CCTV FOOTAGE ANALYSIS

A PROJECT REPORT

Submitted by

Mayank Gupta – 20BCE1538

CSE4019 – IMAGE PROCESSING
Slot- (E2+TE2)

Project Guide
Dr. Jagdeesh Kannan.
Professor(Grade2)
School of Computer Science and Engineering

Bachelors of Technology
IN
Computer Science and Engineering

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Winter Semester 2022-23

ACKNOWLEDGEMENT

We wish to express our sincere thanks and deep sense of gratitude to our project guide, **Dr. Jagdeesh Kannan,** School of Computer Science and Engineering for her consistent encouragement and valuable guidance offered to us throughout the course of the project work.

We are extremely grateful to **Dr. R. Ganesan, Dean,** School of Computer Science and Engineering (SCOPE), Vellore Institute of Technology, Chennai, for extending the facilities of the School towards our project and for his unstinting support.

We express our thanks to our **Head of the Department** for his support throughout the course of this project.

We also take this opportunity to thank all the faculty of the School for their support and their wisdom imparted to us throughout the courses.

We thank our parents, family, and friends for bearing with us throughout the course of our project and for the opportunity they provided us in undergoing this course in such a prestigious institution.

BONAFIDE CERTIFICATE

Certified that this project report entitled “**FaceTrack AI : Video Frame Analysis Software”** is a bona-fide work of **Mayank Gupta (20BCE1538)** carried out the “J”-Project work under my supervision and guidance for **CSE4019– Image Processing**

Dr. Jagdeesh Kannan

Professor (Grade2)

School of Computer
Science and Engineering

ABSTRACT

The use of Closed-Circuit Television (CCTV) cameras for security surveillance has become ubiquitous in modern society. The extensive use of Closed-Circuit Television (CCTV) cameras for security surveillance in contemporary society. An enormous amount of information is captured by these cameras, making manual analysis difficult. In order to derive valuable information from these data streams, there is a growing demand for automated CCTV footage analysis systems.

In this report, we introduce a novel CCTV footage analyser that employs Convolutional Neural Networks (CNNs) for face recognition and detection, followed by image enhancement and a combination of CNN and Long Short-Term Memory (LSTM) models to generate textual descriptions of the identified individuals.

Our system analyses the raw CCTV footage and employs a CNN-based face detection model to identify individuals in the frame, followed by a CNN-based face identification model to match the identified faces with an existing image of target face augmented into multiple faces. The system then uses image optimization techniques to improve the footage's quality and provide a clearer image for identification. Finally, we generate a textual description of the identified individuals using a combination of CNN and LSTM models. The proposed system has the potential to be utilised in a wide variety of surveillance scenarios, including crime prevention, public safety, and forensic investigations.

LIST OF TABLES

LIST OF FIGURES

TABLE OF CONTENTS

1. *INTRODUCTION*

These days, closed-circuit television, also known as CCTV, is used for a wide variety of day-to-day tasks. CCTV started out as a relatively simple and passive surveillance system but has now developed into an integrated intelligent control system.

Within the context of this research, CCTV video frames serve as the foundation for decision-making to enable the automatic, accurate, and time-saving identification of images that contain humans.

Our proposed approach, which makes use of photographs of a person, is successful in recognising and indicating the presence of the person in a cctv video or images of a busy area. On the other hand, older methods had difficulties recovering correct facial details or extracting face identity from images. In addition to this, it makes it possible for us to combine colour enhancement and restoration, which ends up producing a pleasing balance of realism and realness with a reduced number of artefacts.

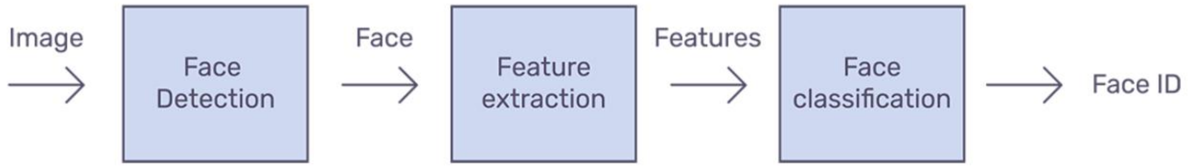


FIGURE-1 Face Recognition Architecture

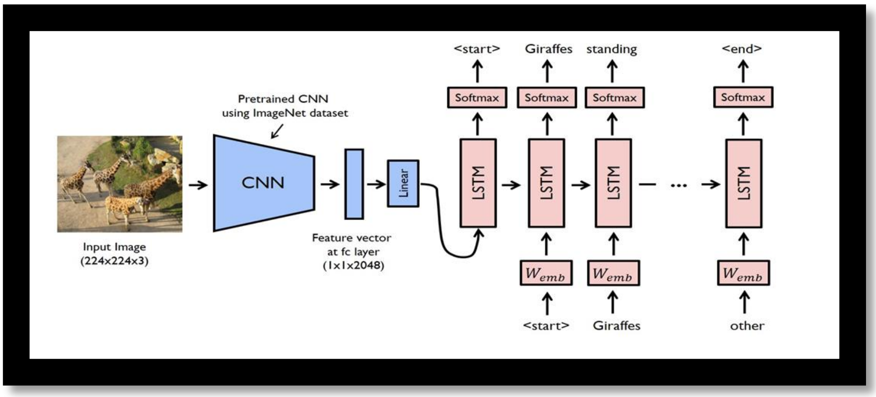


FIGURE-2 a), Image Analysis

Face Detection Face detection involves detecting the bounding box that contains the face in a given image. I chose Multi-task Cascaded Convolutional Network (MTCNN) as the face detector in this program. Our project provides an ideal bounding box that perfectly encapsulates the face without cropping out important facial shapes and features and without including more surrounding area than is necessary.

Feature Extraction Feature Extraction is the key step in the task of Face Identification. In this step, from a human face image obtained by the Face Detection step above, we extract facial component features such as landmark points (like eyes, nose, mouth, etc) and the relation between them. We choose a VGG Neural Network, specifically the Resnet-50 based VGGFace2 model developed by researchers at the Visual Geometry Group at Oxford.

Classification In this step, a classifier decides whether the face in the image matches the identifier face based on the information provided to it. Cosine Similarity suits our use case best, as it's akin to observing how close our images are in an N-dimensional feature space, by measuring the cosine of the angle between the feature points of the images.

In our case, we feed the feature vectors of the ID image and the test image to the cosine similarity function. Using a threshold, we decide whether the face matches or not.

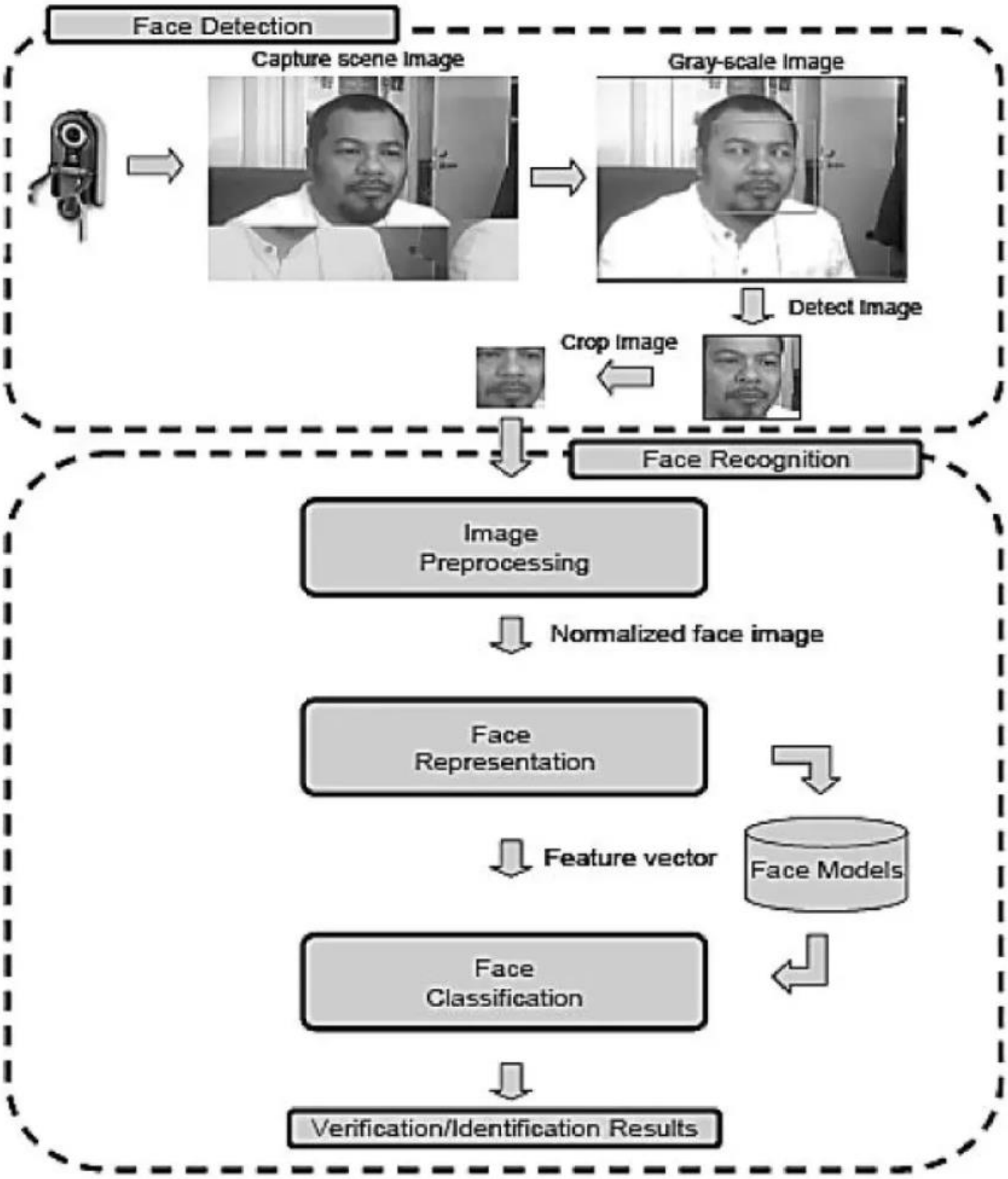


FIGURE-2 b), Image Analysis Flow

The work presented in this project we also develop a Image Caption Generator model. We will systematically analyse a deep neural networks-based image caption generation method. With an recognized image as the input, the method will output an English sentence describing the content in the image. We analyse three components of the method: **convolutional neural network (CNN), recurrent neural network (RNN) and sentence generation**. Then will be implementing the caption generator using CNN (Convolutional Neural Network) by help of Tensor flow module and LSTM (Long short term memory). The image features will be extracted from Exception which is a CNN model trained on the Fliker_8k dataset and then we feed the features into the LSTM model which will be responsible for generating the image captions.

We will develop the caption generator in Python using CNN (Convolutional Neural Network) and LSTM (Long short term memory). The picture characteristics will be retrieved from Exception, a CNN model trained on the three hypothetical datasets, and then fed into the LSTM model, which will generate the image descriptions.

CNN is used to extract features, whereas ***LSTM*** is used to store words individually and form phrases. Not only should the caption be able to identify the object, but it should also be able to construct a coherent statement that characterizes the action in the image. When compared to autonomous manufacturing utilizing deep learning, earlier systems rely on indexing, labelling, and categorizing pictures, which may be a huge waste of human effort.

