/1 [==============================] - 1s 612ms/step
1/1 [==============================] - 0s 123ms/step
1/1 [==============================] - 0s 99ms/step
...
1/1 [==============================] - 0s 66ms/step
1/1 [==============================] - 0s 72ms/step
1/1 [==============================] - 0s 71ms/step
(238, 512)
```

Figure 14: mtCNN model training

```
# Normalize input vectors - We need to normalize the face embeddings
in_encoder = Normalizer(norm="L2")
X_train = in_encoder.transform(newTrainX)
X_val = in_encoder.transform(newTestX)
```
✓ 0.0s

```
# Label encode targets
out_encoder = LabelEncoder()
out_encoder.fit(y_train)
y_train = out_encoder.transform(y_train)
y_val = out_encoder.transform(y_val)
```
✓ 0.0s

```
# fit model
svc_model = SVC(kernel="linear", probability=True)
svc_model.fit(X_train, y_train)
```
✓ 0.2s

```
SVC
SVC(kernel='linear', probability=True)
```

Figure 15: SVM Model Training

Step 5: Face Embedding

Face embedding refers to how a machine stores facial features extracted into a vector array. This vector array is then used to compare faces based on distance, similarity, or for face search purposes.

Load saved embeddings

```python
def load_embeddings(filename):
    with h5py.File(filename, "r") as f:
        embeddings = f["embeddings"][:]
        labels = [label.decode("utf-8") for label in f["labels"][:]]
    return embeddings, labels
```

[6] ✓ 0.0s Python

```python
print("Loading train embeddings...")
train_embeddings, train_labels = load_embeddings("train_embeddings.h5")
print("Loading test embeddings...")
test_embeddings, test_labels = load_embeddings("test_embeddings.h5")
```

[7] ✓ 0.0s Python

```
Loading train embeddings...
Loading test embeddings...
```

Output:

```python
embeddings, labels = load_embeddings("train_embeddings.h5")
```

[8] ✓ 0.0s Python

```python
print(embeddings)
```

[13] ✓ 0.0s Python

```
[[ 0.04590211  0.02315703  0.056639   ...  0.03252606 -0.03999985
  -0.00381483]
 [ 0.01520456  0.03018248 -0.04811997 ... -0.02538411 -0.03673805
  -0.02173346]
 [ 0.03676395  0.00171565  0.0099832  ...  0.03325126 -0.01922811
  -0.08167834]
 ...
 [ 0.0195549  -0.01887512 -0.04664783 ... -0.08956356 -0.00044799
   0.01818916]
 [-0.02421013 -0.01902632 -0.09247315 ... -0.04609853  0.00856372
   0.01690127]
 [-0.01357552 -0.03822539 -0.05966571 ... -0.09271052 -0.01645929
   0.03426121]]
```

Figure 16: Face Embedding

```
for i in range(0, len(embeddings)):
    print(labels[i])
```

[16] ✓ 0.0s Python

```
anne_hathaway
anne_hathaway
anne_hathaway
anne_hathaway
anne_hathaway
anne_hathaway
anne_hathaway
anne_hathaway
anne_hathaway
anne_hathaway
anne_hathaway
anne_hathaway
anne_hathaway
anne_hathaway
anne_hathaway
arnold_schwarzenegger
arnold_schwarzenegger
arnold_schwarzenegger
arnold_schwarzenegger
arnold_schwarzenegger
arnold_schwarzenegger
arnold_schwarzenegger
arnold_schwarzenegger
arnold_schwarzenegger
arnold_schwarzenegger
...
will_smith
will_smith
will_smith
will_smith
```

Step 6: Frame Extraction from CCTV Footage:

After training the models with an existing criminal database, we use the DeepFace library to extract frames from CCTV footage. DeepFace is a hybrid face recognition framework that incorporates state-of-the-art models such as VGG-Face, Google FaceNet, DeepID, ArcFace, Dlib, OpenFace, Facebook DeepFace, and SFace.

```python
def get_facial_frames(video_path):
    faces = []
    cap = cv2.VideoCapture(video_path)  # read video file
    fps = cap.get(cv2.CAP_PROP_FPS)
    fps = math.ceil(fps)
    path = os.path.basename(video_path)

    i = 0
    while cap.isOpened():
        i = i + 1

        ret, frame = cap.read()

        if not ret:
            break

        face_props = DeepFace.extract_faces(
            img_path=frame,
            target_size=(224, 224),
            detector_backend="retinaface",
            enforce_detection=False,
        )

        frame = cv2.cvtColor(face_props[0]["face"], cv2.COLOR_BGR2RGB)
        frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)

        confidence = face_props[0]["confidence"]
        if confidence > 0.990:
            features_dict = {"frames": frame, "confidence": confidence}
            faces.append(features_dict)

        if (i % 10 == 0) and i > 1:
            print(f"Total frames processed {i}")

    return faces, fps, path
```

```python
image_frames = []

for video_frame in video_frames:
    a = video_frame["frames"]
    image_frame = np.interp(a, (a.min(), a.max()), (0, 255)).astype(np.uint8)
    image_frames.append(image_frame)

