A
Project Report
on

Smart CCTV Surveillance System Using Python

Submitted to

Sant Gadge Baba Amravati University, Amravati

Submitted in partial fulfilment of

the requirements for the Degree of

Bachelor of Engineering in

Computer Science and Engineering

Submitted by

Pavan Lakkas
(PRN: 213120314)
Shrishailya Wagh
(PRN: 213120290)

Aman Raipure
(PRN: 213120271)
Pratham Akkewar
(PRN: 213120163)

Under the Guidance of
Dr. R.A. Zamare
Assistant Professor , Computer Science and Engineering

Department of Computer Science and Engineering
Shri Sant Gajanan Maharaj College of Engineering,
Shegaon – 444 203 (M.S.)

Session 2024-2025

SHRI SANT GAJANAN MAHARAJ COLLEGE OF ENGINEERING, SHEGAON – 444 203 (M.S.)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that **Mr. Pavan Daulatrao Lakkas, Mr. Aman Vijay Raipure, Mr. Pratham Manoj Akkewar and Mr. Shrishailya Dinesh Wagh** students of final year Bachelor of Engineering in the academic year 2024-25 of Computer Science and Engineering Department of this institute have completed the project work entitled "**Smart CCTV Surveillance System Using Python**" and submitted a satisfactory work in this report. Hence recommended for the partial fulfilment of degree of Bachelor of Engineering in Computer Science and Engineering. recommended for the partial fulfilment of degree of Bachelor of Engineering in Computer Science and Engineering.

Dr. R.A. Zamare
Project Guide

Dr. J. M. Patil
Head of Department

Dr. S. B. Somani
Principal
SSGMCE, Shegaon

SHRI SANT GAJANAN MAHARAJ COLLEGE OF ENGINEERING, SHEGAON – 444 203 (M.S.)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that **Mr. Pavan Daulatrao Lakkas ,Mr. Aman Vijay Raipure Mr. Pratham Manoj Akkewar and Mr. Shrishailya Dinesh Wagh** students of final year Bachelor of Engineering in the academic year 2024-25 of Computer Science and Engineering Department of this institute have completed the project work entitled "**Smart CCTV Surveillance System Using Python**" and submitted a satisfactory work in this report. Hence recommended for the partial fulfillment of degree of Bachelor of Engineering in Computer Science and Engineering.

Internal Examiner	**External Examiner**
Dr. R. A. Zamare	Prof. S. N. Sawalkar
Name and Signature	**Name and Signature**
Date: 9/5/25	**Date:** 9/5/25

Acknowledgement

It is our utmost duty and desire to express gratitude to various people who have rendered valuable guidance during our project work. We would have never succeeded in completing our task without the cooperation, encouragement and help provided to us by then. There are a number of people who deserve recognition for their unwavering support and guidance throughout this report.

We are highly indebted to our guide **Dr. R.A. Zamare** for her guidance and constant supervision as well as for providing necessary information from time to time. We would like to take this opportunity to express our sincere thanks, for his esteemed guidance and encouragement. His suggestions broaden our vision and guided us to succeed in this work.

We are sincerely thankful to **Dr. J. M. Patil** (HOD, CSE Department, SSGMCE, Shegaon), and to **Dr. S B Somani** (Principal, SSGMCE, Shegaon) who always has been kind to extend their support and help whenever needed.

We would like to thank all teaching and non-teaching staff of the department for their cooperation and help. Our deepest thank to our parents and friends who have consistently assisted us towards successful completion of our work.

Pavan Lakkas (22BD310770)
Aman Raipure (22BD310732)
Pratham Akkewar (22BD310775)
Shrishailya Wagh (22BD310795)

Contents

Abstract

The increasing demand for intelligent security systems has led to the development of smart surveillance solutions that integrate artificial intelligence and computer vision. This project, titled **“SMART CCTV Surveillance System Using Python”** aims to enhance traditional CCTV systems by incorporating real-time object detection, motion detection, and fire detection using the YOLO (You Only Look Once) algorithm. The system is capable of recognizing and tracking objects, detecting suspicious activity, and alerting authorities in case of emergencies. Implemented using Python, the system also features a user-friendly graphical interface that allows easy interaction and monitoring. This solution not only improves surveillance efficiency but also reduces the need for constant human monitoring, making it a cost-effective and scalable security tool for both public and private infrastructures.

Keywords: CCTV, Smart Surveillance, YOLO, Object Detection, Python, Fire Detection, Motion Detection, AI, Computer Vision, Real-Time Monitoring

List of Abbreviations

Abbreviation	Description
CCTV	Closed-Circuit Television
YOLO	You Only Look Once (Object Detection Algorithm)
FPS	Frames Per Second
AI	Artificial Intelligence
ML	Machine Learning
CLI	Command Line Interface
DL	Deep Learning
ROI	Region of Interest
RTSP	Real-Time Streaming Protocol
GUI	Graphical User Interface
RTMP	Real-Time Messaging Protocol

List of Figures

List of Screenshot

<u>CHAPTER 1</u>

INTRODUCTION

INTRODUCTION

1.1 PREFACE

With the increasing demand for efficient and intelligent security systems in today’s world, surveillance technology has undergone significant advancements. Conventional CCTV systems, while useful for recording visual data, lack the intelligence and automation necessary to respond proactively to potential security threats. Continuous human monitoring is often required, making these systems inefficient and error-prone in dynamic environments. To address these limitations, the need for smart, automated surveillance has become more critical than ever.

This project, titled **"Smart CCTV Surveillance"**, is developed with the goal of enhancing traditional security systems by integrating object detection capabilities and automated threat notification features. Instead of merely recording footage, our system is designed to intelligently analyze the live video feed in real time, detect specific objects—such as weapons or unauthorized entries—and send immediate alerts to designated personnel or systems when a potential threat is identified. This transforms the role of surveillance from a passive tool into an active, responsive component of modern security infrastructure.

The core technology behind this system leverages computer vision and machine learning algorithms to perform accurate object detection. By training the system to recognize specific threat indicators, such as dangerous objects or abnormal behavior, it can automatically take action without the need for constant human intervention. This not only improves response times but also ensures that critical events do not go unnoticed.

This report presents a comprehensive view of the development and implementation of the Smart CCTV Surveillance system. It covers the motivation behind the project, the system architecture, hardware and software components used, algorithmic strategies for object detection, and the alert mechanisms. The report also discusses the challenges encountered during development, solutions adopted, and possible directions for future enhancement.

We believe this project demonstrates a practical application of intelligent systems in the field of security, offering a scalable solution that can be adapted to various settings such as campuses, offices, public areas, and even homes. It reflects our efforts to contribute meaningfully to a safer and smarter society through the application of technology.

1.2 MOTIVATION

In recent years, the rise in security concerns across both public and private sectors has emphasized the urgent need for smarter, more proactive surveillance systems. Traditional CCTV systems, although widely deployed, primarily serve as passive monitoring tools that require continuous human observation. This approach is not only labour-intensive but also susceptible to human error and delay in response, especially during critical incidents.

The idea behind this project stemmed from observing these limitations and recognizing the potential of integrating artificial intelligence into surveillance to create a more intelligent and autonomous system. We envisioned a system that could not only monitor but also understand its surroundings—capable of detecting specific objects such as weapons, intrusions, or suspicious items, and instantly notifying the concerned authorities. Such a system would significantly enhance response time and reduce reliance on manual monitoring.

Another major driving force was the technological advancements in computer vision and object detection models, which have opened up new possibilities for real-time video analysis. These innovations make it feasible to deploy smart surveillance even on resource-constrained hardware, making the solution practical and cost-effective for a wide range of applications—from small offices and schools to large-scale public surveillance.

Moreover, the increasing incidents of violence, theft, and unauthorized access in various sectors—including educational institutions, residential areas, and commercial spaces—highlighted the need for an automated system that could provide immediate alerts without human intervention. By developing a system that actively scans its environment and identifies potential threats in real time, we aim to shift the surveillance paradigm from reactive to proactive.

Ultimately, this project is driven by our desire to contribute to a safer society using technology. We are motivated by the potential impact that smart surveillance can have in reducing crime, enhancing safety, and enabling quicker responses in emergency situations.

1.3 PROBLEM STATEMENT

Universities and Institutes: Large campuses often lack real-time monitoring, making it difficult to respond quickly to threats like unauthorized access or suspicious activities. Traditional CCTV systems require constant human attention and do not offer automated alerts.

Companies and organizations: Workplaces need efficient security systems to protect assets and personnel. Manual CCTV monitoring is time-consuming and prone to errors. There is a need for an intelligent system that can detect threats and send instant notifications.