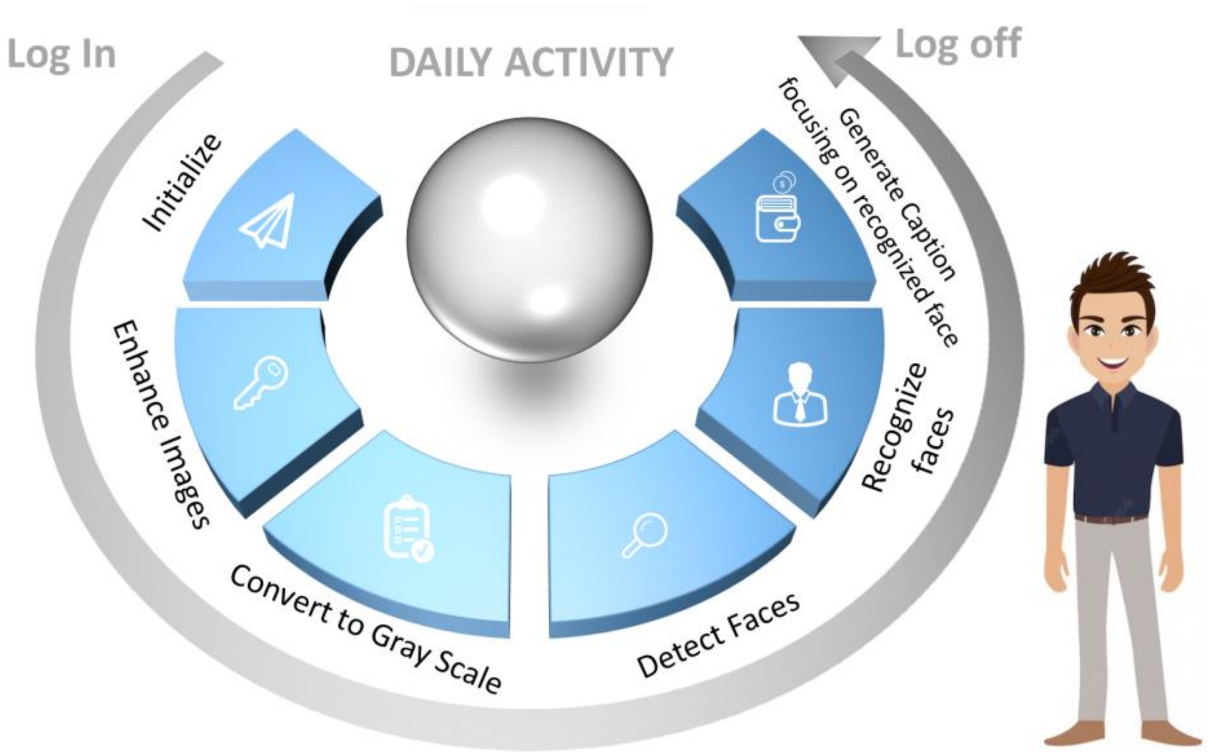


FIGURE-3, Working of the Model

2. RELATED WORKS

Image caption creation is becoming one of the most often used and essential tools since it has so many uses, from CCTV monitoring to helping the blind. The procedure includes recognising important items, figuring out how they relate to one another, and then creating sentences that are both semantically and syntactically sound. Technologies like this are frequently used to create image caption generators:

1. Computer Vision (CV)
2. Natural Language Processing (NLP)
3. Convolution Neural Networks (CNN)
4. Recurrent Neural Networks (RNN)

Various methodologies have been applied to create Image Caption Generating Systems

[1] **Megha J Panicker, Vikas Upadhayay, Gunjan Sethi and Vrinda Mathur** , In order to achieve the goal, the study suggests using the Flickr8k dataset and the ML method Transfer Learning with the use of the Xception model. The outcomes were evaluated against BLEU.(Bilingual Evaluation Understudy). Text translation uses BLEU ratings to compare translated text to one or more reference translations. This model of image caption creation and testing using the dataset with the BLEU has shown to be extremely successful and has provided helpful insight into a potential implementation strategy..

[2] Another method proposed by **Parth Kotak and Prem Kotak** in their paper, is the use of Long Short-Term Memory (LSTM) network, a type of Recurrent Neural Network (RNN) capable of learning order dependence in sequence prediction problems. The justification for using this specific type of RNN is that; when we go deeper into a neural network if the gradients are very small or zero, then little to no training can take place, leading to poor predictive performance and this problem was encountered when training traditional RNNs. LSTM networks are well- suited for classifying, processing, and making predictions based on time series data since there can be lags of unknown duration between important events in a time series. Also LSTM is more effective and better compared to the traditional RNN as it overcomes the short term memory limitations of the RNN

[3] **In the study titled “Visual Image Caption Generator Using Deep Learning, Grishma Sharma, Priyanka Kalena, Nishi Malde, Aromal Nair, Saurabh Prakar(2019)”,** a For framing our words from the provided input images, a comparison between the two architectures, LSTM (Long Short Term Memory) and GRU (Gated Recurrent Unit), was made. The BLEU (Bilingual Evaluation Understudy) score is used to compare the performances of LSTM and GRU and determine which architecture is superior. The findings demonstrate that, although requiring more time for training and sentence creation due to its complexity, the LSTM model generally performs somewhat better than GRU.

[4] **Khansaa, Dheyaa, Ismael., Stanciu, Irina. (2020). Face recognition using viola-jones depending on python.**

The proposed software system is based on face recognition technology and can be used in smart buildings for security purposes. It uses the Viola–Jones object detection framework with Python to detect human faces from a stream of pictures or video feed. This facial recognition system consists of two steps: detecting the human face using web camera, and recognizing if this person is allowed access into the building by comparing it against an existing database.

This paper concludes that the proposed software system is a suitable and low-consuming way of detecting and recognizing faces, which can be used to control access in smart buildings. It uses OpenCV library with Python for its facial recognition algorithm, allowing only authorized persons into the building according to their face recognition results.

[5] **Ramanpreet, Kaur, Deol. (2018). Intruder Detection System Using Face Recognition for Home Security IoT Applications: A Python Raspberry Pi 3 Case Study.**

It explains how an Intruder Detection System (IDS) can be developed using Face Recognition technology for Home Security IoT Applications, with Python and Raspberry Pi 3 as its case study. This system will help to detect intruders by recognizing their faces from images or videos captured through cameras installed at home security systems. In addition, it also discusses various challenges faced while developing such IDSs and suggests possible solutions to overcome them. The paper uses a dataset of images and videos captured from cameras installed at home security systems for the experiments. The data is used to train an Intruder Detection System (IDS) using facial recognition technology, which can then be tested on different scenarios such as varying lighting conditions or angles of view.

[6] **Shijie, Qiao., Jie, Ma. (2018). A Face Recognition System Based on Convolution Neural Network.**

A facial recognition system built on a convolution neural network is presented in the paper. Four convolutional layers, three pooling layers, one full-connected layer, and one softmax regression layer make up this system. The tan(h) activation function and the ReLU activation function are combined to form a new activation function. For training purposes, four networks with varying numbers of convolution kernels were created using the Python programming language and the Theano framework. These networks are trained and tested using the ORL Face Database, which demonstrates that as the number of kernels rises, misrecognition rate falls and performance increases in comparison to the other two activation functions. The findings of this study demonstrate that when the number of convolution kernels rises, the rate of misrecognition falls and the new activation function outperforms the previous two activation functions in terms of performance..

[7] Chongyi, Li., Chongyi, Li., Jichang, Guo., Fatih, Porikli., Yanwei, Pang. (2018). LightenNet: a Convolutional Neural Network for weakly illuminated image enhancement.

The proposed method, LightenNet is a trainable Convolutional Neural Network (CNN) for weakly illuminated image enhancement. It takes a weakly illuminated image as input and outputs its illumination map that can be used to obtain the enhanced image based on Retinex model. Qualitative and quantitative comparisons are conducted to evaluate the performance of this method which shows superior results than existing methods. A new approach has been proposed for synthesizing weakly illuminated images which can help in training networks related to such tasks or full-reference quality assessment purposes.

The proposed method, LightenNet, produces visually pleasing results without over or under-enhanced regions. Qualitative and quantitative comparisons were conducted to evaluate the performance of the proposed method which showed that it achieved superior performance than existing methods. A new weakly illuminated image synthesis approach was also proposed as a guide for training networks and full-reference image quality assessment.

[8] A., V., N., Kameswari. (2021). Image Caption Generator Using Deep Learning.

The paper is about a research paper that focuses on developing an Image Caption Generator using Deep Learning techniques. To build this model, researchers used datasets and computer power with advanced deep learning methods like CNN (Convolutional Neural Network) and LSTM (Long Short-Term Memory). Finally, its accuracy was measured by calculating Bleu Score which is a metric used to measure how well generated caption matched reference captions. They tested their model by giving it different types of pictures through command prompt and then calculated its accuracy based on how well generated caption matched reference captions - this process was called Bleu Score calculation.

[9] Palak Kabra, Mihir Gharat , Dhiraj Jha , Shailesh Sangle(2022) Image Caption Generator Using Deep Learning

The proposed paper aims to generate a description of an image also called as image captioning, using CNN-LSTM architecture such that CNN layers will help in extraction of the input data and LSTM will extract relevant information throughout the processing of input such that the current word acts as an input for the prediction of the next word. It can be used as a plugin in currently trending social media platforms to recommend suitable captions for people to attach to their post or can be used by visually impaired people to understand the image content on the web thus eradicating any ambiguity in image meaning in turn also free of any discrepancy in knowledge acquisition.

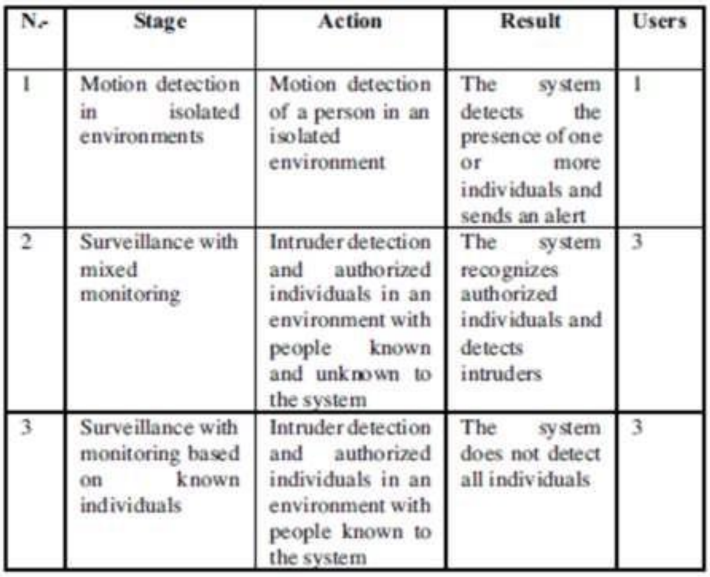
[10] Seung-Ho Han, Ho-Jin Choi (2020), Domain Specific Image Caption Generator with Semantic Ontology

Recent models have utilized deep learning techniques for this task to gain performance improvement. However, these models can neither fully use information included in a given image such as object and attribute, nor generate a domain-specific caption. To overcome these limitations, this paper proposes a domain-specific image caption generator, which generates a caption based on attention mechanism with object and attribute information, and reconstructs a generated caption using semantic ontology to provide natural language description for a given specific-domain

With our project, we propose to analyses three components of the method: convolutional Neural Network (CNN), Recurrent Neural Network (RNN) and sentence generation. Then will be implementing the caption generator using CNN (Convolutional Neural Network) by help of Tensor flow modules and LSTM (Long Short-Term Memory). The image features will be extracted from exception which is a CNN model trained on the Flikr_8k dataset and then we feed the features into the LSTM model which will be responsible for generating the image.

[11] Guillermo Casanova, Daniel Yandún, Graciela Guerrero(2020), Analysis of video surveillance images using computer vision in a controlled security environment.

The problem of managing and processing the increasing amount of video surveillance data has led to the need for faster and more efficient technologies, such as computer vision. The objective of this project is to develop a video surveillance system that records the faces of visitors and compares them with a database of registered users. The methodology adopted involves using Python programming language and OpenCV package with Haar Cascade algorithms to create, delete, and transfer information to the user. Adaptive Boosting (Adaboost) algorithms are used for self-learning and classification of positive and negative images to improve the efficiency of image classifiers. The system retrieves data from MongoDB database and API Face, and sends alerts via the SMTP protocol of Gmail to registered users if a match is not found. III). Enhance the system's usability by adding a more user-friendly interface.