# Create a video clip from the frames
clip = ImageSequenceClip(image_frames, fps=fps)

# Write the video clip to a file
direct_path = r"D:\\"
clip.write_videofile(direct_path + path)
```

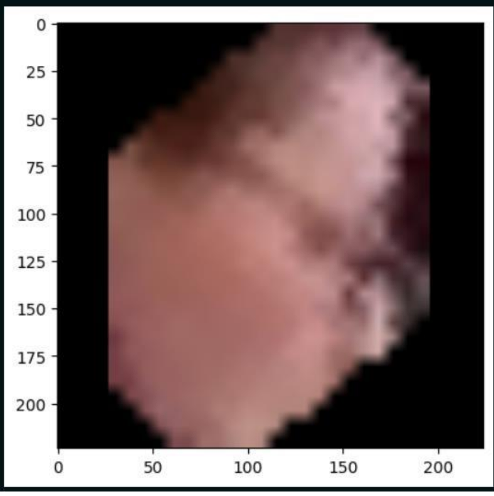
Moviepy - Building video D:\\0.mp4.
Moviepy - Writing video D:\\0.mp4

```python
print(f"Total number of frames in the video: {len(video_frames)}")
print(f"The frame rate per second: {fps}")
print(f"The file name: {path}")
frame_number = 50
print(f"Displaying frame number {frame_number}...")

image = video_frames[frame_number]["frames"]
plt.imshow(image)
```

```
Total number of frames in the video: 217
The frame rate per second: 30
The file name: 0.mp4
Displaying frame number 50...

<matplotlib.image.AxesImage at 0x1e988bac910>
```




```
Total frames processed 150
Total frames processed 160
Total frames processed 170
Total frames processed 180
Total frames processed 190
Total frames processed 200
Total frames processed 210
Total frames processed 220
Total frames processed 230
Total frames processed 240
Total frames processed 250
...
Total frames processed 1760
Total frames processed 1770
Total frames processed 1780
Total frames processed 1790
```

Figure 17: Frame Extraction

Step 7: Image Upscale

Add an image upscale algorithm to raise the quality of older photos or those that have a lower resolution. The image upscaling function enhances the quality of the image by augmenting its resolution. It can be done using dedicated applications. By increasing the resolution of an image, it can improve its quality and make it look more like the original.

```python
# Initialize FSRCNN model
model = cv2.dnn_superres.DnnSuperResImpl_create()
model.readModel("models/FSRCNN_x4.pb")
model.setModel("fsrcnn", 4)  # Set scaling factor to 4x
```
[2] ✓ 0.0s Python

```python
def super_resolve(image):
    return model.upsample(image)

def process_image(input_path, output_path):
    image = cv2.imread(input_path)
    upscaled_image = super_resolve(image)
    cv2.imwrite(output_path, upscaled_image)

def process_video(input_path, output_path):
    cap = cv2.VideoCapture(input_path)
    out = cv2.VideoWriter(output_path,
                          cv2.VideoWriter_fourcc(*'mp4v'),
                          cap.get(cv2.CAP_PROP_FPS),
                          (int(cap.get(cv2.CAP_PROP_FRAME_WIDTH) * 2),
                           int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT) * 2)))

    while True:
        ret, frame = cap.read()
        if not ret:
            break

        upscaled_frame = super_resolve(frame)
        out.write(upscaled_frame)

    cap.release()
    out.release()
```

Output:

```python
print("Size of the image before upscaling: ", cv2.imread(input_path).shape)
print("Size of the image after upscaling: ", cv2.imread(output_path).shape)
```
5] ✓ 0.3s