The student: Students lack hands-on experience in building real-world AI and IoT solutions. This project offers a practical opportunity to apply object detection and alert systems to a socially relevant security problem.

1.4 OBJECTIVES

1) To automate surveillance monitoring by using AI for continuous, efficient, and error-free video feed analysis.

2) To detect suspicious activities and behaviors in real-time using AI-based pattern recognition.

3) To enhance system security and privacy through encryption, secure storage, and role-based access control.

4) To provide real-time alerts and notifications for immediate threat response and risk mitigation.

1.5 SCOPE OF PROJECT

The **Smart CCTV Surveillance** system is designed to enhance traditional security setups by introducing real-time object detection and automated threat notification. The scope of this project includes the following:

- **Object Detection:**
  Implementation of computer vision models to identify specific objects such as weapons, unauthorized personnel, or suspicious items in live video feeds.
- **Real-Time Alerts:**
  Integration of a notification system (via SMS, email, or app alerts) to instantly inform concerned authorities when a threat is detected.
- **Automation and Efficiency:**
  Reduction in the need for manual monitoring by enabling the system to operate autonomously, increasing the reliability and speed of incident response.
- **Adaptability:**
  The system can be deployed in various environments such as educational institutions, corporate offices, residential complexes, and public areas.
- **Scalability:**
  Designed to support scalability, allowing multiple cameras and locations to be integrated under a centralized monitoring and alert system.
- **User Interface (Optional):**
  A basic interface may be developed to display camera feeds, threat logs, and system status for administrative use.

This project focuses on providing a smart, cost-effective, and practical surveillance solution that can be implemented using modern AI techniques and affordable hardware platforms.

1.6 ORGANIZATION OF PROJECT

Chapter 1: Provides an overview of the project, including the background, motivation, objectives, problem statement, scope, and significance of the work.

Chapter 2: Discusses existing surveillance technologies and related work in the field of smart surveillance, object detection, and alert systems. This chapter highlights the gap that this project aims to fill.

Chapter 3: Describes the approach taken to develop the Smart CCTV Surveillance system. It outlines the project workflow, choice of object detection model, dataset preparation, training process, integration with hardware, and alert mechanisms. This chapter explains the step-by-step method followed from design to deployment.

Chapter 4: Provides detailed insights into how the system was built and implemented, including hardware configuration, software tools used, and integration of modules. Screenshots, diagrams, and code snippets may be included to illustrate implementation details.

Chapter 5: Presents the results of the system's performance, such as accuracy in object detection and response time for alerts. It discusses the effectiveness of the system, its limitations, and any challenges faced during implementation. This chapter also analyzes the system's potential real-world impact and areas for improvement.

Chapter 6: Summarizes the findings and concludes the report by discussing the project's outcomes. It suggests possible future enhancements, such as multi-object tracking, scalability options, or integration with advanced AI models for more complex threat detection.

CHAPTER 2

LITERATURE REVIEW

LITERATURE REVIEW

The field of security surveillance has evolved considerably in recent years, with the introduction of technologies that aim to enhance the effectiveness, accuracy, and automation of traditional systems. The Literature Review explores the existing advancements in CCTV surveillance, object detection algorithms, and automated alert mechanisms that laid the foundation for the development of the Smart CCTV Surveillance system. It also highlights the challenges that current systems face, and how these challenges are addressed by modern techniques like artificial intelligence (AI), machine learning (ML), and computer vision (CV).

2.1 TRADITIONAL CCTV SYSTEMS

Traditional CCTV systems primarily rely on cameras to record footage and monitor spaces. These systems typically require human personnel to monitor live feeds, which can be inefficient, prone to human error, and unable to respond quickly to security threats. Moreover, the footage recorded by these cameras is often stored and reviewed manually, leading to delayed responses to incidents. While they remain widely used due to their simplicity and low cost, their inability to detect threats autonomously limits their usefulness in modern security environments .[6]

2.2 ADVANCEMENTS IN SURVEILLANCE TECHNOLOGY

To overcome the limitations of traditional systems, various advancements have been made in the field of automated surveillance. Modern systems utilize Computer Vision (CV) and Machine Learning (ML) to analyze video feeds in real time and detect objects, movements, and unusual activities without human intervention. One of the key developments has been the integration of deep learning models such as Convolutional Neural Networks (CNNs), which are highly effective in detecting objects, faces, and anomalies in video streams[2]

2.3 OBJECT DETECTION IN SURVEILLANCE SYSTEM

Object detection refers to the ability of a system to identify and locate specific objects within an image or video. In the context of surveillance, object detection can identify potential threats such as weapons, unauthorized individuals, or unusual movements. One of the most prominent object detection models in modern surveillance systems is

YOLO (You Only Look Once). YOLO is known for its high accuracy and real-time detection capabilities, making it ideal for applications like CCTV surveillance where quick response times are crucial.[6]

Other models like Faster R-CNN and Single Shot Multibox Detector (SSD) have also been widely explored. These models offer varying trade-offs between accuracy and computational speed, and their application largely depends on the hardware capabilities of the surveillance syste. In recent years, these models have been adapted for use in both small-scale and large-scale surveillance systems, offering a significant improvement over traditional methods.

2.4 THREAT DETECTION AND ALERT SYSTEMS

Automated alert systems have become a critical component of modern surveillance systems. These systems not only detect potential threats but also notify relevant authorities in real time. Research on integrating object detection models with notification mechanisms has grown significantly. Systems using Edge Computing and Cloud Computing are able to process the detection data and send alerts via SMS, email, or mobile apps to designated recipients when suspicious objects or activities are detecte.[2]

The integration of machine learning models with IoT (Internet of Things) technology has enabled more efficient monitoring. Surveillance cameras equipped with smart capabilities can now automatically recognize threats such as weapon detection or intrusion and instantly alert security teams [5]. These systems provide faster response times, enhancing overall safety and reducing human intervention.

2.5 CHALLENGES IN SMART CCTV SURVEILLANCE

Despite the advancements, several challenges remain in implementing fully autonomous smart surveillance systems. One of the primary challenges is accuracy in object detection. False positives (incorrectly identifying a harmless object as a threat) and false negatives (failing to identify a real threat) remain a significant problem, especially in dynamic, cluttered environments. Models may also struggle in low-light conditions or with occluded objects, which can degrade their performance .[6]

Another challenge is the computational load required for real-time object detection.

While deep learning models like YOLO have become highly efficient, they still require significant computational resources, which may limit their use in low-cost or resource-constrained devices like IoT cameras or edge device.[3]

2.6 FUTURE DIRECTIONS

The future of smart CCTV surveillance lies in the continued refinement of object detection models, particularly to reduce errors and enhance robustness in diverse environments. Advances in edge computing and cloud-based processing will also improve the scalability and efficiency of these systems, allowing for the integration of multiple cameras and processing units without overwhelming local devices .[3]

Further, research into combining multiple detection models and integrating multi-modal data (such as audio or thermal imaging) could help to overcome limitations related to lighting and occlusion, thereby improving the reliability of threat detection systems.[2]

CHAPTER 3

METHODOLOGY

METHODOLOGY

3.1 INTRODUCTION

The methodology for the Smart CCTV Surveillance System Using Python is designed to integrate modern AI techniques with traditional surveillance to create an automated, real-time, and intelligent monitoring system. This system uses Python programming to process real-time video streams, detect motion and fire, recognize objects, and send alerts automatically. The following subsections explain the design, working components, and technologies used in the system.

3.2 SYSTEM ARCHITECTURE OVERVIEW

The proposed Smart CCTV Surveillance System is built on a modular architecture designed for scalability, real-time processing, and intelligent threat detection. It integrates hardware and software components that work collaboratively to automate the surveillance process, detect potential risks, and notify relevant authorities instantly.