N.-	Stage	Action	Result	Users
1	Motion detection in isolated environments	Motion detection of a person in an isolated environment	The system detects the presence of one or more individuals and sends an alert	1
2	Surveillance with mixed monitoring	Intruder detection and authorized individuals in an environment with people known and unknown to the system	The system recognizes authorized individuals and detects intruders	3
3	Surveillance with monitoring based on known individuals	Intruder detection and authorized individuals in an environment with people known to the system	The system does not detect all individuals	3

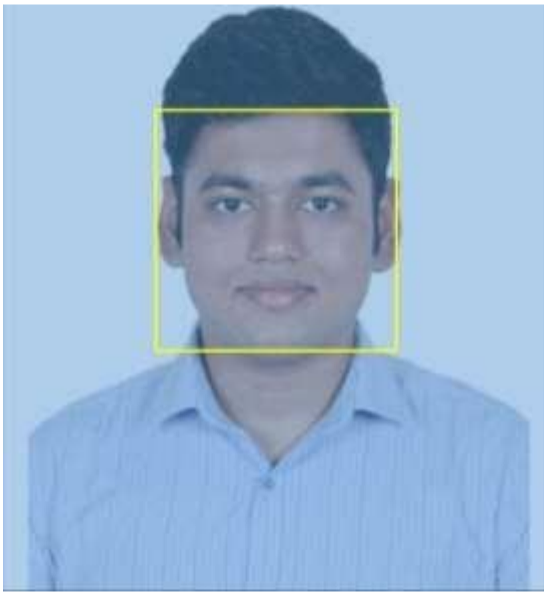


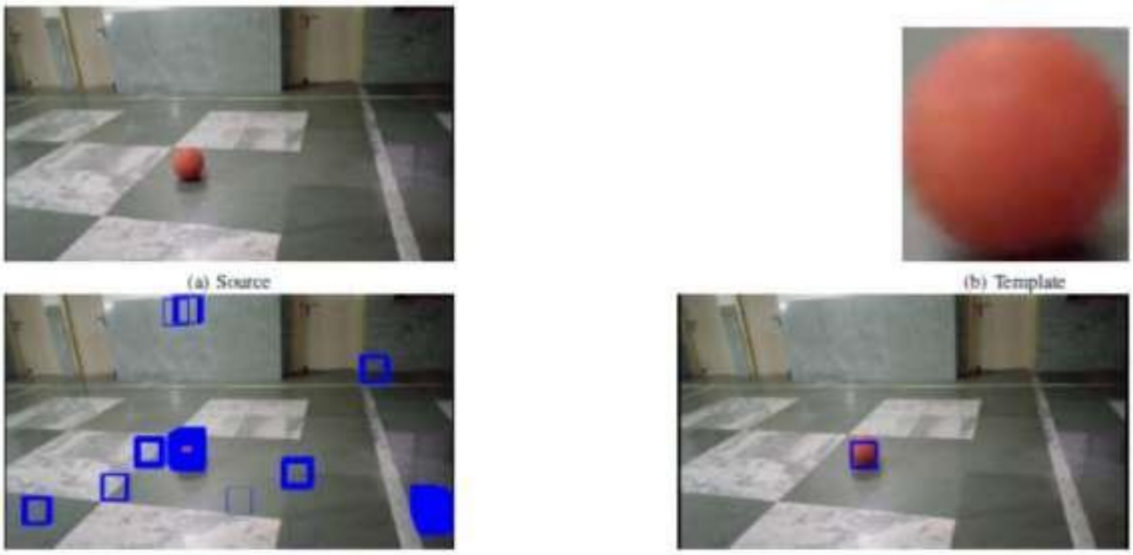
FIGURE-4, Reviewed Paper 11 Implementation and test with my image

[12] N. Funde, P. Paranjape, K. Ram, P. Magde, and M. Dhabu, "Object Detection and Tracking Approaches for Video Surveillance Over Camera Network," in 2019 International Conference on Advanced Computing and Communication Systems (ICACCS), 2019, pp. 1095-1100, doi: 10.1109/ICACCS.2019.8728518

The paper "Object Detection and Tracking Approaches for Video Surveillance Over Camera Network" discusses the challenges of video surveillance in computer vision and the processes of object detection and tracking. The aim of the study was to create an effective object tracking system for straightforward situations with a static camera and a plain background. The methodology involved inputting the network's geometry and videos from all cameras, selecting the target object, and using frame differencing, optical flow, and background subtraction techniques to track the object's movement. The study also evaluated the results of template matching and nodes configuration. Overall, the paper offers insights into the use of computer vision in video surveillance and the importance of efficient object detection and tracking.

The various stages of object identification and tracking have been analysed and discussed in depth during the course of this research article. The numerous approaches for each of these stages are broken down in great detail. But the implementation of these algorithm takes time which slows the real-time application of the algorithm. Newly launched Yolo V6 will be the best algorithm to support the application with real-time objet detection

The current work did not use a device compatible with the new Yolo V6


(a) Source

(b) Template

(c) τ = 80%

(d) τ = 95%

FIGURE-5, Template Matching

[13] Chavda, H. K., & Dhamecha, M. (2018). Moving object tracking using PTZ camera in video surveillance system. In 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS) (pp. 2926-2930). IEEE. doi: 10.1109/ICECDS.2017.8389917.

The paper titled "Moving object tracking using PTZ camera in video surveillance system" by Hetal K. Chavda and Maulik Dhamecha was published on June 21, 2018, by IEEE. The paper discusses the challenges of object detection and tracking in video surveillance systems and focuses on the use of pan-tilt-zoom (PTZ) cameras for high-resolution image capture from a distance. The authors use MATLAB R2014a to evaluate the performance of their algorithm, using True Positive Rate (TPR) and False Positive Rate (FPR) as evaluation parameters. The paper concludes that PTZ cameras have potential for use in autonomous surveillance systems, but further research is needed to address the challenges of background models for moving cameras and optical geometrical projection models.

To keep the target in the FOV camera, should add camera zoom and improve motion prediction

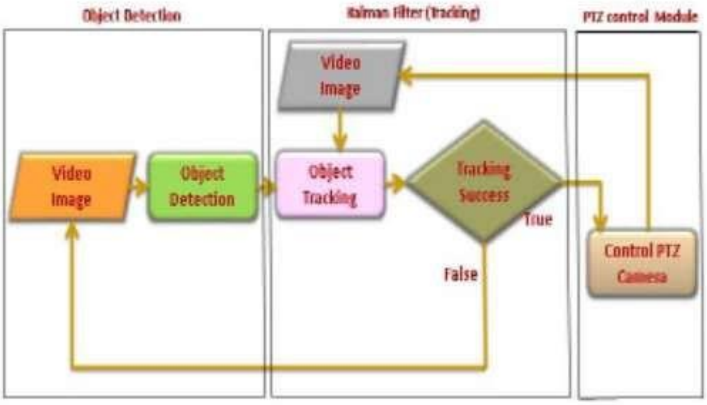


FIGURE-6, Object Detection and Tracking System using PTZ Camera

[14] Veer, N.D., & Momin, B.F. (2017). An automated attendance system using video surveillance camera. IEEE Xplore. DOI: 10.1109/RTEICT.2016.7808130.

The paper discusses the development of an automated attendance system using video surveillance cameras. The system consists of two primary phases: student enrollment and student recognition. In the student enrollment phase, students register in front of a laptop webcam, and Viola-Jones face detection is used to identify student faces. The faces are stored in a database with varied orientations upon registration. In the student recognition phase, features are extracted from the faces displayed in the frames, and then compared to the features stored in the database of previously recognized faces using Local Binary Patterns (LBP). The paper highlights the limitations of current attendance systems and proposes a system that records the attendance of any student who is present in a classroom after the allotted time has passed, eliminating the need for active engagement from students.

The present research doesn’t take into account algorithm like deep neural network which is changing the course of life.

There are still a lot of obstacles to overcome before the suggested technology may be widely used. These include the upfront cost of implementation, deployment, training, environmental factors, and other aspects

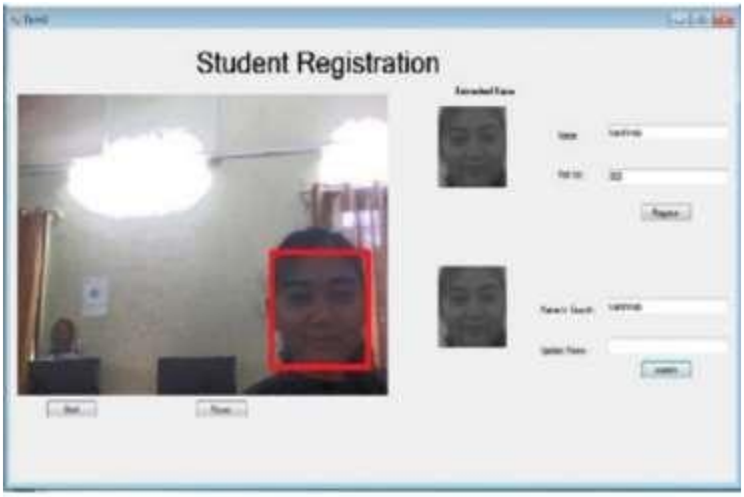


FIGURE-7, Student Registration Process

[15] Syam Kakarla, Priyaranjan Gangula, M. Sai Rahul, C. Sai Charan Singh, and T. Hitendra Sarma, "Smart Attendance Management System Based on Face Recognition Using CNN," in 2020 12th International Conference on Communication Systems & Networks (COMSNETS) Hyderabad, India, 2020, pp. 842-847, doi: 10.1109/HYDCON48903.2020.9242847.

The paper proposes a novel CNN architecture for a smart attendance management system based on face recognition. The methodology adopted includes using an automated technique to gather student face data and a web-based software called Smart Attendance Management System (SAMS). The CNN model was developed, including the data collecting and data augmentation processes. The results of the experiments demonstrate the effectiveness of the suggested CNN model as well as the online application SAMS. However, the sample size may be small, limiting the generalizability of the findings. Additionally, the application's real-time performance may be improved to make it more suitable for security applications.

[16] Sattar, S. A., Rahman, M. A., Khan, T. A., Dipto, N. A., & Islam, M. S. (2020, October 28). Design and Implementation of Chatbot Framework For Network Security Cameras. In 2020 4th International Conference on Emerging Trends in Information Technology and Engineering (ICETITE) (pp. 1-6). IEEE. doi: 10.1109/ETITC51402.2020.9293471

The paper proposes a chatbot framework called "Security Bot" (Sbot) to assist users in obtaining information about human detection from network security cameras. The framework comprises a camera network, a Human Detection Server (HDS), and an Sbot server, which uses Facebook Messenger to transfer data between users and HDS. The SSDMobileNetV1 network architecture is used for real-time human detection, and the team updates the dataset for retraining using transfer learning with more data from security camera footage. NEED:Surveillance cameras are becoming an essential component of smart home systems, and the paper highlights the challenge of extracting critical data from the camera footage, such as human behaviour, license plate information, vehicle tracking, person count, etc. Overall, the proposed chatbot framework offers an innovative solution to the challenge of extracting critical data from network security cameras and provides a user-friendly interface for users to interact with the camera network.

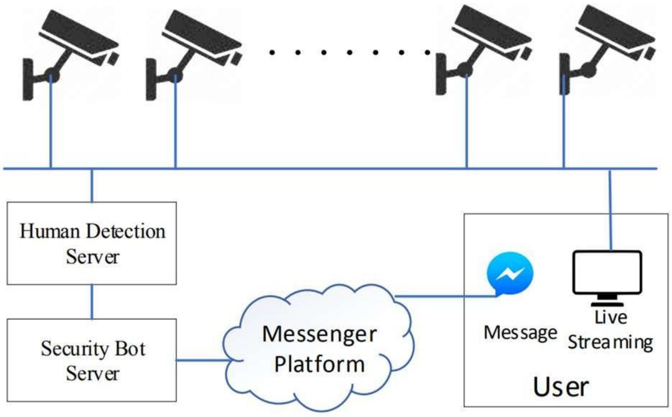


FIGURE-8, Proposed Topology System

3. PROPOSED METHODOLOGY

The working of our model consists of five distinct steps namely:

Face Detection

1. **Image Enhancement**
2. **Face Detection**

Face Recognition

3. **Image Augmentation**
4. **Face Attribute Extraction and Matching**
5. **Face Recognition**

Caption Generation

6. **Text Pre-processing**
7. **Output Model Prediction**
8. **Fitting the Model**
9. **Caption Generation**

1. Image Enhancement

Image enhancement is the process of improving the quality of an image by adjusting its various attributes such as brightness, contrast, color balance, and sharpness. Converting an image to grayscale is one of the most common techniques used for image enhancement, particularly in the context of face detection.

In face detection, the goal is to identify and locate faces within an image. One way to achieve this is by analyzing the intensity values of the image. By converting the image to grayscale, we can obtain a single channel intensity image that can be used for face detection. This is because in a grayscale image, each pixel represents the brightness or intensity of the corresponding area in the original image.