```
Size of the image before upscaling:  (1280, 960, 3)
Size of the image after upscaling:  (5120, 3840, 3)
```

Figure 18: Image Upscaling

Step 8: Model Evaluation

The Criminal Identification System (CIS) model was evaluated, showcasing their performance through accuracy, precision, recall, f-1 score and auc-roc curve plots.

Performance Metrics:

- **TP**: True Positive: Predicted value is correctly identified as an actual positive.
- **FP**: False Positive: Predicted value is incorrectly identified as an actual positive.
- **FN**: False Negative: Positive value is incorrectly identified as a negative.
- **TN**: True Negative: Predicted value is correctly identified as an actual negative.

Accuracy:

This is one of the widely used performance metrics for assessing the proportion of correct outcomes. Accuracy is calculated by dividing the total number of correct outcomes by the total number of outcomes.

$$Accuracy = \frac{Number\ of\ correct\ Predictions}{Total\ number\ of\ Predictions}$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

Precision:

It measures how close results are to one another.

$$Precision = \frac{TP}{TP + FP}$$

Recall:

It measure how system is identifying true positive correctly

$$Recall = \frac{TP}{TP + FN}$$

F1-score:

The F1-score is an indicator that balances precision and recall, producing a single score that reflects the harmonic mean of the two. It is especially useful when dealing with asymmetrical datasets in which one class vastly outweighs the other.

$$F1 - score = \frac{2 * precision * recall}{precision + recall}$$

AUC-ROC Curve:

The Receiver Operating Characteristic (ROC) curve shows how a model performs across various threshold levels. The Area Under the ROC Curve (AUC-ROC) serves as a measure of the model's overall performance. It represents a plot of recall versus the fall-out rate.

$$TPR = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{TN + FP}$$

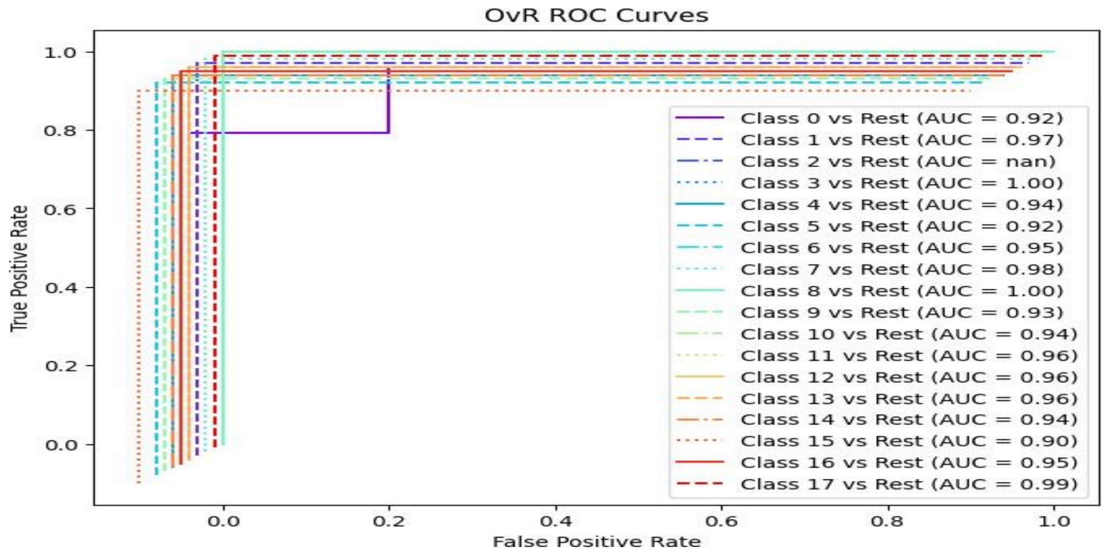


Figure 19: AUC-ROC Curve

Confusion Matrix:

It is a table used to assess the efficacy of a classification algorithm. It provides a comprehensive view of the algorithm's predictions compared to the actual outcomes. The matrix contrasts the predicted classifications against the actual classifications to show the number of correct and incorrect predictions for each class.

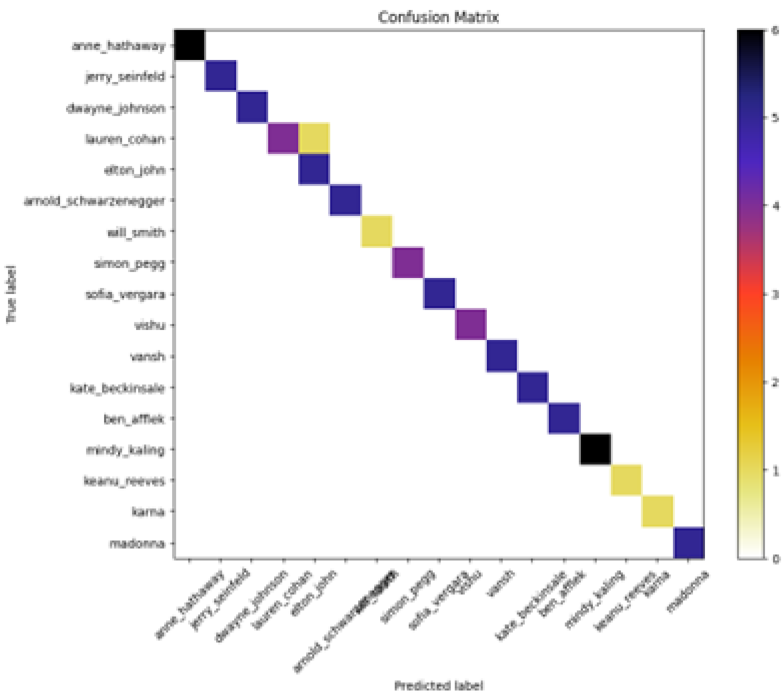


Figure 20: Confusion Matrix

Performance Matrices Comparison

We compare different types of performance matrices like accuracy, precision, recall, f1-score to see how our model is performing for training dataset as well as test dataset.

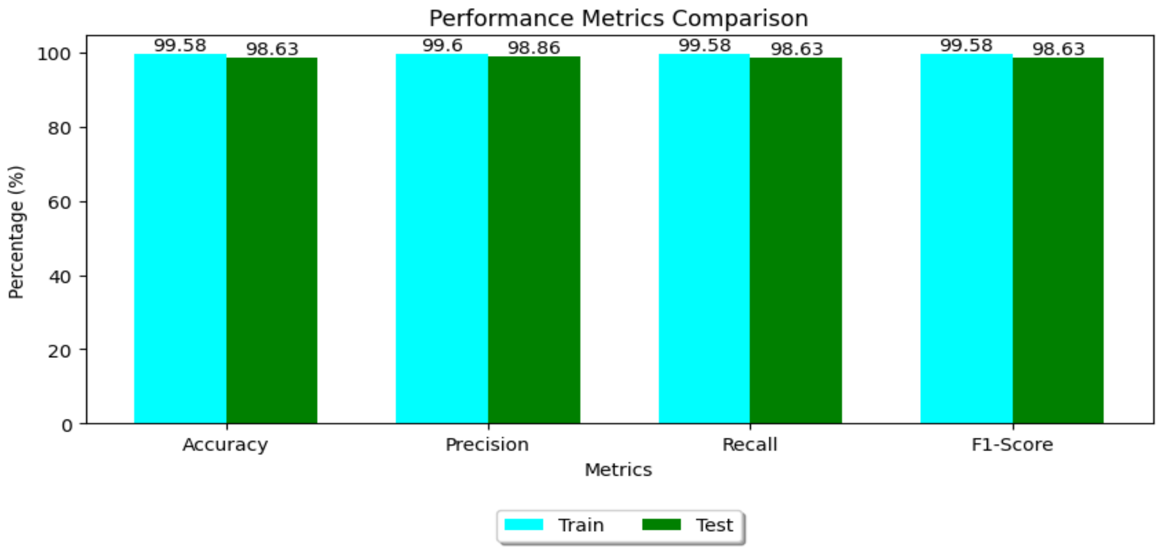


Figure 21: Performance Matrices Comparison

Chapter 4: Testing

4.1. Testing Strategies

We have done Tests while done before testing our model.

1. Ensured the correctness of functions which we are tested in our model like OpenCV, MTCNN, DeepFace, dlib, and FaceNet.
2. Conducted comprehensive testing of the entire multiple face recognition.
3. Using various datasets, we validated system functionality under various scenarios.
4. Evaluated system performance using metrics like accuracy, precision, recall, and F1 score.
5. Monitored and analyzed the results to measure effectiveness of the face recognition model.
6. Summarized the testing process and outcomes.

4.2. Test Cases and Outcomes

4.2.1. Detecting Face in live footage

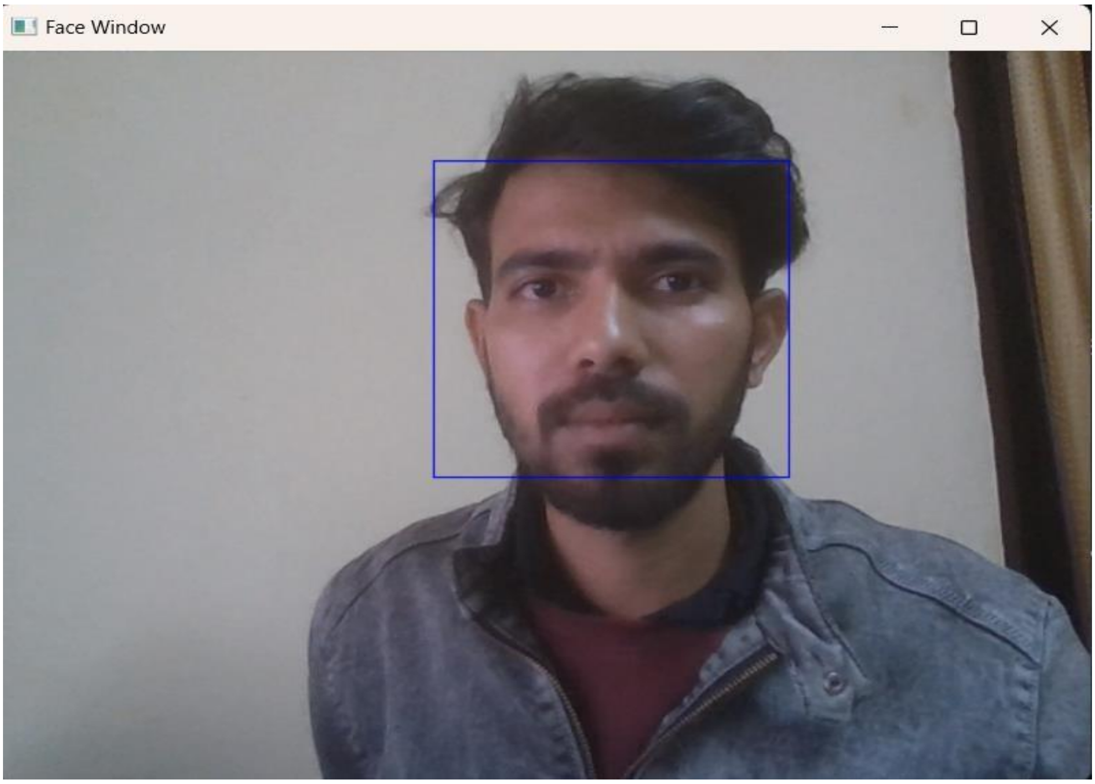