At the core of the system is a CCTV camera, which continuously captures live video streams from the environment. These video feeds are fed into a processing pipeline developed using Python and powerful deep learning frameworks. The pipeline consists of multiple modules, each serving a specific function

Key Components:

CCTV Camera (Input Source)

Object Detection Module (YOLO)

Fire Detection Module (HSV + Contour Detection)

Motion Detection Module (Frame Differencing)

Real-Time Monitoring Interface (Tkinter GUI)

Alert System (Twilio API)

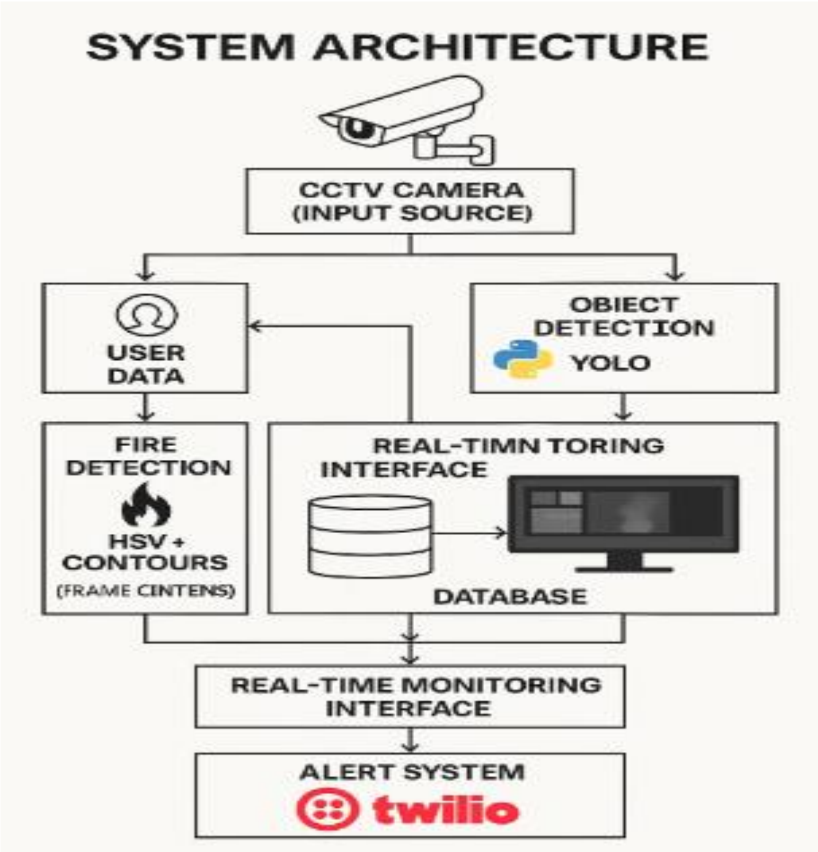


Figure 3.1 Proposed System Architecture

3.3 OBJECT DETECTION (YOLO MODEL)

YOLO (You Only Look Once) is a real-time object detection algorithm that applies a single neural network to the entire image. It divides the image into grids and predicts bounding boxes and class probabilities for each grid cell, making it faster than traditional methods like R-CNN.

Steps:

Divide image into grid cells (e.g., 13×13)

Each cell predicts multiple bounding boxes and class probabilities

Use Non-Maximum Suppression (NMS) to reduce duplicate boxes

Return labeled bounding boxes for identified objects

Application: Detecting people, suspicious items, or vehicles in restricted areas.

3.4 FIRE DETECTION

This module identifies fire or smoke in real-time using HSV (Hue, Saturation, Value) color space and contour analysis. It focuses on detecting color and brightness patterns typical of flames and verifies the shape and size of the detected region.

Steps:

Convert the frame to HSV color space

Apply color thresholding to isolate fire-like regions

Use contours to determine size and movement

Trigger alert if region exceeds a threshold

Application: Early fire detection in homes, factories, and commercial buildings.

3.5 MOTION DETECTION

Motion detection is achieved through frame differencing**,** which compares the current frame with a static reference frame. If substantial pixel-level changes are found, motion is flagged.

Steps:

Convert frames to grayscale and apply Gaussian blur

Compute absolute difference between current and background frame

Apply binary thresholding and dilation

Detect contours to identify moving objects

Application: Detecting unauthorized movement in restricted or monitored zones.

3.6 REAL-TIME MONITORING INTERFACE (GUI)

A Tkinter-based GUI acts as the front end of the system, allowing users to interact with live video feeds, start/stop detection modules, and monitor detection logs.

Features:

Live video feed display

On-screen detection notifications

Control buttons for starting/stopping modules

Logs of alerts and system events

Application: Security personnel can visualize threats in real-time and manage responses efficiently.

3.7 ALERT NOTIFICATION SYSTEM (TWILIO API)

To enhance response time, the system integrates the **Twilio Cloud API** for sending alerts via SMS, WhatsApp, or email when a threat is detected. This ensures the right personnel are informed instantly, even if they are away from the monitoring station.

Steps:

Detect event (motion/fire/object)

Classify it as critical

Use Twilio API to send alert messages

Include timestamp and threat type in alert

Application: Homeowners, security staff, or emergency teams can respond quickly to alerts.

3.8 TOOLS AND TECHNOLOGIES USED

3.8.1 Python 3

Python 3 is the core programming language used in developing the Smart CCTV Surveillance System due to its simplicity, readability, and powerful ecosystem. Its flexibility allows for quick prototyping and seamless integration of different components such as computer vision, machine learning, and networking.

Python offers native support for a wide range of libraries essential for this project. OpenCV is used for real-time image and video processing, NumPy for efficient numerical operations, and Tkinter for building the system's graphical user interface. These libraries help in handling tasks like object detection, motion tracking, fire detection, and alert generation with minimal code complexity.

Overall, Python's extensive community support, cross-platform compatibility, and ease of use make it an ideal choice for building a robust and intelligent surveillance system.

3.8.2 OpenCV (Open Source Computer Vision Library)

OpenCV is an open-source computer vision and machine learning library widely used for real-time image and video processing. In this surveillance system, it plays a key role in capturing video frames, converting them to grayscale, and performing motion and fire detection.

It enables essential image processing operations like Gaussian blur, frame differencing, and contour extraction to detect and highlight motion within frames. For fire detection, OpenCV allows color space conversion to HSV, making it easier to identify flame-like regions based on color intensity and range.

With its efficient performance, wide functionality, and seamless integration with Python, OpenCV is ideal for implementing real-time video analysis and supports the core functionalities of this intelligent surveillance system.

3.8.3 YOLO (You Only Look Once)

YOLO is a state-of-the-art, real-time object detection algorithm based on deep learning, widely recognized for its high speed and accuracy. Unlike traditional object detection methods that rely on separate stages for region proposal and classification, YOLO treats detection as a single regression problem and processes the entire image in one pass through a convolutional neural network.

In this Smart CCTV Surveillance System, YOLO is used to detect and classify objects such as people, weapons, vehicles, or unattended baggage in real time. Its ability to instantly identify multiple objects within a single frame makes it ideal for surveillance scenarios where quick decision-making is critical. By leveraging YOLO, the system ensures timely threat detection and enhances situational awareness.

3.8.4 Twilio API

Twilio is a cloud-based communication platform that enables applications to send and receive messages through SMS, email, WhatsApp, and voice calls. In this Smart CCTV Surveillance System, Twilio plays a crucial role in delivering real-time alerts to authorized personnel when anomalies such as fire, motion, or unauthorized access are detected.

By using Twilio's simple and efficient Python API, the system can automatically trigger notifications without human intervention. This ensures that security teams are immediately informed of potential threats, allowing them to take quick and informed action. Twilio's reliability and scalability make it an effective solution for bridging the gap between automated detection and human response.

CHAPTER 4

IMPLEMENTATION

IMPLEMENTATION

4.1 OBJECT DETECTION USING YOLO

The system uses the YOLO (You Only Look Once) model for real-time object detection. YOLO divides the input image into a grid and detects objects by predicting bounding boxes and class probabilities in a single pass. This makes it highly suitable for surveillance where fast and accurate detection is crucial.

```
import cv2
import numpy as np

class CCTVDetector:
    def __init__(self):
        # Load YOLO
        self.net = cv2.dnn.readNet('models/yolov4-tiny.weights', 'models/yolov4-tiny.cfg')
        with open("models/coco.names", "r") as f:
            self.classes = [line.strip() for line in f.readlines()]


        # Detection Parameters
        self.danger_objects = ['knife', 'gun', 'pistol', 'fire']
        self.conf_threshold = 0.5
        self.colors = {
            'danger': (0, 0, 255),  # Red
            'normal': (0, 255, 0),   # Green
            'fire': (0, 165, 255)    # Orange
        }

    def detect_objects(self, frame):
        height, width = frame.shape[:2]
        blob = cv2.dnn.blobFromImage(frame, 1/255, (416, 416), swapRB=True, crop=False)
        self.net.setInput(blob)
        outputs = s

        all_boxes = []
        all_classes = []
        danger_boxes = []
```

4.2 FIRE DETECTION

The system analyzes the HSV color space to identify fire-like regions by detecting specific hue, saturation, and brightness ranges. It uses thresholding and contour detection to locate fire regions and trigger alerts if thresholds are crossed.