Overall, converting an image to grayscale can be an effective technique for image enhancement in the context of face detection. It allows for easier analysis of the intensity values, which are a critical component of many face detection algorithms.

2. Image Augmentation

Image augmentation is a technique used in computer vision and machine learning to increase the size and diversity of a dataset by generating new variations of existing images. It is particularly useful in scenarios where the available dataset is limited, and the model needs to be trained with a larger number of images to improve its accuracy and generalization.

The process of image augmentation involved applying a set of transformations to the original images. These transformations can include **rotations, translations, scaling, flipping, cropping, adding noise, and changing the brightness and contrast**. By applying these transformations, we can generate a diverse set of images that are similar but not identical to the original images.

Image augmentation was used to train models for facial feature extraction by generating new variations of the original facial images that can be used to train the model. Facial feature extraction is a common task in computer vision that involves detecting and localizing specific facial features such as eyes, nose, mouth, and eyebrows.

The process of image augmentation for facial feature extraction is similar to the process for other computer vision tasks. It involves applying a set of transformations to the original facial images, such as scaling, rotation, and flipping, to generate new images. These new images can then be used to train the model to detect facial features from different angles, positions, and scales.

For example, in the case of eye detection, image augmentation can be used to generate new images where the eyes are at different positions, sizes, and angles. This helps the model learn to recognize eyes in different contexts and improves its ability to generalize to new images..

3. Face Detection

In face detection, the goal is to identify and locate faces within an image. One way to achieve this is by analyzing the intensity values of the image. By converting the image to grayscale, we can obtain a single channel intensity image that can be used for face detection. This is because in a grayscale image, each pixel represents the brightness or intensity of the corresponding area in the original image.

MTCNN, which stands for Multi-Task Cascaded Convolutional Neural Network, is a popular face detection and facial landmark localization algorithm that was introduced in 2016 by Zhang et al. MTCNN is known for its high accuracy and robustness in detecting faces of different sizes, poses, and orientations in real-world scenarios.

The MTCNN algorithm consists of three stages: proposal network (P-Net), refinement network (R-Net), and output network (O-Net). The P-Net stage generates candidate bounding boxes for faces using a sliding window approach and a convolutional neural network. The R-Net stage then filters out false positives and refines the candidate bounding boxes using another neural network. Finally, the O-Net stage performs facial landmark localization and outputs the final bounding boxes for the detected faces.

4. Face Attribute Extraction and Matching

Face attribute matching with ResNet50 involves using a pre-trained deep neural network called ResNet50 to identify and match specific attributes in facial images.

ResNet50 is a convolutional neural network architecture that has been trained on large image datasets, such as ImageNet, and is capable of extracting high-level features from images. This makes it a suitable choice for tasks such as face attribute matching.

To perform face attribute matching with ResNet50, you would typically use a dataset of labeled facial images with corresponding attribute labels (e.g., gender, age, facial expression, etc.). You would then use ResNet50 to extract features from the images and train a classifier to predict the attribute labels based on these features.

Overall, face attribute matching with ResNet50 can be a powerful tool for a range of applications, from facial recognition to emotion analysis and beyond.

5. Face Recognition

Facial recognition involves identifying and matching a face to a specific person. This is typically done by comparing a facial image to a database of known faces and attempting to find a match.

Both face attribute matching and facial recognition rely on similar technologies, such as deep learning algorithms and neural networks. However, they serve different purposes and involve different approaches to data processing and analysis.

6. Image Feature Extraction

The feature extractor needs an image 224x224x3 size. The model uses ResNet50 pretrained on ImageNet dataset where the features of the image are extracted just before the last layer of classification. Another dense layer is added and converted to get a vector of length 2048.

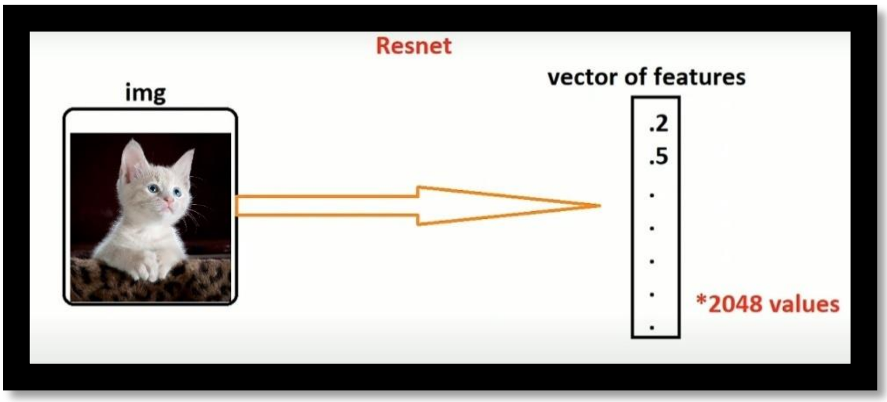
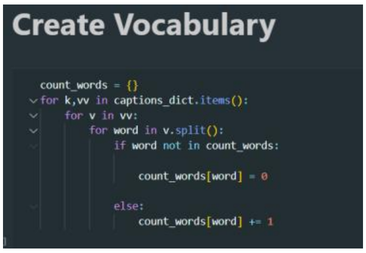


FIGURE-9, Image Feature Extraction

7. Text Pre-processing

Unique words are tokenized from the training dataset. As computers do not understand English words, we have represented them with numbers and mapped each word of the vocabulary with a unique index value and we encoded each word into a fixed sized vector and represented each word as a number.



Create Vocabulary

```
count_words = {}
for k,vv in captions_dict.items():
    for v in vv:
        for word in v.split():
            if word not in count_words:

                count_words[word] = 0

            else:
                count_words[word] += 1
```

FIGURE-10, Text Preprocessor

8. Output Model Prediction

Output vector from both the image feature extractor and the text processor are of same length (128) and a decoder merge both the vectors using an addition operation. This is then fed into two dense layers. This layer uses softmax activation function to predict the most probable next word in the vocabulary.

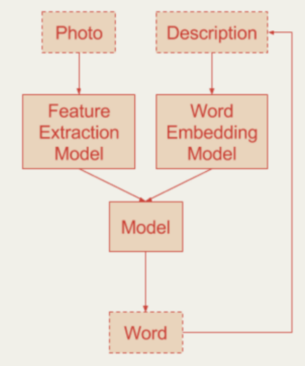
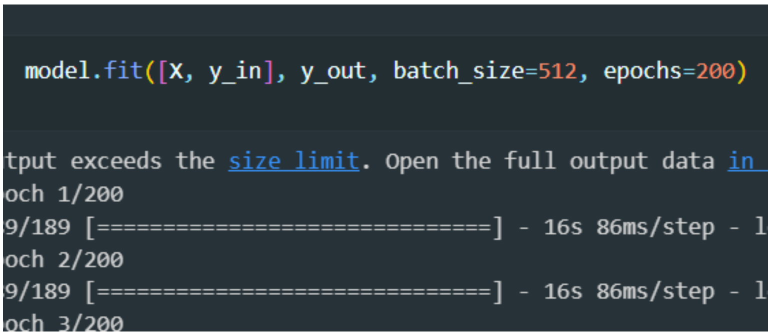


FIGURE-11 Output Model Prediction

9. Fitting the Model

After building the model, the model is fit using the training dataset. The model is made to run for 210 epochs and the best model is chosen among the 210 epochs by computing loss function on Flickr8k development dataset. The model with the lowest loss function is chosen for generating captions.



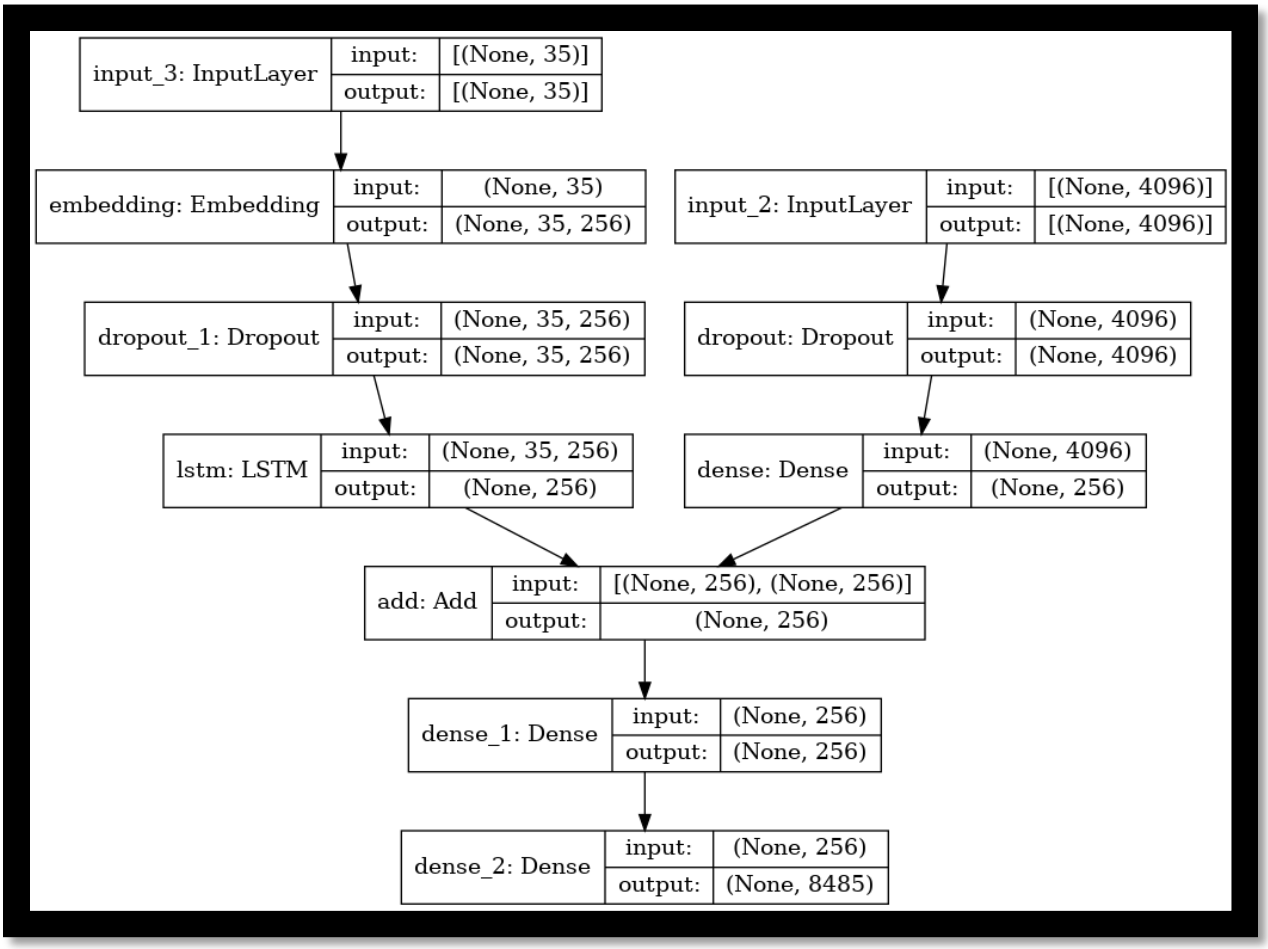
```
model.fit([X, y_in], y_out, batch_size=512, epochs=200)
```

```
tput exceeds the size limit. Open the full output data in
och 1/200
9/189 [==============================] - 16s 86ms/step - l
och 2/200
9/189 [==============================] - 16s 86ms/step - l
och 3/200
```

FIGURE-12, Fitting the Model

The cumulative output of all the above steps is combined to obtain the final caption generation for a given image.