Figure 22: Single Face Detection using openCV

4.2.2. Detecting multiple Faces in Still Image

Figure 23: Multiple Face Detection

Chapter 5: Results

5.1. Discussion on the Results Achieved

Our project "Multiple Face Detection" where we used 3 different models: mtCNN, FaceNet and DeepFace for training of our model. During the training process, our model discovers features and patterns and learns them for the identification purpose. The provided table presents the performance results for the system.

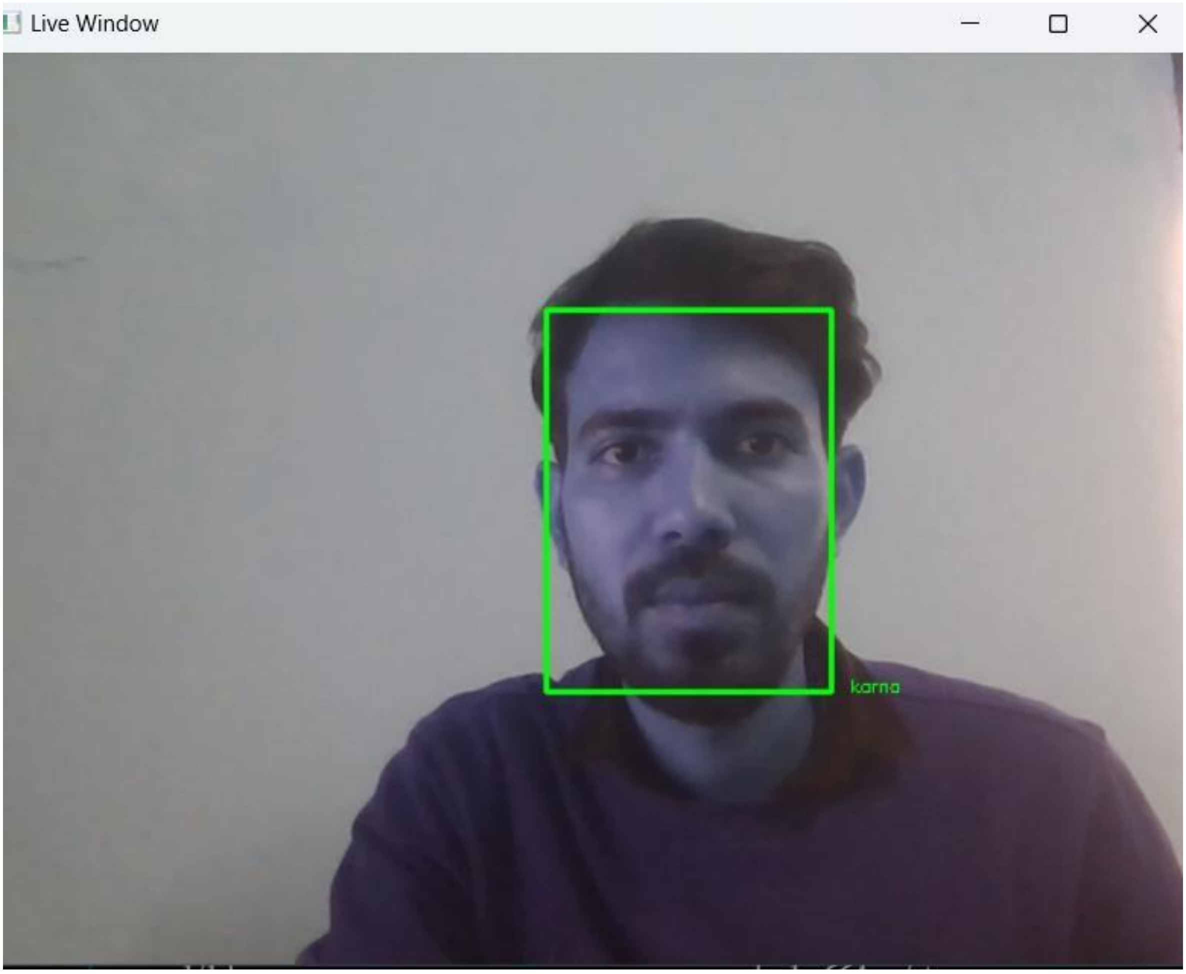


Figure 24: Live Identification

Table 2: Performance Report

S. No.	Performance Measure	Dataset	
		Train (%)	Test (%)
1	Accuracy	99.58	98.63
2	Precision	99.60	98.86
3	Recall	99.58	98.63
4	F1-score	99.57	98.63

5.2. Comparison with existing model

5.2.1. Accuracy comparison of different algorithm

We compare our Multiple Face Recognition system with different face recognition models to see how our model is performing compared to them.

Table 3: Accuracy Comparison Table

S.No.	**Model**	**Accuracy (%)**
1.	OpenCV	93.00
2.	Haar-cascade	83.00
3.	Convolutional Neural Network (CNN)	96.00
4.	MultiTask Convolutional Neural Network (mtCNN)	98.20
5.	FaceNet	99.50
6.	Multiple Face Recognition (MFR)	99.61

5.2.2. Application of the Major or Project

Multiple Face Recognition has various applications across different domains due to its potential to enhance human-computer interaction, improve user experience, and provide valuable insights into human face recognition. Here are some notable applications of Multiple Face Recognition:

1. Human-Computer Interaction (HCI): MFR can be integrated into systems to authenticate user.
2. Education: MFR system can used to automate the attendance task.
3. Access Control Systems: Can be used to manage access based on recognized faces in business buildings, government institutions, research facilities, as well as to get entry to residential areas.
4. Security and Surveillance: Can be used in public spaces such as banks, railway stations, airports, bus stations and stadiums for enhanced security and monitoring, as well as crowed surveillances.
5. Customer Recognition: Improving the shopping experience for customers by identifying and personalizing interactions based on previous visits or preferences.
6. Healthcare:
    a. Patient Identification: Ensuring accurate patient identification for

electronic health records in hospitals and clinics.

   b. Security in Healthcare Facilities: Improving healthcare facility security by recognizing legitimate users.

7. Traffic Management: Identifying and monitoring drivers for traffic management and law enforcement purposes.
8. Law Enforcement:

   a. Criminal Identification: Assisting the police department in identifying and tracking individuals of interest.
   b. Surveillance Operations: Conducting surveillance with the ability to recognize multiple faces in real-time.

9. Social Media and Online Platforms:

   a. Photo Tagging: Automatically tagging individuals in photos on social media platforms.
   b. User Verification: Improving online platform security by using facial recognition for user verification.

As technology continues to advance, the applications of Multiple face Recognition are likely to expand, providing new opportunities for improving various aspects of human-machine interaction and understanding.

5.2.3. Limitation of the Project