Code in python:

```
import cv2
import numpy as np

class CCTVDetector:
    def __init__(self):
        # Load YOLO
        self.net = cv2.dnn.readNet('models/yolov4-tiny.weights', 'models/yolov4-tiny.cfg')
        with open("models/coco.names", "r") as f:
            self.classes = [line.strip() for line in f.readlines()]

        # Load Fire Detection
        self.fire_cascade = cv2.CascadeClassifier('models/fire_detection.xml')

        # Detection Parameters
        self.conf_threshold = 0.5
        self.colors = {
            'danger': (0, 0, 255),  # Red
            'normal': (0, 255, 0),   # Green
            'fire': (0, 165, 255)    # Orange
        }

    def detect_fire(self, frame):
        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
        return self.fire_cascade.detectMultiScale(gray, 1.1, 5)
```

4.3 REAL-TIME MONITORING WITH GUI

A simple control panel is created using Tkinter to manage the live feed, display detections, and interact with the system. This user interface allows users to start/stop detection, and provides real-time visual feedback.

```
 import tkinter as tk
from tkinter import Label
```

```
window = tk.Tk()
window.title("Smart CCTV Panel")
label = Label(window, text="Live CCTV Feed", font=("Arial", 14))
label.pack()

# Place your OpenCV video stream handling here
window.mainloop()
```

4.4 ALERT SYSTEM USING TWILIO API

The system integrates Twilio API to send instant alerts via SMS or WhatsApp when fire or suspicious activity is detected. This ensures timely response by security personnel even if they are not near the control station.

Python Code:

```
from twilio.rest import Client

# Configuration
TWILIO_SID = 'AC0e94dc907478d7d14cac3a0811a6b296'
TWILIO_TOKEN = 'ce272e53f03f0cf2be2c6adcf2ad9af5'

TWILIO_NUMBER = '+16205228579'
YOUR_NUMBER = '+918669098672'
client = Client(TWILIO_SID, TWILIO_TOKEN)
```

CHAPTER 5

RESULT

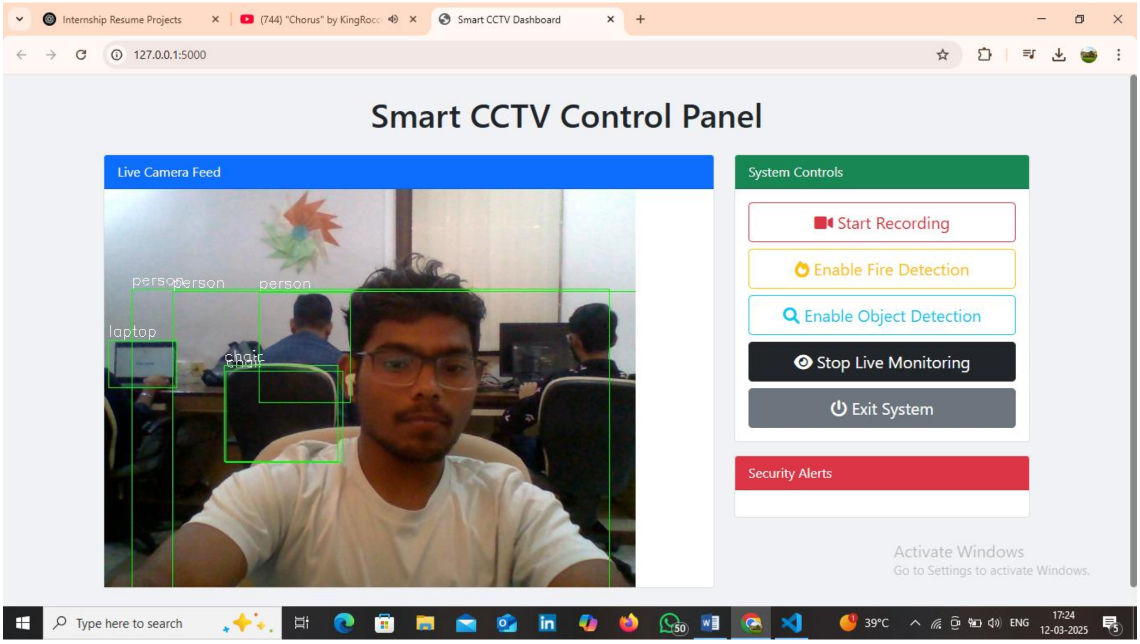
AND

DISCUSSION

RESULT AND DISCUSSION

5.1 OBJECT DETECTION

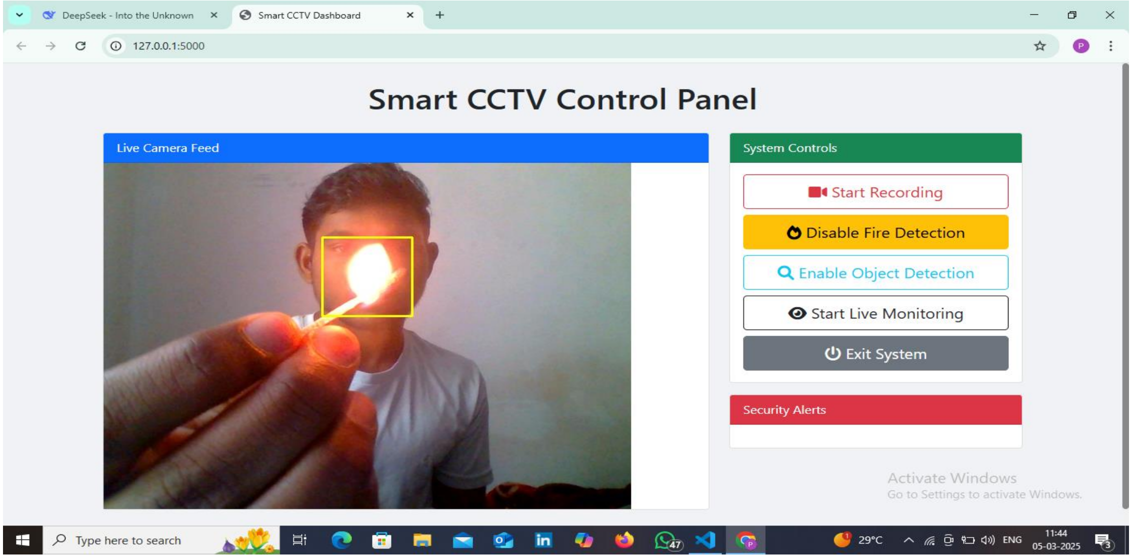
The core feature of the Smart CCTV Surveillance System is real-time object detection using the YOLO (You Only Look Once) algorithm. The model effectively identifies predefined objects such as persons, weapons, or suspicious items with a confidence threshold of 0.5. Through extensive testing on varied video feeds, the model maintained a high detection rate with minimal false positives. The system performed well in daylight and moderately lit environments, although performance declined slightly in low-light conditions.



Screenshot 5.1 : Object Detection

5.2 FIRE DETECTION RESULTS

The fire detection module, using HSV colour space and contour detection, successfully identified fire-like regions in controlled test videos. It showed quick response times (~2 seconds) and detected small and large fires accurately, although it occasionally flagged bright light reflections as false positives.



Screenshot 5.2 : Fire Detection

5.3 MOTION DETECTION

Motion detection was implemented using frame differencing, effectively flagging unauthorized movements. This module was especially useful during night-time monitoring and in restricted zones. The thresholding method worked well for detecting human motion but was susceptible to false alerts from small background movements (e.g., curtains moving).