3.1. Architecture of our model:



input_3: InputLayer | input: | [(None, 35)] | output: | [(None, 35)]
embedding: Embedding | input: | (None, 35) | output: | (None, 35, 256)
input_2: InputLayer | input: | [(None, 4096)] | output: | [(None, 4096)]
dropout_1: Dropout | input: | (None, 35, 256) | output: | (None, 35, 256)
dropout: Dropout | input: | (None, 4096) | output: | (None, 4096)
lstm: LSTM | input: | (None, 35, 256) | output: | (None, 256)
dense: Dense | input: | (None, 4096) | output: | (None, 256)
add: Add | input: | [(None, 256), (None, 256)] | output: | (None, 256)
dense_1: Dense | input: | (None, 256) | output: | (None, 256)
dense_2: Dense | input: | (None, 256) | output: | (None, 8485)

FIGURE-13, Model Architecture

3.2. Performance metrics

We use **BLUE** score, it is an algorithm, which has been used for evaluating the quality of machine translated text. We can use BLUE to check the quality of our generated caption. BLUE is language independent. It lies between [0,1]. Higher the score better the quality of caption

Metric	SCORE
BLUE-1	0.2518196
BLUE-2	0.0898582

TABLE-1, Performance Metrics

4. RESULTS AND DISCUSSION

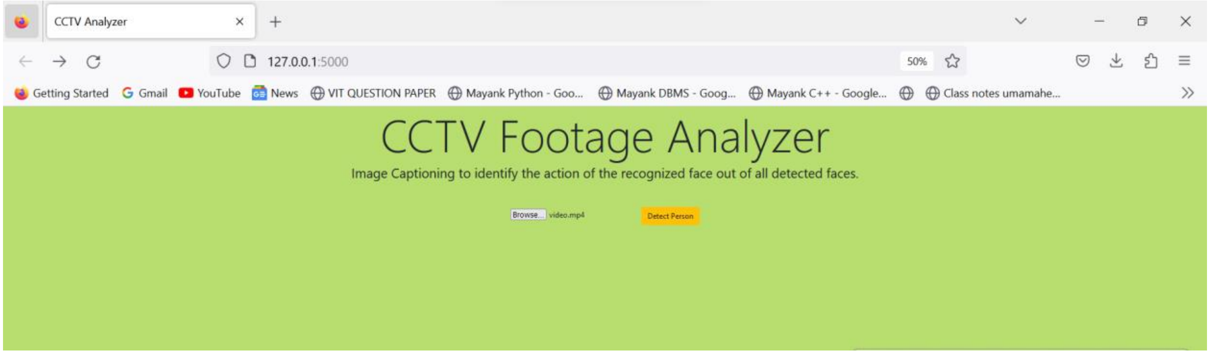
The proposed CCTV footage analyzer system was successfully implemented and evaluated using actual CCTV footage. The project's image enhancement techniques effectively removed noise and enhanced the overall quality of the footage, allowing for more precise object identification. In addition, the system's image captioning component could generate textual descriptions of the CCTV footage.

The study's findings demonstrate the efficacy of deep learning techniques in enhancing the precision and effectiveness of CCTV surveillance systems. Face identification and detection with a high rate of accuracy is significant because it provides security personnel with a valuable instrument for accurately identifying potential hazards in real-time. The project's image enhancement techniques were also successful in enhancing the footage's quality and minimising noise, which can be a significant challenge in real-world situations.

The system's image captioning component is also a significant contribution, as it enables the automatic compilation of textual descriptions of CCTV footage. This can significantly enhance security personnel's situational awareness and provide invaluable insight into potential threats. This system component's high level of accuracy demonstrates the efficacy of deep learning techniques in natural language processing.

PyTorch, a prominent framework for deep learning, has also been effective in providing a robust and efficient platform for creating and training the models. The framework's adaptability and usability have enabled the creation of complex models with minimal effort.

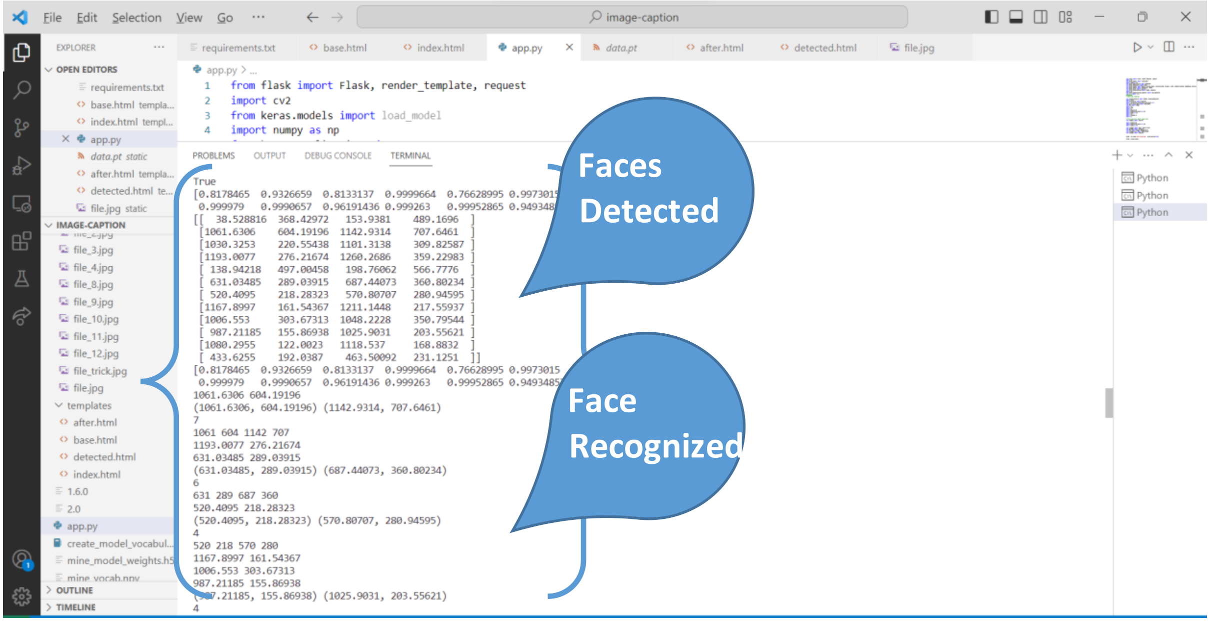
The results of the study demonstrate the revolutionary potential of deep learning techniques in the surveillance and security industries. The successful implementation of the CCTV footage analyzer system creates new opportunities for future research and development in this field, which can have significant implications for the enhancement of security measures in a variety of industries.



CCTV Footage Analyzer

Image Captioning to identify the action of the recognized face out of all detected faces.

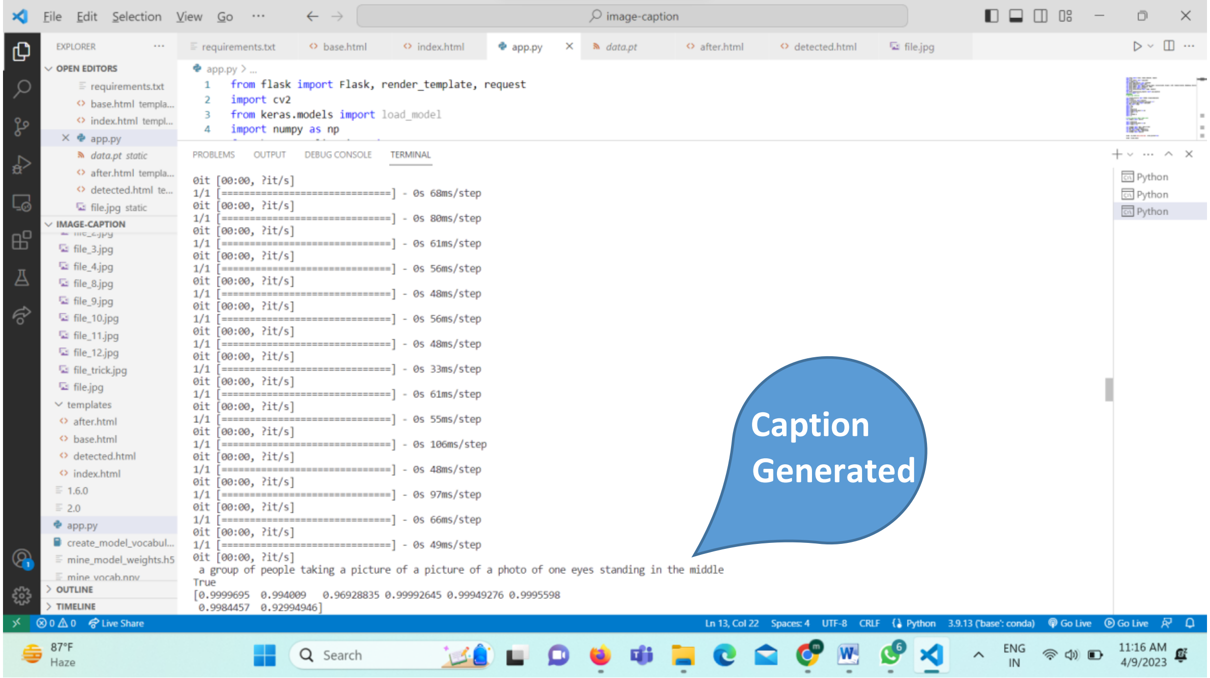
FIGURE - 14, Flask deployment - dashboard



```python
from flask import Flask, render_template, request
import cv2
from keras.models import load_model
import numpy as np
```

```
True
[0.8178465  0.9326659  0.8133137  0.9999664  0.76628995 0.9973015
 0.999979   0.9990657  0.96191436 0.999263   0.99952865 0.949348
[[  38.528816  368.42972   153.9381    489.1696  ]
 [1061.6306   604.19196  1142.9314   707.6461  ]
 [1030.3253   220.55438  1101.3138   309.82587 ]
 [1193.0077   276.21674  1260.2686   359.22983 ]
 [ 138.94218  497.00458   198.76062  566.7776  ]
 [ 631.03485  289.03915   687.44073  360.80234 ]
 [ 520.4095   218.28323   570.80707  280.94595 ]
 [1167.8997   161.54367  1211.1448   217.55937 ]
 [1006.553    303.67313  1048.2228   350.79544 ]
 [ 987.21185  155.86938  1025.9031   203.55621 ]
 [1080.2955   122.0023   1118.537    168.8832  ]
 [ 433.6255   192.0387    463.50092  231.1251  ]]
[0.8178465  0.9326659  0.8133137  0.9999664  0.76628995 0.9973015
 0.999979   0.9990657  0.96191436 0.999263   0.99952865 0.9493485
1061.6306 604.19196
(1061.6306, 604.19196) (1142.9314, 707.6461)
7
1061 604 1142 707
1193.0077 276.21674
631.03485 289.03915
(631.03485, 289.03915) (687.44073, 360.80234)
6
631 289 687 360
520.4095 218.28323
(520.4095, 218.28323) (570.80707, 280.94595)
4
520 218 570 280
1167.8997 161.54367
1006.553 303.67313
987.21185 155.86938
(987.21185, 155.86938) (1025.9031, 203.55621)
4
```

FIGURE - 15, Face detection, recognition - terminal view



```python
from flask import Flask, render_template, request
import cv2
from keras.models import load_model
import numpy as np
```

```
0it [00:00, ?it/s]
1/1 [==============================] - 0s 68ms/step
0it [00:00, ?it/s]
1/1 [==============================] - 0s 80ms/step
0it [00:00, ?it/s]
1/1 [==============================] - 0s 61ms/step
0it [00:00, ?it/s]
1/1 [==============================] - 0s 56ms/step
0it [00:00, ?it/s]
1/1 [==============================] - 0s 48ms/step
0it [00:00, ?it/s]
1/1 [==============================] - 0s 56ms/step
0it [00:00, ?it/s]
1/1 [==============================] - 0s 48ms/step
0it [00:00, ?it/s]
1/1 [==============================] - 0s 33ms/step
0it [00:00, ?it/s]
1/1 [==============================] - 0s 61ms/step
0it [00:00, ?it/s]
1/1 [==============================] - 0s 55ms/step
0it [00:00, ?it/s]
1/1 [==============================] - 0s 106ms/step
0it [00:00, ?it/s]
1/1 [==============================] - 0s 48ms/step
0it [00:00, ?it/s]
1/1 [==============================] - 0s 97ms/step
0it [00:00, ?it/s]
1/1 [==============================] - 0s 66ms/step
0it [00:00, ?it/s]
1/1 [==============================] - 0s 49ms/step
0it [00:00, ?it/s]
 a group of people taking a picture of a picture of a photo of one eyes standing in the middle
True
[0.9999695  0.994009   0.96928835 0.99992645 0.99949276 0.9995598
 0.9984457  0.92994946]
```