While Multiple face Recognition (MFR) holds promise for various applications, it also has several limitations and challenges that need to be considered:

1. Privacy concern: Multiple face recognition includes the possible breach of privacy in public settings, as well as the risk of unauthorized access and misuse of stored facial photos for sensitive personal data.
2. Real-time Processing: Real-time processing is crucial for many applications, but some MFP models may have high computational demands, limiting their effectiveness in real-time scenarios, especially on resource-constrained devices.

Chapter 6: Conclusion and Future Scope

6.1. Conclusion

Evidence is available, and using sketches and other evidence, it is simple to identify the criminal at the crime scene. However, when a crime happens without evidence, the facial recognition system can be used to identify the criminals. These models play a crucial role in identifying the criminal after the incident has occurred.

6.2. Future Scope

Future work includes:

- Training the given dataset on federated learning platform
- Implementing Identification on recorded CCTV footage.

The field of Multiple Face Recognition (MFR) is dynamic, and there are several areas for future research and development. Some potential areas for future work include:

1. Cross-Cultural Adaptation: Improving the generalizability of MFR models across different background illumination, poses, image size, video length is a significant challenge. Future research could focus on developing models that are more adaptable to diverse images and videos contexts.
2. Deep Learning Architectures: Further exploration of advanced deep learning architectures, such as transformer-based models or neural architectures designed for sequential data i.e., for video, could lead to improvements in MFR performance. These architectures may capture more complex features in the image and video data.
3. Real-time Processing and Edge Computing: Enhancing the efficiency of MFR models for real-time processing and deployment on edge devices is crucial for applications in live verification, Identification, and other real-world scenarios. Future work could focus on optimizing models for speed and resource efficiency.

4. Continual Learning and Adaptation: Developing MFR models that can adapt and learn from new data over time is essential for maintaining relevance in dynamic environments. Continual learning approaches could enable models to adapt to different types of images and videos datasets.
5. Unsupervised and Self-Supervised Learning: Investigating unsupervised or self-supervised learning methods for training MFR models could be beneficial, especially in scenarios where labeled training data is scarce. These approaches could leverage unlabeled data to pretrain models and then fine-tune them on smaller labeled datasets.
6. Robustness and Bias Mitigation: Addressing biases in MFR models and improving their robustness across diverse image background and pose. Future research should focus on developing methods to mitigate biases and ensure fair and unbiased Facial feature extraction across different types of images and videos.

References

[1] N, A., Anusudha, K. Real time face recognition system based on YOLO and InsightFace. *Multimed Tools Appl* (2023). https://doi.org/10.1007/s11042-023-16831-7

[2] M. A. Lazarini, R. Rossi, and K. Hirama, “A Systematic Literature Review on the Accuracy of Face Recognition Algorithms ”, *EAI Endorsed Trans IoT*, vol. 8, no. 30, p. e5, Sep. 2022.