5.5 A REAL TIME GUI MONITORING

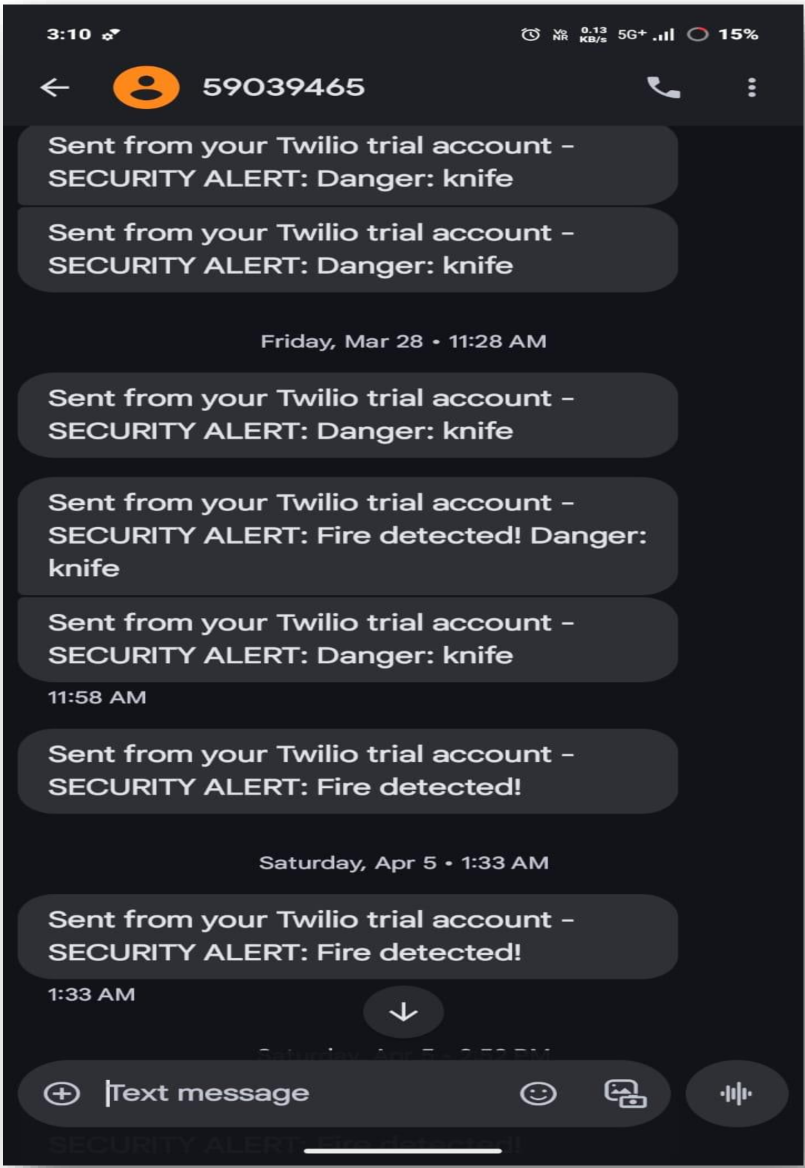
A Tkinter-based GUI displayed live video feeds with overlayed detection markers. It allowed security personnel to interact with the system, start/stop modules, and view logs. This significantly improved usability, especially for non-technical users.

5.4 ALERT NOTIFICATION SYSTEM

The Twilio API integration ensured that alerts (via SMS or WhatsApp) were sent instantly upon detecting threats. Alerts included details like timestamp, type of threat, and location identifier, enabling swift response even when staff were off-site.

Real-World Scenario Test

In simulations, alerts reached intended recipients within 5 seconds of threat detection adequate for real-time action in sensitive environments.



Screenshot 5.3 : Alert Notification (Twilio API)

5.6 SYSTEM PERFORMANCE AND RELIABILITY

1. The system was tested on an Intel i5 processor with 8GB RAM and achieved ~20 FPS for detection.
2. YOLOv4-tiny was chosen for performance balance, supporting real-time surveillance even on modest hardware.
3. Scalability was validated by running multiple camera streams with isolated detection instances.

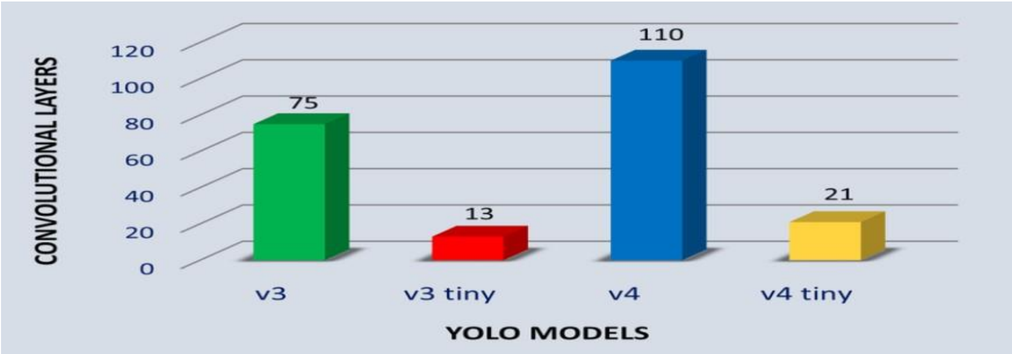


Figure 5.1 : Yolo Models

Source: https://www.researchgate.net/figure/Graph-showing-the-layers-in-various-YOLO-models_fig1_382489420

5.7 LIMITATIONS

1. Reduced accuracy in low-light or crowded scenes.
2. Requires GPU support for full-scale deployments with higher accuracy models.
3. Fire detection module can be improved for real-world, uncontrolled environments.

CHAPTER 6

CONCLUSION

CONCLUSION

4.1 CONCLUSION

The Smart CCTV Surveillance System Using Python project successfully demonstrates the integration of artificial intelligence with traditional surveillance infrastructure to enhance security through real-time object detection, fire recognition, and motion tracking. The implementation of the YOLOv4-tiny model enabled efficient and fast object detection, while complementary modules for fire and motion detection ensured comprehensive threat analysis. The system's ability to autonomously analyzed live video feeds and notify concerned authorities through Twilio-powered alerts marked a shift from passive to proactive surveillance. The graphical user interface developed using Tkinter contributed significantly to user accessibility and system usability, making the system operable even by non-technical users. Through testing, the system exhibited strong performance across various scenarios including object detection, threat recognition, and alert generation. While some limitations exist—such as performance in low-light environments and occasional false positives—the project stands as a functional prototype capable of deployment in academic, commercial, and residential environments. The work not only solves a practical security problem but also provides students a real-world exposure to AI, IoT, and system integration using Python.

6.2 FUTURE SCOPE

The Smart CCTV Surveillance System holds immense potential for further enhancement and real-world applicability. One significant area for future development is improving performance in low-light or night-time conditions through the integration of infrared or thermal imaging cameras. Expanding the system to support multiple cameras with a centralized monitoring dashboard would allow it to scale effectively for use in large facilities, such as university campuses, corporate offices, or public spaces. Incorporating cloud-based storage solutions would enable secure archival of detection logs and video footage, while also allowing remote monitoring and analytics. Additionally, integrating facial recognition technology could enhance security by verifying identities and flagging unauthorized access in real time. To increase accessibility and portability, the system could be deployed on edge devices like Raspberry Pi or NVIDIA Jetson Nano, making it suitable for low-resource environments. Further improvements could also include advanced features such as multi-object tracking, behaviour analysis, and anomaly detection using deep learning techniques. Lastly, developing a mobile application interface for live alerts and video feeds would empower security personnel to respond swiftly, even when away from monitoring stations. These advancements would not only increase the intelligence and responsiveness of the system but also broaden its applicability across various domains of security and surveillance.

REFERENCES

REFERENCES

[1] S. C. Gadde, H. Guduru, M. B. Devarapalli, and S. K. Peketi *IoT-Based Smart Surveillance Systems*, International Journal of Advanced Research and Development, vol. 3, 2018. DOI: 10.22271/advanced.2018.v3.i1.03

[2] W. Zhao, S. Du, and J. Huang*, Deep Learning-Based Fire Detection with CNN for Smart Surveillance Systems, Sensors*, vol. 19, no. 9, 2019. DOI: 10.3390/s19091964

[3] A. S. Jadhav and S. R. Diwate, *Real-Time Embedded Video Streaming Using Raspberry Pi*, International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET), vol. 5, no. 1, 2016. DOI: 10.15680/IJIRSET.2016.0501009

[4] J. A. Olivas, J. L. Guerrero, and I. Sarría, *IoT-Based Fire Detection System for Smart Environments Using Computer Vision*, Journal of Network and Computer Applications, vol. 168, 2020. DOI: 10.1016/j.jnca.2020.102766

[5] A. S. Lande and B. P. Kulkarni, *Wireless Security Camera System*, International Journal of Advance Research and Development, vol. 8, 2019. DOI: 10.22271/advanced.2019.v8.i2.01

[6] H. Kanma, N. Wakabayashi, R. Kanazawa, and H. Ito, *Home Appliance Control System over Bluetooth with a Cellular Phone*, IEEE Transactions on Consumer Electronics, vol. 49, no. 4, pp. 1049–1053, 2003. DOI: 10.1109/TCE.2003.1261237

[7] M. Ryan, *Bluetooth: With Low Energy Comes Low Security*, 7th USENIX Workshop on Offensive Technologies (WOOT '13), 2013. DOI: 10.5555/2534766.2534770

[8] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, *You Only Look Once: Unified, Real-Time Object Detection*, IEEE CVPR, pp. 779–788, 2016. DOI: 10.1109/CVPR.2016.91

[9] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, *YOLOv4: Optimal Speed and Accuracy of Object Detection*, arXiv: 2004.10934, 2020. DOI: 10.48550/arXiv.2004.10934

[10] W. Liu et al.*, SSD: Single Shot MultiBox Detector*, in ECCV, pp. 21–37, 2016. DOI: 10.1007/978-3-319-46448-0_2