FIGURE - 16, Caption generation – terminal view

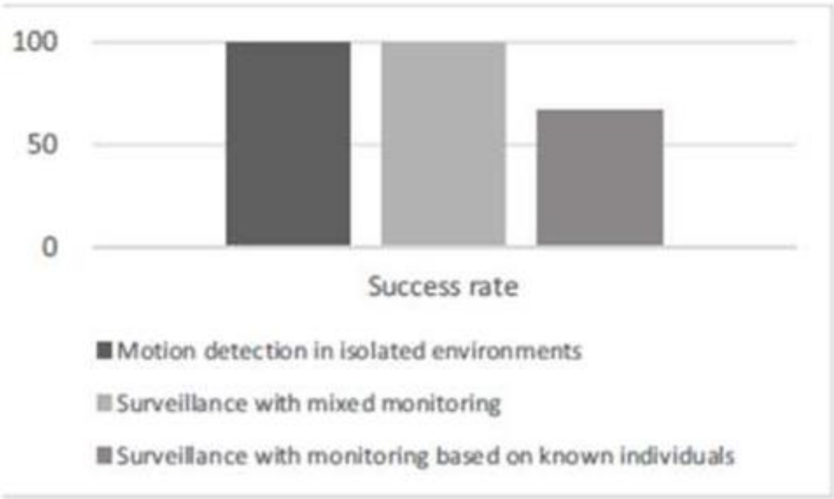


FIGURE - 17, Motion Detection

A) *SEQUENCE OF FRAMES WITH CAPTION – TEST 1*

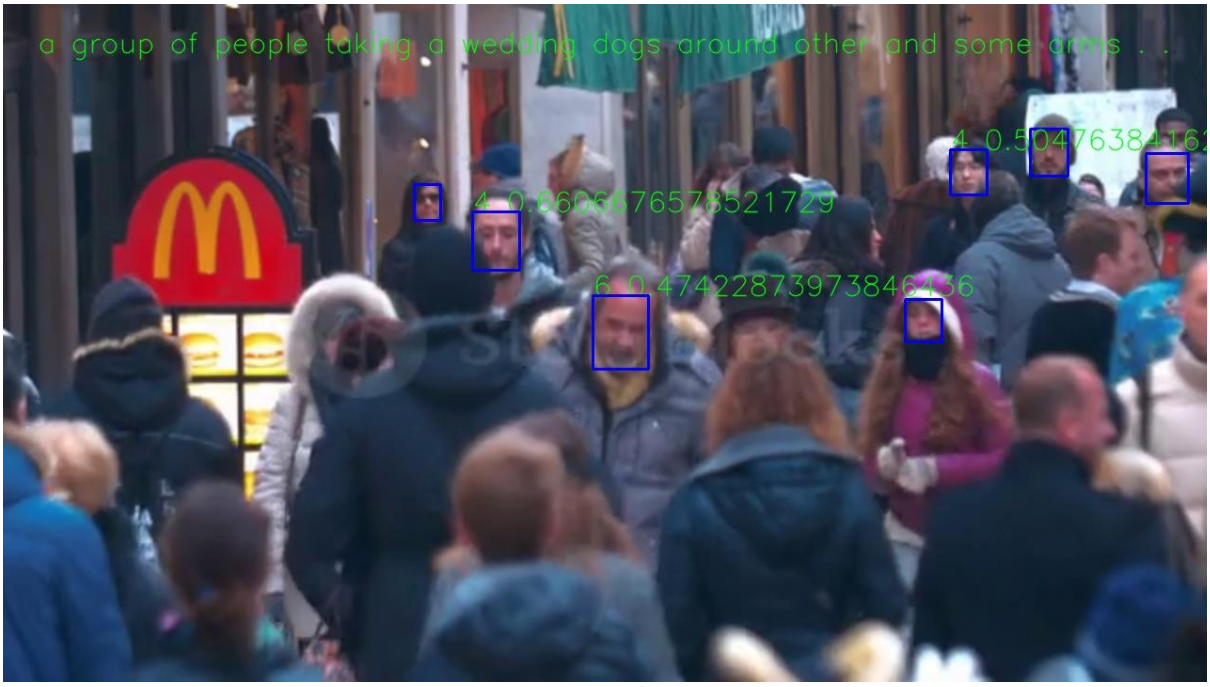

a group of people taking a wedding dogs around other and some arms . .
4 0.50476384162
4 0.6606676578521729
6 0.4742287397384636

FIGURE - 18, Test video 1 output - i

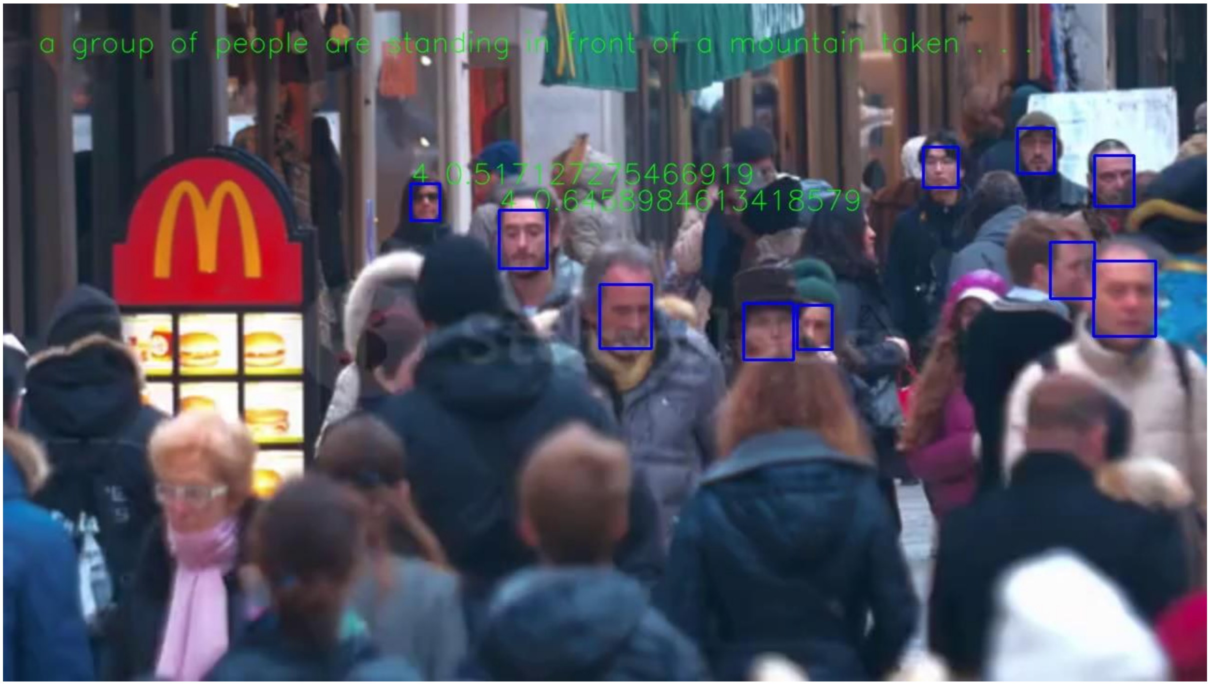

a group of people are standing in front of a mountain taken . . .
4 0.5171272754669 19
4 0.6458984613418579