[3]. Li, Y. Research and application of deep learning in image recognition. In: 2022 IEEE 2nd International Conference on Power, Electronics and Computer Applications (ICPECA),(IEEE). DOI: https://doi.org/10.1109/ICPECA53709.2022.9718847.

[4] Aherwadi, Nagnath B. and Chokshi, Deep and Pande, Dr. Sagar and Khamparia, Aditya, Criminal Identification System using Facial Recognition (July 12, 2021). Proceedings of the International Conference on Innovative Computing & Communication (ICICC) 2021, Available at SSRN: https://ssrn.com/abstract=3884827 or http://dx.doi.org/10.2139/ssrn.3884827

[5] Abhay Pratap Singh “Criminal Face Identification System”. International Journal for Research in Applied Science & Engineering Technology (IJRASET**)** *ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429Volume 8 Issue V May 2020*

[6] S. Sharma, M. Bhatt and P. Sharma, "Face Recognition System Using Machine Learning Algorithm," 2020 5th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 2020, pp. 1162-1168, doi: 10.1109/ICCES48766.2020.9137850.

[7] Shiva Tamrkar, Ayush Gupta "Criminal Face Detection System Using Python". IJIRT Volume 7 Issue 2 ISSN: 2349-6002 (2020)

[8] S. Ayyappan and S. Matilda, "Criminals And Missing Children Identification Using Face Recognition And Web Scrapping," 2020 International Conference on System, Computation, Automation and Networking (ICSCAN), Pondicherry, India, 2020, pp. 1-5, doi: 10.1109/ICSCAN49426.2020.9262390.

[9] Subramanian, Kanaga Suba Raja. (2020). 22 Face Recognition using Haar - Cascade Classifier for Criminal Identification.

[10] Piyush Chhoriya, "Automated Criminal Identification System using Face Detection and Recognition". International Research Journal of Engineering and Technology p-ISSN: 2395-0072 (2019)

[11] Khan M, Chakraborty S, Astya R, Khepra S (2019) Face detection and recognition using OpenCV. In 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), pp. 116-119. IEEE

[12] P. Apoorva., H. C. Impana., S. L. Siri., M. R. Varshitha. and B. Ramesh., "Automated Criminal Identification by Face Recognition using Open Computer Vision Classifiers," *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, Erode, India, 2019, pp. 775-778, doi: 10.1109/ICCMC.2019.8819850.

[13] K. Rasanayagam, S. D. D. C. Kumarasiri, W. A. D. D. Tharuka, N. T. Samaranayake, P. Samarasinghe and S. E. R. Siriwardana, "CIS: An Automated Criminal Identification System," 2018 IEEE International Conference on Information and Automation for Sustainability (ICIAfS), Colombo, Sri Lanka, 2018, pp. 1-6, doi: 10.1109/ICIAFS.2018.8913367.

[14] K. Bogahawatte and S. Adikari, "Intelligent criminal identification system," 2013 8th International Conference on Computer Science & Education, Colombo, Sri Lanka, 2013, pp. 633-638, doi: 10.1109/ICCSE.2013.6553986.

[15] Ajay Gurav, Alireza Chevelwalla, Sachin Desai, Prof. Sumitra Sadhukhan, 2015, Criminal Face Recognition System, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 04, Issue 03 (March 2015), http://dx.doi.org/10.17577/IJERTV4IS030165

[16] Singh, R., & Gupta, P. (2019). A review of face detection and recognition techniques. International Journal of Computer Applications, 161(6), 1-7.

[17] N, A., Anusudha, K. Real time face recognition system based on YOLO and InsightFace. Multimed Tools Appl (2023).

[18] Shiva Tamrkar, Ayush Gupta "Criminal Face Detection System Using Python". IJIRT Volume 7 Issue 2 ISSN: 2349-6002, (2020).