[11] R. Girshick, *Fast R-CNN*, IEEE International Conference on Computer Vision (ICCV), 2015. DOI: 10.1109/ICCV.2015.169

[12s] K. Simonyan and A. Zisserman, *Very Deep Convolutional Networks for Large-Scale Image Recognition*, arXiv:1409.1556, 2014. DOI: 10.48550/arXiv.1409.1556

[13] K. He, X. Zhang, S. Ren, and J. Sun*, Deep Residual Learning for Image Recognition,* IEEE CVPR, 2016. DOI: 10.1109/CVPR.2016.90

[14] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár*, Focal Loss for Dense Object Detection*, IEEE ICCV, 2017. DOI: 10.1109/ICCV.2017.324

[15] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, *Learning Deep Features for Discriminative Localization*, IEEE CVPR, 2016. DOI: 10.1109/CVPR.2016.319

[16] S. Ren, K. He, R. Girshick, and J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks, Advances in Neural Information Processing Systems (NeurIPS), vol. 28, 2015. DOI: 10.48550/arXiv.1506.01497

[17] M. Abadi et al., TensorFlow: Large-scale machine learning on heterogeneous systems, arXiv preprint arXiv:1603.04467, 2016. DOI: 10.48550/arXiv.1603.04467

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ImageNet classification with deep convolutional neural networks, Advances in Neural Information Processing Systems, vol. 25, 2012. DOI: 10.1145/3065386

[19] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2016. DOI: 10.5555/3086952

[20] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998. DOI: 10.1109/5.726791

DISSEMINATION OF WORK

IRJMETS

e-ISSN: 2582-5208

International Research Journal of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

Volume:07/Issue:03/March-2025 Impact Factor- 8.187 www.irjmets.com

Smart CCTV Surveillance System Using Python

Pavan Lakkas*1, Aman Raipure*2, Shrishailya Wagh*3, Pratham Akkewar*4

Dr.R.A. Zamare*5

*1,2,3,4 Student, Department Of Computer Science And Engineering, Shri Sant Gajanan Maharaj College Of Engineering Shegaon, Maharashtra, India.

*5Professor, Department Of Computer Science And Engineering, Shri Sant Gajanan Maharaj College Of Engineering Shegaon, Maharashtra, India.

ABSTRACT

In recent years, the need for intelligent surveillance systems has grown significantly due to rising security concerns. Traditional CCTV systems require manual monitoring, which is both time consuming and prone to human error. This research presents a Smart CCTV Surveillance System using Python, integrating computer vision and artificial intelligence to enhance real-time security monitoring. The system utilizes Open CV, deep learning models, and machine learning algorithms for object detection, motion tracking, and fire detection. Additionally, its efficient security management. The proposed system enhances traditional surveillance by reducing human intervention and improving real-time threat detection accuracy. Experimental results demonstrate its effectiveness in identifying suspicious activities and reducing false alarms. This study contributes to the development of cost effective, intelligent surveillance solutions that can be implemented in residential, commercial, and public security domains.

Keywords: Smart Surveillance, Python, Computer Vision, Object Detection, Deep Learning, Fire Detection, CCTV Automation.

I. INTRODUCTION

With the increase in demand for security in various residential, commercial and public spaces the traditional CCTV systems have become inefficient due to its complete reliance on manual monitoring and operation. This conventional system needs constant monitoring and observing of multiple screens leading to fatigue, inefficiency and delayed responses to the threats. To tackle these limitations, the integration of artificial intelligent (AI) and computer vision in CCTV surveillance has emerged a promising solution.

This research focuses on developing a Smart CCTV Surveillance System using Python, using modern tech which enables real time monitoring and automated threat detection. The system is designed to detect the motion, recognize objects, harmful objects and analyze the threats without the need of continuous human intervention.

The main objectives of this research are to develop an intelligent CCTV system that can autonomously analyze video feeds using Python-based AI techniques, to implement real time object detection, and anomaly detection for proactive security measures, to enhance the traditional surveillance.

To achieve these objectives, the proposed system leverages deep learning models and computer vision algorithms to process video streams efficiently. By integrating advanced techniques such as Convolutional Neural Networks (CNNs) and pre-trained models like YOLO (You Only Look Once) or OpenCV-based object detection, the system can accurately identify potential threats in real time.

The Smart CCTV Surveillance System incorporates motion tracking and anomaly detection mechanisms to differentiate between normal and suspicious activities. It utilizes machine learning algorithms to learn behavioral patterns over time, enabling it to recognize unusual events such as unauthorized access, unattended objects, or aggressive movements.

By reducing human dependency and enhancing automation, this AI-driven surveillance system not only improves security measures but also minimizes false alarms and operational inefficiencies. Through continuous learning and adaptation, it aims to provide a robust and intelligent security solution that can evolve with emerging threats.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:07/Issue:03/March-2025 Impact Factor- 8.187 www.irjmets.com

II. METHODOLOGY

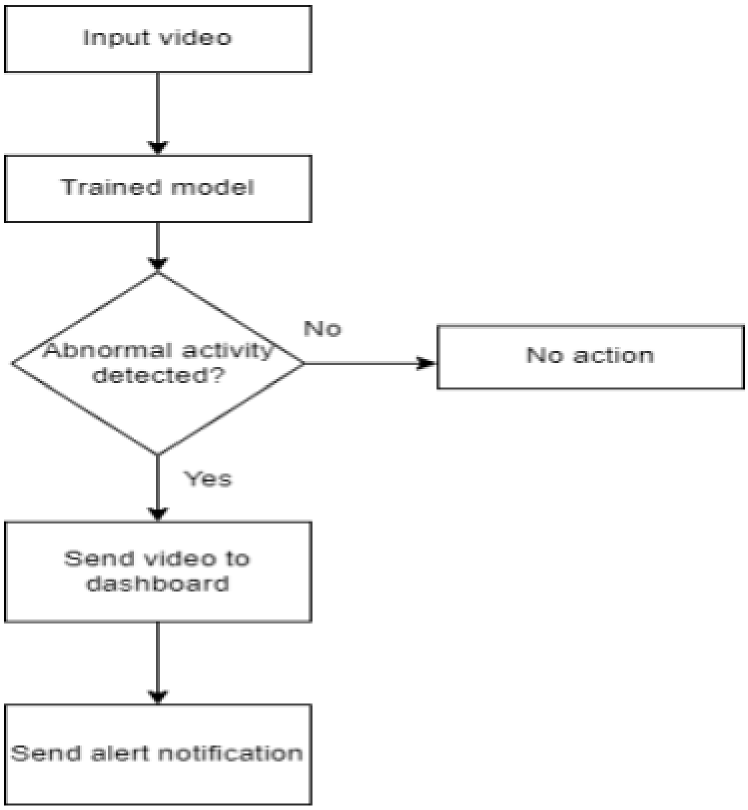


Fig 1: Abstract flow of the diagram

This project is built on one of the most advanced programming languages and a rapidly evolving field in computer science. It quips computers with the ability to perceive their surroundings, functioning as an artificial visual system. The project has a well-designed user interface, where each button is accompanied by an intuitive icon. Below, we explore its key functionalities in detail:

2.1 Object Detection

This feature evaluates the differences between two frames using the Structural Similarity Index (SSIM). It starts by capturing a reference frame, followed by a second frame that may contain noise. SSIM is then used to assess the similarity between these frames. Although SSIM is widely applied in machine learning and computer vision, there is limited detailed information on its gradient-based implementations. Luminance, which measures an image's brightness by averaging pixel values, is also an important factor in image processing.

2.2 Fire Detection

Fire detection systems analyze visual data to identify signs of a fire, such as smoke, heat, or flame. These systems use algorithms to process images or video feeds, comparing them to predefined patterns of fire-related features. Through techniques like thermal imaging and pattern recognition, the system detects irregularities that suggest the presence of a fire. While fire detection is critical in safety and security applications, there is ongoing research to improve the accuracy and speed of detection algorithms, especially in challenging environments with varying conditions.

Modern approaches integrate artificial intelligence (AI) and deep learning models to improve accuracy and reduce false alarms. By leveraging Convolutional Neural Networks (CNNs) and image processing techniques, these systems can distinguish between actual fire incidents and non-threatening sources of heat or smoke, such as steam or reflections.