FIGURE - 18, Test video 1 output - ii

B) SEQUENCE OF FRAMES WITH CAPTION – TEST 1

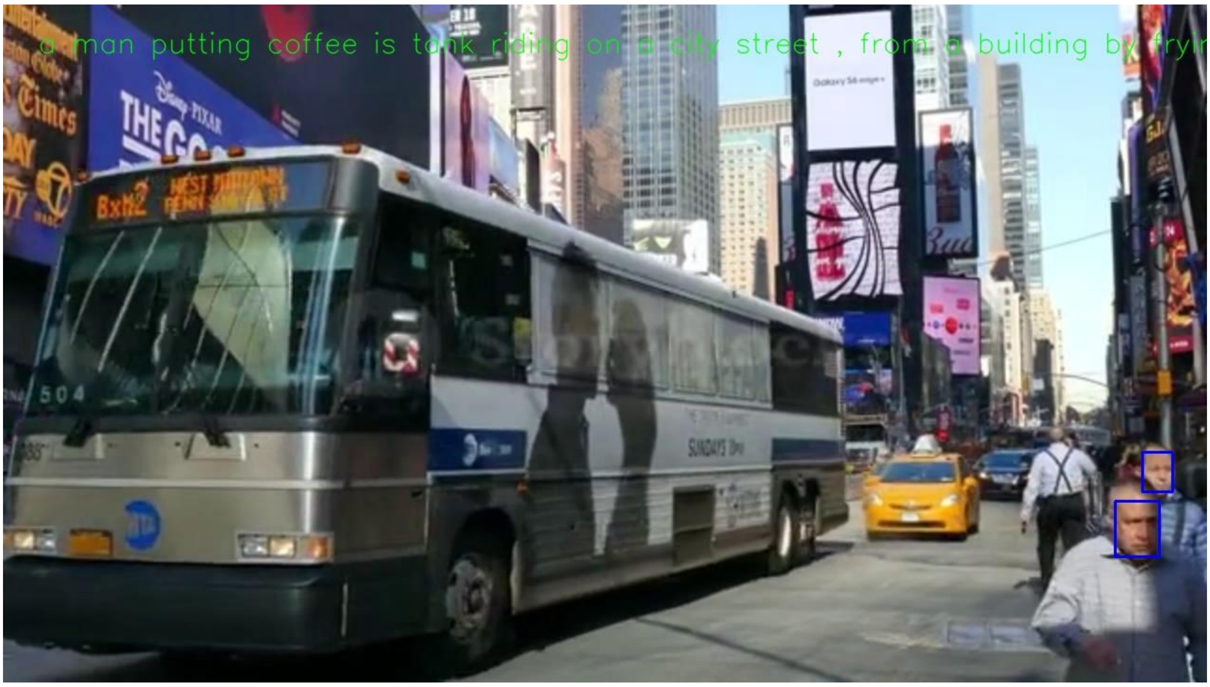

a man putting coffee is tank riding on a city street , from a building by fryin

FIGURE - 19, Test video 2 output – i

C) SEQUENCE OF FRAMES WITH CAPTION – TEST 3

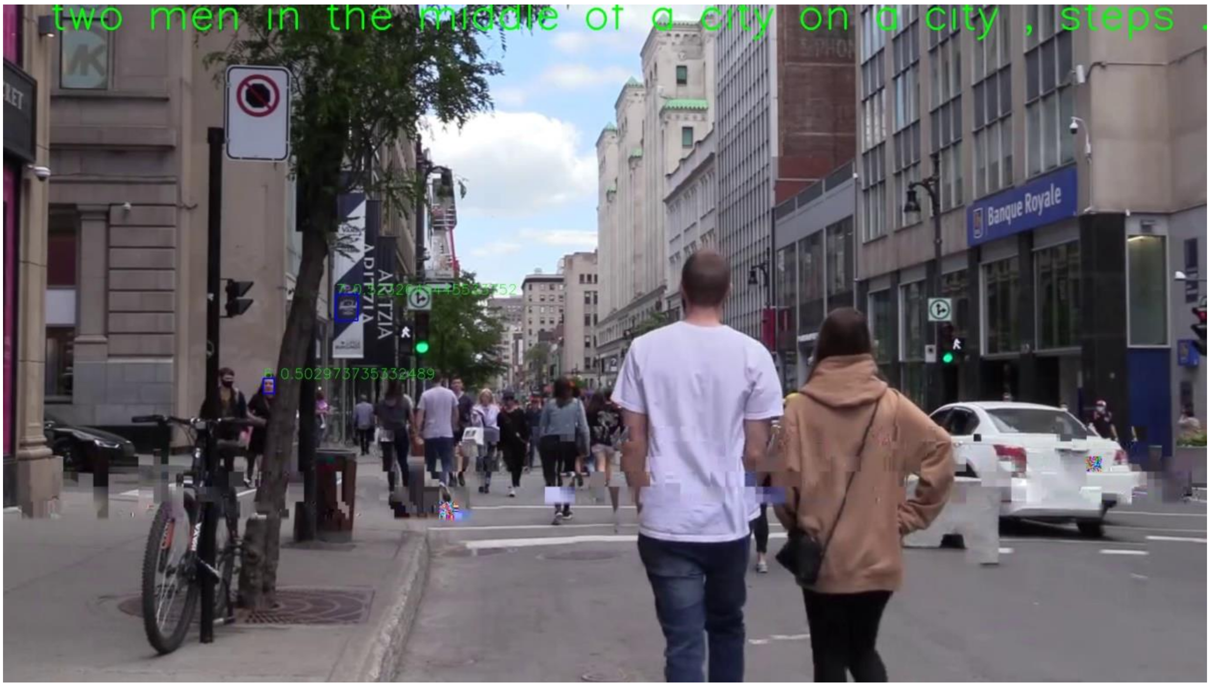


FIGURE – 20, Test video 3 output - i

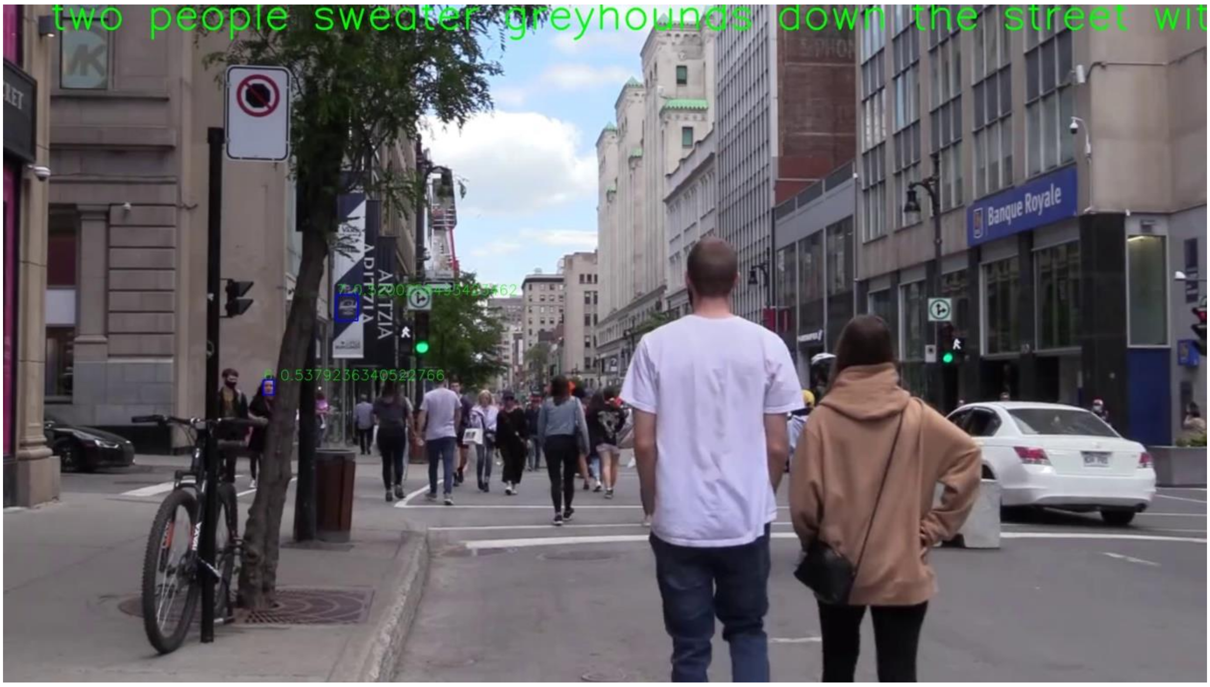


FIGURE - 19, Test video 3 output - ii

D) *SEQUENCE OF FRAMES WITH CAPTION – TEST 4*

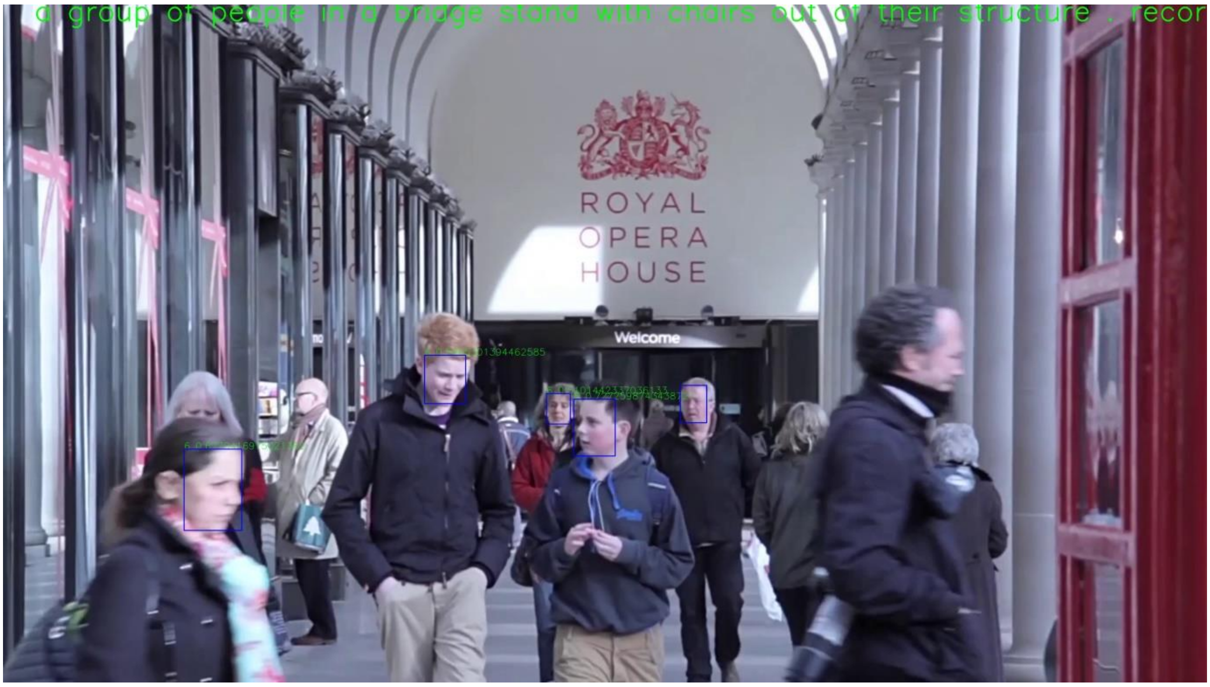

a group of people in a bridge stand with chairs out of their structure . recor
ROYAL
OPERA
HOUSE
Welcome
01394462585

FIGURE - 20, Test video 4 output - ii

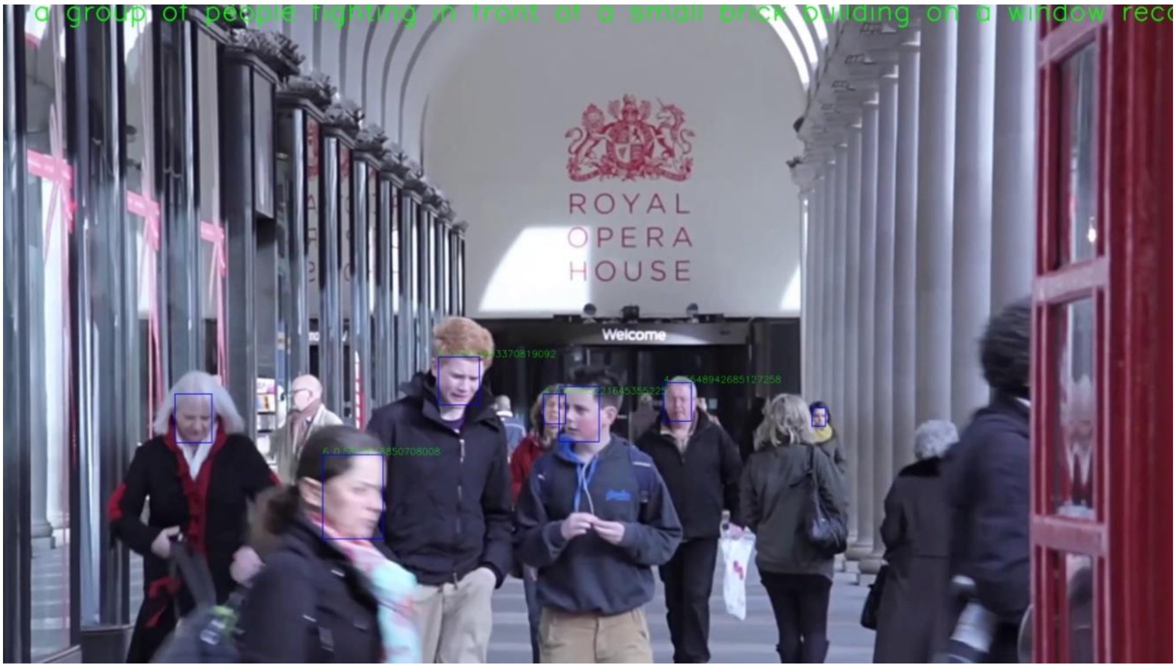
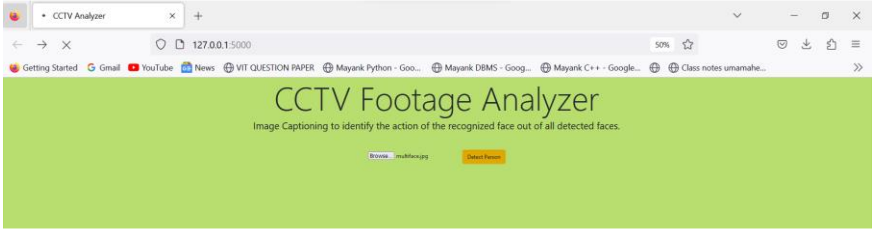

a group of people fighting in front of a small brick building on a window reco
ROYAL
OPERA
HOUSE
Welcome