[19] Subramanian, Kanaga Suba Raja. (2020). 22 Face Recognition using Haar - Cascade Classifier for Criminal Identification.

[20] Piyush Chhoriya, "Automated Criminal Identification System using Face Detection and Recognition". International Research Journal of Engineering and Technology p-ISSN: 2395-0072 (2019).

[21] Khan M, Chakraborty S, Astya R, Khepra S (2019) Face detection and recognition using OpenCV. In 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), pp. 116-119. IEEE.

[22] Alireza Chevelwalla, Ajay Gurav, Sachin Desai, Prof. Sumitra Sadhukhan, "Criminal Face Recognition System", International Journal of Engineering Research & Technology (IJERT) 2015.

[23] Vikram Mohanty, David Thames, Sneha Mehta, and Kurt Luther, "Photo Sleuth: Combining Human Expertise and Face Recognition to Identify Historically Portraits", Conference: the 24th International Conference, March 2019.

[24] Sanika Tanmay Ratnaparkhi, Aamani Tandasi, Shipra Saraswat, "Face Detection and Recognition for Criminal Identification System",2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence) 2021.

[25] P. Apoorva., H. C. Impana., S. L. Siri., M. R. Varshitha. and B. Ramesh., "Automated Criminal Identification by Face Recognition using Open Computer Vision Classifiers," 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2019, pp. 775-778, doi: 10.1109/ICCMC.2019.8819850.

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

PLAGIARISM VERIFICATION REPORT

Date: ……………………….

Type of Document (Tick): **PhD Thesis** | **M.Tech Dissertation/ Report** | **B.Tech Project Report** | **Paper**

Name: ___________________________ __**Department:** _________________ **Enrolment No** _________

Contact No. ______________________________**E-mail.** ______________________________________

Name of the Supervisor: ____

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters): ________________________

____

____

UNDERTAKING

I undertake that I am aware of the plagiarism related norms/ regulations, if I found guilty of any plagiarism and copyright violations in the above thesis/report even after award of degree, the University reserves the rights to withdraw/revoke my degree/report. Kindly allow me to avail Plagiarism verification report for the document mentioned above.

Complete Thesis/Report Pages Detail:

- Total No. of Pages =
- Total No. of Preliminary pages =
- Total No. of pages accommodate bibliography/references =

(Signature of Student)

FOR DEPARTMENT USE

We have checked the thesis/report as per norms and found **Similarity Index** at ……………….(%). Therefore, we are forwarding the complete thesis/report for final plagiarism check. The plagiarism verification report may be handed over to the candidate.

(Signature of Guide/Supervisor) **Signature of HOD**

FOR LRC USE

The above document was scanned for plagiarism check. The outcome of the same is reported below:

<table>
<tr><th>Copy Received on</th><th>Excluded</th><th>Similarity Index (%)</th><th colspan="2">Generated Plagiarism Report Details
(Title, Abstract & Chapters)</th></tr>
<tr><td></td><td rowspan="4">All Preliminary Pages
Bibliography/Images/Quotes
14 Words String</td><td rowspan="2"></td><td>Word Counts</td><td></td></tr>
<tr><td>Report Generated on</td><td>Character Counts</td><td></td></tr>
<tr><td rowspan="2"></td><td>Submission ID</td><td>Total Pages Scanned</td><td></td></tr>
<tr><td></td><td>File Size</td><td></td></tr>
</table>

Checked by
Name & Signature **Librarian**