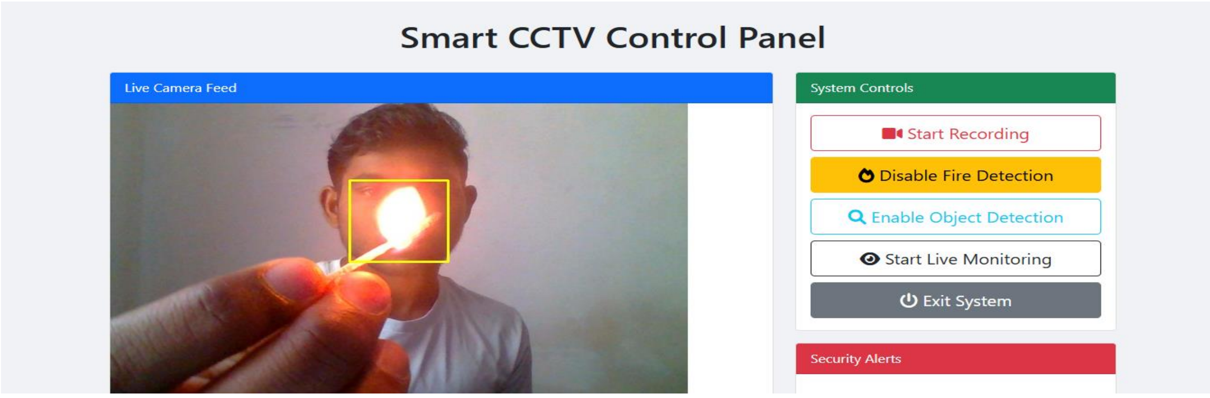


Fig 2: Smart CCTV Control Panel Interface with Fire and Object Detection Features

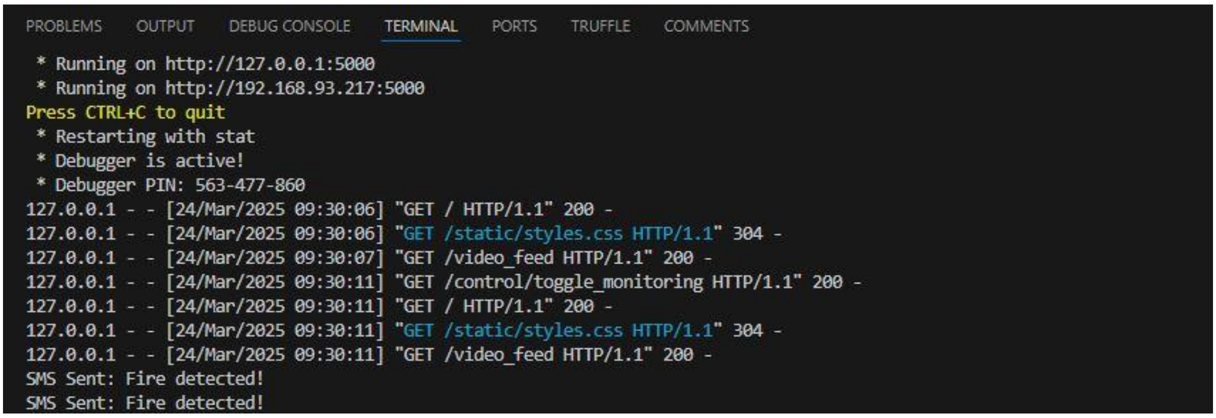


Fig 3: Terminal Output of Smart CCTV System Showing Fire Detection and Alert Notification

2.3 Real Time Monitoring

Real-time monitoring systems continuously track and analyze data to provide instant insights into various conditions or processes. These systems often utilize sensors, cameras, or other data sources to capture real-time information, which is then processed and evaluated to detect changes or anomalies. With applications ranging from industrial equipment monitoring to environmental tracking, real-time monitoring ensures timely responses to any issues. Ongoing advancements aim to enhance the efficiency and accuracy of these systems, particularly in handling large volumes of data and complex scenarios.

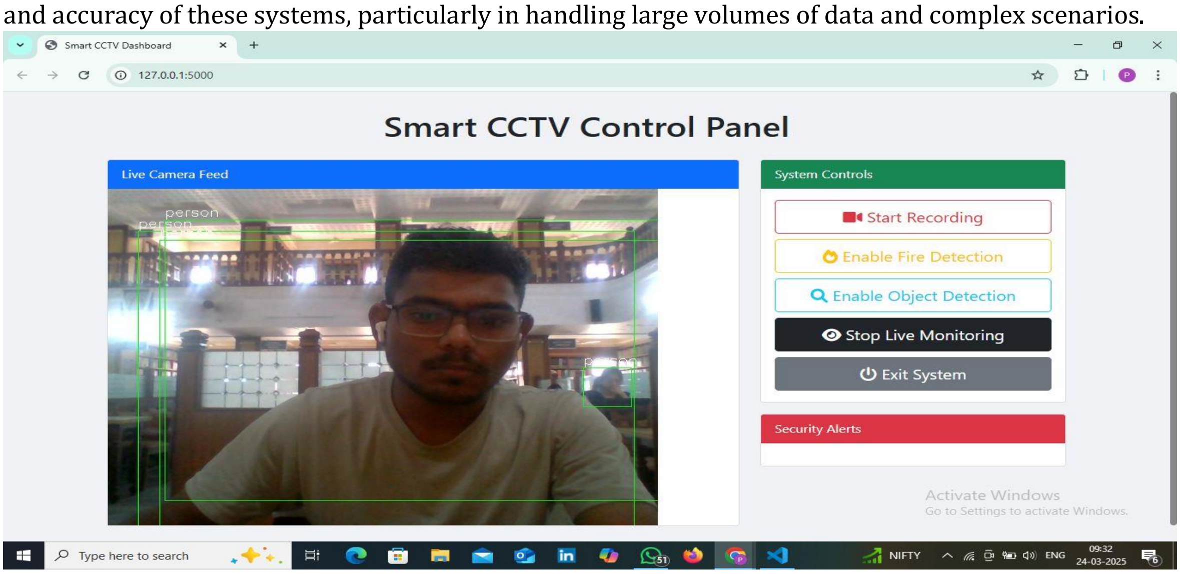


Fig 4: Smart CCTV Control Panel Interface with Real-Time Object Detection

2.4 Motion Detection

Motion detection systems are designed to identify movement within a specific area by analyzing changes in visual or infrared data. These systems use sensors, cameras, or other detection devices to monitor the environment and trigger alerts when motion is detected. Commonly applied in security, surveillance, and automated systems, motion detection helps to respond quickly to any unauthorized activity or changes. Continuous advancements in algorithms and sensor technologies aim to improve detection accuracy, especially in dynamic and cluttered environments.

III. MODELING AND ANALYSIS

4.1 YOLO (You Only Look Once) – A Real-Time Object Detection Model

YOLO (You Only Look Once) is a deep learning-based object detection algorithm that is widely used for real-time applications. Unlike traditional object detection models, which use region proposal methods (such as R-CNN), YOLO treats object detection as a single regression problem, making it extremely fast and efficient.

Working Mechanism:

The YOLO model divides an input image into a fixed grid of **S×S** cells. Each cell is responsible for detecting objects that have their centre within that cell. The model predicts bounding boxes, confidence scores, and class probabilities for objects in the image in a single forward pass of the neural network.

1. **Single Neural Network Approach**: YOLO applies a single convolutional neural network (CNN) to the image, making it faster than region-based approaches.
2. **Grid-Based Detection**: The image is divided into a grid, where each grid cell predicts multiple bounding boxes and confidence scores.
3. **Bounding Box Prediction**: Each bounding box consists of coordinates (x, y), width (w), height (h), and confidence scores.
4. **Non-Maximum Suppression (NMS)**: To filter overlapping predictions and retain the most accurate ones, YOLO applies NMS, improving detection performance.

4.2 TWILIO - A Cloud Communication Platform

A Smart CCTV Surveillance System uses AI-powered cameras to monitor activities, detect anomalies, and notify security personnel in real-time. Twilio is integrated into the system to provide instant communication via SMS, voice calls, emails, and WhatsApp alerts whenever suspicious activity is detected.

Working Mechanism:

1. Real-Time Video Capture: CCTV cameras continuously monitor the environment and send video feeds to a processing server.
2. AI-Based Object Detection and Anomaly Recognition: AI models (e.g., YOLO, OpenCV, TensorFlow) analyze video frames for unusual activity such as unauthorized entry, loitering, suspicious object detection and fire or smoke detection.
3. Event Triggering & Decision Making: If an anomaly is detected, the system generates an event trigger based on predefined rules. The backend server processes the detected event and classifies it as critical or non-critical.
4. Twilio API Activation: The server communicates with Twilio's API to send alerts through various channels such as SMS Alert, Whatsapp Notification and email notifications.
5. Security Personnel Response: The security team receives the alert and acts.