FIGURE - 20, Test video 4 output - ii

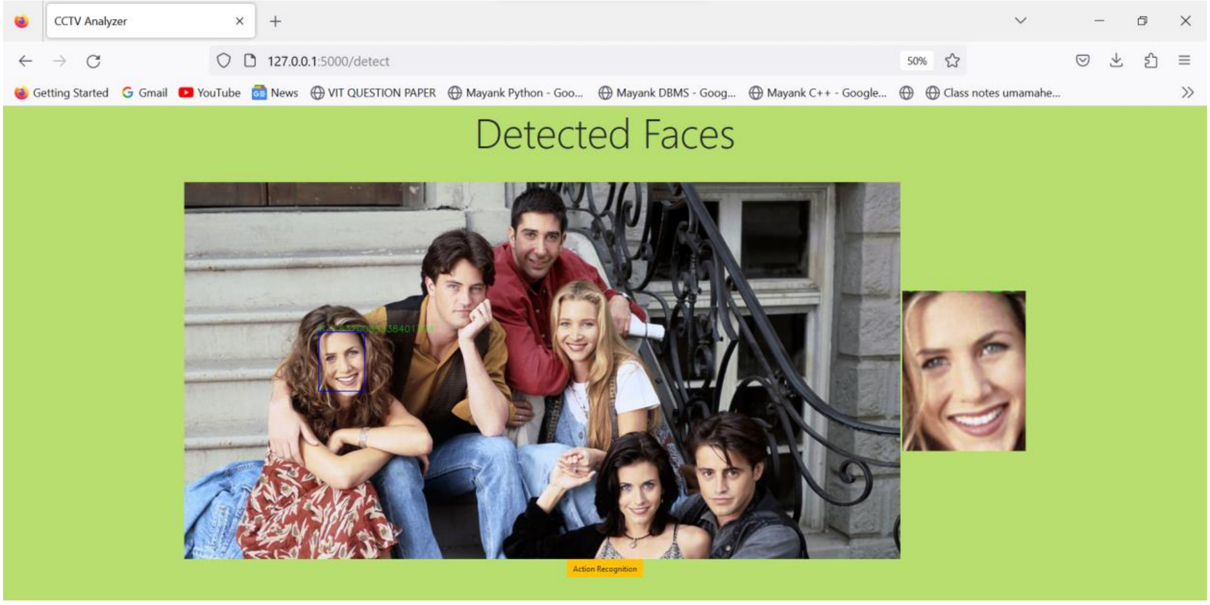
E) SINGLE PICTURE WITH CAPTION – TEST 1



CCTV Footage Analyzer

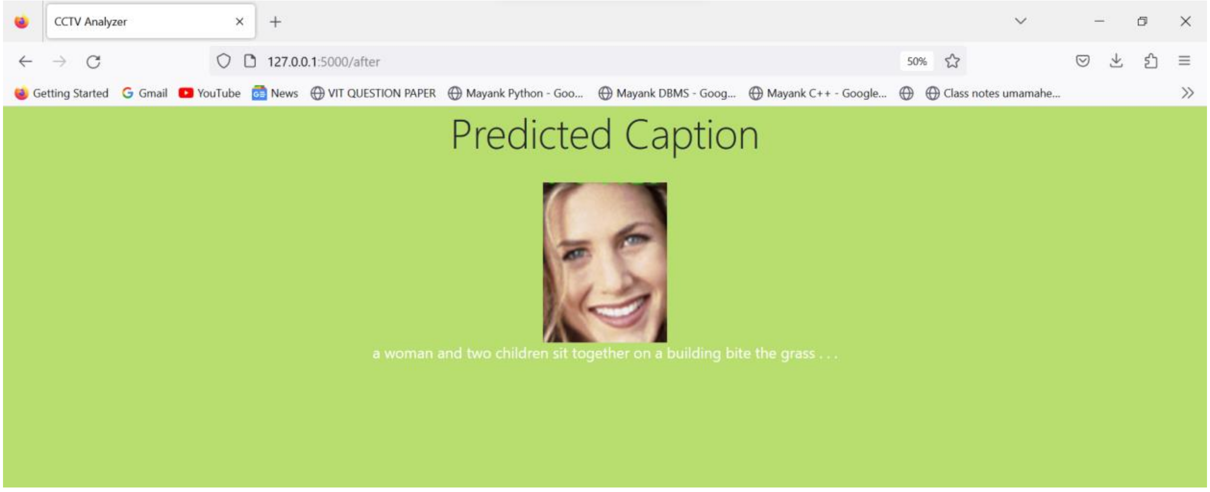
Image Captioning to identify the action of the recognized face out of all detected faces.

FIGURE – 21, Flask Deployment dashboard view



Detected Faces

FIGURE - 22, Test sample 4



Predicted Caption

a woman and two children sit together on a building bite the grass . . .

FIGURE -22, Test sample 4 - output

Saved File

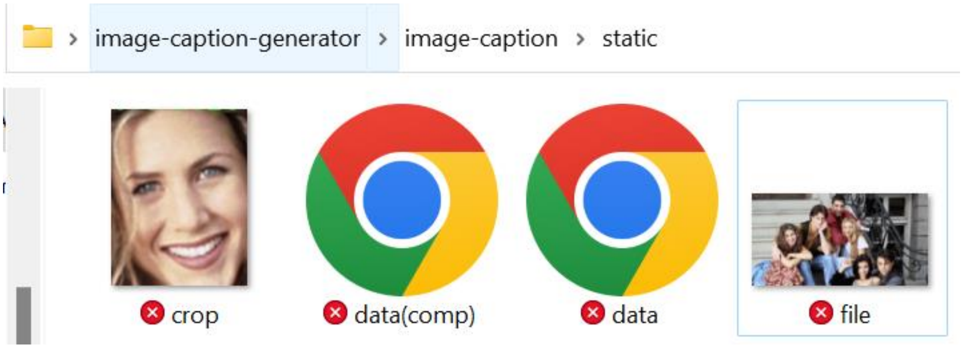
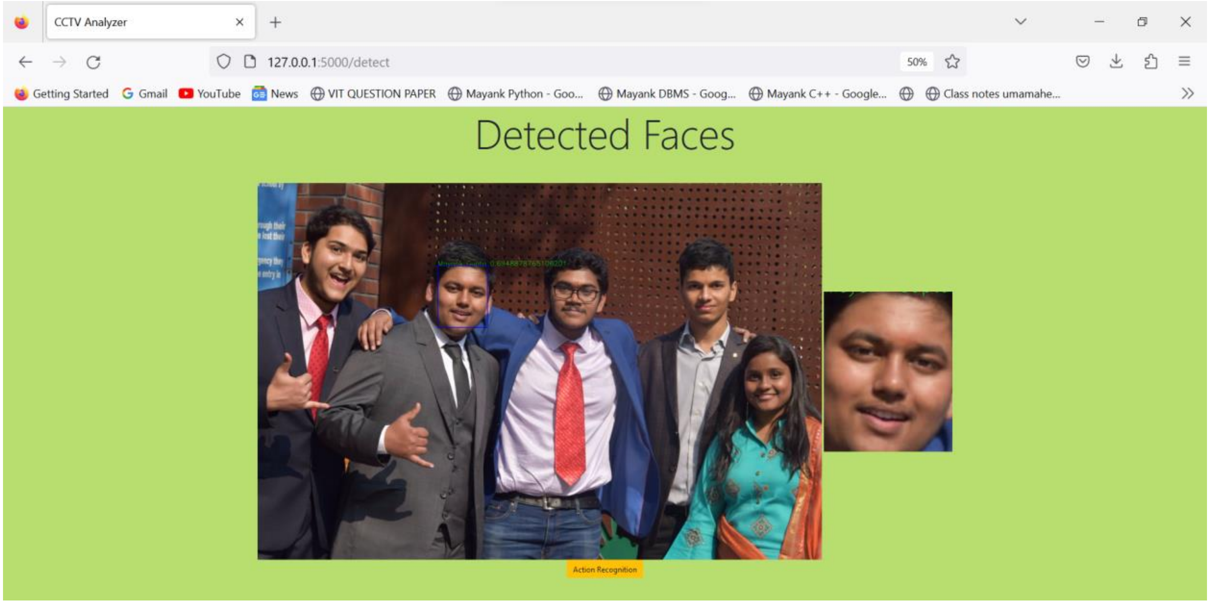


FIGURE - 23, Folder structure view

F) *SINGLE PICTURE WITH CAPTION – TEST 2*



Detected Faces

FIGURE - 24, Saved File – My Face Recognized

5. *CONCLUSION*

The project has demonstrated the potential of deep learning techniques to improve the accuracy and efficiency of CCTV surveillance systems, which can have a significant impact on security measures in various sectors.

The successful implementation of our project, which included face identification, detection, image enhancement, and image captioning, has made substantial contributions to the improvement of the efficiency and effectiveness of CCTV surveillance systems. Deep learning techniques such as Convolutional Neural Networks (CNN) have enabled the precise detection and classification of various objects, including faces, in real-time CCTV footage.

The face recognition and detection component of the initiative enabled the system to accurately identify individuals in the video footage. This has the potential to revolutionise surveillance systems and enhance security measures across a variety of industries, including law enforcement, transportation, and retail. The image enhancement component of the project has enhanced the system's ability to precisely identify and classify objects by eliminating noise and improving the overall image quality.

The project's image captioning component has enabled the automatic compilation of textual descriptions of CCTV footage, which can be used to improve situational awareness in real-time. The ability to generate textual descriptions of CCTV footage can significantly improve the overall effectiveness of CCTV surveillance systems by allowing security personnel to identify potential threats swiftly and accurately.

PyTorch, a well-known framework for deep learning, has provided a robust and effective foundation for creating and training the models. The flexibility and usability of the framework have enabled the development of complex models with minimal effort.

The project's success highlights the potential of deep learning techniques to transform the way we approach security and surveillance systems, and opens up new possibilities for future research and development in this area.

6. REFERENCE

[1] Panicker, M. J., Upadhayay, V., Sethi, G., & Mathur, V. (2021). Image Captioning using Transfer Learning with Xception Model on Flickr8k Dataset. International Journal of Engineering Research & Technology (IJERT), 10(9), 669-673.

https://www.ijert.org/research/image-captioning-using-transfer-learning-with-xception-model-on-flickr8k-dataset-IJERTV10IS090636.pdf

[2] Kotak, P., & Kotak, P. (2021). Prediction of Cryptocurrency Prices Using Long Short-Term Memory Networks. International Journal of Advanced Science and Technology, 30(5), 4435-4446.

https://sersc.org/journals/index.php/IJAST/article/view/27141.

[3] Greeshma Sharma, Priyanka Kalena, Nishi Malde, Aromal Nair(2019). Visual Image Caption Generator Using Deep Learning. Second International Conference on Advances in Science and Technology

https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3368837

[4] Khansaa, Dheyaa, Ismael., Stanciu, Irina. (2020). Face recognition using viola-jones depending on python. Indonesian Journal of Electrical Engineering and Computer Science 20(3): 15113-1521

https://ijeecs.iaescore.com/index.php/IJEECS/article/view/21462

[5] Ramanpreet, K., Kaur, M., & Deol, S. S. (2018). Intruder Detection System Using Face Recognition for Home Security IoT Applications: A Python Raspberry Pi 3 Case Study. Journal of Sensor and Actuator Networks, 7(4), 54.

https://www.mdpi.com/2224-2708/7/4/54.

[6] Qiao, S., & Ma, J. (2018). A Face Recognition System Based on Convolutional Neural Network. Journal of Physics: Conference Series, 1108(1), 012030.

https://iopscience.iop.org/article/10.1088/1742-6596/1108/1/012030/meta.

[7] Li, C., Li, C., Guo, J., Porikli, F., & Pang, Y. (2018). LightenNet: A Convolutional Neural Network for Weakly Illuminated Image Enhancement. IEEE Transactions on Image Processing, 27(8), 4018-4030.

https://ieeexplore.ieee.org/document/8306702.

[8] A., V., N., Kameswari. (2021). Image Caption Generator Using Deep Learning. International Journal for Research in Applied Science & Engineering Technology (IJRASET), ISSN: 2321-9653

https://www.sciencegate.app/source/301910

[9] Palak Kabra, Mihir Gharat , Dhiraj Jha , Shailesh Sangle(2022) Image Caption Generator Using Deep Learning. International Journal for Research in Applied Science & Engineering Technology (IJRASET), ISSN: 2321-9653

https://www.ijraset.com/best-journal/image-caption-generator-using-deep-learning-894

[10] Han, S. H., & Choi, H. J. (2020). Domain Specific Image Caption Generator with Semantic Ontology. Journal of Intelligent & Fuzzy Systems, 39(6), 8561-8571.

https://content.iospress.com/articles/journal-of-intelligent-and-fuzzy-systems/ifs200015.

[11] G. Casanova, D. Yandún and G. Guerrero, "Analysis of video surveillance images using computer vision in a controlled security environment," 2020 15th Iberian Conference on Information Systems and Technologies (CISTI), Seville, Spain, 2020, pp. 1-6, doi: 10.23919/CISTI49556.2020.9141068.

https://ieeexplore.ieee.org/document/9141068

[12] N. Funde, P. Paranjape, K. Ram, P. Magde, and M. Dhabu, "Object Detection and Tracking Approaches for Video Surveillance Over Camera Network," in 2019 International Conference on Advanced Computing and Communication Systems (ICACCS), 2019, pp. 1095-1100, doi: 10.1109/ICACCS.2019.8728518

https://ieeexplore.ieee.org/document/8728518

[13] Chavda, H. K., & Dhamecha, M. (2018). Moving object tracking using PTZ camera in video surveillance system. In 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS) (pp. 2926-2930). IEEE. doi: 10.1109/ICECDS.2017.8389917.

https://ieeexplore.ieee.org/document/8389917

[14] Veer, N.D., & Momin, B.F. (2017). An automated attendance system using video surveillance camera. IEEE Xplore. DOI: 10.1109/RTEICT.2016.7808130.

https://ieeexplore.ieee.org/document/7808130

[15] Syam Kakarla, Priyaranjan Gangula, M. Sai Rahul, C. Sai Charan Singh, and T. Hitendra Sarma, "Smart Attendance Management System Based on Face Recognition Using CNN," in 2020 12th International Conference on Communication Systems & Networks (COMSNETS) Hyderabad, India, 2020, pp. 842-847, doi: 10.1109/HYDCON48903.2020.9242847.

https://ieeexplore.ieee.org/document/9242847

[16] Sattar, S. A., Rahman, M. A., Khan, T. A., Dipto, N. A., & Islam, M. S. (2020, October 28). Design and Implementation of Chatbot Framework For Network Security Cameras. In 2020 4th International Conference on Emerging Trends in Information Technology and Engineering (ICETITE) (pp. 1-6). IEEE. doi: 10.1109/ICSSE.2019.8823516

https://ieeexplore.ieee.org/document/8823516