………

Please send your complete thesis/report in (PDF) with Title Page, Abstract and Chapters in (Word File) through the supervisor at plagcheck.juit@gmail.com

gfdgdfgdgh

ORIGINALITY REPORT

15%	11%	7%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.ijert.org Internet Source	2%
2	S. Ayyappan, S. Matilda. "Criminals And Missing Children Identification Using Face Recognition And Web Scrapping", 2020 International Conference on System, Computation, Automation and Networking (ICSCAN), 2020 Publication	1%
3	Fatima Dakalbab, Manar Abu Talib, Omnia Abu Waraga, Ali Bou Nassif, Sohail Abbas, Qassim Nasir. "Artificial intelligence & crime prediction: A systematic literature review", Social Sciences & Humanities Open, 2022 Publication	1%
4	link.springer.com Internet Source	1%
5	www.americaspg.com Internet Source	1%
6	publications.eai.eu Internet Source	1%

7	Kavushica Rasanayagam, S.D.D.C Kumarasiri, W.A.D.D Tharuka, N.T Samaranayake, Pradeepa Samarasinghe, Samanthi E.R. Siriwardana. "CIS: An Automated Criminal Identification System", 2018 IEEE International Conference on Information and Automation for Sustainability (ICIAfS), 2018 Publication	1 %
8	techdigipro.com Internet Source	1 %
9	www.coursehero.com Internet Source	<1 %
10	ir.juit.ac.in:8080 Internet Source	<1 %
11	Submitted to The Robert Gordon University Student Paper	<1 %
12	papasearch.net Internet Source	<1 %
13	www.ukessays.com Internet Source	<1 %
14	ijsrst.com Internet Source	<1 %
15	Anjeana N, K. Anusudha. "Real timeface recognition system based on YOLO and	<1 %

InsightFace", Multimedia Tools and Applications, 2023
Publication

16	Submitted to Auckland Institute of Studies Student Paper	<1%
17	Khawla Almazrouei, Tamer Rabie. "Survey of Face Recognition Using Eigenfaces", 2023 5th International Conference on Robotics and Computer Vision (ICRCV), 2023 Publication	<1%
18	Submitted to Liverpool John Moores University Student Paper	<1%
19	www.ucl.ac.uk Internet Source	<1%
20	huggingface.co Internet Source	<1%
21	medium.com Internet Source	<1%
22	Submitted to HELP UNIVERSITY Student Paper	<1%
23	ijsrset.com Internet Source	<1%
24	P Apoorva., H.C Impana., S.L Siri., M.R Varshitha., B Ramesh.. "Automated Criminal Identification by Face Recognition using Open	<1%

Computer Vision Classifiers", 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), 2019
Publication

25	Submitted to University of Bedfordshire Student Paper	<1%
26	Submitted to NorthWest Samar State University Student Paper	<1%
27	www.ijcseonline.org Internet Source	<1%
28	kylo.tv Internet Source	<1%
29	Submitted to Sabanci Universitesi Student Paper	<1%
30	ijsrcseit.com Internet Source	<1%
31	dspace.daffodilvarsity.edu.bd:8080 Internet Source	<1%
32	www.ir.juit.ac.in:8080 Internet Source	<1%
33	"Economic Recovery, Consolidation, and Sustainable Growth", Springer Science and Business Media LLC, 2023 Publication	<1%

34	bmcbioinformatics.biomedcentral.com Internet Source	<1 %
35	www.i-asem.org Internet Source	<1 %
36	www.techscience.com Internet Source	<1 %
37	Submitted to University of North Texas Student Paper	<1 %
38	Raja Khurram Shahzad, Syed Imran Haider, Niklas Lavesson. "Detection of Spyware by Mining Executable Files", 2010 International Conference on Availability, Reliability and Security, 2010 Publication	<1 %
39	ijact.in Internet Source	<1 %
40	www.jointdrs.org Internet Source	<1 %
41	www.researchgate.net Internet Source	<1 %

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		