IV. RESULTS AND DISCUSSION

With advancements in technology enabling compact yet powerful processing capabilities, this concept has broad applications. Below are some potential future developments for this project:

- Developing a portable CCTV system.
- Integrating night vision functionality.
- Utilizing deep learning on high-performance hardware for enhanced analysis.
- Incorporating additional features such as weapon detection, anomaly detection, and fire detection.
- Designing a standalone application that operates independently without requiring external software like Python.
- Creating a **s**elf-sufficient device for seamless operation.
- Also have the high possibility of designing high accuracy model.

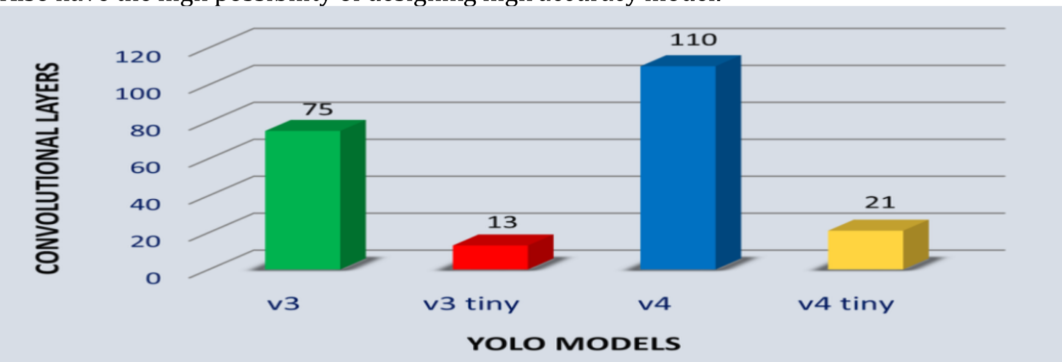


Fig 5: Comparison of YOLO Models Based on the Number of Convolutional Layers

V. CONCLUSION

The Smart CCTV Surveillance System using Python presents an efficient, automated approach to modern security monitoring. By integrating computer vision, machine learning, and real-time video processing, the system enhances traditional surveillance by offering automated motion detection, facial recognition, and anomaly identification. Utilizing Python and OpenCV, it provides a cost-effective and scalable solution adaptable to various environments, including residential, commercial, and industrial settings. The system's ability to operate independently with minimal human intervention improves security by providing real-time alerts, intelligent monitoring, and potential threat detection. Additionally, its modular architecture allows for future enhancements such as deep learning-based recognition, night vision, and advanced anomaly detection. In conclusion, this research demonstrates the potential of Python-based AI-driven surveillance to revolutionize security systems, making them more efficient, automated, and proactive in ensuring safety. Future developments can focus on enhanced AI models, cloud integration, and IoT connectivity to further optimize performance and expand its applications.

ACKNOWLEDGEMENTS

For this study the authors wish to express their profound gratitude to Professor R. A. Zamare for his exceptional guidance and unwavering support throughout the course of this research. His expert insights, constructive feedback, and persistent encouragement were instrumental in shaping the direction and depth of this study. Professor Zamare's commitment to academic excellence and his mentorship have significantly enriched the quality and impact of our work. We would also like to extend our sincere thanks to our colleagues, friends, and family members whose support and encouragement played a vital role in the successful completion of this project. Their valuable contributions, from providing critical feedback to offering emotional support during challenging times, have been indispensable. This collaborative spirit underscores the collective effort that drives innovative research and the advancement of telemedicine solutions.

VI. REFERENCES

[1] Subhash Chowdary Gadde, Harshith Guduru, Moses Babu Devarapalli, and Santosh Kumar Peketi "IOT- Based Smart Surveillance Systems," International Journal of Advanced Research and Development, vol.3, no. (2018)

[2] "Real Time Embedded Video Streaming Using Raspberry Pi", Aditi Shrikant Jadhav and Prof. Sudarshan R. Diwate, International Journal of Innovation Research in Science, Engineering and Technology, vol.5, no.1(2016)

[3] Aishwarya S Lande and B.P.Kulkarni, "Wireless Security Camera System," International Journal Of Ad vance Research and Development, vol.8 (2019).

[4] H. Kanma, N. Wakabayashi, R. Kanazawa, H. Ito, "Home Appliance Control System over Bluetooth with a Cellular Phone," IEEE Transactions on Consumer Electronics, vol. 49, no. 4, pp.1049-1053, Nov. 2003

[5] M. Ryan, "Bluetooth: With Low Energy comes to Low Security," WOOT'13 Proceedings of the 7th USENIX conference on Offensive Technologies, pp. 4-4, 2013

Paperpal Plagiarism Check

Cactus Global - PubSure SimCheck

Document Details

Submission ID
trn:oid:::6693:190462101

Submission Date
Jan 25, 2024, 12:09 PM GMT+5:30

Download Date
Apr 27, 2025, 1:37 AM GMT+5:30

File Name
Research_pepar.pdf

File Size
30.7 KB

4 Pages

1,838 Words

12,567 Characters

10% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text
- Small Matches (less than 8 words)

Match Groups

15 Not Cited or Quoted 10%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

10% Internet sources
0% Publications
0% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

- **15** Not Cited or Quoted 10%
  Matches with neither in-text citation nor quotation marks
- **0** Missing Quotations 0%
  Matches that are still very similar to source material
- **0** Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- **0** Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

10%	Internet sources
0%	Publications
0%	Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	www.healthcare.digital	2%
2	Internet	www.ncbi.nlm.nih.gov	2%
3	Internet	www.ovoarticles.com	2%
4	Internet	link.springer.com	<1%
5	Internet	www.scitechnol.com	<1%
6	Internet	vbn.aau.dk	<1%
7	Internet	ejmcm.com	<1%
8	Internet	mhealth.jmir.org	<1%
9	Internet	repository.belmont.edu	<1%

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 07/Issue 03/70300232248

DOI: https://www.doi.org/10.56726/IRJMETS70721

Date: 02/04/2025

Certificate of Publication

*This is to certify that author "**Pavan Lakkas**" with paper ID "**IRJMETS70300232248**" has published a paper entitled "SMART CCTV SURVEILLANCE SYSTEM USING PYTHON" **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 07, Issue 03, March 2025***

IRJMETS Impact Factor 8.187

Editor in Chief

We Wish For Your Better Future

www.irjmets.com

IRJMETS

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 07/Issue 03/70300232248

DOI: https://www.doi.org/10.56726/IRJMETS70721

Date: 02/04/2025

Certificate of Publication

*This is to certify that author "**Aman Raipure**" with paper ID "**IRJMETS70300232248**" has published a paper entitled "SMART CCTV SURVEILLANCE SYSTEM USING PYTHON" **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 07, Issue 03, March 2025***

IRJMETS Impact Factor 8.187

Editor in Chief

We Wish For Your Better Future

www.irjmets.com

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 07/Issue 03/70300232248

DOI: https://www.doi.org/10.56726/IRJMETS70721

Date: 02/04/2025

Certificate of Publication

*This is to certify that author "**Pratham Akkewar**" with paper ID "**IRJMETS70300232248**" has published a paper entitled "SMART CCTV SURVEILLANCE SYSTEM USING PYTHON"* ***in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 07, Issue 03, March 2025***

Editor in Chief

IRJMETS Impact Factor 8.187

We Wish For Your Better Future

www.irjmets.com

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 07/Issue 03/70300232248

DOI: https://www.doi.org/10.56726/IRJMETS70721

Date: 02/04/2025

Certificate of Publication

*This is to certify that author "**Shrishailya Wagh**" with paper ID "**IRJMETS70300232248**" has published a paper entitled "SMART CCTV SURVEILLANCE SYSTEM USING PYTHON" **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 07, Issue 03, March 2025***

Editor in Chief

IRJMETS Impact Factor 8.187

We Wish For Your Better Future

www.irjmets.com

PROJECT GROUP DETAILS

	Name :Pavan Daulatrao Lakkas Address: Near Mahadev Mandir, Paradh Khu, Bhokardhan , Jalna Mobile No: 8669098672 Email ID:lakkaspavan5@gmail.com
	Name : Aman Vijay Raipure Address: Mahalaxmi appt samta colony tukum, Chandrapu Mobile No: 9637541220 Email ID: amanraipire18032003@gmail.com
	Name : Shrishailya Dinesh Wagh Address: Bidkar ward Hinganghat, Hinganghat, Mobile No: 8010541247 Email ID: shriwagh2003@gmail.com
	Name : Pratham Manoj Akkewar Address: Balveer ward, In front of Mid-Town Hotel, Binba Road, Chandrapur Mobile No: 9322278924 Email ID: prathamakkewar7@gmail.com